

SedonaCloud - Technical Configuration - Changed API Endpoints [1.42.0]

01/02/2024 6:43 pm EST

(for #23045, 23046, 23047, 23048, 23625, 23997, and 29556)

SOLUTION SUMMARY AND IMPACT OF CHANGES:

Note: These new endpoints correct many problems found in their corresponding obsolete endpoints. Customers that have an existing integration using the obsolete endpoints will not benefit from these changes until they modify their integration to use the new endpoints. If customers are experiencing problems with an obsolete endpoint, they must modify their integration to use the new endpoint.

Added the following endpoints:

- POST /api/Customer2
- PUT /api/Customer
- POST /api/CustomerBill2
- PUT /api/CustomerBill
- POST /api/CustomerSite2
- PUT /api/CustomerSite
- POST /api/CustomerSystem2
- PUT /api/CustomerSystem

Marked the following endpoints as obsolete:

- POST /api/app/Customers
- POST /api/Customer
- PUT /api/Customer/{id}
- PATCH /api/Customer/{id}

- POST /api/CustomerBill
- PUT /api/CustomerBill/{id}
- POST /api/CustomerSite
- PUT /api/CustomerSite/{id}
- POST /api/CustomerSystem
- PUT /api/CustomerSystem/{id}

Added full documentation to the model properties of the new create and update models associated with the new endpoints. This is seen in swagger.

Changed POST /api/CustomerSystemSwap to not allow that on a system that is integrated with an alarm monitoring system.

Moved several property validations to their respective service so the validation for a given property can be called from anywhere without being duplicated. Added PropertyValidation enum and several formatting methods for consistent reporting of validation errors.

Added IntegrationId, IntegrationDescription, and CSSystemLabel properties to AlarmCompany model. Need integration information for GET /api/CSAlarmCompany so an external user can determine the proper alarm company to supply for a customer system.

Added AutoCreateCustomerNumber, AutoCreateInvoiceNumber, AutoCreateJobNumber, CustomerGroupSecurity, DfltCustomerGroupId, RequireSystemAccount, and RequireUniqueSystemAccount to ARSetup model (GET /api/setup/AR). The test suite has to know this information. But also useful information for external users as well.

Added endpoint DELETE /api/Customer2/TestAPI/{customerId} so the customers added by the test suite can also be deleted by the test suite. This is hidden from public view and is only for internal use! The API does not allow deletion of customers, so I wanted a way for the testing to be able to remove added customers so the database does not get cluttered with test data. To keep the test data in the database, the test suite can be run up to the point where the deletes start. As some of the tests fill string fields to their max capacity, the test data could be useful for testing client front ends, etc. The SO client can then be used to delete the customers if so desired.

Corrected CollectionStatus view model which had the Inactive boolean property tagged as a string, which caused the model validation to throw an error every time POST or PUT /api/CollectionStatus was hit, which resulted in a 500 error to the caller. Obviously no customers have ever tried that (or at least have not complained about it).

ServiceLevel view model had same problem as CollectionStatus view model and was corrected.

ContractForm view model had same problem as CollectionStatus view model and was corrected.

Warranty view model had same problem as CollectionStatus view model and was corrected.