SedonaWeb/SedonaAPI 2.0 Release Notes

November 2026

Version 1.55.0

Contents

SedonaWeb 2.0 2
Enhancements 2
Sub item of Roll Up of RMR in SedonaOffice Reflects on Invoices in SedonaWeb 2.0 (SedonaWeb 1.0 parity) 2
Migrating FSU Technicians to the Sedona-X Mobile App 2
Users with the default password must now change it at first sign-in to SedonaWeb 2.0 3
Application Corrections 4
Security of Sedona connection strings in SedonaCloud database [00166118] 4
SedonaAPI 2.0 5
Enhancements/Features 5
ADD, UPDATE, and DELETE Endpoints for new JobPrevailingWage Model 5
Barcode Lookup API Support 5
Application Corrections 6
Time stamps from the technicians show Eastern time even though in Pacific time [42069, 00131306, 00135587] 6
API - Cannot send Aging Date or Posting Date in the POST for api/CustomerInvoice [55480, 121200] 6
API allowed invoices to be imported with an invoice type of M, which is not a valid invoice type [57006] 7
SedonaWeb2 behavior - cross session authentication/contamination [00167220] 7
Inventory API creates records with invalid ValueOnHand state 8
Allow SedonaOffice empty date placeholder for optional purchase order dates 8
Special Upgrade/Installation Instructions 9
Supported Environments 9

SedonaWeb 2.0

Enhancements

Sub item of Roll Up of RMR in SedonaOffice Reflects on Invoices in SedonaWeb 2.0 (SedonaWeb 1.0 parity)

Recurring items set as sub items in SedonaOffice are no longer listed as individual items on the invoice in SedonaWeb 2.0. The invoice in SedonaWeb (print view only) only displays the roll up item, not the sub items. The structure of invoice items in SedonaWeb 2.0 matches the structure of items on the SedonaOffice invoice.

Migrating FSU Technicians to the Sedona-X Mobile App

A host administrator can now migrate FSU technicians to the Sedona-X mobile app in bulk from the Edit Company page. This replaces the previous **Migrate FSU Technicians** checkbox, which did not work as intended because technicians need an email address and a role group to be migrated. The Edit Company page now has a Migrate FSU Technicians section with a **Download** button and an **Upload** button.

Step 1: On the Edit Company page, the administrator clicks the **Download** button and receives an Excel file listing the company's FSU technicians. The file shows each technician's first name, last name, FSU username, any email address already on file, and a Role Group column. The names and the FSU username cannot be edited. The Role Group column is a dropdown with six choices: Junior Technician, Junior Technician II, Subcontractor, Senior Technician, Project Manager, and Manager. Each technician starts with Junior Technician II unless a different role group is already on file for that technician.

Step 2: Fill in the Excel file. The administrator enters an email address for each technician to migrate and can change the role group. Only the technicians to migrate need an email address. A technician with no email address is skipped and does not stop the others from being migrated. The same email address may be used for more than one technician.

Step 3: Upload the file. The administrator clicks the **Upload** button to upload the completed file. The file can be uploaded more than once, so technicians can be migrated in groups over time. Only technicians with newly provided email addresses are migrated. The system checks the file when it is uploaded and shows warnings for rows that do not pass.

What happens to each technician: Each migrated technician receives a Sedona-X mobile account in the selected role group. The technician's FSU settings carry over to the new account. These include the days of appointments the technician can see, the maximum dispatch, arrival, and departure minutes, the

access level, and permissions such as rescheduling appointments, manual labor, system updates, subcontractor charges, final charges, removing contacts, credit card and bank access, closed tickets, and time and attendance. The role group can be changed later to customize permissions. Each technician is also linked to the matching employee record. Each migrated technician is sent an email titled “Welcome to Sedona-X — Set Your Password” with a link to set a password.

Usernames: Each technician keeps the FSU username when possible. If a technician has no FSU username, the email address is used. Spaces in a username are replaced with underscores; characters that Sedona-X mobile does not allow are removed. Each username must be unique across all companies. If a username is already taken, the system adds the lowest available number to the end, for example “john1”. If the username already ends in a number, the system uses the next higher available number, for example “john2”. A technician whose username is not taken keeps it and is not changed by another technician in the same upload. After the migration, the results list shows the Sedona-X username for each technician, which is the username the technician uses to sign in to the Sedona-X Mobile app.

Results: After the upload, a message states how many technicians were migrated. A list shows the result for each technician, including the technician’s name, FSU username, Sedona-X username, email address, role group, and status.

- Repaired: A technician from an earlier incomplete migration was finished.
- Up to date: The technician was already migrated.
- Skipped: The technician had no email address or an invalid one.
- Failed: The account could not be created, or its roles or settings could not be assigned.

If a welcome email could not be sent, the technician’s status says so. The list can be filtered by group and exported to Excel. Technicians who are already migrated are not changed when a file is uploaded again.

License check: If the company has a software license key on file, the system checks it before it creates any accounts. The migration stops and no technicians are migrated, if the license does not include Sedona-X mobile users or if migrating the technicians would exceed the number of mobile users the license allows.

Users with the default password must now change it at first sign-in to SedonaWeb 2.0

SedonaWeb now requires a new user who signs in with the standard default password to change it before doing anything else. This helps protect the data of their customers. It applies to new accounts only. Existing accounts keep working and are not prompted.

After a user signs in with the default password, SedonaWeb takes them to the Change Password screen. The user’s email address is filled in and the screen explains: “You are using a default password. You must change it before you can continue.” The user cannot open any other page until the password is changed.

The only other action available is Logout. The explanation stays visible if the user submits the form with an error. After a successful change, the user is signed out and signs in again with the new password.

This requirement applies to these new accounts:

- HostAdmin accounts created from this release on.
- Company administrators and users that an administrator creates without entering a password. These accounts receive a default password automatically.
- Field service users (FSU) created by the field service user migration.
- Customer web logins created from the Sedona Office desktop application with the default password. These logins are created the first time the customer signs in to SedonaWeb, and the customer is asked to change the password right away. If a customer's SedonaWeb password is reset to the default in Sedona Office, the customer must change it at the next sign-in.

If an administrator enters a password of their own choice for a new account, the user is not prompted to change it.

The default password can no longer be chosen anywhere in SedonaWeb. This applies when adding a user, adding a company, changing a password, resetting a password (including an administrator reset from the Users grid), and registering. A user who tries to use it sees the message: "This password is the product default and cannot be used. Please choose a different password." When an administrator resets another user's password from the Users grid, that user is not asked to change it a second time.

If an administrator changes an account while a user is already signed in, the requirement can take up to about 5 minutes to appear or clear. A new sign-in always reflects the current state.

The requirement applies to sign-ins through a web browser only. Sign-ins that use an API token, including integrations, are not affected.

Application Corrections

Security of Sedona connection strings in SedonaCloud database [00166118]

Previously, SedonaWeb stored the database connection information for each company in three separate settings: the SedonaOffice, SedonaMaster, and SedonaDocument connection strings. When a company or its connection settings were edited from the SedonaWeb site, only the SedonaOffice connection string was protected. The other two settings could remain stored as readable text.

This release corrects that issue. All three connection strings are now protected with encryption when a company is added or edited. Each company's own existing encryption key is used, so no new setup or configuration is required in your environment.

Background jobs that run for a company now properly unlock the protected SedonaOffice connection string before using it, so those jobs continue to connect to the database as expected. The system also records a warning message whenever it finds a connection string that is not yet protected. This helps the support team identify any affected records.

SedonaAPI 2.0

Enhancements/Features

ADD, UPDATE, and DELETE Endpoints for new JobPrevailingWage Model

Users can now create, edit, and delete prevailing wage records for a job. These records let them decide which wage labor grades are available to use for that job.

When users add or change a record, they can identify the job by its job number or its job code and can identify the labor grade by its identifier or its description. The system checks that the job and the labor grade are valid. A job cannot have more than one prevailing wage record for the same labor grade. If users try to create a duplicate, the system shows the message "A prevailing wage record already exists for this job and labor grade." If they try to edit or delete a record that cannot be found, the system also shows an error message.

Users can also view all prevailing wage records for a specific job.

Barcode Lookup API Support

Users can look up parts by barcode value using the existing part list and the existing warehouse parts list. The warehouse parts list shows only the parts in the selected warehouse. This release makes lookups easier by showing the available filter options on the API documentation page. This also supports the ability to look up parts by barcode in the Sedona-X mobile app.

Endpoints that support filtering now list the available query options, which are $filter, $select, $expand, $orderby, $top, and $skip. Each option includes a short description and an example. For example, the Warehouse list shows how to filter by name and how to limit the number of results.

The documentation page has moved to a new address, which ends in "/swagger". The previous address, which ended in "/api/help", automatically redirects to the new page.

Every endpoint now shows a lock icon to indicate that sign-in is required. When users select Execute on an endpoint and are not yet signed in, a sign-in window asks for their username and password. After signing in, they can run other endpoints without signing in again until refreshing the page.

If an unexpected error occurs while the API is processing a request, the API now returns a consistent message, "An unexpected error occurred." The error is also recorded in the system logs.

Application Corrections

Time stamps from the technicians show Eastern time even though in Pacific time [42069, 00131306, 00135587]

Technicians were entering notes in Sedona-X Mobile. The techs were in Pacific time as was the customer. Time zones should be showing Pacific in SedonaOffice, but they were showing as Eastern time.

The Sedona-X Mobile app was pulling timestamps from the server, which was set to Eastern time, leading to the incorrect timestamp display. We changed the API to set the edit date to now if it was not supplied in the payload. This ensures consistency and avoids reliance on server time zones.

API - Cannot send Aging Date or Posting Date in the POST for api/CustomerInvoice [55480, 121200]

This release improves how the invoice date, aging date, and posting date are handled when a customer invoice is created through the API.

The posting date is always set to the same value as the invoice date. The posting date cannot be set separately, and if a posting date is included in the request, it is ignored.

The aging date can be included in the request. If it is not included, the aging date is set to the invoice date.

If the invoice date is not included in the request, the system uses today's date for the invoice date and the posting date. Because of this, the invoice date should always be included when importing invoices with earlier dates, so that those invoices are not given today's date. The accounting period that contains the invoice date must be open.

Invoice dates and aging dates should be sent in the format year-month-day, for example 2026-03-04. This is the only format that is read the same way on every server. Other formats are read using the regional settings of the server, so a date such as 03/04/2026 could be read as either March 4 or April 3.

If an invoice date or aging date is not a valid date, or is earlier than January 1, 1753, the invoice is no longer processed with an unclear failure. Instead, the system returns a message that names the date that was not valid and asks for the year-month-day format.

When an invoice item creates a customer equipment record, the date on that equipment record is the same validated invoice date used for the invoice.

The API documentation for creating a customer invoice has been updated to describe all of these date rules.

API allowed invoices to be imported with an invoice type of M, which is not a valid invoice type [57006]

Previously, the POST /api/CustomerInvoice API accepted any text as the invoice type, including values that SedonaOffice does not recognize, such as “M”. This could cause data problems later. The POST /api/CustomerInvoice API now checks the invoice type before an invoice is created. The only accepted invoice types are Job (J), Service (S), Cycle (C), and Other (O). If any other value is submitted, the invoice is not created. The API returns an error message that names the value submitted and lists the valid values.

This release also fixes an error that could occur in the response returned after a new invoice was created through the POST /api/CustomerInvoice API.

SedonaWeb2 behavior - cross session authentication/contamination [00167220]

A customer reported a security concern in SedonaWeb. Sessions could cross over between users, so a person could end up signed in as someone else.

This was caused by how SedonaWeb created new user accounts from SedonaOffice. When a new account was created, SedonaWeb also signed that new user in. This could happen while an administrator was already signed in and viewing the list of users. In that case, the administrator's session was replaced by the new customer's session.

This release corrects the issue. SedonaWeb no longer signs a user in when it creates or updates user information from SedonaOffice. The login and registration screens continue to sign users in on their own, as before.

Inventory API creates records with invalid ValueOnHand state

There were three issues in the Inventory API. All three had the same cause. When a record was updated, the API did not first load the existing stored record, so any value the request left out was treated as zero instead of keeping its stored value.

1. Creating a record: A new inventory record was saved with no Value On Hand at all, which left the record in a state that SedonaOffice would never produce.
2. Updating a record: If an update did not include a Standard Cost, the API treated it as zero. This caused an unintended cost adjustment to be recorded for the item.
3. Deleting a record: Those unintended adjustments caused the API to refuse to delete the affected records, because it does not allow deletion of an inventory record that has journal activity.

This release makes the following corrections:

1. A new inventory record is now created with a Value On Hand of zero. This matches the behavior of SedonaOffice.
2. An update now keeps the existing Standard Cost, Minimum Quantity, and Maximum Quantity when the request does not include them. An update that explicitly sets Standard Cost to zero still saves zero.
3. Value On Hand is a calculated balance that builds up through inventory transactions, so it cannot be set directly. If a request to create or update an inventory record includes Value On Hand, the API now returns a 400 error. Previously, an update that included Value On Hand returned a 204 success response even though the value was not saved. The 400 error replaces that misleading response.

Records that were created through the API before this release may already have unintended adjustments that prevent deletion. This release does not remove them. Sites experiencing this should contact Bold Group support for assistance in identifying the tables and records that need to be corrected.

Allow SedonaOffice empty date placeholder for optional purchase order dates

The Purchase Order feature of the SedonaOffice API now accepts the SedonaOffice “no date” placeholder, December 30, 1899, in optional date fields. SedonaOffice uses that date to mean that no date has been set.

A recent validation rule rejected any date before 1980. That rule blocked the placeholder date. A customer that needed to leave an optional date empty could not do so. Leaving the date out did not work either, because SedonaOffice returned Error 94 when it loaded an empty date.

The placeholder date is now accepted in the optional Due Date, Shipping Date, Entered Date, and Edit Date fields of a purchase order. When a Due Date or Shipping Date is left empty, the system now fills in the placeholder date, so SedonaOffice receives a value it can process.

The placeholder is not accepted in the required date fields. These are the Order Date, Acknowledged Date, and Vendor Acknowledged Date. The 1980 minimum still applies to them. The checks that a Due Date or Shipping Date must be on or after the Order Date now apply only when both dates are real dates, not the placeholder.

The related documentation for the API was also updated.

Special Upgrade/Installation Instructions

If your company uses SedonaWeb 2.0/SedonaAPI 2.0, IT will update the version at the same time as your SedonaOffice upgrade. This is to ensure compatibility with all modules using SedonaWeb 2.0/SedonaAPI 2.0.

.NET Framework 4.8 automatically uses TLS 1.2. TLS 1.2 must be the only TLS version Enabled in the Registry. TLS 1.0 and TLS 1.1 must be disabled. Verify that they are disabled and that TLS 1.2 is enabled.

Supported Environments

- This version of SedonaWeb 2.0/SedonaAPI 2.0 requires SedonaOffice version 6.2.0.24 or later.
- Server (where SedonaWeb 2.0 is installed) has Microsoft .NET 4.8 installed.