



# **Product: DQ Order Manager Release Notes**

Subject: DQ Order Manager v7.1.29

Version: 1.0  
January 20, 2017

Distribution: ODT Customers



## DQ OrderManager v7.1.29

\*\*\* requires db update 20170120 or newer \*\*\*

Changes:

- Added option to 'Expire Orders' job step, to set the date field used for Age comparison. dTimeStamp or ReqDate
- Fixed issue with 'Selected' flag not getting set correct when location changed.
- Changed the way scripts are saved to the database. The previous way didn't allow for new variables.
- Fixed issue with how SMO Server establishes connection – the credentials were not being used.
- Added modified flag and Apply button to forms.

## DQ OrderManager v7.1.26

Changes:

- New option under "Create Order; job step: 'Set invoice num same as order num'
  - When selected, the invoice number field in the order\_tbl will be set to the same value as the generated order number.

## DQ OrderManager v7.1.25

Changes:

- Added option to Move Orders job step – 'Update order req date to match trip date'
  - If selected, the order's ReqDate field will be updated to match whatever the new trip date value is for that order.

## DQ OrderManager v7.1.24

Changes:

- Expire Order:
  - Orders will be searched for based on the following criteria:
    - Trip location – perhaps should be by order location but for now it's the trip location.
    - ORDER status – can either be an exact status or the selected status or below (based on the status display order)
    - Age (days) – based on the order's dTimeStamp (the time that the order record was originally created) and the current day that the job is running on. The query will look for orders whose dTimeStamp is older than the specified age.
  - The found order's status will then be updated to the specified status.



- For Move Order step:
  - The ability to search by selected status or below (based on the status display order) has been added
  - The update status list box has been replaced by a combo box since only one selection can be made anyways.

## **DQ OrderManager v7.1.23**

Changes:

- New option added for Move Orders job step: Update order date to match trip date.
- Tooltips added for Move Orders job step options.

## **DQOrderManager v7.1.22**

\*\*\* requires db script 20150317 or newer \*\*\*

Changes:

- DQOrderGenerator has now been renamed as DQOrderManager.
- New job step has been added – ‘Move Orders’
  - Select criteria to move orders around.
  - Set status of trips and orders after they have been moved.

## **DQOrderGenerator v7.1.21**

Changes:

- The schema for function and stored procedures, whose comments must be removed, is not explicitly set when searching for them. Previously, the schema was not explicitly specified, and so that the default schema would need to be determined

## **DQ Order Generator v7.1.20**

Changes:

- Bug fix. At startup, the code turns on certain db features that must be turned on in order for the order generator to work. One of these features, ‘enabling the broker’, had a syntax error in it that was causing it to fail. It has been corrected.



## DQ Order Generator v7.1.19

\*\*requires db script 20120314 or newer

Changes:

- Bug fix: On order import, if numeric fields duration, weight or volume are blank, they should be saved to the db as NULL, but this conversion is not automatic. It has to be specified in transact sql code, which it is now.
- An enabled column has been added to job steps. An individual job step can now be excluded from a job if needed.
- Grid column sizes are now saved after every form is closed. So, the next time the grid is displayed, it should have the previous column sizes.
- The selected grid row is preserved between grid refreshes. If a certain row is selected and F5 (refresh) is pressed, the grid data will be re-queried from the db and the grid will be repopulated with the latest data and the same row will still be selected.

## DQ Order Generator v7.1.18

Requires db version 20120302 or newer.

Changes:

- Order import routine has been updated:
  - CSV files with quotes and commas in the column can now be handled without erring.
  - Error handling has been improved.
  - Order number generation logic has been improved - %N% can now be anywhere in the order number, not just at the end
  - If order already exists, it will be updated instead of creating a new order number every time – this makes it more like a standard importer.
  - Same thing for trip records, notes and time windows.
- Bug fix: If the File Location textbox was blank and then you clicked on the ellipses (...) button, an unhandled error would be thrown and the form would close. This is now fixed – if the textbox is blank, the root is set as the default location in the file location browser.
- A default status is now required on the setup form for both generated orders and import orders tasks.

## DQ Order Generator v7.1.17

Changes:



- Modified order import query (OI\_SQL\_1) to provide better error handling and rolling back transactions in the case of an error.

## DQ Order Generator v7.1.16

Changes:

- When setting the location of an import file, the File directory local to the SQL Server is now shown. Previously, the local machine directory was being shown which would only work if the app was being run on the server. If it were run from a client then the returned path would be wrong relative to the server.

## DQ Order Generator v7.1.15

Changes:

- The import file step generator option added 6 additional fields to the output file:
  - Order number – if the order number is specified, it will be used. If it already exists, it will still be used, but the PKImport will have a different shipment number appended to the end to make it unique. If the order number field for a particular order is blank, then the order number (or order number format) specified in the setup will be used.
  - Time Window – if the time window specified already exists as a time window in the database, it will be set for the generated order record.
  - Order Type – if the order type specified exists in the db, it will be set to the order. If it does not exist, it will not be created.
  - Duration – the order duration. Can be decimals.
  - Weight – the order weight. The db record is an integer, so if a decimal is passed in, it will be converted to an int. e.g., 123.2 will get saved as 123.
  - Volume – the order volume. Also stored in the db as an integer.
- For importing orders from a CSV file, there is now a place to indicate on which line to start importing records from. Typically, the first line will contain the column names and the data will start on the second line, but in the case where there are no columns, the value could be changed to 1. 2 is the default.
- Notes:
  - The import script does not handle having commas (,) in the data of a particular column very well, even if the data is wrapped in quotes. For example, for Weight, the default export creates



“1,120.03”. The quotes get added because of the comma. The script will choke on this and throw an error. However, if the commas are removed, then and the data looks like this – “1120.03”, the script will not have a problem – the quotes will be removed and the value will be imported.

- All columns must exist in their correct sequence even if they are blank. So if a particular customer never specifies Volume because they only measure by weight, the volume column must still be present in the output file in its correct spot, or else the script will error.

The CSV file format for order import right now is:

CustomerCode,Name,Address,City,State,Zip,Order#,Time Window,Order  
Type,Durration,Weight,Volume

## DQ Order Generator v7.1.14

Changes:

- Modified the Step creation form.
  - Added a Type to the step form that allows for various methods for generating orders.
    - There are now 2 types –the original create order method , now called ‘Create Order’ and the new type added called ‘Import Orders From Customer File’
      - The ‘Import Orders From Customer File’ method looks for a file in a specified directory in a specific format. It creates the job account and customer account if they don’t exist and creates 1 order for each line in the file. The order number for the order is generated based on the specified order number format. To generate unique, incrementing order numbers, use %N%.
        - The format for the CSV customer file is:  
CustomerCode,Name,Address,City,State,Zip
    - File Location – The path to the customer file to be imported – s relative to the Sql server. Can be a UNC path but proper permissions are required to access and delete the file.
    - Delete file after successful import’ – If the file is successfully processed and there were no errors, the file will be deleted.