ONESOURCE INTEGRATION FOR ORACLE FUSION TAX
PRODUCT UPDATES
VERSION 2.6.0.12

This document provides the following information.

For additional information, see the following on the ONESOURCE Customer Center:

- Integration for Oracle Fusion Tax Documentation
- Integration for Oracle Fusion Tax Product Updates

VERSION 2.6.0.12

UPDATE SUMMARY

Thomson Reuters is pleased to provide a fix version of ONESOURCE Integration for Oracle Fusion Tax on top of the 2.6.0.11 release. The 2.6.0.12 version supports the following enhancements:

1. Attribute Ordering

The integration code has been enhanced to ensure that all User element attributes are consistently displayed in the expected sequence within both the Determination Request XML and Determination Response XML.

Previous Behavior

- User element attributes were included in the XML payloads, but their order was not consistent.
- The attribute sequence could vary between requests and responses, making XML comparison, validation, troubleshooting, and downstream processing more difficult.

Updated Behavior

- User element attributes are now explicitly generated in a predefined and consistent order.
- The same attribute ordering is maintained across both:
  - Determination Request XML
  - Determination Response XML
- This improves XML readability, simplifies debugging, and ensures alignment with schema or consumer expectations where attribute ordering is required for review or comparison purposes.

Benefits

- Consistent XML structure across all transactions.
- Easier request/response comparison during troubleshooting.
- Improved maintainability and readability of integration payloads.
- Better compatibility with systems or validation processes that expect a standardized attribute sequence.

Summary: The enhancement standardizes the ordering of User element attributes in both Determination Request and Response XMLs, resulting in more predictable, readable and easier-to-maintain integration payloads.

2. Integration UI Credentials Sync

An issue was identified with the **Credentials Sync** functionality in the integration UI, where credential updates were not consistently synchronized on the first save. In some cases, users had to save the credentials multiple times before the updated passwords were properly synced.

Previous Behavior

- Credential updates were not always synchronized immediately after saving.
- Users occasionally needed to save the credentials more than once for the changes to take effect.
- Inconsistent synchronization could result in authentication issues during batch processing and calculation runs.

Updated Behavior

- The credential save and synchronization process has been enhanced to ensure credentials are properly persisted and synchronized on the first save.
- Password updates are now reliably propagated to the integration components without requiring additional save attempts.
- Batch calculations can now utilize the updated credentials immediately after a successful save operation.

Benefits

- Eliminates the need for multiple save attempts.
- Improves reliability of credential synchronization.
- Reduces authentication-related failures during batch calculations.
- Provides a more seamless and predictable user experience.
- Enhances overall stability of integration processing.

Summary: The Credentials Sync process has been improved to ensure that updated credentials are successfully saved and synchronized in a single operation, eliminating the need for repeated saves and improving the reliability of batch calculation executions.

3. Dynamic Mapping

The Dynamic Request Mapping functionality has been enhanced to support Split Mapping, providing greater flexibility when transforming and mapping data from the Fusion request payload to the OneSource payload.

Previous Behavior

- Dynamic Request Mapping supported only a one-to-one mapping relationship.
- A single source field could be mapped directly to only one target field in the destination payload.
- When multiple target attributes required values derived from the same source field, additional source fields or custom BIP queries were often needed to populate the required data.

Updated Behavior

- Dynamic Request Mapping now supports Split Mapping.
- A single source field can be split into multiple values using a configured delimiter (for example, comma, pipe, or other supported separators).
- The resulting values can then be mapped to multiple target elements within the OneSource destination payload.
- This transformation occurs dynamically during request processing, eliminating the need for additional source fields or custom data extraction logic.

Benefits

- Enables more flexible and powerful request payload transformations.
- Reduces dependency on additional Fusion source fields and custom BIP queries.
- Simplifies integration configuration and maintenance.
- Improves efficiency when populating multiple related target attributes from a single source value.
- Enhances scalability for complex mapping requirements while minimizing customization effort.

Summary: The Dynamic Request Mapping enhancement introduces Split Mapping support, allowing a single source field to be divided into multiple values and mapped to multiple target elements. This streamlines request transformations, reduces integration complexity, and provides greater flexibility in handling complex payload mapping scenarios.