

Automated Mid-Year OAB Benefits Conversion - Functional Considerations for Technical Success

Stewart St. Dennis
Chief Technical Officer
ZAXIA Inc.

Background

In the past few years, Oracle's employee benefit management functionality has grown significantly. Using Oracle Advanced Benefits (OAB) you can now:

- Centrally define all aspects of benefits programs,
- Declaratively define and enforce eligibility requirements,
- Manage life event driven benefits with a rich life event detection and management engine,
- Enable anywhere anytime employee self-service access to their benefit information,
- Record and action certification requirements and their implications,
- Take advantage of tight integration with Oracle HR,
- Take advantage of tight integration with Oracle Payroll,
- Work with tools that have full history and date-tracking capabilities, and
- Much more.

Sounds really good, huh? Sign me up you say! So, now that you own it, how are you going to use it? How are you going to stand it up and make it available? When should you cut-over? What else do you need to stand up to make it work? What about historical data? The questions are plentiful.

The Challenge

This paper will attempt to address some of these questions and, in particular, will focus on the situation where a mid-benefit year conversion is required. By:

- providing an overview of key components of the inter-related data underlying OAB,
- explaining the basics of how OAB works and is configured, and
- reviewing data migration issues and considerations,

this paper will target how functional and technical team members can work together to migrate data into OAB so that it is set up and ready to go on day one.

Unless you are a rapidly growing start up company, the chances are that your company already has a benefit management system of some sort. That system has information about employees, their dependants, employee and dependant coverage (including level, duration, costs, etc.), employee salaries, and a variety of other information needed to manage employee benefits. From a post-cut-over point of view, that legacy "system" will be kept around by the organization for legislative and data access purposes. However, it is likely that some of the information present in that environment needs to move over to the new OAB environment. The question becomes, how much data needs to come over and when should the cut over be done?

Key factors in addressing that question include:

- the amount of existing historical information available,
- the need for historical information to be present in OAB,
- the effort and expense needed to ensure appropriate quality of that data,
- the modules active in the current Oracle Enterprise Business Suite (EBS) implementation,
- the number of new modules being implemented along with OAB,
- the alignment of fiscal and benefit years,
- the breadth, quantity and complexity of existing benefit programs and associated rules,

- the number of employees involved, and
- the level of risk an organization is willing to accept.

Obviously, these factors are interrelated and can lead to contradictory outcomes. To ensure the success of the conversion, it is common practice to consider the contribution of each factor to the conversion.

Common conversion objectives include:

- ensure continuity of employee benefits, including accurate:
 - enrolments
 - enrolment details
 - financial implications
- timely and cost effective execution of the conversion process and
- support effective and efficient long-term maintenance of the converted data.

The easiest and fastest way to deal with the transition to OAB is to cut-over from the existing benefit systems at the beginning of the benefit year, establish an open enrolment, and have employees use OAB self-service capabilities to set up their own benefits. This is a fairly common way of cutting over.

However, what if, after considering the conversion objectives and contributing factors, this strategy is not appropriate for your organization? Examples of some scenarios where this might be the case include:

- multiple products are being implemented simultaneously, the fiscal and benefit years start at different times, and cutting over at the fiscal year end is more important than cutting over at the benefit year end and
- ongoing maintenance of benefit enrolments requires some historical information (like original enrolment dates needed for lock-in management) that may not be directly enterable by employees.

Some of the strengths of OAB, its strong rule base, enforcement engines and tight integration with other EBS data, can become hurdles during conversion of historical data. The absence and/or quality of key historical data can lead to false “rejects” when converting existing data. The functional and technical challenge in doing this type of conversion is striking a balance between data cleansing and short-term “softening” of the rules. The remainder of the paper will cover off the key aspects of what data is considered by OAB, how the rules are established and manipulated, what technical options are available, and some thoughts on how to temporarily stretch the rules without creating downstream conflicts when data changes occur.

The Data

For discussion purposes, the data foundation of OAB can be thought of as covering the following subject areas:

- Integration with core HR information (people, positions, assignments, salaries, contacts, unions/associations, operating units, etc.)
- Benefit program definitions (programs, plans, options, basic rules surrounding them, and eligibility rule components)
- Enrolments (life events, electable choices, and actual enrolments in plans and/or options)
- Integration with payroll (pay elements, rates, pay run integration/aggregation rules, pay out detail)

Figure 1: High Level Data Integration shows the high level data interrelationship of these subject areas. Each of the subject areas will be covered in more detail in the following sections. The diagram shows how, during enrolment processing, both program definition data and core HR data is considered to determine things like “is this person allowed to participate in a given life event” and “what electable choices are available to this person during this life event”. As well, it shows how, during payroll processing, HR, enrolment, and program data are considered, along with Payroll maintained rates, to determine what are the financial implications of current enrolments (addressing questions like “what does coverage of 1% of salary amount to as a period premium for optional life insurance for this employee”).

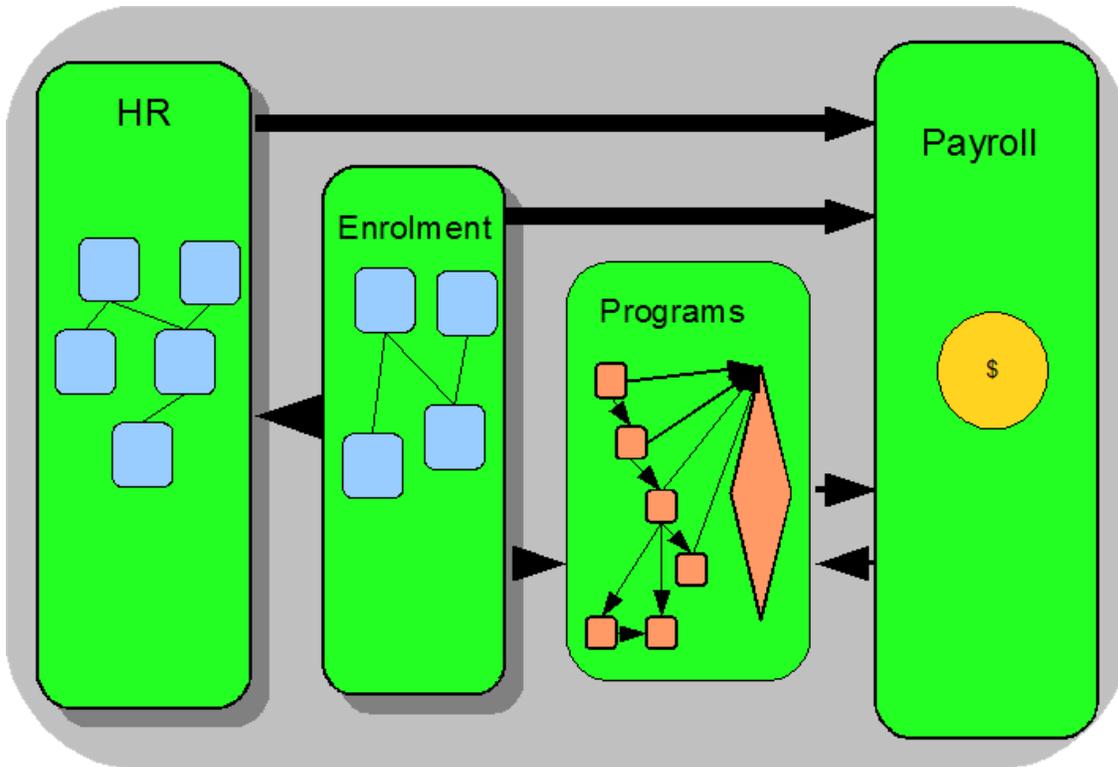


Figure 1: High Level Data Integration

All of the data model figures in this paper are significantly simplified logical representations of the underlying tables. Links from the underlying tables to other tables are present in the database, but are excluded from the figures in order to focus in on the key data items. Oracle's Electronic Technical Reference Manuals (eTRM) can be used by those who are interested in a more complete and accurate understanding of the reality of the complex data model underlying OAB and the other modules with which it interacts.

HR Core Data

The key data considered by OAB from HR is:

- Person (name, gender, age, type, etc.)
- Position (union, operating union, etc.)
- Assignment (operating unit, location, hours of work, ...)
- Salary (amount, frequency, etc.)
- Contacts (dependents, nature of dependency)

Figure 2: Core HR Data Model shows the logical relationships between the underlying tables that store this data. Due to OAB's tight integration with HR, this HR data is regularly referenced by OAB the instant it is needed. If an OAB rule needs to be evaluated, and the rule makes reference to HR data, then the moment the data needed, it is looked up and considered. As a result, all core HR data must be correctly recorded for each person BEFORE that person can participate in a benefit program. The key implication of this behaviour is that if all required data is not already present in HR, then an HR data conversion MUST be completed prior to starting an OAB data conversion. If a mid-benefit-year conversion is being done, with data being recorded retroactive to the beginning of that benefit year, then this means that all HR data must be accurately entered from the beginning of the benefit year to the cut-over date. This can have significant implications on the total conversion effort required and may result in limitations on how much historical election information can be converted.

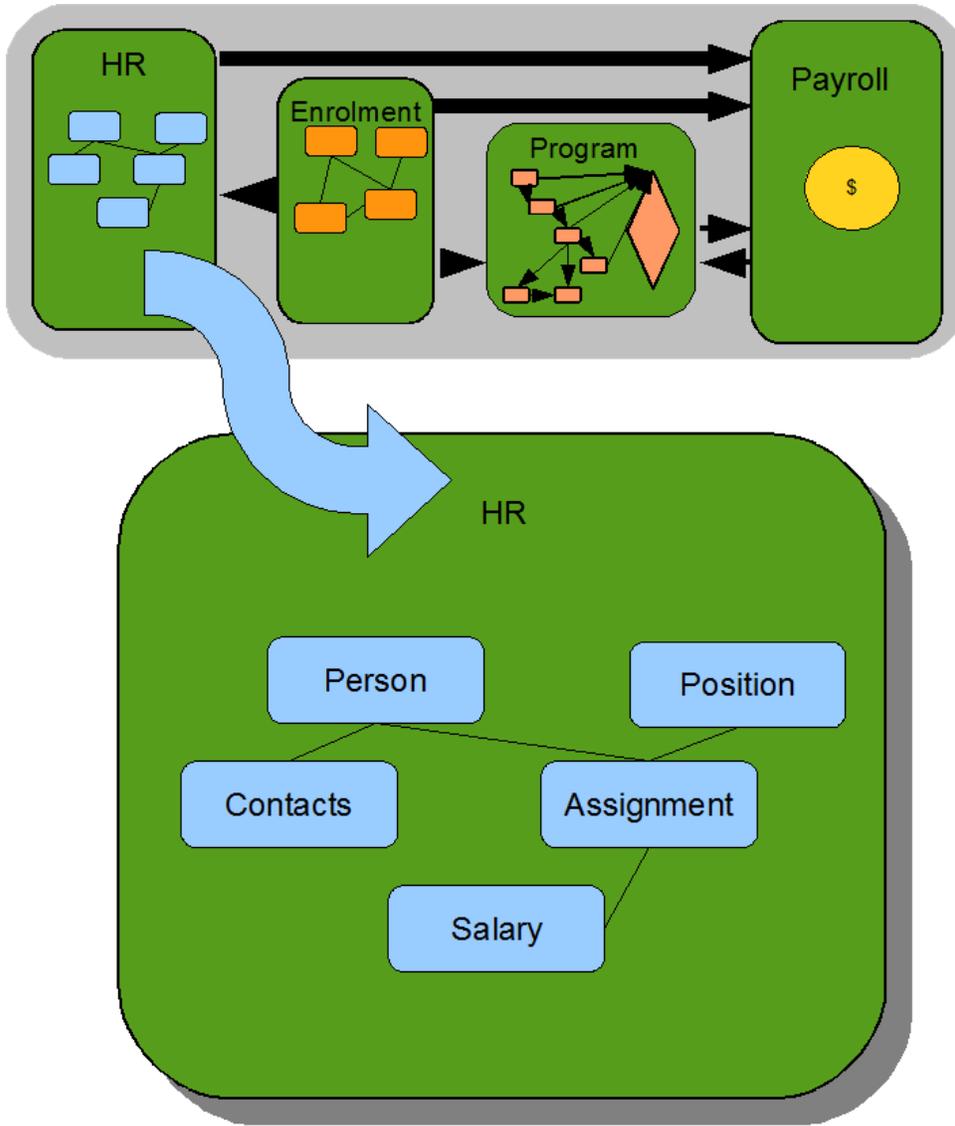


Figure 2: Core HR Data Model

OAB Benefit Definition Data

The key data that defines OAB Benefit offerings is:

- Program (overriding umbrella for clustered benefits)
- Plan Type (grouping of similar plans)
- Plan (the actual benefit made available to employees/dependents)
- Options (choices about the plan possible for employees/dependents)
- Plan/Option rates/behaviours
- The pool of rules governing program participation

Figure 3: OAB Benefit Definition Data Model and Figure 4: Program Eligibility Criteria show the logical relationships between the underlying tables that store this data. These figures show how the hierarchical nature of benefit programs are recorded, how linkages are established between program components and payroll elements and rates, and that eligibility rules are linked to key program component. Since an understanding of how eligibility rules are stored contributes no value to the focus of this paper, no table

details are provided for the eligibility rules aspect of the data model. Interested readers can review the Compensation and Benefits Management Guide and relevant eTRM information.

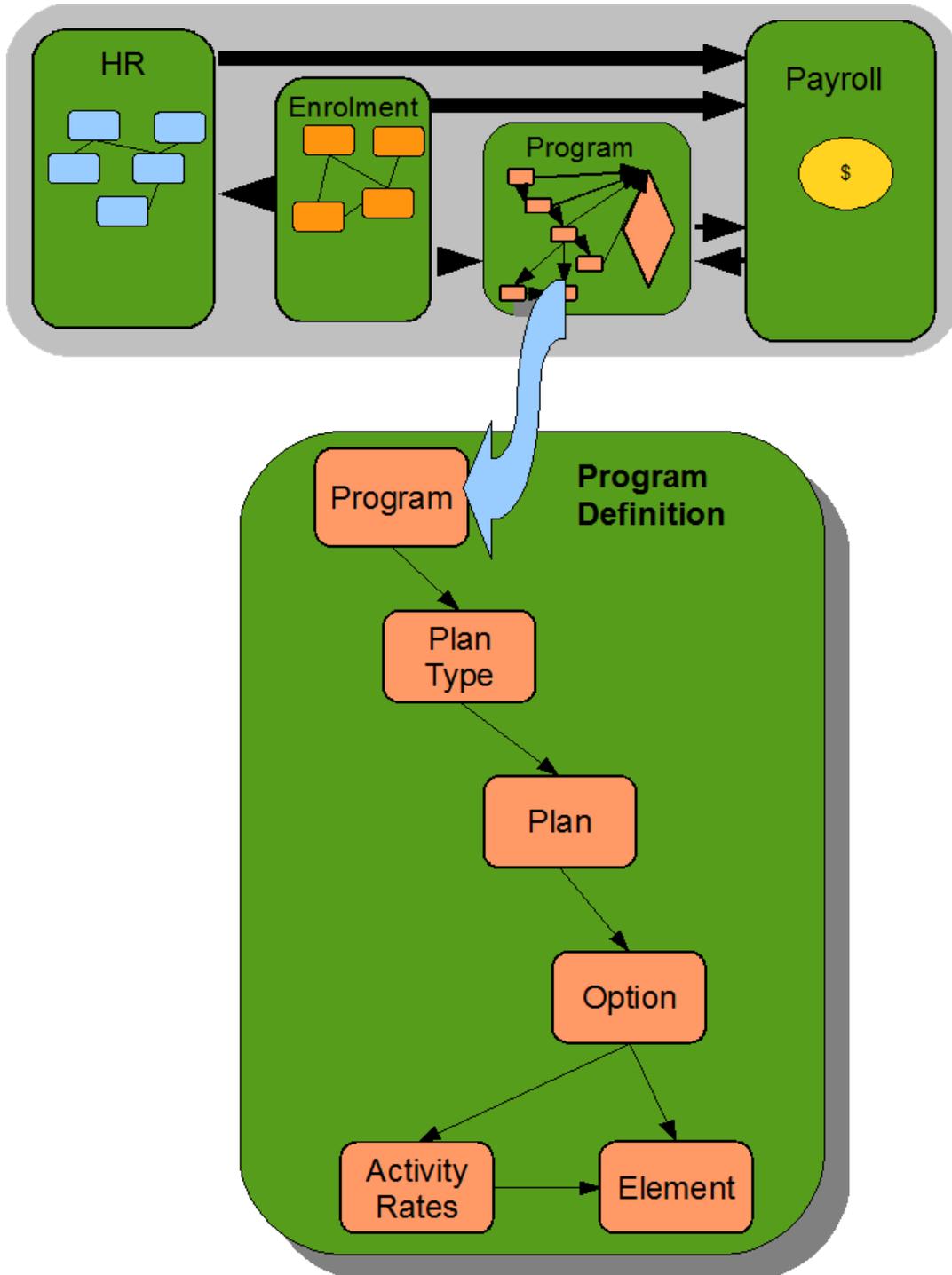


Figure 3: OAB Benefit Definition Data Model

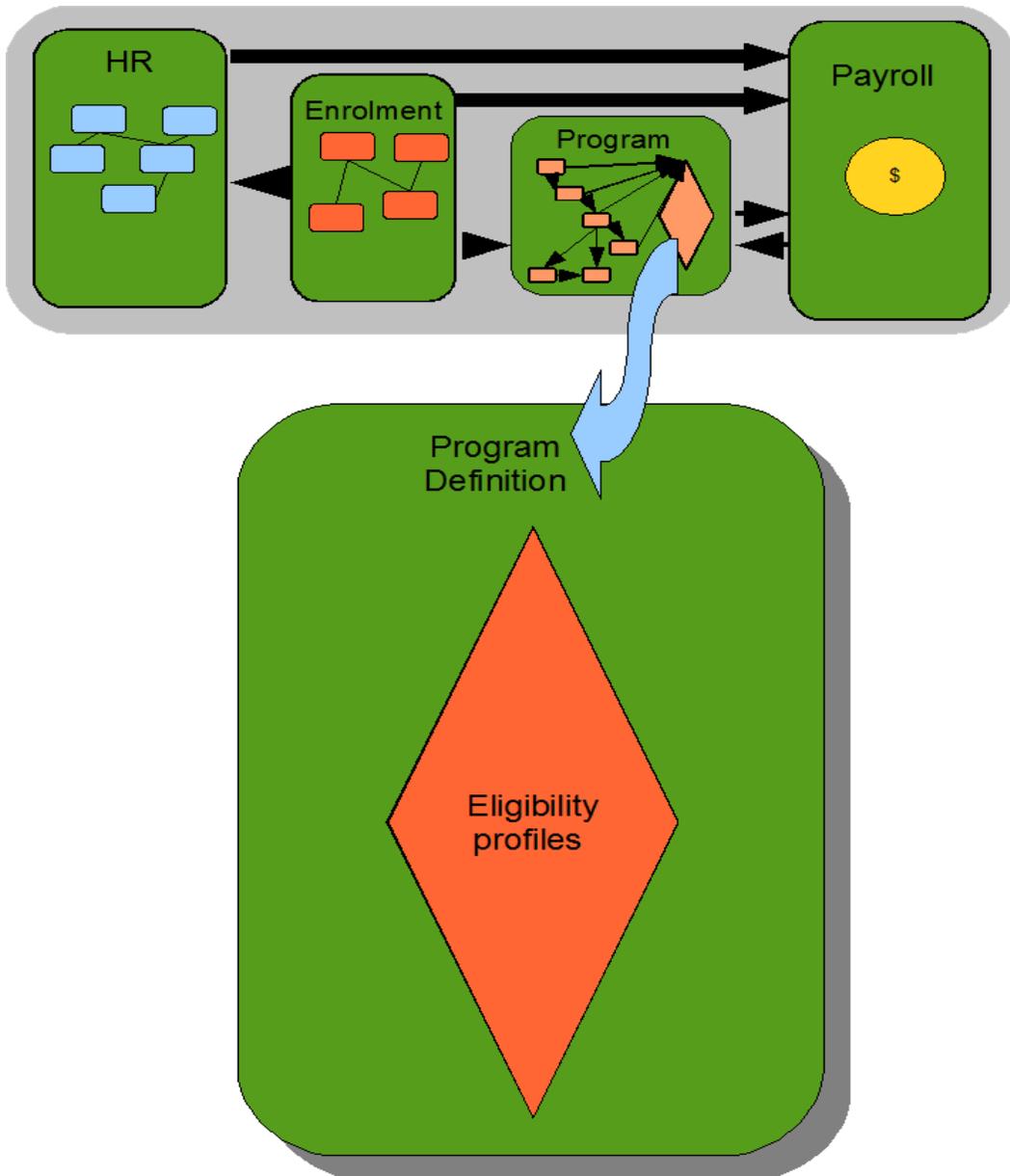


Figure 4: Program Eligibility Criteria

Program Participation Data

The key data that defines how people participate in individual benefits is:

- Life Events (nature, duration, etc.)
- Life event participation rules/behaviours (defaulting, rules to apply to each program/plan/option)
- Eligible dependants (family members who can participate in a given life event)
- Electable choices (plans/options available to employee/dependent participating in a given life event)
- Elections (actual enrolments made in plans/options as a result of the life events, including associated values (e.g. \$100,000 optional spousal life insurance))

Figure 5: Program Participation Data Model below shows the logical relationships between the underlying tables that store this data. This figure shows that people participating in a life event end up with electable

choices available to them during that life event and that those electable choices may be converted into actual elections during that life event. As well, it shows that during a life event dependent electable choices may be derived and then considered for enrolment.

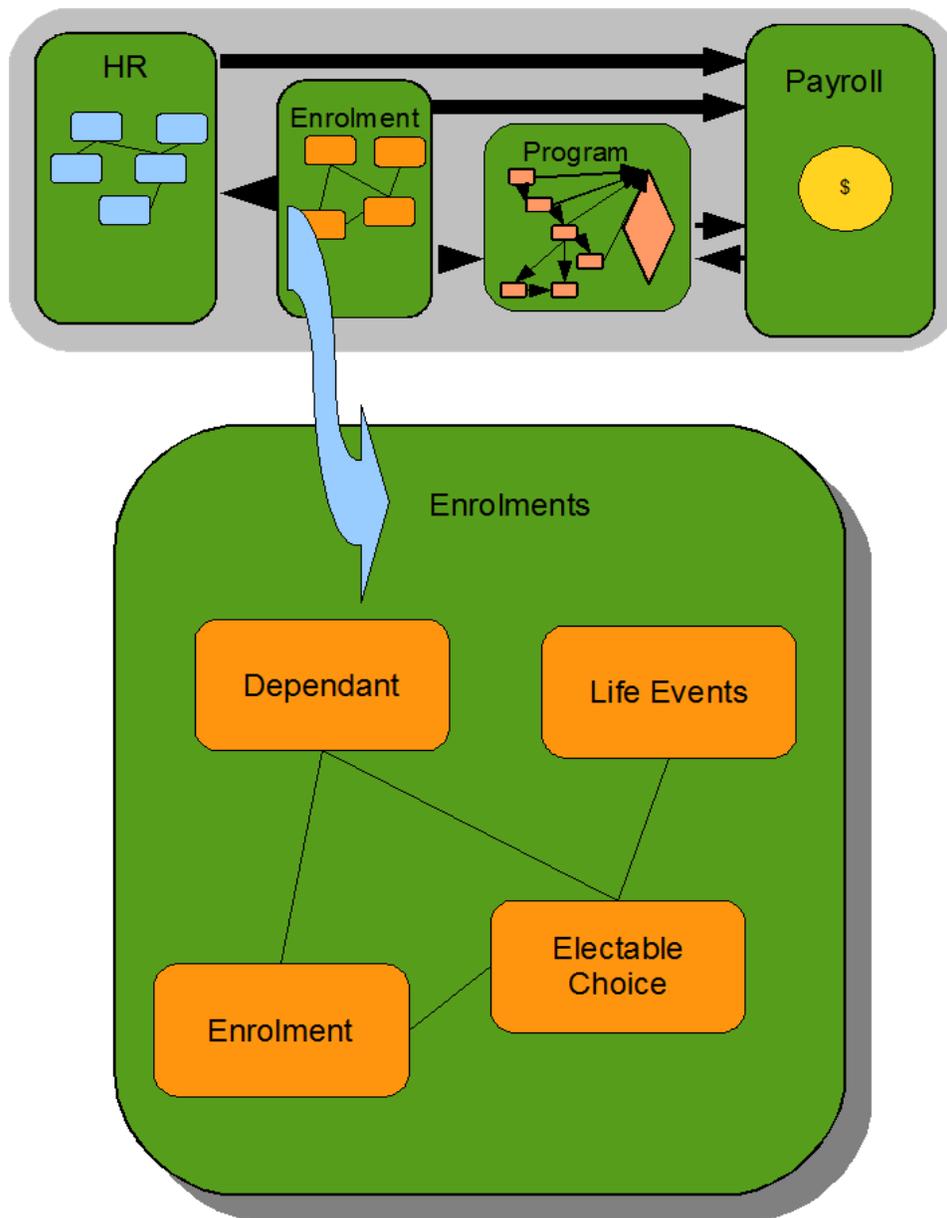


Figure 5: Program Participation Data Model

Payroll Data

The key data considered by OAB from Payroll is:

- Elements
- Rates
- Pay Periods
- Pay Runs
- Individual pay results

Figure 6: Payroll Data shows the logical relationships between the underlying tables that store this data. The underlying payroll data model is complex. Understanding that model adds no value to the focus of this paper. Therefore, no data model details are provided. Interested readers can review the Payroll Processing Management Guide and appropriate eTRM documentation.

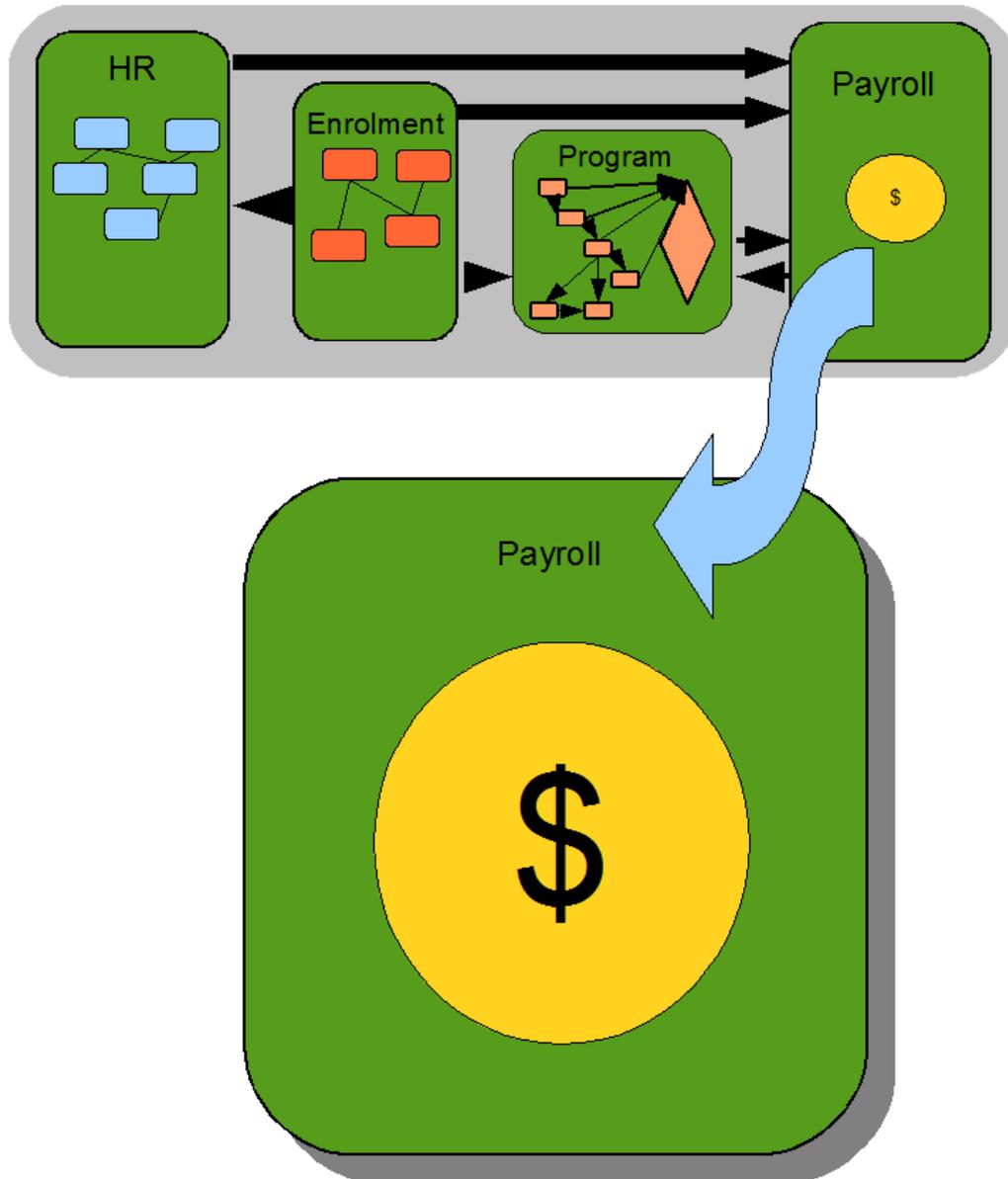


Figure 6: Payroll Data

Date Tracking

One of the hardest mental adjustments people have to make when dealing with Oracle HR and OAB is to really understand the point-in-time history capabilities of the products (i.e. date tracking). Most definitional and transactional data has effective date components. For each date tracked table, each core item can have multiple occurrences of itself over time. The system ensures that effective date ranges do not

overlap so that, at any given point in time, there is either zero or one active record for that item (person, assignment, etc.).

Figure 7: Sample Date-Tracked Timeline below shows a possible timeline diagram for a person and their assignment information over a fixed period of time. All OAB and HR forms have an effective date that they are working with when data is queried or updated. Depending when that point-in-time effective date is set, you would see either the first or second occurrence of the person, the corresponding first, second or third occurrence of that person's assignment, and the first or second occurrence of the salary associated to that assignment.

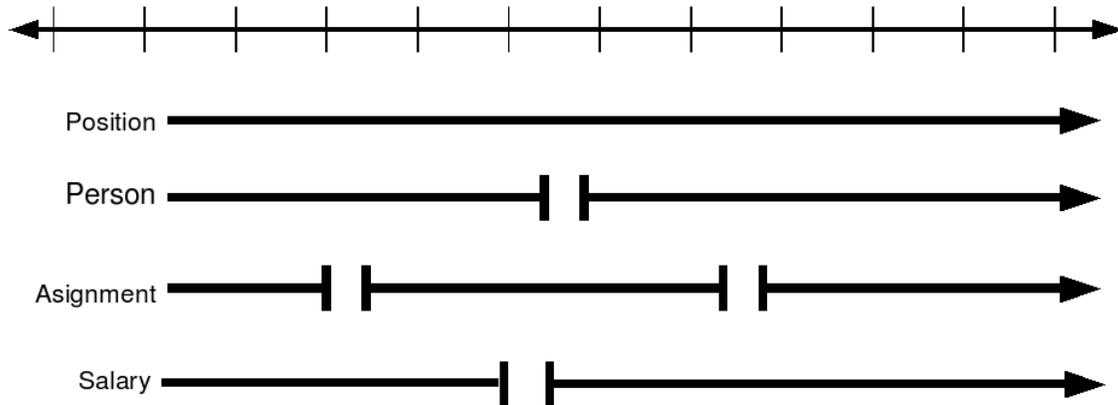


Figure 7: Sample Date-Tracked Timeline

If the effective date is moved far enough back in time, say prior to the hiring of that person, then that person will not be present in the database at that time. This has major implications for OAB enrolment conversions, as when the effective date of an enrolment is set, the database is queried for person related information AT THAT TIME. If the person does not have a salary record as of that time and there is a percentage of salary type rule used by a plan in the program, then they will not be able to participate in the life event. Furthermore, care must be used in setting effective dates for program information itself. If the plan does not exist at the time of intended enrolment, then enrolment in that plan is not possible.

Set Up 101

A full-blown discussion of OAB set up is beyond the scope of this paper. Instead, a quick summary will be provided to point out the areas where functional control is available. See the appropriate Oracle provided manuals for a more complete understanding the setup and recommended steps to complete it effectively. OAB set up can be classified as two separate but interrelated activities: functional set up and technical coding.

The functional setup defines the organization's programs, plan types within programs, plans within plan types, options for plans, life events, life event instantiation rules, linkages between life events and plans/options, the natures of programs, plan types, plans and options, eligibility rules, the use of those rules within life events, and integration between OAB and Oracle Payroll.

Technical coding is done within Oracle's Fast Formula framework and may require underlying custom packages and code. See documentation such as the Application Developers Guide and the Fast Formula Guide for further information on this topic. Fast formulas are program units built by technical staff that become integrated into the runtime OAB environment. They can be used to derive complex values such as lock-in dates, prorate flex credits, or convert various pay schemes into equivalent hourly or annual salaries. Fast Formulas can be used in a wide variety of situations where more straight forward declarative ways of deriving values is not possible. Once fast formulas are completed and made available, functional staff can integrate them into the OAB rule base.

The overall role of the OAB functional lead is to analyze the existing benefits programs and rules, clarify and codify those rules (rationalizing inconsistencies and gaps) and then record the resulting Programs and their rules within OAB. Because the rules defined within OAB are part of an “active dictionary”, those rules not only define the Program and its behaviour, they also are the foundation for the “runtime” behaviour of those Programs. Accordingly, a rule such as “a person must work 30 or more hours per week to participate in program X” is enforced consistently by the system. If a person’s work week is 29.95 hours, they do not qualify for Program X and they are excluded from it until explicit overrides are applied.

Many of the benefit systems that OAB replaces are not based on active dictionary type technology and are often designed such that the rules are in procedure manuals and benefits staff interpret/enforce the rules during data entry. OAB’s use of active dictionary type technology is what makes broad-based self-service benefits viable for organizations. However, that same technology means that OAB Program rules may be more strenuously enforced than were corresponding rules in the original system. This improved rule enforcement is a major source of issues when converting data from existing systems to OAB.

Conversion Methods

Oracle Metalink document 365034.1 (Oracle Benefits Enrollment Conversion) is the definitive source on the alternatives available for converting benefit enrolment data. Once the provided guidance is used to select a conversion method, this document contains significant discussion of specific processes to follow for various data scenarios. In summary, there are three basic conversion methods; manual, the HRMS Workbench or custom coding. OAB’s Application Program Interface (API) is set up so that at the lowest level, the same code is ultimately used no matter which method is chosen. This ensures that data integrity and rule compliance is ultimately ensured, regardless of method. The major differences between the three approaches are: accuracy, execution time/effort, complexity, participant team member profiles and flexibility. If the source data is clean enough and the mappings between the source and target data are clear enough, and a high degree of accuracy is required, then manual conversion is likely to be a poor choice. If case-by-case manual data interpretation is regularly required, then manual conversion may be unavoidable.

Assuming that manual conversion can be avoided, then the choice comes down to custom coding or using the HRMS Workbench. Custom coding relies heavily on qualified technical staff to create the custom code based on published public OAB API’s. The HRMS Workbench uses Excel based templates, Web ADI and Data Pump Technology, so it is more likely that functional-staff-only solutions may be possible. Depending on the staff available to the project, the complexity of the transform rules, the volumes to be converted and the reliability standards the conversion must meet, one or the other may be best suited for any given project. When a mid-year conversion is dictated and core HR data is incomplete, it is likely that qualified technical staff will be needed to implement a custom coded solution, but each implementation is different.

Regardless of the method selected, the basic steps used to record converted enrolment data remain the same. Once appropriate pre-requisites have been met (such as core HR data is present, programs are fully defined, life events are fully defined, and eligibility and derivation rules are fully defined), the overall process is:

1. Establish a potential life event for the employee
2. Evaluate the life event implications: potentially establishing eligible dependents, employee and dependent electable choices
3. Record individual employee and/or dependent enrolments
4. Validate enrolments
5. Close the employee’s life event.

Conversion Strategy

No matter how much care and effort is put into conversion design and planning, Murphy’s Law will ensure that it will not work cleanly the first time (or the second, third or subsequent times). Assuming that

continuity of post-conversion employee benefits is one of the key measures of success for the conversion, then a run-and-refine cycle should be used to refine the entire conversion process to ensure a smooth and accurate cut-over. By using this iterative approach, with each pre-cut-over pass through the conversion process, errors can be identified, reviewed and resolved. This approach can be used to validate the OAB setup, support data cleansing, and validate the conversion process itself. Using a run-and-refine cycle in this manner reduces the risk of errors or failures during the actual cut-over conversion. The most likely sources of error in any given conversion run are:

- Incomplete or incorrect:
 - program definition
 - source data
 - source data extracts
 - source/target data mappings/translations
 - Core HR data
 - Payroll setup
 - Payroll integration rules
- Conversion program errors.

OAB has life event back out functionality that can be highly valuable during both cut-over and run-and-refine conversion runs. A key functional contribution to the success of the conversion is the selection and definition of an appropriate life event (or life events) that will allow the desired behaviour and support on-demand back-out of partial or incorrect enrolments. OAB life event back-out functionality marks data changed during that life event as backed out. The resulting records are ignored by subsequent on-line and batch OAB and Payroll processes. The use of back-out functionality is a supported “undo” mechanism that can significantly reduce instance management requirements, as a full database and/or instance restores are not required between each test run.

Conversion Scenario

So far the tone of this paper has been more general. In order to get more specific about individual functional configurations adjustments that can ease the conversion coding effort, a specific conversion scenario will be created. Within that scenario, specific issues will be discussed and configuration adjustment options to address the issue will be discussed.

The Program

The OurFlexCredit Program is a Flex Credit based program where each employee is given a pool of credits that can be used to “pay for” various benefits within their means. This fairly common model is used by a wide variety of organizations where the employee is free to apply their credits towards benefits that are of value to them. Unused credits can be applied to a health spending account or paid out on a pro-rated basis over the course of the benefit year, which starts on May 1. The hierarchy of program, plan type, plan, and options is represented below by the indentation of the program components.

- OurFlexCredit Program
 - Basic Health
 - Basic Health
 - Provincial/State Healthcare
 - Locale 1 Healthcare
 - Single
 - Couple
 - Family
 - Opt Out
 - Locale 2 Healthcare
 - Single
 - Couple
 - Family

- Opt Out
 - ...
- Extended Healthcare
 - Essential Healthcare
 - Single
 - Couple
 - Family
 - Enhanced Healthcare
 - Single
 - Couple
 - Family
 - Extended Health Opt Out
- Extended Dental
 - Essential Dental
 - Single
 - Couple
 - Family
 - Enhanced Dental
 - Single
 - Couple
 - Family
 - Extended Dental Opt Out
- Optional Life Insurance
 - Employee Life Insurance
 - Spousal Life Insurance
 - Child Life Insurance
- Savings Plan
- Health Spending Account
 - HSA
 - HSA Opt Out

Various wait periods are in effect for new employees. Lock-in rules limit how often specific elections can be changed. Pro-rating rules apply to mid-year enrolments and enrolment changes. A life event model is used to control what changes can be made at various points in an employee's enrolment period. Enrolments can be without an amount associated to them or amounts can be recorded as absolute values or a percentage of things like employee's annual salary. Documentation is required for things like life insurance and over-age dependant coverage, including proof of good health, proof of age and proof of disability. Failure to provide required documentation results in suspension of the benefit. Retirees can enjoy some of the benefits after departure from the organization. Covered dependents can continue to receive benefits for a short period of time after the death of the employee.

The Situation

Oracle HR, Payroll and OAB are being installed simultaneously on top of an existing EBS environment where a shared HR environment has been in use for many years. The data contained in the shared HR environment is incomplete (it has no dependents and not all employees) and has only current minimal information needed for the modules it was installed to support. The cut-over date is driven by the fiscal year start and is Jan 1.

The source HR, benefits and payroll system has limited historical information. The benefits system has a limited set of enrolment rules enforced by the system, so much of the rule enforcement responsibilities fall on the shoulders of benefits staff, including life event detection and handling (things like marriage and change of salary are not automatically processed by the system). Most of the benefits maintained by the system do not have explicitly recorded opt outs. Over time, as new benefits were added to the system some of the financial implications of those benefits were not well supported by the system, so regular

intervention was required to ensure each pay run reflected the appropriate financial results for each employee. Beneficiary data is not stored in the source system.

Conversion Process

Regardless of whether the HRMS Workbench or custom coding is used, the same basic process is followed. However, for the purpose of this example, it is assumed that custom coding is done using supported OAB API's. Without stating error handling, the overall conversion process is:

1. Extract source system data
2. Transfer and load data into custom interface tables
3. Process each employee:
 - 3.1. Get appropriate employee information
 - 3.2. Create the appropriate potential life event for the employee
 - 3.3. Process the life event
 - 3.4. For each expected enrolment to be loaded:
 - 3.4.1. Validate and map source plan and option descriptors to corresponding OAB component identifiers
 - 3.4.2. Deal with skipped enrolments by establishing appropriate opt out mechanisms
 - 3.4.3. Transform any plan or option values into appropriate OAB units/values
 - 3.4.4. Get appropriate electable choice information for the enrolment
 - 3.4.5. Process the enrolment
 - 3.4.6. Record enrolment results in appropriate tables and log files
 - 3.4.7. Address any needed certification requirements
 - 3.5. Validate Employee enrolments and log results
4. For each successfully processed employee:
 - 4.1. Get appropriate employee and life event information
 - 4.2. For each employee's dependents:
 - 4.2.1. Get appropriate dependent information
 - 4.2.2. For each expected enrolment to be loaded:
 - 4.2.2.1. Validate and map source plan and option descriptors to corresponding OAB component identifiers
 - 4.2.2.2. Deal with skipped enrolments by establishing appropriate opt out mechanisms
 - 4.2.2.3. Transform any plan or option values into appropriate OAB units/values
 - 4.2.2.4. Get appropriate electable choice information for the enrolment
 - 4.2.2.5. Process the enrolment
 - 4.2.2.6. Record enrolment results in appropriate tables and log files
5. Produce appropriate completion reports

The OAB enrolment conversion is executed as one of the last conversions. Because of the pre-requisites created by HR core data dependencies, all HR related conversions (person, assignment, salary, and dependents) need to be completed and validated prior to starting the benefit enrolment conversion. For retirees with no active assignments, benefit assignments need to be created.

Note about Dates

With all of the dates involved in establishing life events and enrolments, a brief discussion of the key dates and their meanings is very valuable. The key dates are:

- Effective start date: The earliest date that a given record will show up as a date-track effective date moves back in time.
- Enrolment start date: The most recent start date that the enrolment became active on.
- Original enrolment date: The first in the series of contiguous start dates that have occurred over the life of the employee's enrolment in a given election.

Conversion Issues and Associated Functional Adjustments

Potential conversions issues, and associated possible functional setup adjustment that may resolve them, will be covered in the top-to-bottom order they would be encountered in the conversion process listed above.

Program Definition

In order to ensure that all enrolments can occur when they need to, all effective dates within all aspects of the Program set up must be before the earliest possible enrolment start date to be recorded.

Life Event Dates

Depending on the definition of the life event, individual election original enrolment dates can occur before the life event start date and individual enrolment start dates can not. For this type of life event, this means that the life event start date must be on or before the earliest enrolment start date across all enrolments to be created through the life event. When creating a life event, the effective date needs to be on or before the life event start date. Since life event eligibility criteria are evaluated as of the effective date, all relevant person and dependent data must be effective on or before the life event start date. There are two major options for the preferred life event dates to be used:

1. The start of the beginning of the benefit year for which mid-year enrolments are being recorded or
2. The cut-over date.

There are pros and cons with each of the options. In this scenario, it is desired that the benefit year start date be used. With the limitations of the source data, only current employment and salary data is available for conversion. For employees who have had assignment or salary adjustments between the beginning of the benefit year and the cut-over date, this means that the desired life event start date is not achievable. This has multiple implications on the conversion that will be discussed later in this paper. Figure 8: Sample Person Enrolment Timeline shows a sample employee's employment and enrolment data that demonstrates this situation.

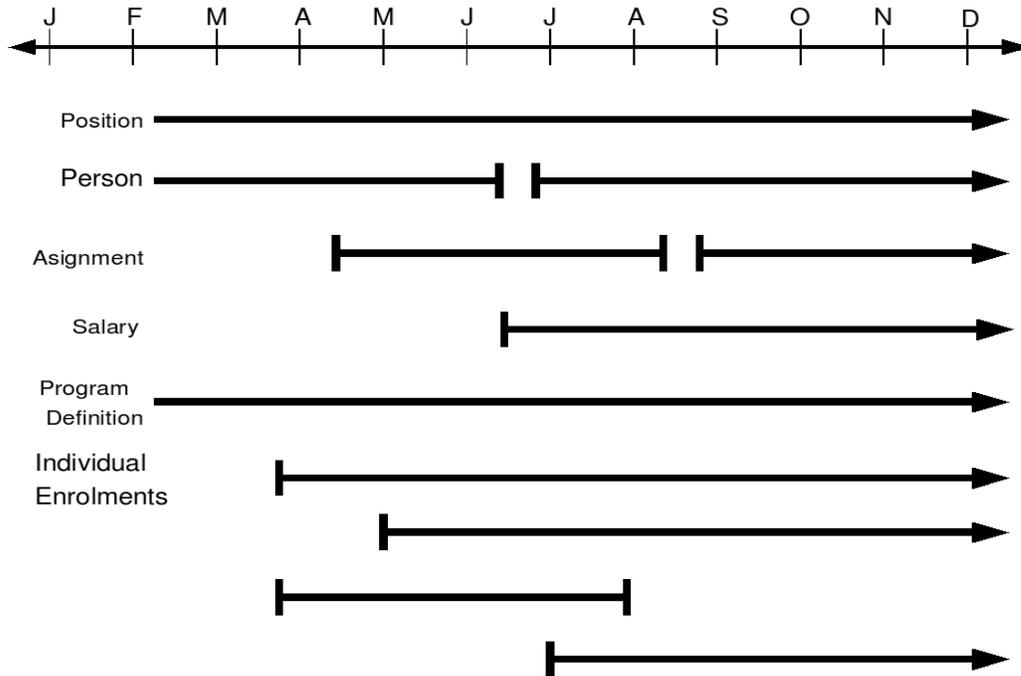


Figure 8: Sample Person Enrolment Timeline

The relevant dates for this employee's scenario are:

- Benefit year start date: May 1
- Benefit cut-over date: Jan 1 (at the end of the timeline)
- Earliest viable employment date: June 10 (driven by the salary effective date)
- Earliest enrolment start date: March 15

Based on those dates the desired life event start date is May 1 and the earliest possible life event start date is June 10.

Potential Life Events

Given the significantly different rules governing the processing of active and retired employees, separate life events should be established for each situation. Since, given the definition of the sample program, benefits may extend to dependents of both active and retired employees; each life event must be configured to process dependents. With the appropriate life events defined, it is possible to process each type of employee by relying on OAB defined rules and not via simulations of those rules imposed upon a more generic and "loose" life event.

Default Enrolment Processing

Depending on what version of OAB is being used, there may be two different enrolment defaulting schemes used when potential life events are converted to started life events (via the appropriate Participation Process concurrent program). In 11i.10 RUP2, OAB default enrolment processing was changed. Up to, and including, 11i.10 RUP1, defaults defined for a life event where created during Participation Process processing. From RUP2 onward, default processing was accomplished via a separate concurrent process and only occurred after the enrolment period passed for an "empty" life event. The complexity of the defaulting behaviour becomes a big factor in determining if default processing should be simulated by the conversion program or if the life event should be configured so the default processing can be done right after the life event is started. If non-trivial defaulting behaviour is required, then the life event can be configured so that the enrolment period is very short (1 day) so that the default enrolment process can be run to establish defaults in a manner consistent with the 11i.10 RUP1 and prior model.

Program Component Mappings

Depending on how close the source and OAB systems were able to get to correctly modelling the organization's benefit programs, the mapping process from source system values to corresponding OAB Program components may be a simple substitution process (replace source code X with OAB component name Y). However, if the source system was unable to correctly model the company's benefit programs, then complex mappings may be required. If hard-to-specify splits and merges are required to convert a block of source benefit elections into their corresponding OAB components, then consideration can be given to carrying forward the "weaknesses" of the source systems set up. While this may be a sub-optimal solution, a balance may need to be struck between the effort needed to migrate to the "correct" structure versus the inconvenience of carrying forward an "inappropriate" program definition.

Opt Out Processing

Depending on how the source system deals with terminated enrolments or people who have opted out of a given benefit, OAB enrolments may need to be created in the face of terminated or missing source system elections. If the OAB Program is configured appropriately, implied source system opt outs can be converted into explicit OAB opt outs at the plan or option level.

Value Transformations

Depending on how the source system records election values, transforms from the source unit of measure values to the new OAB unit/value equivalents may be required. Furthermore, adjustments to flex credit “opening balances” and/or HSA election amounts may be required. This situation may arise depending on flex credit pro-rating rules and whether “compromises” were required on the life event start date. As well, depending on subtle variances between the source and OAB employee election-costing behaviours, negative OAB unused flex credit amounts may occur for individuals who had appropriate zero or positive unused balances in the source system. Depending on the OAB rule base, negative unused balances are normally inappropriate. The simplified formula that governs flex credit balances is:

$$\begin{array}{rcl} & \text{Opening Flex Credit Amount} & \\ \text{minus} & \text{Total non-HSA election costs} & \\ \text{minus} & \text{HSA election amount} & \\ \text{equals} & \text{Unused Flex Credit Amount} & \end{array}$$

Election rate/cost calculations are totally driven by the program component set up and the corresponding integration with Payroll. There is no readily available programmatic way to influence the costs employees incur for their non-HSA elections. Therefore, to ensure an appropriate unused flex credit balance, you can either adjust the opening balance upwards or decrease the HSA amount. Depending on the number of employees and the expected variance between the costs, a simple solution is to make a one-time global increase in the opening flex credit amount. By adding a few percent more to the opening balance, this class of reconciliation issues can be totally avoided.

In the example case presented in the *Life Event Dates* section, the desired life event start date was May 1 and the earliest viable life event start date was June 10 (creating “compromise” mid-year enrolments). Care needs to be taken to ensure that unused flex credit and/or HSA amounts are appropriate for mid-benefit-year enrolments. Automatically applying or not applying pro-rating rules to the conversion life events can have implications for both “real” and “compromise” mid-year enrolments. For “real” mid-benefit-year enrolments, the source system values for HSA election amounts have been determined based on pro-rated flex credit opening balances. If automatic pro-rating is disabled, then artificially high unused flex credit amounts will occur. If automatic pro-rating is enabled, then unused balances will be appropriate. For “compromise” mid-year enrolment, inappropriate unused flex credit balances will like occur whether pro-rating is enabled or not. The easiest technical solution is to work with pro-rating enabled. That way HSA values can be adjusted to get appropriate balances. For those employees subjected to “compromise” dates, their HSA amount will be artificially low as the pro-rating formula can be used to reduce it to an amount needed to get an appropriate unused flex credit amount (and special instructions will be needed to explain to them what happened). With this approach, all employees will experience the appropriate financial behaviour on their pay checks.

Electable Choice Issues

If core HR dependent data is not set up correctly, then some option-in-plan expected electable choices will not be present (e.g. If the rules engine concludes the employee has no dependents, then Single will be the only available option for a plan like Enhanced Health). There is no functional work around for this that makes sense. The solution to this issue lies in pre-requisite conversion issue resolution (data cleansing and/or bug fixes).

Depending on an employee’s primary address, place of work, and provincial/state health care plan rules, people who live near province/state boundaries may have source system enrolments that are not among derived electable choices. Some of these issues can be addressed through data cleansing. Others are legitimate and may involve “temporal exemptions” due to plan transition rules (employees may have a period of time after they move to a new locale before health care needs to be switched to the new province/state). There may be functional set up approaches to address this issue. Otherwise post-conversion manual overrides may be required to fix the data on a case-by-case basis.

Lock-in Date Management

OAB lock-in date determination and management is highly configurable. Flex formulas can be used to calculate lock-in/earliest de-enrolment dates from original enrolment dates using plan specific rules. Depending on “compromise” life event dates and earliest de-enrolment date processing rules, inappropriate lock-in dates (those that disagree with the source system) may be recorded. Enabling earliest de-enrolment date override processing can make it possible for the conversion process to override the derived date.

Certification Processing

A number of plans and options require certification documentation of certain information. Examples include proof of good health and proof of age for life insurance. General enforcement practice is that missing required certifications result in suspension of the benefit until documentation is received. Certification compliance can be inferred in OAB by virtue of the fact that an active enrolment is present in the source system. Accordingly, removing the certification requirement during the conversion life events eliminates inappropriately suspended OAB enrolments.

Generalized Issue Resolution Process

The above issues and associated functional set up adjustments were identified by using the run-and-refine testing strategy discussed above. By running the conversion over and over, various issues were identified and resolved with each test run. The importance of this strategy can not be overstressed if employee benefit continuity is a major goal of the conversion process. Making the effort to create an environment where a baseline environment is readily re-established makes on-demand full-population test runs viable. This will allow issues to be classified as source data errors, conversion errors, set up errors or cases for special case handling. By getting benefit staff, project functional and project technical staff working together, appropriate solutions to each implementations specific issues can be identified and actioned.

Conclusion

Oracle Advanced Benefits is a powerful and feature rich product that is highly configurable. In some implementations, benefit enrolment conversion is required. In those implementations where mid-year conversion is required, accurate data conversion can be quite challenging. By taking advantage of the configuration power of OAB and creating a good working environment between benefit staff, project functional staff and project technical staff, various issues can be resolved in the appropriate domain. Wise use of life event configuration can significantly ease the technical effort to correctly migrate the data. Approached correctly, a high quality conversion can be achieved at the lowest possible cost.