Release Notes for 2023-08-01

Highlighted Changes

- Addition of a “validation only” mode to action requests. Setting this flag within the `header` section of a mutation will skip the submission of the request to Oracle. Requests which would have otherwise submitted to Oracle and received a `PENDING` status will instead return with status `VALIDATED` to indicate that they were not actually sent.
- *Addition of recognition of use of Sandbox vs. Production API keys.*
    - In the API Gateway, each boundary system has two sets of credentials. One labeled Sandbox, the other Production. In practice, the back-ends for both will point to the Oracle instance indicated by the endpoint path you are using. (`/ait-sit/1`, `/ait-test/1`)
    - However, in the upcoming test cycle, there will be a behavior change when using your Sandbox credentials which will persist into production.
    - When using your Sandbox credentials submit a GraphQL mutation operation, the request will be made with validate-only semantics as above. This will allow the testing of systems against current production data without the possibility of submitting test data to the ERP system. And, it will allow smoke testing of the production connections during our final cutover.

- Please note that there are some breaking changes noted below. We once again had to perform some cleanup for schema consistency. We apologize for the inconvenience if any of these fields are in use.
- Considerable updates to accounting period validation logic. If you are using the GlAccountingPeriod status for determining which period to use, or ever supply an accountingDate on a glJournalRequest, you may be affected by these changes.
    - **First:** *For All Practical Purposes, Accounting Period Name is Ignored.*
    - Oracle uses the accountingDate as the master identifier of a period. The accountingDate’s calendar month determines the accounting period. If you send both, and they did not match, Oracle will reject the journal. As of this update, we detect this and reject it for you. `:-)` (Recommendation…never send in accountingPeriodName…a lot fewer rules will get in your way if you leave it out.)
    - If no accountingDate is specified, we default in the current date for you. (This was happening already - but now we do it earlier and run validations against it.)
    - **The accounting period used for checking open/closed status is the new PpmAccountingPeriod.** This is because this period closes first of the five parallel accounting periods in Oracle. Once it closes, we need to stop posting journals of any content to that period. *However, in practice, this will be unlikely to ever take effect…read the next bullet point.*
    - **There is now a cutoff date for each period.** As we had with KFS, there is a calendar date after which you can no longer post to each period. The timing of this date is unknown, but the dates will be set by Finance based on the reporting deadlines for each period. This date can be found as the `journalCutoffDate` on the `GlAccountingPeriod` and `PpmAccountingPeriod` objects.

Types

- **ActionRequestHeaderInput** : Added the `validateOnly Boolean` property to this header object. Prevents action request from being submitted.
- **ApInvoice** : The `paymentAmount` was corrected to be a Float in this release. `invoiceAmount Float` was also added.
- **ErpAccount** : `ppmAllowed Boolean!` was added to indicate that the given natural account is also an eligible Expense Type when used as part of a PPM segment string.
- **GlAccountingPeriod** : Added `journalCutoffDate LocalDate` field to the object as informational as to when the period will no longer allow posting. This is necessary to allow for time to perform period closing activities within reporting deadlines.
- **PpmAccountingPeriod** : Same as `GlAccountingPeriod` but for the Projects module of Oracle. (There are actually 5 different period calendars in Oracle…we just didn't know this one was important to boundary systems until last week.)
- **PpmAward** : *(BREAKING CHANGE)* Changed the name of field `lastUpdatedBy` to `lastUpdateUserId` for consistency with all other objects. Not sure anyone cares about this field. But in case you were a completist when pulling fields from the award, this one may affect you.
- **PpmCfdaAward** : *(BREAKING CHANGE)* It was fated to be that we would change the name of field `creationFate` to `creationDate`.
- **PpmExpenditureType**
  - Added field `revenue Boolean!` to indicate that the "expense type" is really a revenue natural account which will be accepted by the PPM integration code on faculty projects. (Integration code applies such to the project via GL, and routes documents to adjust the budget of the project to the central office for validation and further processing.)
  - *(BREAKING CHANGE)* Changed the name of field `lastUpdatedBy` to `lastUpdateUserId` for consistency with all other objects.
- **PpmFundingSource** : *(BREAKING CHANGE)* Changed the name of field `lastUpdatedBy` to `lastUpdateUserId` for consistency with all other objects.
- **PpmProject* : Added `legalEntityCode` field. (You shouldn't need this for anything…but we did.) Seriously - this code matches the ErpEntityCode value to which the GL transactions associated to this project will eventually post.
- **PpmSponsor** : Overhaul…and given the limited use for this, I will let a comparison of the schema be the docs.
- **PpmTask** : *(BREAKING CHANGE)* Changed the name of field `lastUpdatedBy` to `lastUpdateUserId` for consistency with all other objects.
- **PpmTerms** : *(BREAKING CHANGE)* Changed the name of field `lastUpdatedBy` to `lastUpdateUserId` for consistency with all other objects.
- **RequestStatus (enum)** : Added `VALIDATED` value for validate-only mutation calls.
- **ScmPurchasingCategory** : Added `expenditureTypes: [PpmExpenditureType!]!` to provide the ~~object codes which are allowed to be used with a commodity code~~ PPM expenditure types which are allowed to be used with a given purchasing category.

Operations

- **Added**: `ppmSponsorByNumber(sponsorNumber: NonEmptyTrimmedString30!): PpmSponsor` : It does what it says…
- **Added**: `ppmAccountingPeriod / ppmAccountingPeriodByDate / ppmAccountingPeriodSearch` : They do what they say…