

Relational Efficiencies: Part II

Renee Teatro
Information Builders
F.I.R.E. Conference
October 2008

Relational Efficiencies

Agenda

- Optimization Overview
- Aggregation Scenarios
- Selection Scenarios

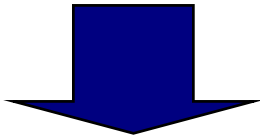
Relational Efficiencies: Optimization Overview

Optimization Hierarchy

JOIN



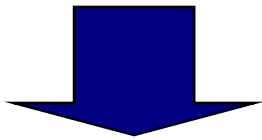
If the JOIN optimization fails,
so will the SORT and
AGGREGATION.



SORT



If the SORT optimization fails,
so will AGGREGATION.



AGGREGATION



AGGREGATION will occur only if
JOIN and SORT operations were
optimized (and AGGREGATION
requested).

Relational Efficiencies: Optimization Overview

Data Adapter TRACE Facility

- SET TRACEON=component//destination
- Component
 - SQLDI - FSTRACE - All Adapter-RDBMS activity
 - SQLAGGR - FSTRACE3 - Optimization messages
 - STMTRACE - FSTRACE4 - SQL only
 - SQLCALL - commands and data exchange between the physical and the logical layers of the data adapter
- Destination
 - FSTRACE - allocation for the ddname of FSTRACE
 - CLIENT - displays client session to the screen

NOTE: Make sure to SET TRACEUSER=OFF afterwards

Relational Efficiencies: Optimization Overview

Data Adapter TRACE Facility

```
SET TRACEOFF=ALL
SET TRACEUSER=CLIENT
SET TRACEON=SQLAGGR//CLIENT
SET TRACEON=STMTRACE//CLIENT
SET TRACESTAMP=OFF
SET XRETRIEVAL=OFF
TABLE FILE ...
```

SET XRETRIEVAL=[ON | OFF]

- **ON** – the data adapter sends the request to the RDBMS and it processes the request
- **OFF** – the data adapter attempts to optimize the request and trace output is generated but no RDBMS processing is done

Relational Efficiencies: Aggregation Scenarios

Efficient Aggregation

Aggregation translation is important

- RDBMS aggregation is more efficient: indices
- An aggregated answer set reduces FOCUS-to-RDBMS communication
- A smaller answer set reduces FOCUS local processing

FOCUS	SQL
SUM ..., WRITE ... BY field	SELECT SUM(...) GROUP BY column ORDER BY column
SUM., CNT., MIN., MAX., AVE.	SUM(...), COUNT(*), MIN(...), MAX(...), AVG(...)

Relational Efficiencies: Aggregation Scenarios

Translatable Aggregation

- Verbs: SUM, COUNT, WRITE
- Direct operators: MIN., MAX., AVE.
- Aggregating DEFINEd fields:
 - Constant DEFINEd fields translated with CNT.

The following defined expressions can be translated

Type of expressions	Expression components	Examples
Arithmetic Valued (Expressions that return a single number)	<ul style="list-style-type: none">• Real fields of datatype I, P, D, or F• Numeric constants• Arithmetic operators (+, -, *, /)	<pre>DEFINE FILE ORAEMP NEW_SAL/D12.2=(CSAL * 0.10) + CSAL ; END</pre>
Character String Valued (Expressions that return a character string)	<ul style="list-style-type: none">• Real fields of datatype A• String constants• Concatenation operator ()• EDIT of alphanumeric fields	<pre>DEFINE FILE ORAEMP NAME/A18=EDIT(FN, '9.\$') LN; END</pre>

Relational Efficiencies: Aggregation Scenarios

Non-Translatable Aggregation

Aggregation is not translated to SQL and optimization is automatically disabled when:

- Optimization was set off by user
- JOINS were not passed to RDBMS (and consequently optimization was disabled by the interface)
- FOCUS sort phrase is not translated
- Some screening conditions not passed to RDBMS
- Some non-SQL operators are used
- Multi-verb requests
- COUNT with MISSING=ON

NOTE

If the verbs PRINT or LIST are used, no aggregation is requested and FSTRACE3 returns the following message:

```
(FOC2590) AGGREGATION NOT DONE FOR THE FOLLOWING REASON:  
(FOC2594) AGGREGATION IS NOT APPLICABLE TO THE VERB USED
```

Relational Efficiencies: Aggregation Scenarios

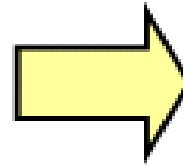
Key Points for Aggregation Translation

- All IF or WHERE statements in the request are translated to SQL WHERE clause.
- FOCUS / WebFOCUS must generate a single SQL statement
- Only these aggregation commands or operators are used: SUM, COUNT, WRITE, MIN., MAX., AVE., SUM., DST., CNT. or CNT.DST.

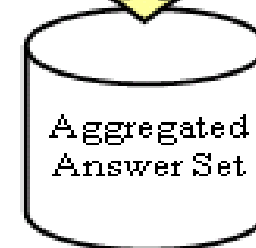
Relational Efficiencies: Aggregation Scenarios

Aggregation Management by the RDBMS

```
TABLE FILE EMPINFO  
SUM ED_HRS  
BY DEPARTMENT  
BY LAST_NAME  
BY FIRST_NAME  
END
```



```
SELECT SUM(ED_HRS),  
DEPARTMENT, LAST_NAME,  
FIRST_NAME FROM EMPINFO T1  
ORDER BY DEPARTMENT,  
LAST_NAME, FIRST_NAME  
GROUP BY DEPARTMENT,  
LAST_NAME, FIRSTNAME;
```



Relational Efficiencies: Aggregation Scenarios

SUM on ALPHANUMERIC Field

```
JOIN STOR_ID IN STORES TO ALL STOR_ID IN SALES AS J1
TABLE FILE STORES
SUM STOR_NAME QTY
    ORD_DATE TITLE_ID
BY STOR_ID
END
```

FOC2590 - AGGREGATION NOT DONE FOR THE FOLLOWING REASON:

FOC2591 - AVE ON ALPHA, DATE OR DATETIME CANNOT BE AGGREGATED

```
SELECT T1."stor_id",T1."stor_name",T2."stor_id",T2."ord_num",
T2."title_id",T2."ord_date",T2."qty" FROM pubs.dbo.stores T1,
pubs.dbo.sales T2 WHERE (T2."stor_id" = T1."stor_id")
ORDER BY T1."stor_id",T2."stor_id",T2."ord_num",T2."title_id";
```

Relational Efficiencies: Aggregation Scenarios

SUM on ALPHANUMERIC Field

```
JOIN STOR_ID IN STORES TO ALL STOR_ID IN SALES AS J1
TABLE FILE STORES
SUM STOR_NAME QTY
    MAX.ORD_DATE MIN.TITLE_ID
BY STOR_ID
--*BY TITLE_ID
--*BY ORD_DATE
END
```

```
FOC2589 - AGGREGATION DONE BUT MAY PRODUCE INCONSISTENT RESULTS
FOC2612 - OVERFLOW MAY OCCUR WHEN SUMMING AN I2 FIELD WITH USAGE I, D OR P
SELECT T1."stor_id", MAX(T1."stor_name"), SUM(T2."qty"),
MAX(T2."ord_date"), MIN(T2."title_id") FROM pubs.dbo.stores T1,
pubs.dbo.sales T2 WHERE (T2."stor_id" = T1."stor_id") GROUP BY
T1."stor_id" ORDER BY T1."stor_id";
```

```
FOC2589 - AGGREGATION DONE BUT MAY PRODUCE INCONSISTENT RESULTS
FOC2612 - OVERFLOW MAY OCCUR WHEN SUMMING AN I2 FIELD WITH USAGE I, D OR P
SELECT T1."stor_id",T2."title_id",T2."ord_date",
MAX(T1."stor_name"), SUM(T2."qty") FROM pubs.dbo.stores T1,
pubs.dbo.sales T2 WHERE (T2."stor_id" = T1."stor_id") GROUP BY
T1."stor_id",T2."title_id",T2."ord_date" ORDER BY T1."stor_id",
T2."title_id",T2."ord_date";
```

Relational Efficiencies: Aggregation Scenarios IF/WHERES Translated for Aggregation

```
JOIN STOR_ID IN STORES TO ALL STOR_ID IN SALES AS J1
TABLE FILE STORES
SUM QTY BY STOR_ID BY STOR_NAME
WHERE STOR_ID GT '7130'
WHERE ZIP LIKE '9%'
WHERE QTY EQ EDIT('10')
-*WHERE QTY GT 10
WHERE PAYTERM LIKE 'Net %'
END
```

FOC2598 - FOCUS IF/WHERE TEST CANNOT BE PASSED TO SQL : WHERE expression

FOC2590 - AGGREGATION NOT DONE FOR THE FOLLOWING REASON:

FOC2596 - ONE OR MORE EXPRESSION(S) CAN NOT BE TRANSLATED TO SQL

```
SELECT T1."stor_id",T1."stor_name",T1."zip",T2."qty",
T2."payterms" FROM pubs.dbo.stores T1,pubs.dbo.sales T2 WHERE
(T2."stor_id" = T1."stor_id") AND (T1."zip" LIKE '9%') AND
(T1."stor_id" > '7130') AND (T2."payterms" LIKE 'Net %') ORDER
BY T1."stor_id",T1."stor_name";
```

...

```
WHERE (T2."stor_id" = T1."stor_id") AND (T1."zip" LIKE '9%') AND (T1."stor_id" >
'7130') AND (T2."payterms" LIKE 'Net %') AND (T2."qty" > 10) GROUP BY
T1."stor_id",T1."stor_name" ORDER BY T1."stor_id", T1."stor_name";
```

Relational Efficiencies: Aggregation Scenarios

DEFINE Field Logical Expression: Optimized

```
JOIN STOR_ID IN STORES TO ALL STOR_ID IN SALES AS J1
```

```
DEFINE FILE STORES
```

```
LOGICAL_NUM/I1=IF QTY GT 20 THEN 1 ELSE 0;
```

```
LOGICAL_CHR/A1=IF QTY GT 20 THEN 'Y' ELSE 'N';
```

```
END
```

```
TABLE FILE STORES
```

```
SUM QTY
```

```
LOGICAL_NUM
```

```
-*LOGICAL_CHR
```

```
BY STOR_ID BY STOR_NAME
```

```
END
```

```
FOC2589 - AGGREGATION DONE BUT MAY PRODUCE INCONSISTENT RESULTS
```

```
FOC2612 - OVERFLOW MAY OCCUR WHEN SUMMING AN I2 FIELD WITH USAGE I, D OR P
```

```
SELECT T1."stor_id",T1."stor_name", SUM(T2."qty"), SUM((CASE  
WHEN (T2."qty" > 20) THEN 1 ELSE 0 END)) FROM pubs.dbo.stores  
T1,pubs.dbo.sales T2 WHERE (T2."stor_id" = T1."stor_id") GROUP  
BY T1."stor_id",T1."stor_name" ORDER BY T1."stor_id",  
T1."stor_name";
```

Message in earlier releases:

```
(FOC2590) AGGREGATION NOT DONE FOR THE FOLLOWING REASON:
```

```
(FOC2597) USE OF DEFINE FIELD THAT CANNOT BE AGGREGATED : LOGICAL
```

Relational Efficiencies: Selection Scenarios

Non-Translatable Screening Conditions

Expressions using	Example
User-written subroutines	<pre>DEFINE... FNL/I3 = ARGLEN(15, LN, FNL); TABLE... IF FNL LE 6</pre>
Self-referential expressions	<pre>DEFINE... CPT/I2=CPT+1; TABLE... IF CPT NE 0</pre>
EDIT for field format conversions	<pre>WHERE EDIT(ID) GT 20</pre>
Strong concatenation (II)	<pre>DEFINE... NAME/A27=FN (' ' LN); TABLE... IF NAME EQ 'DANIEL VALINO'</pre>
DECODE function	<pre>DEFINE... DEVISION/A11=DECODE CDIV ('CORP' 'CORPORATE' 'NE' 'NORTH-EAST' ELSE 'NA'); TABLE... IF DIVISION EQ 'CORPORATE' OR 'NORTH-EAST'</pre>
Non-SQL relational operators (INCLUDES, EXCLUDES)	<pre>IF LN INCLUDES 'VALINO'</pre>
FOCUS subroutines (ABS, INT, MAX, MIN, LOG, SQRT)	<pre>WHERE SQRT(CSAL) GT 260</pre>
Expressions using fields with ACTUAL=DATE	<pre>DEFINE... HDAT2/YYMD=HDAT+365; TABLE... IF HDAT2 GT '1990/03/01'</pre>

Relational Efficiencies: Selection Scenarios Variable with Subroutine Reference

```
-DEFAULT &EMPVAL='pma42628m';  
DEFINE FILE EMPLOYEE  
EMPTEST/A9=UPCASE(9, '&EMPVAL', 'A9');  
END  
TABLE FILE EMPLOYEE  
PRINT JOB_ID JOB_LVL HIRE_DATE  
BY EMP_ID  
WHERE EMP_ID EQ UPCASE(9, '&EMPVAL', 'A9')  
-*WHERE EMP_ID EQ EMPTEST  
END
```

FOC2598 - FOCUS IF/WHERE TEST CANNOT BE PASSED TO SQL : WHERE
expression

FOC2590 - AGGREGATION NOT DONE FOR THE FOLLOWING REASON:

FOC2594 - AGGREGATION IS NOT APPLICABLE TO THE VERB USED

```
SELECT T1."emp_id",T1."job_id",T1."job_lvl",T1."hire_date"  
FROM pubs.dbo.employee T1 ORDER BY T1."emp_id";
```

Relational Efficiencies: Selection Scenarios Variable with Subroutine Reference

```
-DEFAULT &EMPVAL='pma42628m';  
-SET &UPEMPVAL=UPCASE(9, '&EMPVAL.EVAL', 'A9');  
DEFINE FILE EMPLOYEE  
EMPTEST/A9 WITH EMP_ID='&UPEMPVAL';  
END  
TABLE FILE EMPLOYEE  
PRINT JOB_ID JOB_LVL HIRE_DATE  
BY EMP_ID  
WHERE EMP_ID EQ '&UPEMPVAL'  
-*WHERE EMP_ID EQ EMPTEST  
END
```

```
FOC2590 - AGGREGATION NOT DONE FOR THE FOLLOWING REASON:  
FOC2594 - AGGREGATION IS NOT APPLICABLE TO THE VERB USED  
SELECT T1."emp_id",T1."job_id",T1."job_lvl",T1."hire_date"  
FROM pubs.dbo.employee T1 WHERE (T1."emp_id" = 'PMA42628M')  
ORDER BY T1."emp_id";
```

Relational Efficiencies: Selection Scenarios

DATEs

FILENAME=EMPLOYEE, SUFFIX=SQLMSS , \$

SEGMENT=EMPLOYEE, SEGTYPE=S0, \$

...

FIELDNAME=HIRE_DATE, ALIAS=hire_date, USAGE=HYYMDs, ACTUAL=HYYMDs, \$

```
DEFINE FILE EMPLOYEE
```

```
Hire_Year/YY=HPART(HIRE_DATE, 'YEAR', 'YY');
```

```
END
```

```
TABLE FILE EMPLOYEE
```

```
PRINT COMPUTE NAME/A55=LNAME || ', ' | FNAME | ' ' | MINIT; HIRE_DATE
```

```
BY LNAME NOPRINT BY FNAME NOPRINT BY MINIT NOPRINT
```

```
WHERE Hire_Year GE 1990 and Hire_Year LT 1991
```

```
END
```

FOC2598 - FOCUS IF/WHERE TEST CANNOT BE PASSED TO SQL : Hire_Year

FOC2598 - FOCUS IF/WHERE TEST CANNOT BE PASSED TO SQL : Hire_Year

FOC2590 - AGGREGATION NOT DONE FOR THE FOLLOWING REASON:

FOC2594 - AGGREGATION IS NOT APPLICABLE TO THE VERB USED

SELECT T1."fname",T1."minit",T1."lname",T1."hire_date" FROM

pubs.dbo.employee T1 ORDER BY T1."lname",T1."fname",T1."minit";

Relational Efficiencies: Selection Scenarios

DATES

```
FILENAME=EMPLOYEE, SUFFIX=SQLMSS , $  
SEGMENT=EMPLOYEE, SEGTYPE=S0, $
```

...

```
FIELDNAME=HIRE_DATE, ALIAS=hire_date, USAGE=HYYMDs, ACTUAL=HYYMDs, $  
FIELDNAME=HIRE_YEAR, ALIAS=Hire_Year, USAGE=YY, ACTUAL=DATE, $
```

```
TABLE FILE EMPLOYEE
```

```
PRINT COMPUTE NAME/A55=LNAME || ', ' | FNAME | ' ' | MINIT; HIRE_DATE  
BY LNAME NOPRINT BY FNAME NOPRINT BY MINIT NOPRINT  
WHERE Hire_Year GE 1990 and Hire_Year LT 1991  
END
```

```
FOC2590 - AGGREGATION NOT DONE FOR THE FOLLOWING REASON:  
FOC2594 - AGGREGATION IS NOT APPLICABLE TO THE VERB USED  
SELECT T1."fname",T1."minit",T1."lname",T1."hire_date",  
T1."Hire_Year" FROM pubs.dbo.employee T1 WHERE (T1."Hire_Year"  
< '19910101') AND (T1."Hire_Year" >= '19900101') ORDER BY  
T1."lname",T1."fname",T1."minit";
```

Relational Efficiencies: Selection Scenarios

Case Sensitive/Insensitive Selection: Coming Soon

- WebFOCUS / FOCUS Selection values are case sensitive
- RDBMS environment could allow for case insensitive selection
- Answer set returned by RDBMS could allow for all permutations resulting in case insensitive values
- WebFOCUS / FOCUS will reapply selection so that final result will be case sensitive
- Workarounds
 - Direct SQL Passthru
 - UPCASE both the FIELD and VALUE in WHERE clause
 - Selection will not be optimized and will impact aggregation if aggregation is part of the request
- New feature to allow for user to specify if selection should be case sensitive or case insensitive

Relational Efficiencies

Thank You !

Relational Efficiencies

Appendix

```
FILENAME=EMPLOYEE, SUFFIX=SQLMSS , $
SEGMENT=EMPLOYEE, SEGTYPE=S0, $
  FIELDNAME=EMP_ID, ALIAS=emp_id, USAGE=A9, ACTUAL=A9, $
  FIELDNAME=FNAME, ALIAS=fname, USAGE=A20, ACTUAL=A20, $
  FIELDNAME=MINIT, ALIAS=minit, USAGE=A1, ACTUAL=A1,
  MISSING=ON, $
  FIELDNAME=LNAME, ALIAS=lname, USAGE=A30, ACTUAL=A30, $
  FIELDNAME=JOB_ID, ALIAS=job_id, USAGE=I6, ACTUAL=I2, $
  FIELDNAME=JOB_LVL, ALIAS=job_lvl, USAGE=I6, ACTUAL=I4,
  MISSING=ON, $
  FIELDNAME=PUB_ID, ALIAS=pub_id, USAGE=A4, ACTUAL=A4, $
  FIELDNAME=HIRE_DATE, ALIAS=hire_date, USAGE=HYYMDs, ACTUAL=HYYMDs, $

SEGNAME=EMPLOYEE, TABLENAME=pubs.dbo.employee, CONNECTION=SQLSRV, KEYS=1,$
```

Relational Efficiencies

Appendix

```
FILENAME=AUTHORS, SUFFIX=SQLMSS, $
SEGMENT=AUTHORS, SEGTYPE=S0, $
  FIELDNAME=AU_ID, ALIAS=au_id, USAGE=A11, ACTUAL=A11, $
  FIELDNAME=AU_LNAME, ALIAS=au_lname, USAGE=A40, ACTUAL=A40, $
  FIELDNAME=AU_FNAME, ALIAS=au_fname, USAGE=A20, ACTUAL=A20, $
  FIELDNAME=PHONE, ALIAS=phone, USAGE=A12, ACTUAL=A12, $
  FIELDNAME=ADDRESS, ALIAS=address, USAGE=A40, ACTUAL=A40,
  MISSING=ON, $
  FIELDNAME=CITY, ALIAS=city, USAGE=A20, ACTUAL=A20,
  MISSING=ON, $
  FIELDNAME=STATE, ALIAS=state, USAGE=A2, ACTUAL=A2,
  MISSING=ON, $
  FIELDNAME=ZIP, ALIAS=zip, USAGE=A5, ACTUAL=A5,
  MISSING=ON, $
  FIELDNAME=CONTRACT, ALIAS=contract, USAGE=I11, ACTUAL=I4, $

SEGNAME=AUTHORS, TABLENAME=pubs.dbo.authors, CONNECTION=SQLSRV, KEYS=1,$
```

Relational Efficiencies

Appendix

```
FILENAME=TITLES, SUFFIX=SQLMSS , $
SEGMENT=TITLES, SEGTYPE=S0, $
  FIELDNAME=TITLE_ID, ALIAS=title_id, USAGE=A6, ACTUAL=A6, $
  FIELDNAME=TITLE, ALIAS=title, USAGE=A80, ACTUAL=A80, $
  FIELDNAME=TYPE, ALIAS=type, USAGE=A12, ACTUAL=A12, $
  FIELDNAME=PUB_ID, ALIAS=pub_id, USAGE=A4, ACTUAL=A4,
  MISSING=ON, $
  FIELDNAME=PRICE, ALIAS=price, USAGE=P21.4, ACTUAL=P10,
  MISSING=ON, $
  FIELDNAME=ADVANCE, ALIAS=advance, USAGE=P21.4, ACTUAL=P10,
  MISSING=ON, $
  FIELDNAME=ROYALTY, ALIAS=royalty, USAGE=I11, ACTUAL=I4,
  MISSING=ON, $
  FIELDNAME=YTD_SALES, ALIAS=ytd_sales, USAGE=I11, ACTUAL=I4,
  MISSING=ON, $
  FIELDNAME=NOTES, ALIAS=notes, USAGE=A200, ACTUAL=A200,
  MISSING=ON, $
  FIELDNAME=PUBDATE, ALIAS=pubdate, USAGE=HYMDs, ACTUAL=HYMDs, $

SEGNAME=TITLES, TABLENAME=pubs.dbo.titles, CONNECTION=SQLSRV, KEYS=1, $
```

Relational Efficiencies

Appendix

```
FILENAME=TITLEAUTHOR, SUFFIX=SQLMSS , $
SEGMENT=TITLEAUTHOR, SEGTYPE=S0, $
  FIELDNAME=AU_ID, ALIAS=au_id, USAGE=A11, ACTUAL=A11, $
  FIELDNAME=TITLE_ID, ALIAS=title_id, USAGE=A6, ACTUAL=A6, $
  FIELDNAME=AU_ORD, ALIAS=au_ord, USAGE=I6, ACTUAL=I4,
  MISSING=ON, $
  FIELDNAME=ROYALTYPER, ALIAS=royaltyper, USAGE=I11, ACTUAL=I4,
  MISSING=ON, $

SEGNAME=TITLEAUTHOR, TABLENAME=pubs.dbo.titleauthor, CONNECTION=SQLSRV,
KEYS=2, $
```

Relational Efficiencies

Appendix

```
FILENAME=ROYSCHED, SUFFIX=SQLMSS , $
```

```
SEGMENT=ROYSCHED, SEGTYPE=S0, $
```

```
FIELDNAME=TITLE_ID, ALIAS=title_id, USAGE=A6, ACTUAL=A6, $
```

```
FIELDNAME=LORANGE, ALIAS=lorange, USAGE=I11, ACTUAL=I4,
```

```
MISSING=ON, $
```

```
FIELDNAME=HIRANGE, ALIAS=hirange, USAGE=I11, ACTUAL=I4,
```

```
MISSING=ON, $
```

```
FIELDNAME=ROYALTY, ALIAS=royalty, USAGE=I11, ACTUAL=I4,
```

```
MISSING=ON, $
```

```
SEGNAME=ROYSCHED, TABLENAME=pubs.dbo.roysched, CONNECTION=SQLSRV, KEYS=1,$
```

Relational Efficiencies

Appendix

```
FILENAME=STORES, SUFFIX=SQLMSS , $  
SEGMENT=STORES, SEGTYPE=S0, $  
  FIELDNAME=STOR_ID, ALIAS=stor_id, USAGE=A4, ACTUAL=A4, $  
  FIELDNAME=STOR_NAME, ALIAS=stor_name, USAGE=A40, ACTUAL=A40,  
    MISSING=ON, $  
  FIELDNAME=STOR_ADDRESS, ALIAS=stor_address, USAGE=A40, ACTUAL=A40,  
    MISSING=ON, $  
  FIELDNAME=CITY, ALIAS=city, USAGE=A20, ACTUAL=A20,  
    MISSING=ON, $  
  FIELDNAME=STATE, ALIAS=state, USAGE=A2, ACTUAL=A2,  
    MISSING=ON, $  
  FIELDNAME=ZIP, ALIAS=zip, USAGE=A5, ACTUAL=A5,  
    MISSING=ON, $  
  
SEGNAME=STORES, TABLENAME=pubs.dbo.stores, CONNECTION=SQLSRV, KEYS=1, $
```

Relational Efficiencies

Appendix

FILENAME=SALES, SUFFIX=SQLMSS, \$

SEGMENT=SALES, SEGTYPE=S0, \$

FIELDNAME=STOR_ID, ALIAS=stor_id, USAGE=A4, ACTUAL=A4, \$

FIELDNAME=ORD_NUM, ALIAS=ord_num, USAGE=A20, ACTUAL=A20, \$

FIELDNAME=TITLE_ID, ALIAS=title_id, USAGE=A6, ACTUAL=A6, \$

FIELDNAME=ORD_DATE, ALIAS=ord_date, USAGE=HYMDs, ACTUAL=HYMDs, \$

FIELDNAME=QTY, ALIAS=qty, USAGE=I6, ACTUAL=I2, \$

FIELDNAME=PAYTERMS, ALIAS=payterms, USAGE=A12, ACTUAL=A12, \$

SEGNAME=SALES, TABLENAME=pubs.dbo.sales, CONNECTION=SQLSRV, KEYS=3, \$