AssetCloud & InventoryCloud: Improving receipt performance

Scott Leonard - 2022-07-28 - Cloud

July 2022

Wasp is continually improving performance of our Cloud products. We are making improvements to the performance of builtin receipts, but customers will need to make manual changes to any receipts they may have customized to see the same performance improvement.

Based on our experience, the following changes improve performance by many factors.

==========

Instructions:

1. Find a customized receipt based off a stock transaction receipt (all receipts are actually small reports).
2. On-Premise customers only: Look at the SQL Query in the MRT file.
3. Cloud-web customers only: Look at the SQL Query in the Report Designer for the report. **See below for screenshots.**
4. Check if the where clause contains "*some-view-name*.**transaction_batch_no = {BatchNo}**".
5. If it does, change that to

   some-view-name .transaction_id in (Select asset_transaction_id
   from {schema}.[asset_transaction]
   where asset_trans_batch_no = {BatchNo}
   and (asset_transaction.asset_trans_batch_no<>(0)))

OP only: Note the use of < for "<" and > for ">".
Web only: use the < and > characters.

==========

Examples:

For example, based off the "Inventory Move Receipt.mrt".

Find the SQL query and note the where clause. In this example,

<SqlCommand>select * from {schema}.vItemMoveTransaction
where vItemMoveTransaction.**transaction_batch_no = {BatchNo}**</SqlCommand>

This "where" clause should be fine, but SQL's default choice is not efficient. Changing the where clause will provide a significant performance improvement.

where vItemMoveTransaction.transaction_id in (Select asset_transaction_id
from {schema}.[asset_transaction]
where asset_trans_batch_no = {BatchNo}
and (asset_transaction.asset_trans_batch_no<>(0)))

The SQL query would now look like the following:

<SqlCommand>select * from {schema}.vItemMoveTransaction
where vItemMoveTransaction.transaction_id in (Select asset_transaction_id
from {schema}.[asset_transaction]
where asset_trans_batch_no = {BatchNo}
and (asset_transaction.asset_trans_batch_no<>(0)))</SqlCommand>

==========

Another example would be a receipt based on “Inventory Reconcile Receipt.mrt”.

```
<SqlCommand>select * from {schema}.vItemReconcileTransaction
where vItemReconcileTransaction.transaction_batch_no = {BatchNo}</SqlCommand>
```

Would change to:

```
<SqlCommand>select * from {schema}.vItemReconcileTransaction
where vItemReconcileTransaction.transaction_id in (Select asset_transaction_id
from {schema}.[asset_transaction]
where asset_trans_batch_no = {BatchNo}
and (asset_transaction.asset_trans_batch_no&lt;&gt;(0)) )</SqlCommand>
```

==========

Another example is “Inventory Remove Receipt.mrt”:

```
<SqlCommand>select * from {schema}.vItemRemoveTransaction
where vItemRemoveTransaction.transaction_batch_no = {BatchNo}
and vItemRemoveTransaction.asset_trans_other_trans_id != 0</SqlCommand>
```

Becomes:

```
<SqlCommand>select * from {schema}.vItemRemoveTransaction
where vItemRemoveTransaction.transaction_id in (Select asset_transaction_id
from {schema}.[asset_transaction]
where asset_trans_batch_no = {BatchNo}
and (asset_transaction.asset_trans_batch_no&lt;&gt;(0)) )
and vItemRemoveTransaction.asset_trans_other_trans_id != 0</SqlCommand>
```

==========

Another example is “Inventory Check Out Receipt.mrt”:

```
<SqlCommand>select * from {schema}.vItemCheckOutTransaction
where vItemCheckOutTransaction.transaction_batch_no in ({BatchNo})</SqlCommand>
```

Becomes:

```
<SqlCommand>select * from {schema}.vItemCheckOutTransaction
where vItemCheckOutTransaction.transaction_id in (Select asset_transaction_id
from {schema}.[asset_transaction]
where asset_trans_batch_no = {BatchNo}
and (asset_transaction.asset_trans_batch_no&lt;&gt;(0)) )</SqlCommand>
```

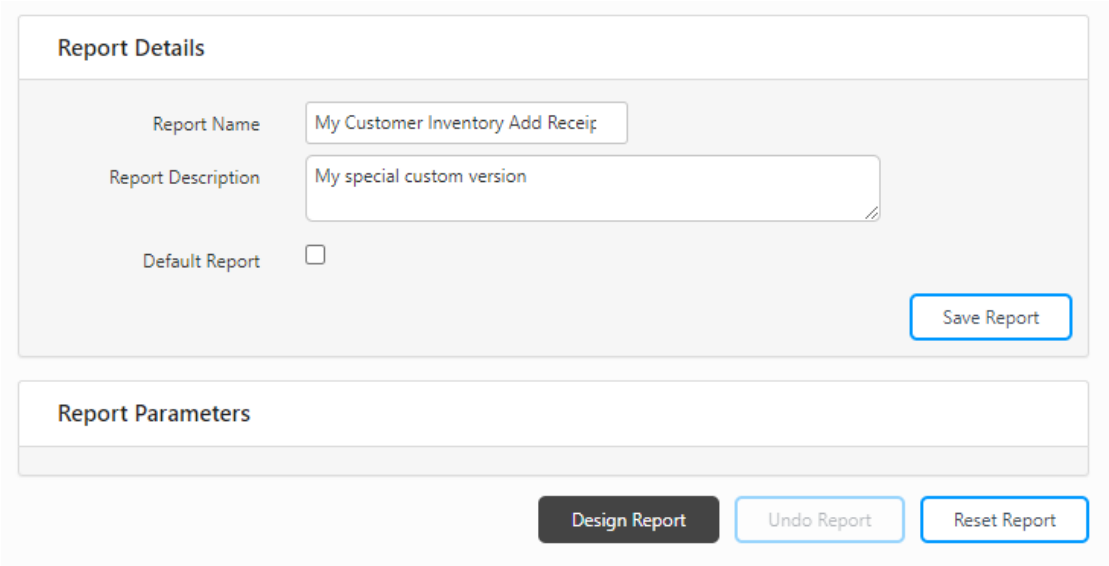
==========

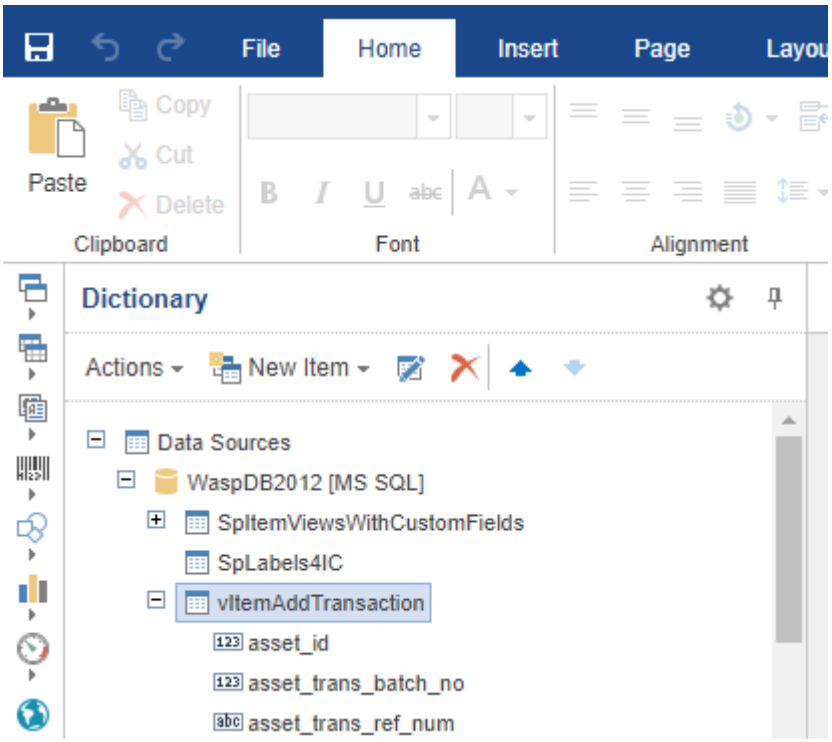
Other improvable receipts include:

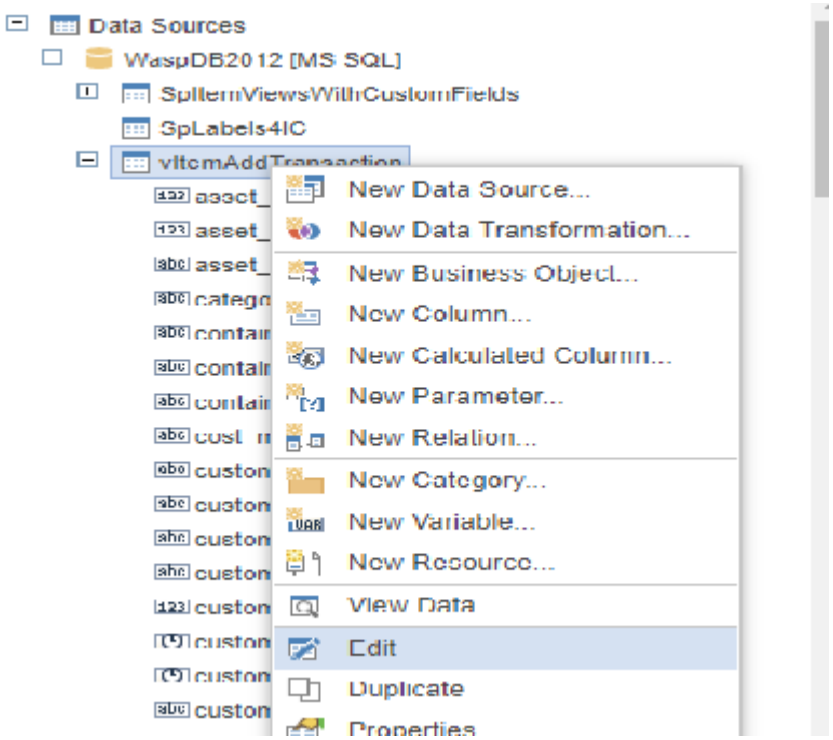
“Add Transaction Label.mrt”
“Inventory Add Receipt.mrt”
“Inventory Adjust Receipt.mrt”
“Inventory Build Receipt.mrt”
“Inventory Check In Receipt.mrt”

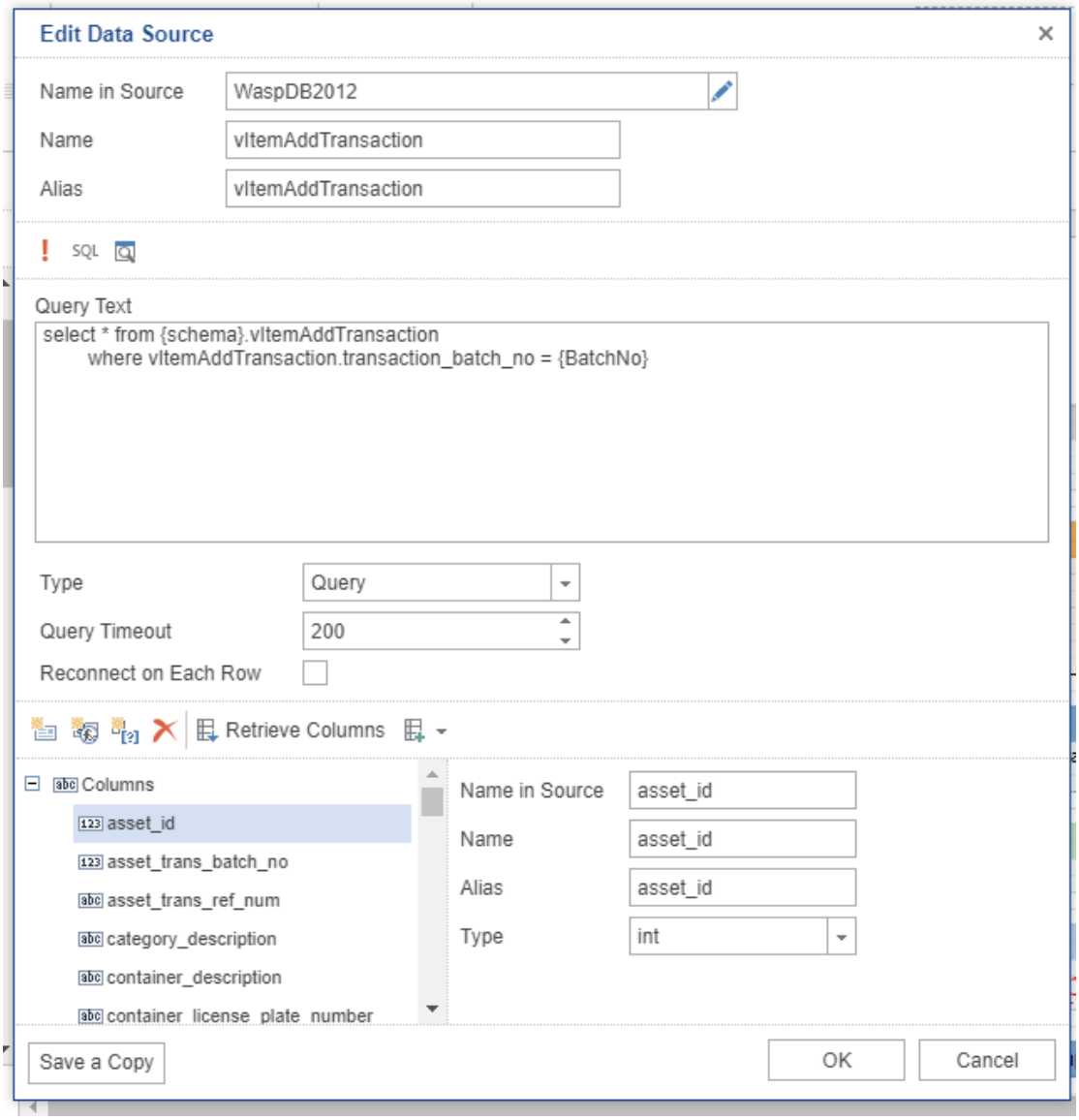
==========

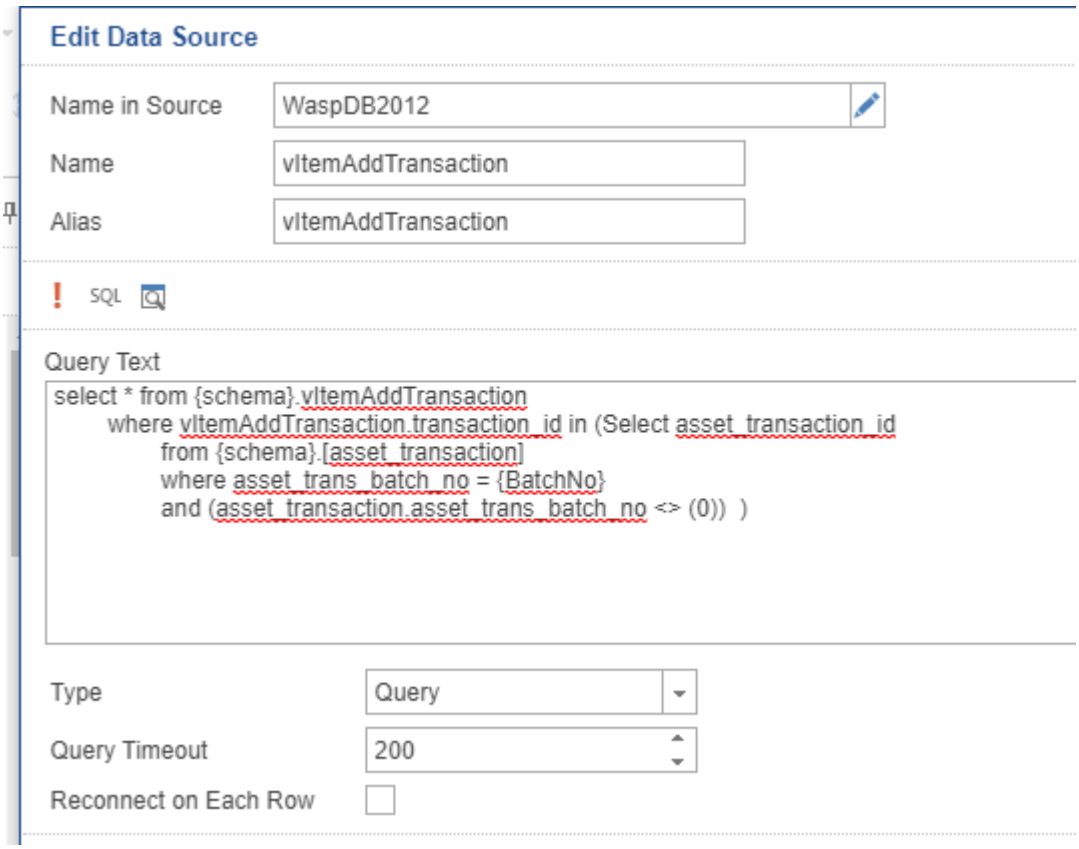
Screenshots for editing receipt reports in the Cloud-Web products:


Report Details
Report Name
My Customer Inventory Add Receip
Report Description
My special custom version
Default Report
Save Report
Report Parameters
Design Report
Undo Report
Reset Report


File
Home
Insert
Page
Layou
Copy
Cut
Delete
Paste
Clipboard
Font
Alignment
Dictionary
Actions
New Item
Data Sources
WaspDB2012 [MS SQL]
SpItemViewsWithCustomFields
SpLabels4IC
vItemAddTransaction
asset_id
asset_trans_batch_no
asset_trans_ref_num


Data Sources
WaspDB2012 [MS SQL]
SpItemViewsWithCustomFields
SpLabels4IC
vItemAddTransaction
New Data Source...
New Data Transformation...
New Business Object...
New Column...
New Calculated Column...
New Parameter...
New Relation...
New Category...
New Variable...
New Resource...
View Data
Edit
Duplicate
Properties


Edit Data Source
Name in Source
WaspDB2012
Name
vItemAddTransaction
Alias
vItemAddTransaction
SQL
Query Text
select * from {schema}.vItemAddTransaction
where vItemAddTransaction.transaction_batch_no = {BatchNo}
Type
Query
Query Timeout
200
Reconnect on Each Row
Retrieve Columns
Columns
asset_id
asset_trans_batch_no
asset_trans_ref_num
category_description
container_description
container_license_plate_number
Name in Source
asset_id
Name
asset_id
Alias
asset_id
Type
int
Save a Copy
OK
Cancel



>>> Note this way does NOT use the < and >

Click the ! to run the query to check syntax.

SQL

Query Text

```
select * from {schema}.vItemAddTransaction
      where vItemAddTransaction.transaction_id in (Select asset_transaction_id
            from {schema}.[asset_transaction]
            where asset_trans_batch_no = {BatchNo}
            and (asset_transaction.asset_trans_batch_no <> (0))  )
```

Type

Query Timeout

Reconnect on Each Row

SQL Expressions

schema	[tbellis]
BatchNo	4611686018429387905

OK Cancel

Query Text

```
select * from {schema}.vItemAddTransaction
      where vItemAddTransaction.transaction_id in (Select asset_transaction_id
            from {schema}.[asset_transaction]
            where asset_trans_batch_no = {BatchNo}
            and (asset_transaction.asset_trans_batch_no <> (0))  )
```

Type

Query Timeout

Reconnect on Each Row

Designer

SQL statement executed successfully

OK