

Syncing Your System With Rider Rewards

Last Modified on 09/04/2018 12:02 pm CDT

UNDERSTANDING THE 6 TYPES OF DATA WE NEED

If your dealer system wishes to support the Rider Rewards customer loyalty platform you must upload the following data files to activate the entire rider rewards platform. Some files are optional however will call the Rider Rewards platform to operate in a limited fashion.

- **THE COMPLETE CUSTOMER LIST - REQUIRED - FULL UPLOAD**

** THIS IS FULL NIGHTLY REFRESH OF ALL RECORDS*

This list contains all customers within the dealer system and allows us to link sales to reward numbers and phone numbers, send emails, text notifications and more. This includes ALL customers not just reward customers. Typically you need to refresh this entire list every night unlike the sale transactions which are just for a date range, this data is for all customer records.

- **FEED FILE TYPE C - Customer List**

- **DAILY SALES TRANSACTIONS FROM SOLD ITEM - REQUIRED - PARTIAL UPLOAD**

** THIS IS PARTIAL NIGHTLY UPLOAD BASED ON A DATE RANGE*

We have 3 sales logging APIs used for Part/Accessories Sales, Vehicle Sales, Service Sales

- **FEED FILE TYPE P - Part Sales**
- **FEED FILE TYPE V - Vehicle Sales**
- **FEED FILE TYPE S - Service Sales**

- **COMPLETE VEHICLE / BIKE INVENTORY LIST - NOT REQUIRED BUT RECOMMENDED**

** DEALERS WOULD PROBABLY NOTICE IF YOU SKIPPED*

This data is used in several features such as email alerts, report screens and lead generation tools. For example if "a customer of a dealership" was searching for select models on their rewards portal they can locate the dealer inventory or create a "favorite" in their "garage". Where dealership sales manager might enable a notification automatic marketing alert which sends them a email alert when a particular stock unit type is back in inventory or found at their sister store location. If you don't upload bike inventory the button says "Your Dealer System Does Not Support This Feature,". So a dealership would probably notice your not syncing bike inventory because they often will turn on and use all the digital marketing tools. . You don't need it for the rewards sync to work but it's used by many key marketing modules so disabling it could be viewed negatively by the dealership if you don't upload bike inventory.

- **FEED FILE TYPE B - Vehicle Bike Inventory**

- **COMPLETE PART INVENTORY LIST - NOT REQUIRED**

** NO ONE WOULD PROBABLY NOTICE IF YOU SKIPPED IT - SKIP IT IF YOU WANT TO CUT SCOPE*

Similar to the vehicle inventory, by not uploading you'll disable a feature. However unlike the bike inventory no one will probably notice. The reason is because this is used just for 1 feature called "wish list". Customers can create "wish lists" on their portal but if there is no part inventory then the entire wish list module is hidden. So no one would probably notice, most dealer systems will do this module last or skip it if they are looking for ways to cut scope in the integration project this is a good one to remove if needed.

- **FEED FILE TYPE I - Part Inventory**
-

HOW TO UPLOAD USING OUR 4 STEP BATCH METHOD

UPLOADING YOUR DATA USING OUR RESTFUL API

We have found that typically when uploading a file doing a single POST with JSON can be problematic when dealing with large amounts of data can be very slow due to many factors from file size time timing of the upload. After managing millions of records for many years we have determined that the most effective way to "upload data" to our system is doing it in "batches" of 1000 records at a time then to import all those batches at once as a single "transaction". *When uploading each one of the feed types listed above will need to be done in a single 4 step batch transaction as described below.*

AUTHENTICATION - DO THIS FIRST

- YOU NEED 3 BITS OF DATA WHEN AUTHENTICATING TO OUR API.
 - PartnerAPIKey = This is the ID assigned to the partner and authorizes them select certain dealers.
 - RRStoreID = This is the unique store id store give out 3rd party data access.
 - RRStoreAPIKey = This is the unique store id store give out 3rd party data access.

Always look up these RRStoreID and RRStoreAPIKey using the "GetMyDealerList " API

NEVER STORE THEM - BECAUSE THEY CHANGE

The reason you want to look them up via this API instead of storing them in a config file or setting is because the RR values change monthly!

FOLLOW OUR 4 STEP UPLOAD

PROCESS...

• STEP 1 - CALL THE **READY FEED UPLOAD FUNCTION** **POSTED ONCE**

FIRST PURPOSE of this is for us to tell YOU the next range of records to upload.

- For "sale transactions" these are typically sale records within a date range.
- For "Customers/Inventory" these are complete lists of data that will not have a date range.

SECOND PURPOSE of this is for you to get a full record count of what you're about to upload and

• STEP 2 - CALL THE **BEGIN TRANSACTION** **POSTED ONCE**

The purpose of this is to TELL US how many records you're about to upload, it also tells us what feed type you're about to upload (C,S,V,I,P,B) and the date range you're about to upload. This allows us to do a final check when you end the transaction to make sure all the records are in fact uploaded as planned. As we will use the TOTAL RECORD you told us to check to make sure you uploaded everything right. ** The idea with this is we are going to give you a unique ID which you'll use to tag each block of 1000 records you're going to upload in a "batch/transaction". You'll use this same transaction ID over and over again in Step 3 until all the rows have been uploaded.*

• STEP 3 - SYNC/UPLOAD RECORDS **POSTED MULTIPLE TIMES**

The purpose of this is to actually upload the records to us, each post we support up to 1000 records in the body of the post. Please see an example of the JSON that will be sent in the body in each URL below. Please upload each batch of 1000 one at a time in sequence vs trying to do them all at once. Then when ya get to step 4 we'll merge them all together in one upload on the back end.

ONLY PICK 1 FEED TYPE BELOW PER TRANSACTION DO NOT MIX FEED TYPES

Uploading New List Each Night - REMOVED AND ADDED

Meaning when this transaction is imported we delete EVERY record and re-import EVERY record.

Each night we're basically just refreshing the full list each night.

- Feed **Type "C"** - Full Customer List - [UploadCustomerProfile](#)
- Feed **Type "B"** - Full Bike Inventory - [UploadVehicleInventory](#)
- Feed **Type "I"** - Full Part Inventory - [UploadPartInventory](#)

Adding New Records From Last Upload Date Range - JUST ADDED

This transaction you'll use the date range returned in STEP 1 to fetch all the records within that date range.

- Feed **Type "P"** - Part / Acc Sales Since Date - [LogPartDate\(startdate,endate\)](#)
- Feed **Type "S"** - Service Sales Since Date - [LogServiceDate\(startdate,endate\)](#)
- Feed **Type "V"** - Vehicle Sales Since Date - [LogVehicleDate\(startdate,endate\)](#)

Our system will reject any posts which the body contains more than 1000 records in it and the entire transaction will be flagged. Instead you should be doing multiple posts of blocks of 1000 using the same transaction ID, then commit it. The only post that should

have less than 1000 rows in it is the last post before the entire thing is committed using the "end transaction" function.

" Also, remember these are FEED TYPE specific, there fore you should not be trying to upload multiple feeds using the same TransID this will cause problems. Instead you want to do them one feed type at a time (you can run multiple stores at the same time but only 1 feed type per store at a time). This helps control server load and often does not effect speed."

Your account could be automatically THROTTLED if you try to post multiple posts with less than a 1000 records within the same Transaction ID in the LogPartSales, LogVehicleSales, LogServiceSales, UploadCustomers, UploadVehicleInventory, UploadPartInventory.

For example; if you try to reduce the amount of records in the body to say 1 per body post then post to us a bunch of times within the same transaction, our system will notice this pattern and flag your account. and begin to throttle your API calls.

• **STEP 4 - END TRANSACTION POSTED ONCE**

AS the final step in the transaction once your done uploading you'll call this passing into the transactionID. This tells us your done uploading and we do record check to make sure everything looks good. What we'll do is get a count of all the records you just inserted in that transaction and compare it against the total you said in step 2 as "spot check" to make sure everything jives. Assuming the counts are the same and all the other parms are good we'll close out all these records and lock the feed until its done importing.

WE RECOMMEND YOU DO A RETRY ON A TIMEOUT

In most cases things work fine however during peak times our API could time out between the hours of 1AM and 4AM. The best times to "upload" is actually during the day

between 6AM to 8AM as this is the time that the system is not used as much. We mention this because previously we did not have "Timeout Retry" and noticed that we would have maybe 1 dealer a month who did not update because of time outs. As soon as we added retry code to the upload logic we never had another random failure, therefore we would like to pass the tip to you

WE RECOMMEND YOU CAN ADJUST THE UPLOAD TIME IN CRON/SCHEDULER

When you're building your "uploaded" application we strongly recommend you make it in such a way that it can be scheduled to run at select times during the day or evening. If you're building a windows console app or unix app you can make it so you can change the schedule using windows scheduler or a cron job. If you're building a windows service we recommend you have some kind of Config File which stores the "start time".

- **IN MOST CASES THE ENTIRE UPLOAD TAKES ABOUT 15 MINUTES PER LOCATION FOR ALL 6 FEED TYPES**
-