

Internals of Retrieval for TABLE/TABLEF/MATCH FILE

Noreen Redden

Information Builders

FIRE 2008 User Conference

Presentation Information

- **Author:** Cesare Petrizio
- **Company:** Information Builders
- **Presentation Title:** Internals of Retrieval for TABLE/TABLEF/MATCH FILE
- **Abstract:** This presentation looks at what FOCUS is doing under the covers when a TABLE is executed in FOCUS on any platform, or through WEBFOCUS. Also, the TABLE could be against a FOCUS file, DB2 TABLE, VSAM, etc. Application developers can create more efficient requests when they understand how FOCUS sets up to retrieve data from FOCUS files, Relational TABLEs, and other non-FOCUS data sources. Learn the answers to such questions as: When would an index be more efficient? One pass or two? DEFINE vs. COMPUTE? What is AUTOPATH and AUTOINDEX, and what do they mean in terms of efficiency?

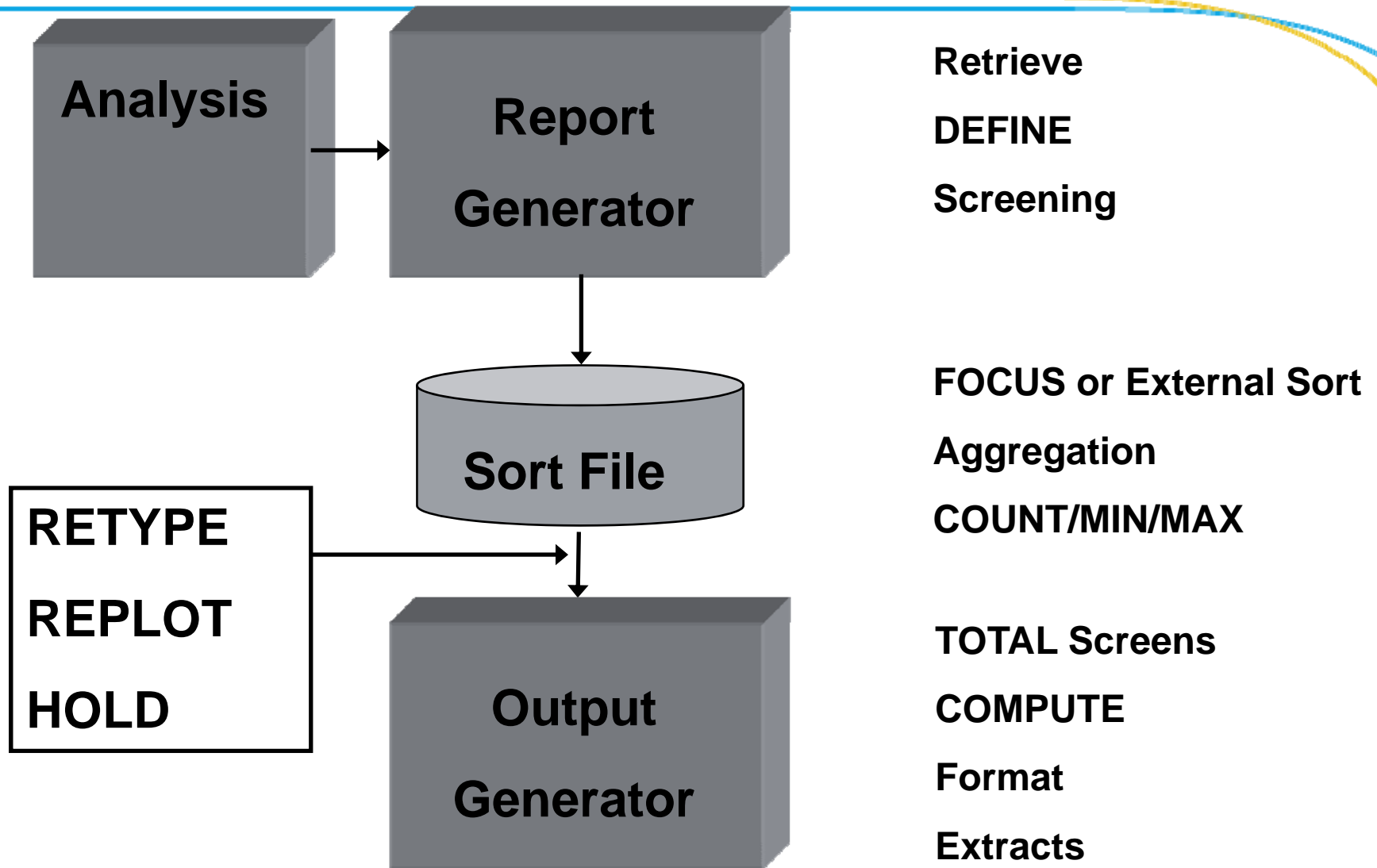


Information Builders Summit 2004
User Conference

Agenda

- **Reporting Internals**
 - **Retrieval Internals**
 - **Analysis Phase**
 - **Retrieval Phase**
 - **Internal Matrix Generation**
- **http://documentation.informationbuilders.com/masterindex/html/html_s390_76/mf_snf76/mf_snf76.pdf**

Retrieval Internals



Retrieval Internals

The Files

```
FILE=KIDS,SUFFIX=DB2
SEGNAME=SEGK,SEGTYPE=S2
FIELD=EMP_ID,      EID ,A9,$
FIELD=CHILD_DOB,   ,YYMD,$
FIELD=CHILD_NAME,  ,A20,$
```

```
FILE=SPICE,SUFFIX=FOC
SEGNAME=SEGS,SEGTYPE=S1
FIELD=EMP_ID, EID, A9, INDEX=I,$
FIELD=SPOUSE,SPOUSE_NME,A20,$
```

```
FILE=EMPLOYEE,SUFFIX=FOC
SEGNAME=SEG1,SEGTYPE=S1
FIELD=EMP_ID,      EID,A9,$
FIELD=FIRST_NAME,  FN,A15,$
FIELD=LAST_NAME,   LN,A15,$
SEGNAME=INSSEG,SEGTYPE=SH2
FIELD=COVER_DT,    ,YMD,$
FIELD=TYPE_COVER,  ,A3,$
FIELD=FAMILY,      ,A1,$
```

Retrieval Internals

The Files XFOCUS

Are Any Of The Following Different?

- DEFINE's
- Selection Test
- JOIN'S

```
FILE=SPICE,SUFFIX=FOC  
SEGNAME=SEGS,SEGTYPE=S1  
FIELD=EMP_ID, EID, A9, INDEX=I,$  
FIELD=SPOUSE,SPOUSE_NME,A20,$
```

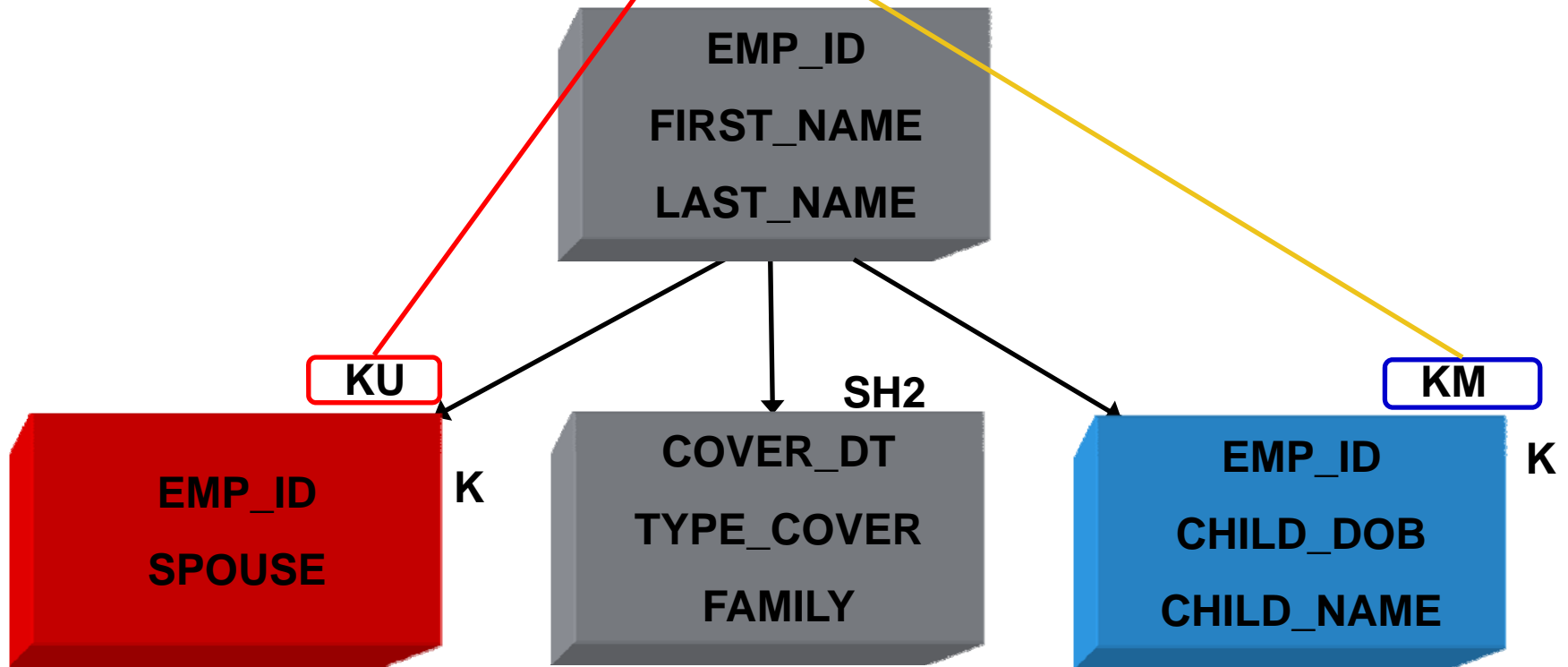
```
FILE=SPICE,SUFFIX=XFOC  
SEGNAME=SEGS,SEGTYPE=S1  
FIELD=EMP_ID, EID, A9, INDEX=I,$  
FIELD=SPOUSE,SPOUSE_NME,A20,$
```

Retrieval Internals

Example

JOIN EMP_ID IN EMPLOYEE TO EMP_ID IN SPICE AS AJ

JOIN EMP_ID IN EMPLOYEE TO ALL EMP_ID IN KIDS AS BJ



DEFINE

- Each DEFINE field is associated with a specific segment
- Segment is determined by:
 - WITH field
 - Fields used in expression
- Constant expressions are evaluated when file is opened (segment 0)
- DEFINES are evaluated **ONLY** if required by request
- DEFINES are evaluated at detail level only

Analysis

- Open and CHECK Master File Descriptions
 - Verify DBA access to file(s)
- Determine I/O access modules required
- Retrieve DEFINE field expressions for pertinent files
- Read and parse the request
- Check DBA and set up access restrictions at SEGMENT and FIELD levels

Analysis

- Activate segments: referenced subtree(s) – smallest subtree which contains “root” and every segment referenced directly or indirectly (HEADING PARENT AND GRAND CHILD)
- Activate fields (real and DEFINEd)
- Read USE list and/or check Access Files and FILEDEF/ALLOCATES
- Open and Verify files

Retrieval Internals

The Files

KIDS

```
          SEGK
01      S2
*****
*EMP_ID      **
*CHILD_DOB   **
*CHILD_NAME  **
*            **
*            **
*****
*****
```

EMPLOYEE

```
          SEG1
01      S1
*****
*EMP_ID      **
*FIRST_NAME  **
*LAST_NAME   **
*            **
*            **
*****
*****
          I
          I
          I I INSSEG
02      I SH2
*****
*COVER_DT    **
*TYPE_COVER  **
*FAMILY      **
*            **
*            **
*****
*****
```

SPICE

```
          SEGS
01      S1
*****
*EMP_ID      **I
*SPOUSE      **
*            **
*            **
*            **
*****
*****
```

Retrieval Internals

```

SEG1
  01      S1
*****
*EMP_ID   ** JOIN EMP_ID IN EMPLOYEE TO      EMP_ID IN SPICE AS J1
*FIRST_NAME ** JOIN EMP_ID IN EMPLOYEE TO ALL EMP_ID IN KIDS  AS J2
*LAST_NAME **
*         **
*         **
*****
*****
      I
      +-----+-----+
      I           I           I
      I SEGS      I SEGKEG      I INSSEG
02      I KU      03      I KM2      04      I SH2
.....          *****          *****
:EMP_ID       :K   *EMPERDDT      **:K   *COVER_DT      **
:SPOUSE       :   *EMPEDCOVER    **:   *TYPE_COVER   **
:             :   *EMLDYNAME     **:   *FAMILY        **
:             :   *             **:   *             **
:             :   *             **:   *             **
:.....          *****          *****
JOINED SPICE   *****          *****
              *****          *****
              JOINED KIDS

```



Retrieval Internals

```
DEFINE FILE EMPLOYEE
INS_TYPE/A6 = DECODE TYPE_COVER(
                DDS    'DENTAL'
                EYE    'VISION'
                MDS    'HEALTH'
                ELSE 'OTHER');
TODAY/YYMD = '&YYMD';
AGE/I3 = (TODAY - CHILD_DOB) / 365.25;
END
TABLE FILE EMPLOYEE
PRINT  INS_TYPE
       CHILD_NAME
       AGE
       BY LAST_NAME
       BY FIRST_NAME
       BY SPOUSE
       IF AGE LE 20
END
```

Seg 4 Define Based on
TYPE_COVER

Seg 0 Define (No Reference)
Seg 3 Define Based on
CHILD_DOB

Retrieval Internals Segment Activation

Seg 1 Activated by
FIRST_NAME and
LASTNAME

Seg 2 Activated by
SPOUSE

Seg 3 Activated by
CHILD_DOB and
CHILD_NAME

Seg 4 Activated by
TYPE_COVER Via
INS_TYPE DEFINE

```

SEG1
  01      S1
*****
*EMP_ID      **
*FIRST_NAME  **
*LAST_NAME   **
*            **
*            **
*****
                                CHILD_NAME
                                AGE
                                BY LAST_NAME
                                BY FIRST_NAME
                                BY SPOUSE
                                IF AGE LE 20
                                END
                                I
                                +-----+-----+
                                I          I          I
                                I SEGS    I SEGK    I INSSEG
  02      I KU          03      I KM          04      I SH2
*****
:EMP_ID      :K      :EMP_ID      ::K      *COVER_DT      **
:SPOUSE      :      :CHILD_DOB     ::      *TYPE_COVER    **
:            :      :CHILD_NAME    ::      *FAMILY        **
:            :      :            ::      *            **
:            :      :            ::      *            **
:.....:      :.....:      :.....:      *****
JOINED  SPICE      :.....:      *****
                                JOINED  KIDS

```



Retrieval Internals

Field Activation

- **Fields Mentioned in the Body of TABLE are ACTIVATED**

- **FIRST_NAME**
- **LASTNAME**
- **SPOUSE**
- **CHILD_NAME**
- **CHILD_DOB** (Via AGE DEFINE)
- **TYPE_COVER** (Via INS_TYPE DEFINE)

```
DEFINE FILE EMPLOYEE
INS_TYPE/A6 = DECODE TYPE_COVER(
                DDS   'DENTAL'
                EYE   'VISION'
                MDS   'HEALTH'
                ELSE 'OTHER');

TODAY/YYMD = '&YYMD';
AGE/I3 = (TODAY - CHILD_DOB) / 365.25;
END

TABLE FILE EMPLOYEE
PRINT  INS_TYPE
      CHILD_NAME
      AGE
      BY LAST_NAME
      BY FIRST_NAME
      BY SPOUSE
      IF AGE LE 20

END
```

Retrieval

- Retrieval is “Top Down, Left Right” within the referenced sub-tree
- Unique segments are promoted into path of parent and are never missing
- IF/WHERE tests on database fields are evaluated upon retrieval
- DEFINE fields are evaluated if needed
- IF/WHERE tests are evaluated on DEFINE fields on segment by segment basis, AFTER all other selection tests are passed
- Each Path is processed independently, and fields are merged on common sort fields

Retrieval

Order of Retrieval – FOCUS Files

- **TABLE/GRAPH/MATCH**
 - Root segment instances are obtained in physical order
 - Lower level instances obtained in **SEGTYPE** order within parent segment
- **TABLEF**
 - Root and child segment instances are obtained in **SEGTYPE** order
 - No sorting

Retrieval

Order of Retrieval – FOCUS Files

- **TABLE FILE filename.fieldname**
 - **When Fieldname is not indexed**
 - **Segment containing “fieldname” becomes ROOT of the view, and is retrieved physically**
 - **When Fieldname is indexed**
 - **Segment containing “fieldname” is accessed via the equality test on INDEX**
 - **Other segments become children of the new root segment**

Retrieval Internals

AUTOPATH - Two Masters And How It Treats Seg1

SEG1 JOIN EMP_ID IN EMPLOYEE TO EMP_ID IN SPICE AS J1

```

01      S1
*****
*EMP_ID      **
*FIRST_NAME  **
*LAST_NAME   **
*            **
*            **
*****
*****

```

```

FILE=EMPLOYEE,SUFFIX=SQL
SEGNAME=SEG1,SEGTYPE=S0
FIELD=EMP_ID,      EID,A9,$
FIELD=FIRST_NAME, FN,A15,$
FIELD=LAST_NAME,   LN,A15,$

```

```

TABLE FILE EMPLOYEE
PRINT SPOUSE
END

```

```

I
+
I
I SEGS
02 I KU

```

HOW MANY SEGMENTS DO WE READ?
Root SEG always gets read

```

.....
:EMP_ID      :K
:SPOUSE      :
:            :
:            :
:            :
:            :
:.....:
JOINED      SPICE

```

```

FILE=SPICE,SUFFIX=SQL
SEGNAME=SEGS,SEGTYPE=S0
FIELD=EMP_ID, EID, A9, $
FIELD=SPOUSE,SPOUSE_NME,A20,$

```



Retrieval

Order of Retrieval – FOCUS Files

- **AUTOPATH = ON**
 - Alternate Physical View is created through the referenced segment highest in the hierarchy
- **AUTOINDEX = ON**
 - Alternate Indexed View is created if there is an Equality test specified for an indexed field on the referenced segment highest in the hierarchy
- **AUTOSTRATEGY = ON**
 - Determines when FOCUS stops the search for a key field specified in a WHERE or IF test. When set to ON, the search ends when the key field is found
- **FIXRETRIEVE = ON**
 - Equality IF/WHERE test on primary specified sort field in single segment FIX file will terminate retrieval as soon as possible

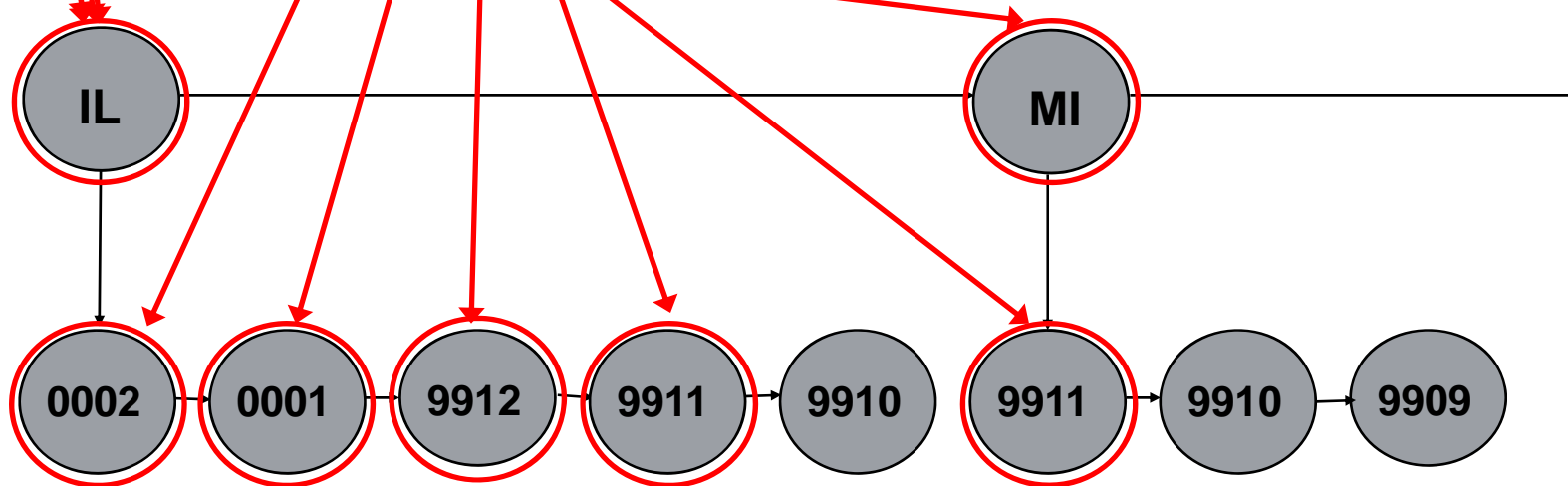
Retrieval Autostrategy – FOCUS Files

WHERE DATE EQ '9912'

- 1 STATE=IL,
- 2 STATE=IL,
- 3 STATE=IL,
- 4 STATE=IL,
- 5 STATE=MI,

- DATE=0002
- DATE=0001
- DATE=9912
- DATE=9911
- DATE=9911

Continue Search
Continue Search
Continue Search
Next Parent
Next Parent



FIXED Files

WHERE COUNTRY EQ 'FRANCE'

ENGLAND JAGUAR

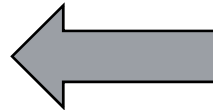
ENGLAND JENSEN

ENGLAND TRIUMPH

FRANCE PEUGEOT

ITALY AUDI

JAPAN.....



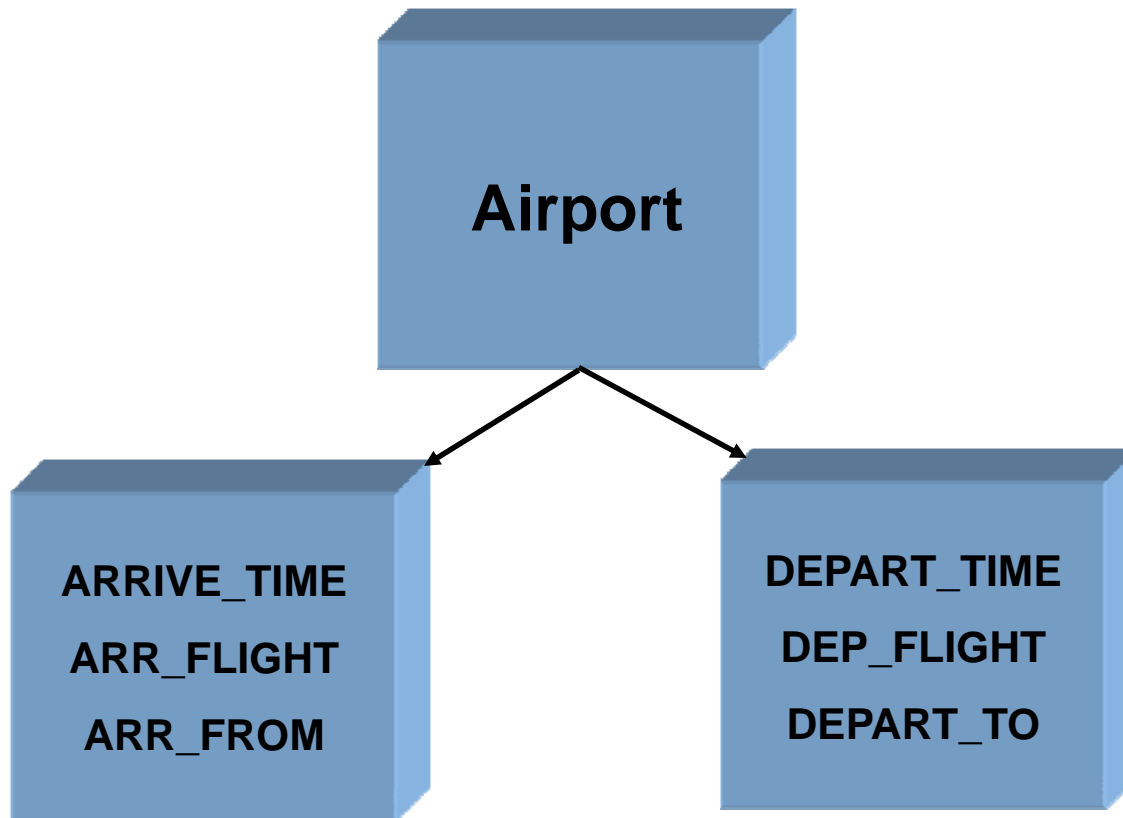
**Retrieval Stops When:
Set FIXRETRIEVAL = ON**

Retrieval

Multiple Paths

- Retrieval is performed for each path separately
 - Unique segments are always in the path of their parent
 - Unique segments are SEGTYPE = U, KU, DKU, KLU
 - JOIN TO creates DKU segments
 - JOIN TO ALL creates DKM segments
- TABLE/MATCH/GRAPH
 - Instances from each path will be merged in the sort process by the common parent(s)
- TABLEF does not merge data from multiple paths
- Alternate file views can be used to create single paths
- SET MULTIPATH controls multiple path retrieval

Multiple Paths



Multiple Paths

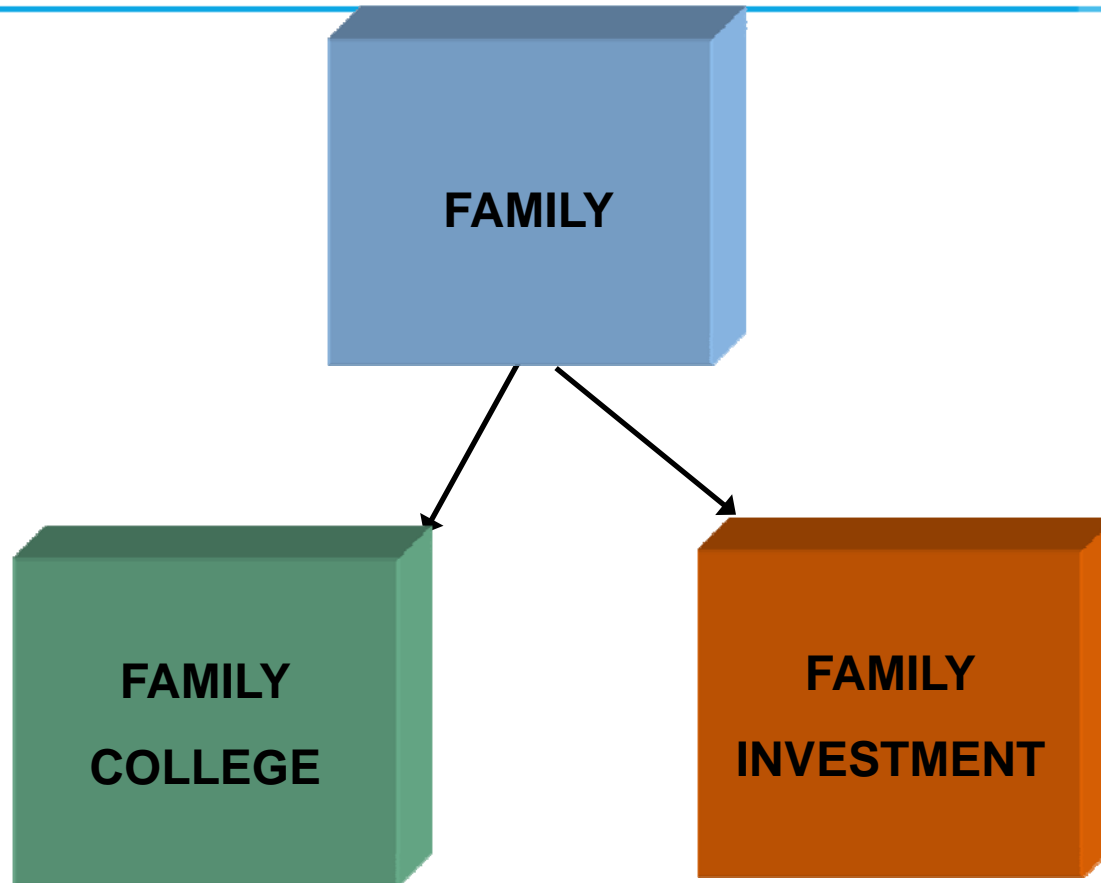
ARRIVE_TIME
ARR_FLIGHT
ARR_FROM

Airport

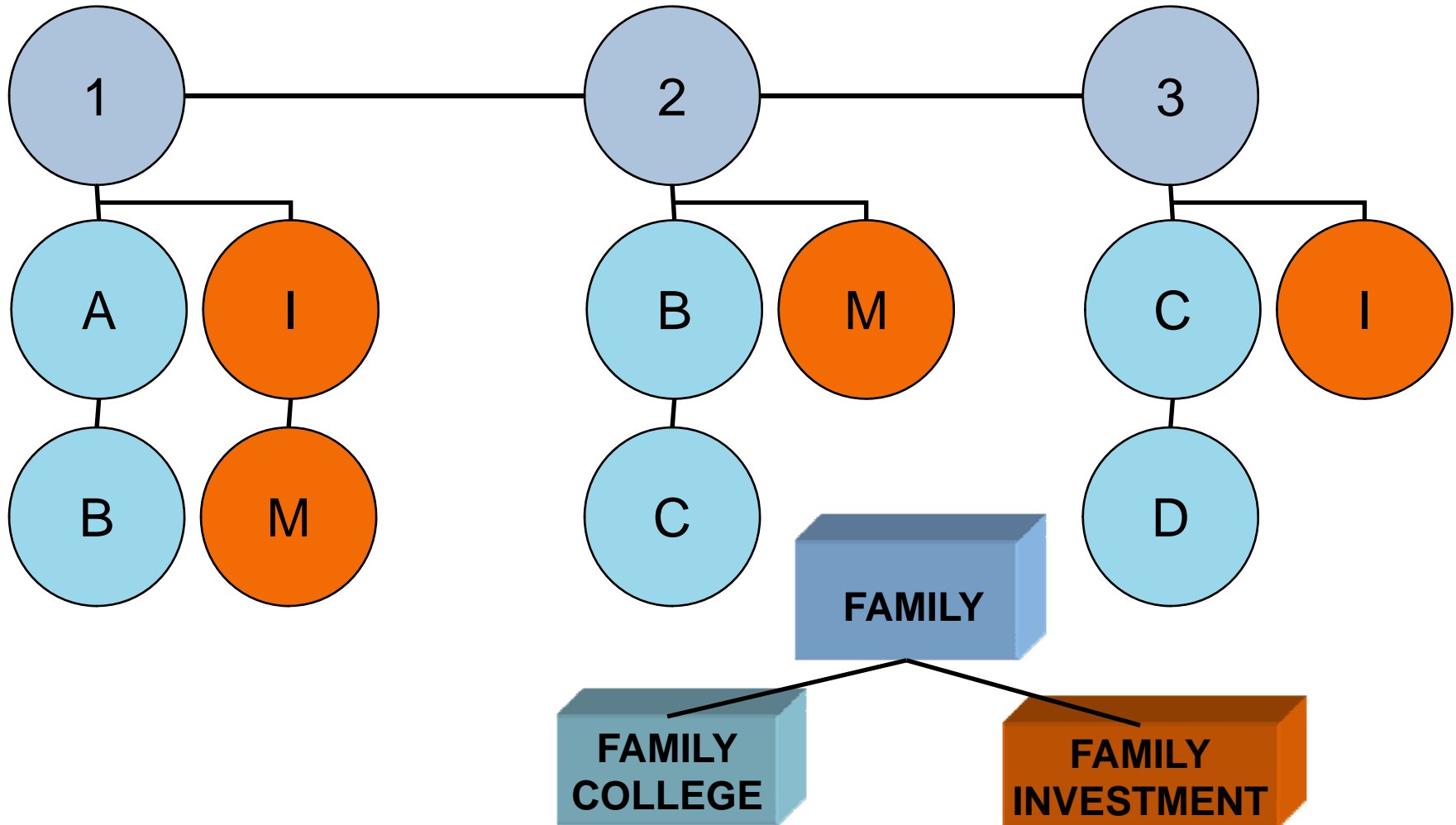
DEPART_TIME
DEP_FLIGHT
DEPART_TO

```
DEFINE FILE AIRPORTS .ARRIVE_TIME  
DELAY=DEPART_TIME - ARRIVE_TIME;  
END  
  
TABLE FILE AIRPORTS .ARRIVE_TIME  
PRINT DEP_FLIGHT DEPART_TIME  
WHERE AIRPORT EQ 'O' 'HARE'  
WHERE DELAY GT 60  
WHERE DEPART_TO EQ 'LAX'  
WHERE ARR_FROM EQ 'TOR'  
END
```

Multiple Paths



Retrieval Multiple Paths Instances



Multiple Paths

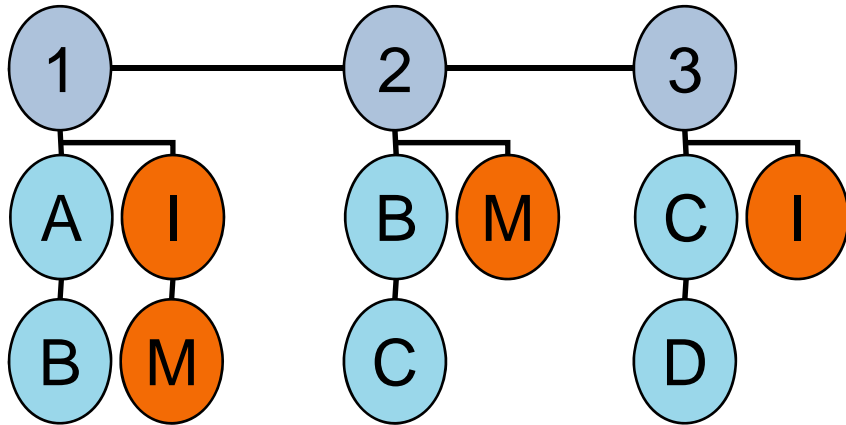


TABLE FILE SAMPLE
 PRINT COLLEGE INVEST
 BY FAMILY
 WHERE COLLEGE EQ 'B'
 WHERE INVEST EQ 'I'
 END

SET MULTIPATH=SIMPLE

FAMILY	COLLEGE	INVEST
1	B	I
2	B	.
3	.	I

SET MULTIPATH=COMPOUND

FAMILY	COLLEGE	INVEST
1	B	I

Retrieval Short Paths

- **SET ALL=OFF**

- High level segments with missing referenced descendents rejected (short path)

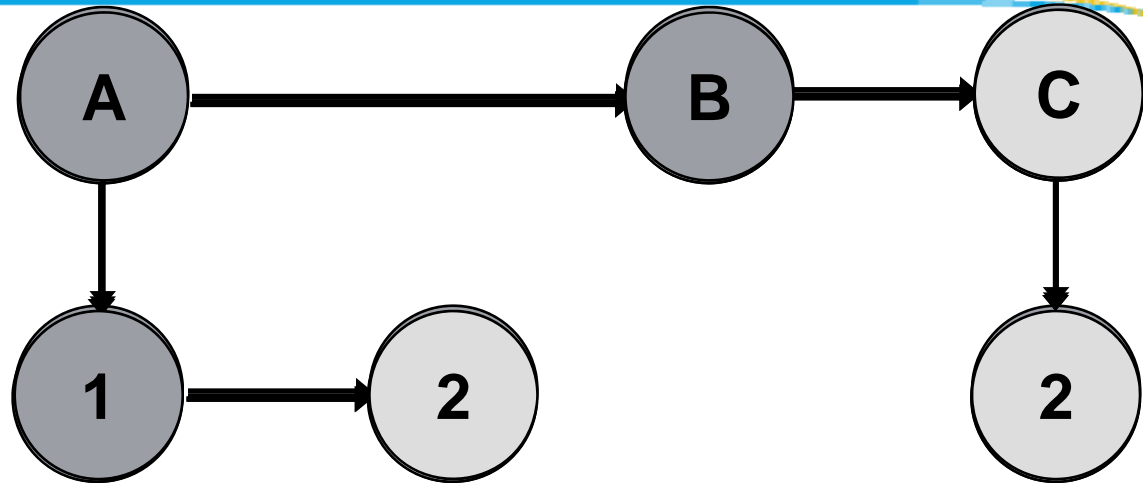
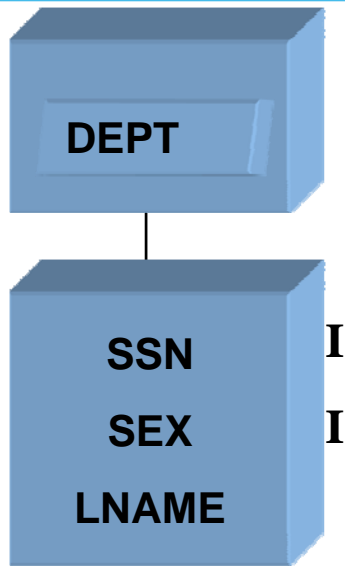
- **SET ALL=ON**

- High level segments with missing referenced descendents accepted (missing fails IF/WHERE tests)

- **SET ALL=PASS**

- High level segments with qualified or missing referenced descendents accepted (missing passes IF/WHERE tests)
- Valid only for FOCUS files

Retrieval Short Paths



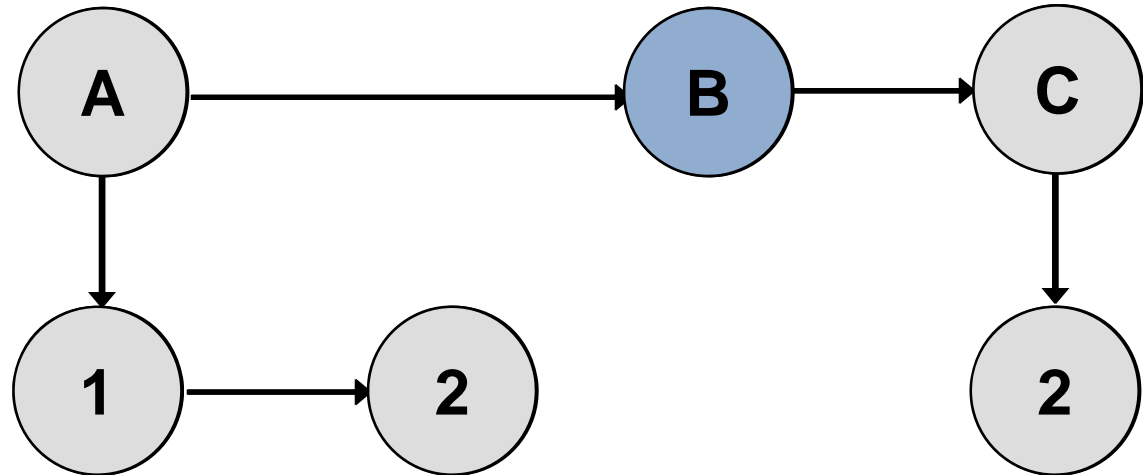
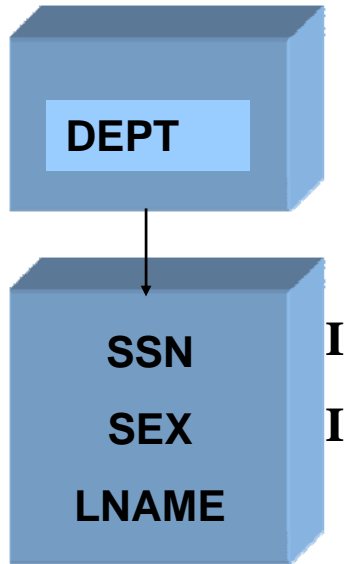
```

&ALL      &IF      RESULT
DEPTS SSN EQ 1, B 1, A 2, B 2,C 2
  
```

```

SET ALL = &ALL
TABLE FILE DEPTS
PRINT *
&IF
END
  
```

Retrieval Short Paths



SET ALL=**PASS**

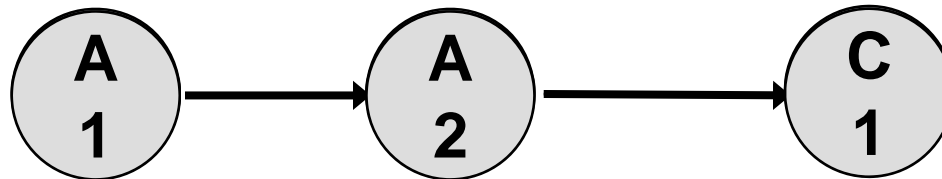
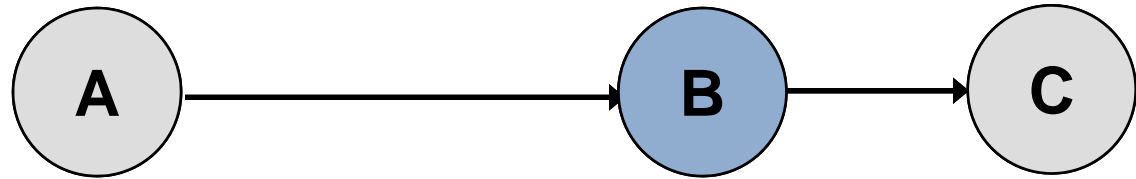
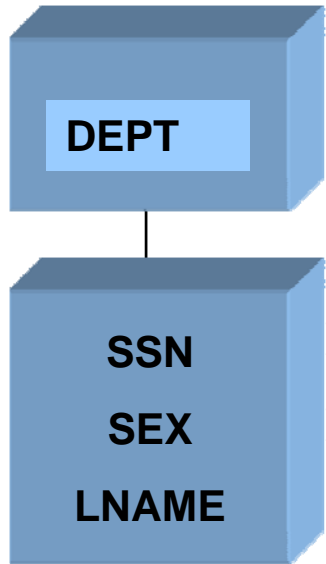
TABLE FILE DEPTS

PRINT DEPT SSN

WHERE **SSN EQ 1 AND SSN EQ 2**

END

Retrieval Short Paths



```
JOIN DEPT IN DEPT1 TO DEPT IN SSNS AS AJ
TABLE FILE DEPT1
PRINT DEPT
WHERE DEPT1.DEPT NE SSNS.DEPT
END
```

Retrieval Short Paths

- **Unique segments (U, KU, DKU, and KLU) are never considered missing**
- **Referenced, not present, Unique Segments are defaulted to blanks for alpha fields, zeros for numerics**
- **Unique segments do not create short paths. Therefore, ALL settings are irrelevant for Unique segments.**

FOCWIZARDISM

Missing Referenced Descendents Of
Missing Unique Segments Revert To:
ALL=OFF logic

If This Segment Is Missing →

Can't get here →

```
EMPDATA
01      S1
*****
*PIN           **I
*LASTNAME     **
*FIRSTNAME    **
*MIDINITIAL   **
*             **
*****
*****
          I
          I
          I
          I PERSONAL
02      I KU
.....
:PIN          :K
:INCAREOF    :
:STREETNO    :
:APT         :
:STATE       :I
:.....:
JOINEDI PERSINFO
          I
          I
          I
          I ONE
03      I KU
.....
:STATE       :K
:FULLNAME    :
:           :
:           :
:           :
:.....:
JOINED      STATENAME
```



Internal Matrix Generation

- **The output of the SORT/MERGE phase is conceptually a matrix**
 - **One row for each distinct combination of sort keys**
 - **One column for each verb object**
- **Computed columns, row and column totals, subtotals, and summaries are not yet done**
- **All data is in internal form**

Internal Matrix Generation

- **Verb Objects (both SUM and COUNT are calculated)**
 - **Fields following a verb**
 - **Fields used in COMPUTEs not previously mentioned**
 - **Fields used in HEADINGS or FOOTINGS**
 - **Fields used in SUBHEADs or SUBFOOTs, not previously mentioned**
 - **MISSING fields are not counted**
- **Sort Fields**
 - **BY fields**
 - **ACROSS fields (used as BY at sort phase)**
 - **FOCLIST (verb is PRINT or LIST)**

Internal Matrix Generation

Multiple Verb Sets

- Up to **16** verbs with associated BY fields can be specified
- Up to **256** verb objects may be specified

SUM SALES BY DIVISION

SUM SALES BY DIVISION BY YEAR

LIST SALES BY DIVISION BY YEAR

Internal Matrix Generation

Multiple Verb Sets

SUM SALES BY DIVISION
 SUM SALES BY DIVISION BY YEAR

LIST SALES BY DIVISION BY YEAR

DIV SALES YEAR SALES FOCLIST SALES

EAST	210	99	100	1	40
				2	60
		00	110	3	110
NORTH	150	99	60	1	60
		00	90	2	90
SOUTH	30	99	30	1	10
				2	20
WEST	10	99	10	1	10



Reporting Internals Output Processing

Attend the Follow-On Session
Output Processing Phase

See You There!



Questions



Thanks for Coming