

# ESS Transaction Tables - Schema and Relationship Guide

Copyright 2014, R. James Holton, All rights reserved  
(January 26, 2015)

## Introduction

The Expense Submittal System (ESS) stores information about an expense report in several tables. Although ESS follows a relational model, some tables have been denormalized to make the creation of informational reports easier.

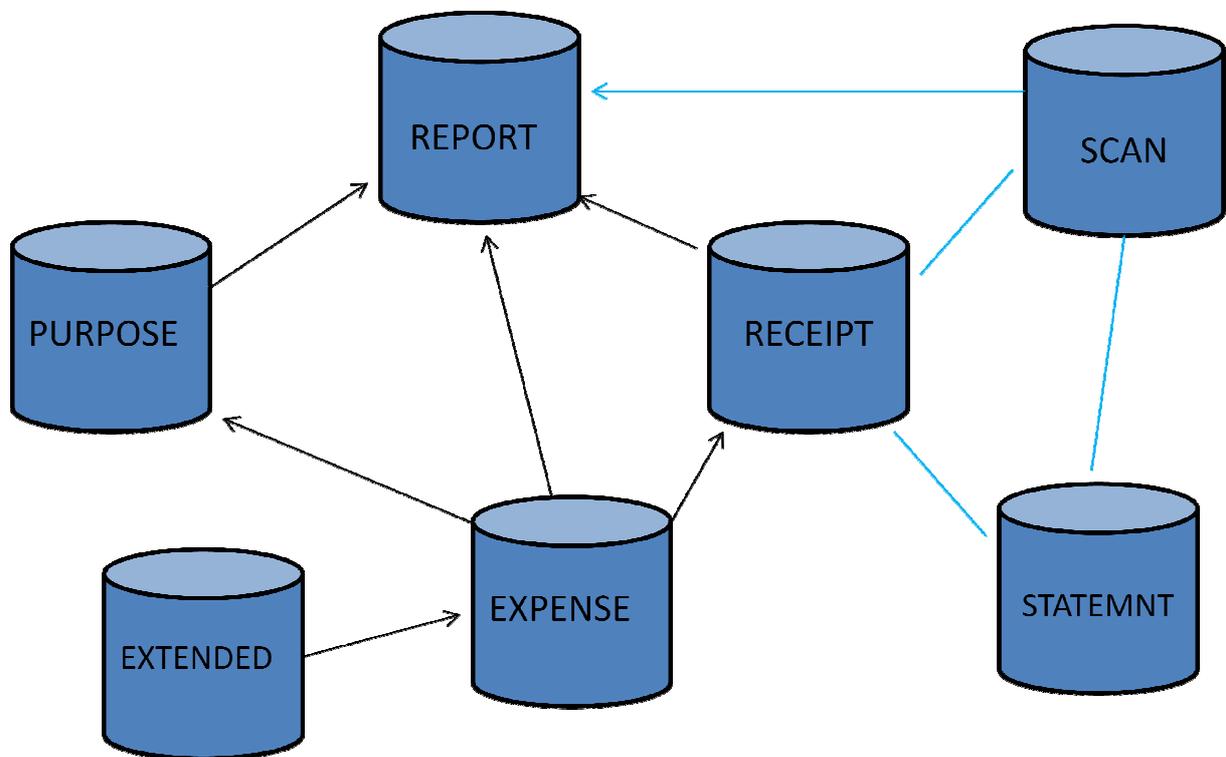


Figure 1 - Transaction Table Relationships

In figure 1 above, the tables and their relationships are visualized. The dark line indicates that a required relationship exist. For example, an item in the RECEIPT table cannot exist without a corresponding item in the REPORT table. The arrow points in the direction of the many towards one direction. If there is a single line it indicates a one to one relationship. The light blue line indicates that relationships are not required.

In the table descriptions below, a row item in red indicates the item is part of the unique identifier for the table. A light blue highlight means the row points to a record in another table.

## EXPENSE Table

The expense table holds information regarding the classification of each expense. This information drives the general ledger feed.

1. **EXPENSE** *varchar(20)* - expense type which is normally assigned by the user during report creation.
2. **AMOUNT** *numeric(12,2)* - expense amount in the report currency.
3. **VOUCHER** *varchar(8)* - voucher from REPORT table row. Ties this expense to a report record. Sometimes referred to as the transaction number
4. **RECEIPT** *varchar(8)* - Receipt number which ties this expense entry to a RECEIPT row. A receipt can have multiple expense lines.
5. **PURPOSE** *varchar(8)* - ties the expense to specific report purpose or reason for incurring the expense.
6. **LINEITEM** *varchar(8)* - unique reference within a report voucher. The VOUCHER plus LINEITEM define a unique entry within the EXPENSE table.,
7. **DEPART** *varchar(30)* - DEPARTMENT row this expense is assigned to
8. **ACCOUNT** *varchar(30)* - site specific, holds information such as client or project number if collected at the expense line level
9. **UP\_DATE** *date* - date that the expense was reimbursed to employee.
10. **UP\_STAT** *varchar(2)* - report status, usually 'H4' if reimbursed.
11. **BILLTYPE** *char(1)* - *reserved for billing operations.*
12. **BL\_DATE** *date* - *reserved for billing operations.*
13. **BL\_STAT** *varchar(2)* - *reserved for billing operations.*
14. **COMMENT** *varchar(60)* - description from user input.
15. **APPROVAL** *varchar(16)* - *currently not used, reserved.*
16. **CRE\_DATE** *date* - date the expense entry was created.
17. **EXP\_DATE** *date* - date the expense was incurred.,
18. **REC\_AMT** *numeric(12,2)* - expense amount in the currency that the expense was incurred. If a mileage type of expense this will be the miles entered.
19. **XRATE** *numeric(12,4)* - exchange rate or mileage reimbursement rate
20. **UNITS** *varchar(20)* - foreign currency or distance units
21. **XSOURCE** *varchar(60)* - Source if the exchange rate or the vehicle/method classification for mileage if applicable to the site
22. **XDATE** *date* - date of the exchange rate info.
23. **COMPANY** *varchar(8)* - *currently not used, reserved.*
24. **PERSPLIT** *numeric(2,0)* - number of persons in the attendee list for expenses such as entertain that require attendees to be entered.
25. **REIMB** *char(1)* - *currently not used, reserved.*

## EXTENDED Table

The Extended table is designed for storage of information that will extend the reason an expense was incurred. It's primary use is to record attendees at events for tax purposes.

1. **VOUCHER** *varchar(8)* - report that this record is tied to.
2. **PURPOSE** *varchar(8)* - *currently not used, reserved.*
3. **LINEITEM** *varchar(8)* - expense line that this record it tied to.
4. **NAME** *varchar(36)* - name of person being entertained
5. **REMARKS** *varchar(20)* - discussion that took place
6. **CLIENT** *varchar(40)* - client name
7. **LOCATION** *varchar(40)* - location of the event.
8. **COMMENT** *varchar(80)* - discussion that took place during event
9. **EXTRAITEM** *varchar(8)* - uniquely identifies an extended item within a report voucher.

## TABLE PURPOSE Table

The PURPOSE table describes the reason for a group of expenses. Normally there is one row per report, but ESS does support multiple purposes.

1. **PURPOSE** *varchar(8)* - uniquely identifies a purpose within a report voucher.
2. **VOUCHER** *varchar(8)* - report voucher that this record is tied to.
3. **CLIENT** *varchar(40)* - normally a client name if present.
4. **LOCATION** *varchar(40)* - normally a location if present.
5. **COMMENT** *varchar(250)* - reason for the expenses.
6. **PROJECT** *varchar(24)* - project, or internal reference such as an order number. It optionally points to the PROJECT table.
7. **BEG\_DATE** *date* - start of the period covered by the associated expenses.
8. **END\_DATE** *date* - end of the the period covered by the associated expenses.
9. **DAYS** *numeric(3,0)* - number of days covered by this purpose. Used in the guideline check.
10. **NIGHTS** *numeric(3,0)* - number of nights covered by this purpose. Used in the guideline check. Normally the numbers of DAYS - 1.
11. **CLIENTNO** *varchar(30)* - ,
12. **FORCED** *char(1)* - *currently not used, reserved.*
13. **STEPNO** *varchar(12)* - intended as a subset of a project. Sites often use this as a general data field.
14. **EXTENDED** *char(1)* - *currently not used, reserved.*
15. **TIMESLIP** *char(1)* - *currently not used, reserved.*

## RECEIPT Table

The RECEIPT table contains information on each report entry, including non-expense items such as advances, returns, etc.

1. RCPTTYPE *varchar(2)* - type of receipt entry.
  - a. 1 - receipt that can have more than one expense line.
  - b. 3 - receipt that has only one expense line.
  - c. 4 - mileage with one expense line.
  - d. 6 - advance, no associated expense line.
  - e. 12 - returned funds, no associated expense line.
2. CHARGE *varchar(20)* - payment method.
3. VOUCHER *varchar(8)* - report voucher that this record is tied to..
4. RECEIPT *varchar(8)* - unique number identifying this receipt within the voucher.
5. REFER *varchar(30)* - reference number. In mileage this is the beginning odometer reading for sites that enter odometer readings for each entry. Some sites use this as a link to the PROJECT table.
6. REC\_DATE *date* - date of the payment.
7. REC\_AMT *numeric(12,2)* - amount in original currency. For mileage this is the miles or kilometers.
8. XRATE *numeric(12,4)* - exchange rate or mileage reimbursement per unit.
9. UNITS *varchar(20)* - currency or distance measure.
10. XSOURCE *varchar(60)* - source of exchange rate or mileage calculation type.
11. XDATE *date* - date the exchange rate is from.
12. AMOUNT *numeric(12,2)* - receipt amount in the report currency
13. ACCOUNT *varchar(30)* - misc info. Sites with the Google mileage calculator use this for storing the beginning postal code.
14. OFFSET *varchar(30)* - misc info. Sites with the Google mileage calculator use this for storing the ending postal code. Sites that capture odometer readings for each mileage entry use this for the ending odometer reading.
15. REIMB *char(1)* - if true this receipt is reimbursable to the employee.
16. FORCED *char(1)* - *currently not used, reserved.*
17. OK\_DATE *date* - date receipt was reconciled to items in the statement file.
18. OK\_STAT *varchar(2)* - status in the reconciliation process.
  - a. XX - user created entry from pre-populated.
  - b. OK - item has been fully processed/reconciled.
19. OK\_REF *varchar(8)* - link to an item in the STATEMNT (pre-populated) table that contains the credit card and/or ETG (email/text gateway) information.
20. CRE\_DATE *date* - date this entry was created.
21. XCHECK *char(1)* - true if the receipt has been accounted for by the auditors.
22. UP\_DATE *date* - date item upload to paid status.
23. UP\_STAT *varchar(2)* - upload status (e.g., H4) is paid.
24. MERCHANT *varchar(30)* - name of merchant
25. SUBTYPE *varchar(2)* - If present this field further classifies the receipt:
  - a. 1 - Hotel receipt
  - b. 2 - mileage entry containing odometer information
26. BEG\_DATE *date* - Used in a hotel stay to mark the beginning of a stay. The REC\_DATE marks the end of a stay.
27. LOCATION *varchar(30)* - expense location for tax purposes if applied by user.
28. SCAN\_REF *varchar(8)* - link to SCAN table for receipt image if entry created through the ETG process.

## REPORT table

The report table is the header record for an expense report and contains information that allows expenses to be classified, an audit trail to be maintained, and fleet vehicle information. Here are a few key points when using the REPORT table information:

- There are two separate tracking reference numbers. VOUCHER, sometimes referred to as the transaction number, always refers to a unique report row. PVOUCHER, the user's number, refers to a expense report in concept and can be duplicated across several report rows depending on the work flow. For instance if a person submits a report and the report is reject and resubmitted there will be two report rows in the REPORT table. One will be in a submitted status and the other will have a rejected status. This is done for audit trail purposes. Both rows will have the same PVOUCHER, but will have unique VOUCHER numbers.
- The meaning of the sign-off elements is dependant upon the workflow being used and are defined in the *Status.xml* file. Some of the definitions giving below are for a "typical" site and may not be the actual definition of these columns in your table.

The REPORT table also has some "extra" columns that may be used by your site.

1. RP\_STAT *varchar(2)* - report status corresponds to the *translation* element value in the *Status.xml* file for a status.
2. PERS\_NUM *varchar(12)* - personnel number from the USER table of the person whom the report is for.
3. CRE\_DATE *date* - date report was created. The create date and time are based on the client date and time. All other dates and time are based on the server date and time.
4. CRE\_TIME *varchar(8)* - time the report was created.
5. VOUCHER *varchar(8)* - uniquely identifies this instance of report submission. Sometimes referred to a the transaction number.
6. PVOUCHER *varchar(8)* - identifies the report. Corresponds to the REGISTER table.
7. SUB\_DATE *date* - date the report was originally submitted for approval/payment.
8. SUB\_TIME *varchar(8)* - time originally submitted for approval/payment.
9. CUR\_DATE *date* - date of submission of this instance.
10. CUR\_TIME *varchar(8)* - time of submission of this instance.
11. ACT\_DATE *date* - date report was accepted. Should be the same as CUR\_DATE.
12. ACT\_TIME *varchar(8)* - time of acceptance. Should be the same as CUR\_TIME.
13. PAYMENT *char(1)* - *currently not used, reserved.*
14. NAME *varchar(36)* - name of reporter.
15. LOCATION *varchar(30)* - reporter's location from USER table.
16. PHONE *varchar(24)* - reporter's phone number from USER table.
17. DEPART *varchar(30)* - department report is charged to.
18. COMPANY *varchar(8)* - company report is charged to.
19. RE\_AMT *numeric(12,2)* - amount reporter will be reimbursed.
20. AD\_AMT *numeric(12,2)* - advance covered amount.
21. RC\_AMT *numeric(12,2)* - gross amount of report.
22. SIGN1 *varchar(12)* - signoff personnel number.
23. SIGN2 *varchar(12)* - signoff personnel number.
24. SIGN3 *varchar(12)* - auditor who approved report

25. XCHECK *char(1)* - currently not used, reserved.
26. XCHARGE *char(1)* - currently not used, reserved.
27. AUD\_DATE *date* - date of approval by auditor.
28. FORCED *char(1)* - currently not used, reserved.
29. UP\_DATE *date* - date report released for payment. Known as the "Batch" date.
30. UP\_STAT *char(2)* - PD if report successfully released for payment.
31. VEHICLE *varchar(20)* - Fleet vehicle number.
32. MONTH *varchar(5)* - Month for reporting purposes.
33. SERVDATE *date* - Date fleet vehicle put in service.
34. START *numeric(7,0)* - Starting odometer reading.
35. FINISH *numeric(7,0)* - Ending odometer reading.
36. BUSINESS *numeric(6,0)* - Business mileage.
37. CLERK *varchar(12)* - Personnel number of person who entered the report
38. SG1\_DATE *date* - date of SIGN1
39. SG2\_DATE *date* - date of SIGN2
40. EXTRA1 *varchar(20)* - reserved.
41. EXTRA2 *varchar(12)* - reserved.
42. UNITS *varchar(20)* - mileage units (??).
43. MANAGER *varchar(12)* - personnel number of the next approver. Otherwise known as the "routed to" field.
44. COMMENT *varchar(80)* - Brief description of report contents entered by the reporter or input admin.
45. EXTRA3 *varchar(8)* - reserved.
46. OLD\_STAT *varchar(2)* - previous
47. PHY\_REC *numeric(4,0)* - physical receipts received by auditors.
48. RECEIVED *date* - as of date accounting received receipts. Used by locations that process physical receipts.
49. EXTRA4 *varchar(12)* - reserved.
50. SIGN4 *varchar(12)* - person who marked report as receipted.
51. RC\_DATE *date* - date report marked as receipted
52. ADMIN1 *varchar(12)* - signoff personnel number.
53. AD1\_DATE *date* - date of ADMIN1 signoff.
54. ADMIN2 *varchar(12)* - signoff personnel number.
55. AD2\_DATE *date* - date of ADMIN2 signoff.
56. WEEKEND *date* - Saturday prior to first submit date.
57. CURRENCY *varchar(20)* - Currency the report is in.
58. HISTORY *varchar(8)* - points to a previous instance of report in the event of audit currency conversion or purchase request process.

## SCAN Table

The purpose of the SCAN table is to hold an image of a physical receipt.

1. **SCAN\_REF** *varchar(8)* - unique identifier,
2. **PERS\_NUM** *varchar(12)* - personnel number from USER table,
3. **PVOUCHER** *varchar(8)* - report number scan is associated with. It can be blank if scan has not been applied to a report yet.
4. **RECEIPT** *varchar(8)* - link into RECEIPT table. Use the SCAN\_REF to tie to RECEIPT table.
5. **FILE\_TYPE** *varchar(6)* - image file type: jpg, gif, bmp, png, pdf.
6. **UP\_DATE** *date* - date scan downloaded by the ETG or uploaded from browser.
7. **UP\_TIME** *varchar(8)* - time scan downloaded by the ETG or uploaded from browser.
8. **CLERK** *varchar(12)* - personnel number of individual who performed upload.
9. **STATUS** *varchar(8)* -
  - a. INCOMING - ETG scan has not been applied to a report yet.
  - b. APPLIED - ETG scan has been applied to a report.
  - c. RECEIVED - scan was uploaded to a report with the browser.
10. **XREF** *varchar(8)* - link to STATEMNT (prepopulated) table.
11. **COMMENT** *varchar(60)* - text associated with image.
12. **ADDRESS** *varchar(60)* - email/phone of sender if sent to ETG.
13. **FILE\_SIZE** *bigint(20)* - size of uncompressed image
14. **IMAGE\_WIDTH** *int* - image width
15. **IMAGE\_HEIGHT** *int* - image height
16. **FILE\_CONTENT** *longblob* - image

## STATEMNT Table

The statement table was originally specified to store incoming credit card charges. It now holds all items that originate from outside ESS report entry screen and allows these items to be selected and added to the report with minimum effort.

1. **CHARGE** varchar(10) - payment type.
2. **CARD** varchar(30) - credit card used. For entries coming through the ETG, this is either a personnel number of the employee sending in the entry.
3. **EMPLOYEE** varchar(25) - name of the credit card holder or employee name derived from the email/phone.
4. **SIC** varchar(20) - Standard Industry Code (SIC) or expense classification derived from email/text message. Defaults to MISC. Is used to search `prepopulatedmap.xml` to create an expense type.
5. **REFERENCE** varchar(23) - Reference number from card vendor or generated by ETG.
6. **AMOUNT** numeric(12,2) - Amount in system currency.
7. **TDATE** date - transaction date.
8. **SDATE** date - statement date.
9. **INFO** varchar(30) - vendor provided description or text from ETG.
10. **STATUS** varchar(2) - processing status of item:
  - a. NW - new item not applied.
  - b. XX - item select and placed on report.
  - c. OK - item fully reconciled as report is marked paid or manually reconciled.
11. **RDATE** date - reconcilment date.
12. **XREF** varchar(8), # XREF - not null unique, consider auto increment.
13. **PDATE** date - *reserved*.
14. **REC\_AMT** numeric(12,2) - amount in foreign currency.
15. **XRATE** numeric(12,4) - exchange rate.
16. **UNITS** varchar(20) - transaction currency.
17. **SCAN\_REF** varchar(8) - link to SCAN table.

###