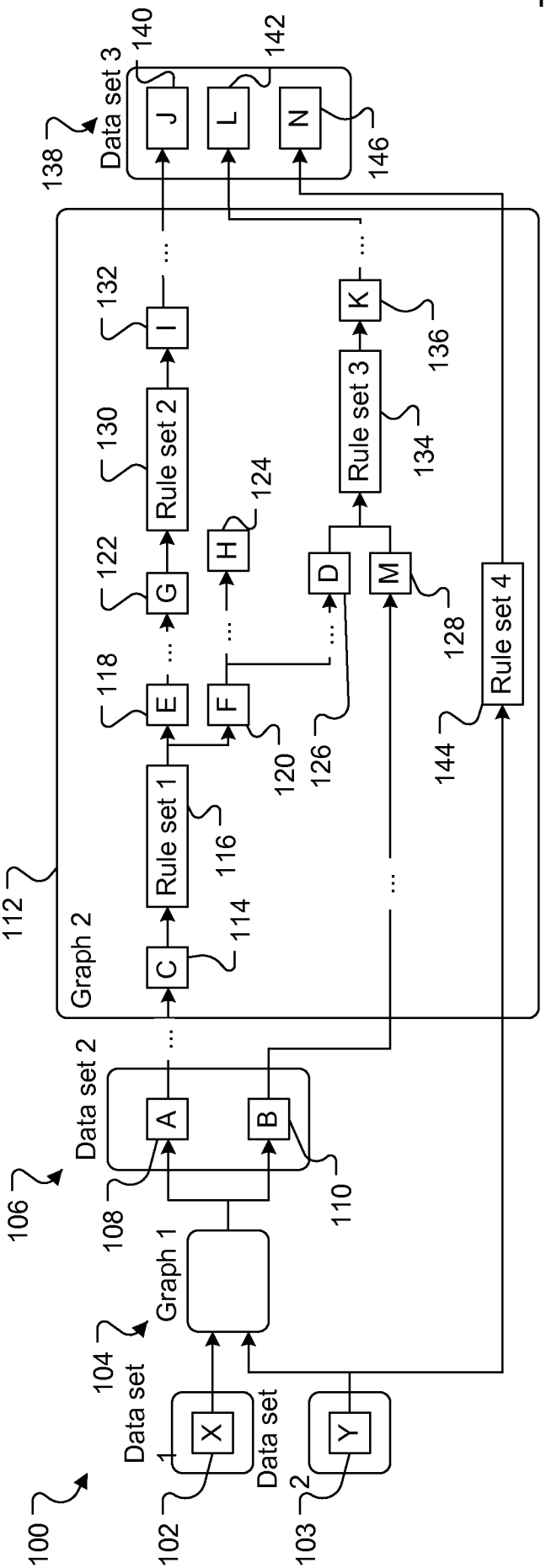
100
102
Data set 1
X
103
Data set 2
Y
104
Data set
Graph 1
106
108
Data set 2
A
B
110
112
Graph 2
C
114
Rule set 1
116
118
E
120
F
122
G
124
H
126
D
128
M
130
Rule set 2
132
I
134
Rule set 3
136
K
138
Data set 3
140
J
142
L
144
Rule set 4
146
N

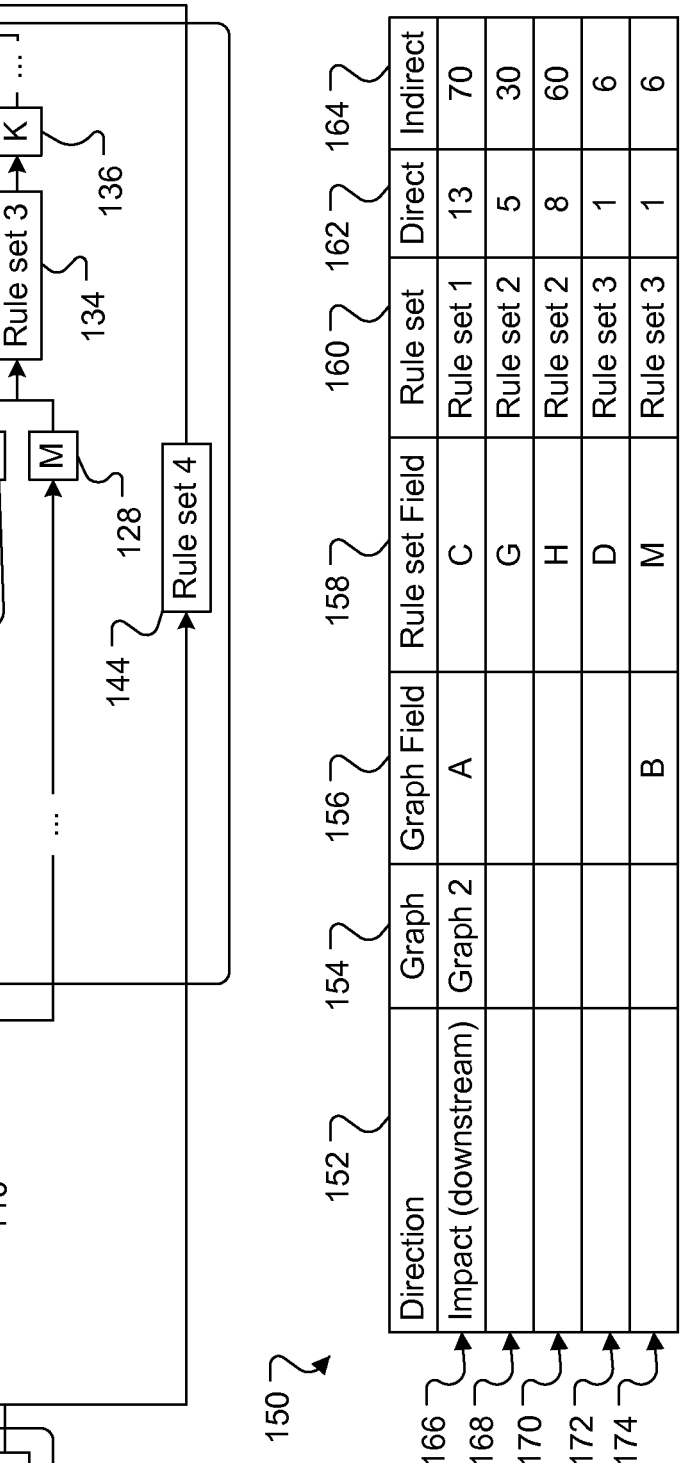
150
152
154
156
158
160
162
164
Direction
Graph
Graph Field
Rule set Field
Rule set
Direct
Indirect
Impact (downstream)
Graph 2
A
C
Rule set 1
13
70
G
Rule set 2
5
30
H
Rule set 2
8
60
D
Rule set 3
1
6
B
M
Rule set 3
1
6
166
168
170
172
174

FIG. 1

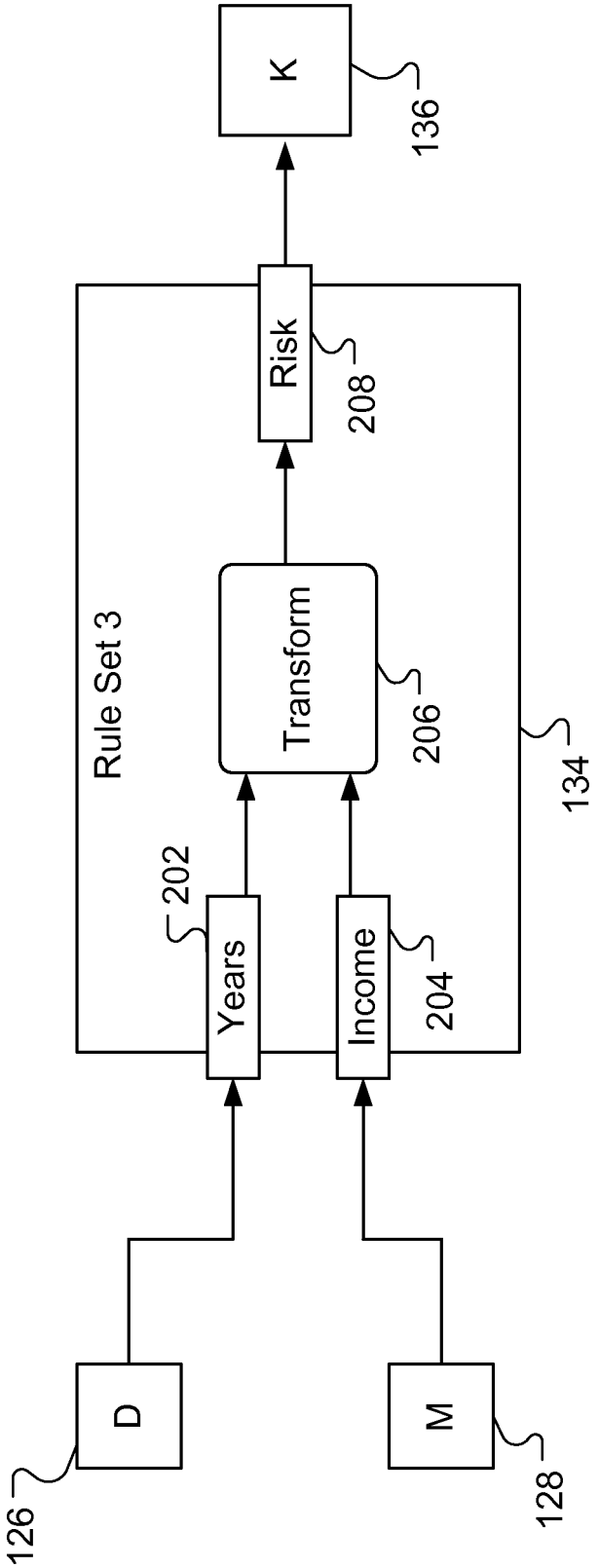
K
136
Risk
208
Rule Set 3
Transform
206
134
202
Years
Income
204
D
M
126
128

FIG. 2

300

	302 Input		304 Output
	306 Years	308 Income	310 Risk
312	>15	any	Low
314	any	>150,000	Low
316	>10	>60,000	Low
318	>5	any	Medium
320	any	>50,000	Medium
322	>3	>40,000	Low
324	any	any	High

FIG. 3

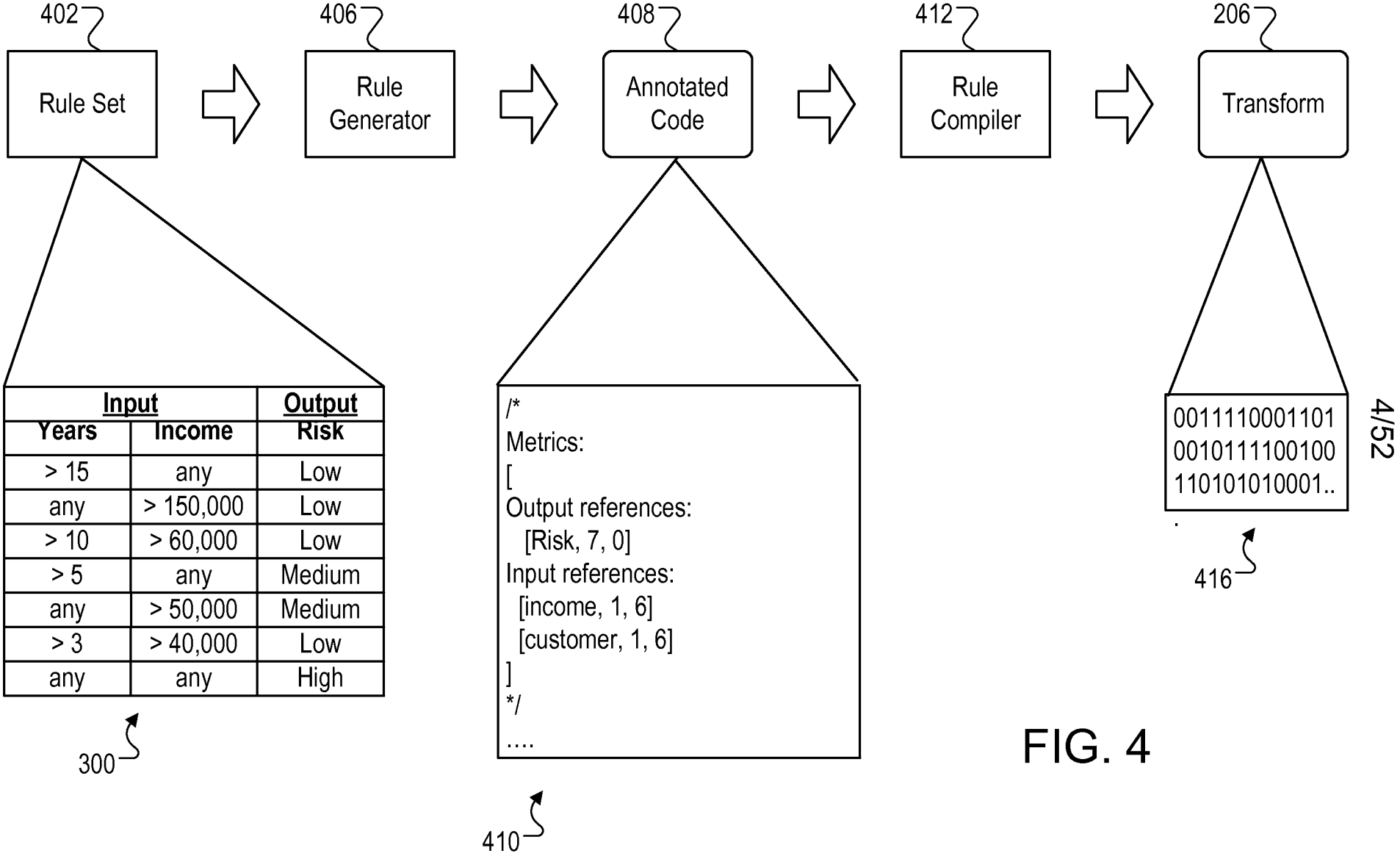
402
Rule Set
406
Rule Generator
408
Annotated Code
412
Rule Compiler
206
Transform
Input
Output
Years
Income
Risk
> 15 any Low
any > 150,000 Low
> 10 > 60,000 Low
> 5 any Medium
any > 50,000 Medium
> 3 > 40,000 Low
any any High
300
/*
Metrics:
[
Output references:
[Risk, 7, 0]
Input references:
[income, 1, 6]
[customer, 1, 6]
]
*/
....
410
0011110001101
0010111100100
110101010001..
416

FIG. 4

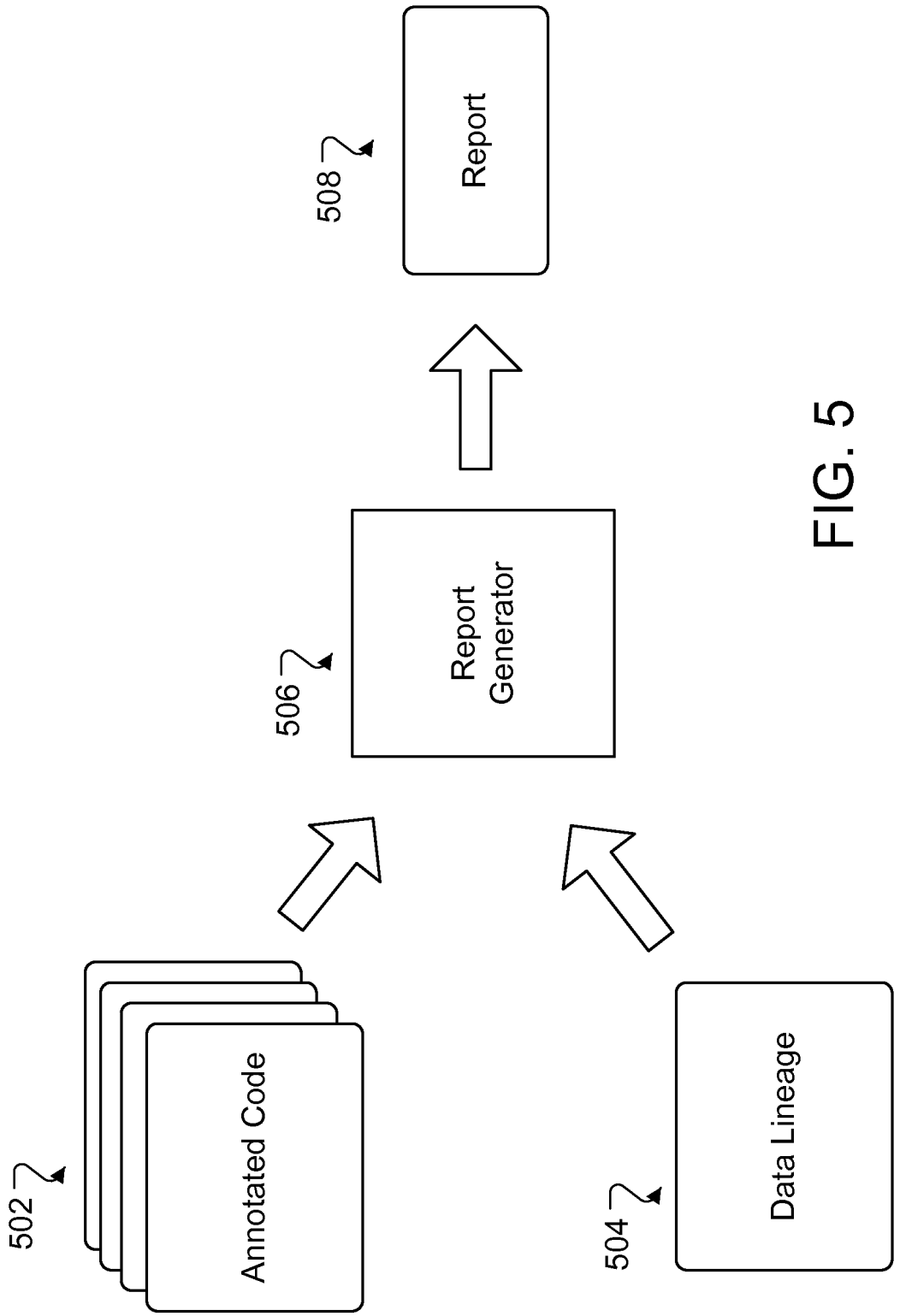
502
Annotated Code
504
Data Lineage
506
Report Generator
508
Report

FIG. 5

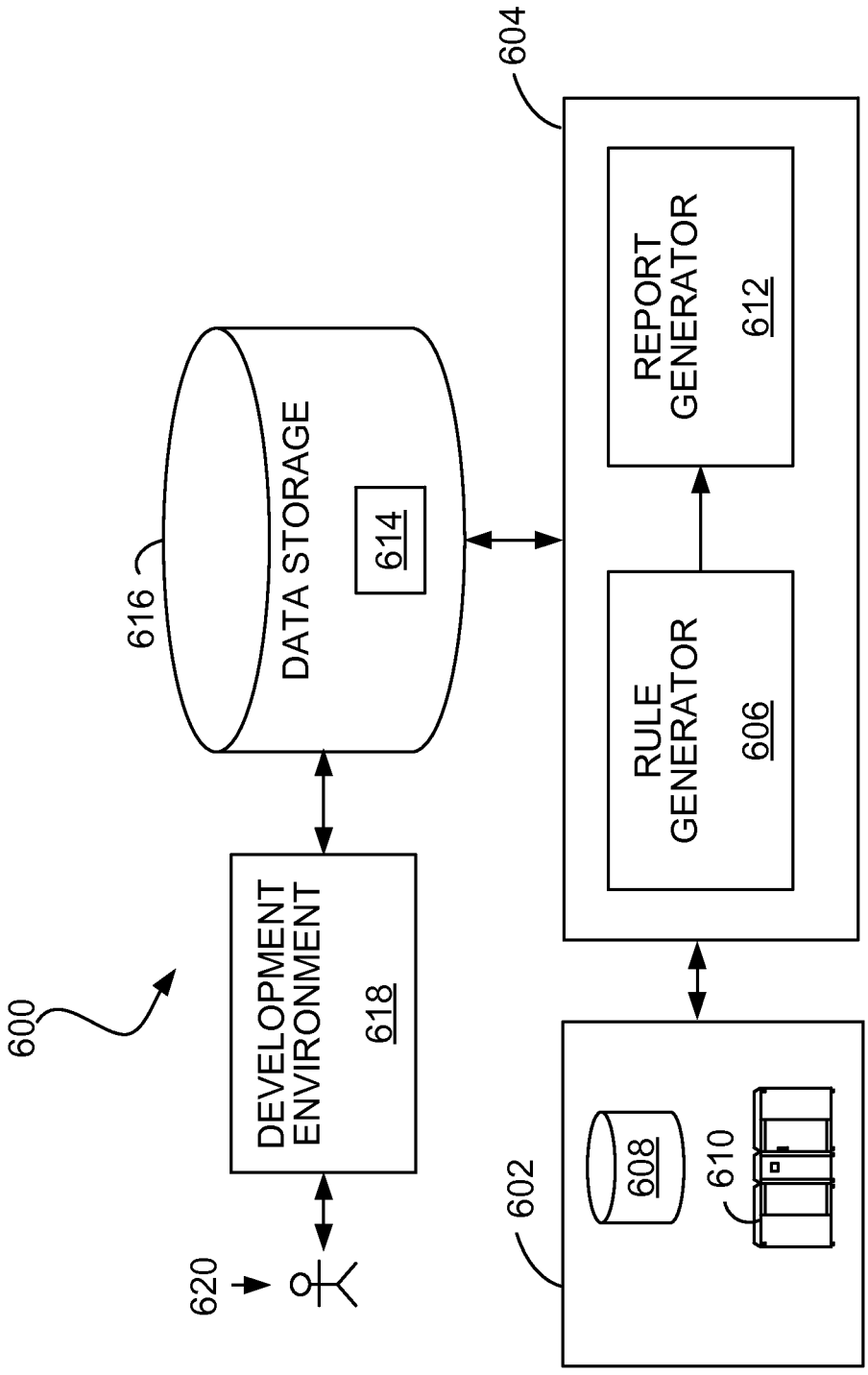
600
620
DEVELOPMENT ENVIRONMENT
618
616
DATA STORAGE
614
604
REPORT GENERATOR
612
RULE GENERATOR
606
602
608
610

FIG. 6

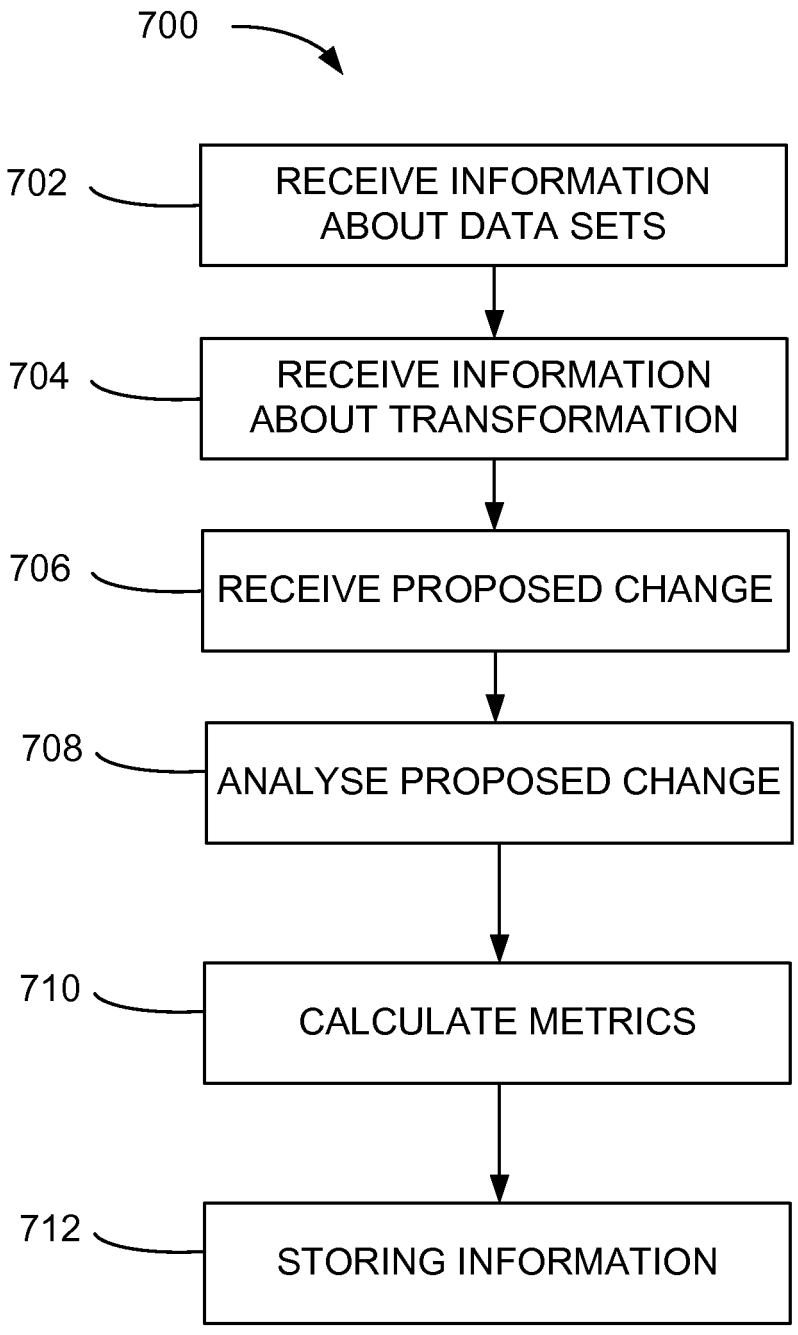
700
702
RECEIVE INFORMATION ABOUT DATA SETS
704
RECEIVE INFORMATION ABOUT TRANSFORMATION
706
RECEIVE PROPOSED CHANGE
708
ANALYSE PROPOSED CHANGE
710
CALCULATE METRICS
712
STORING INFORMATION

FIG. 7

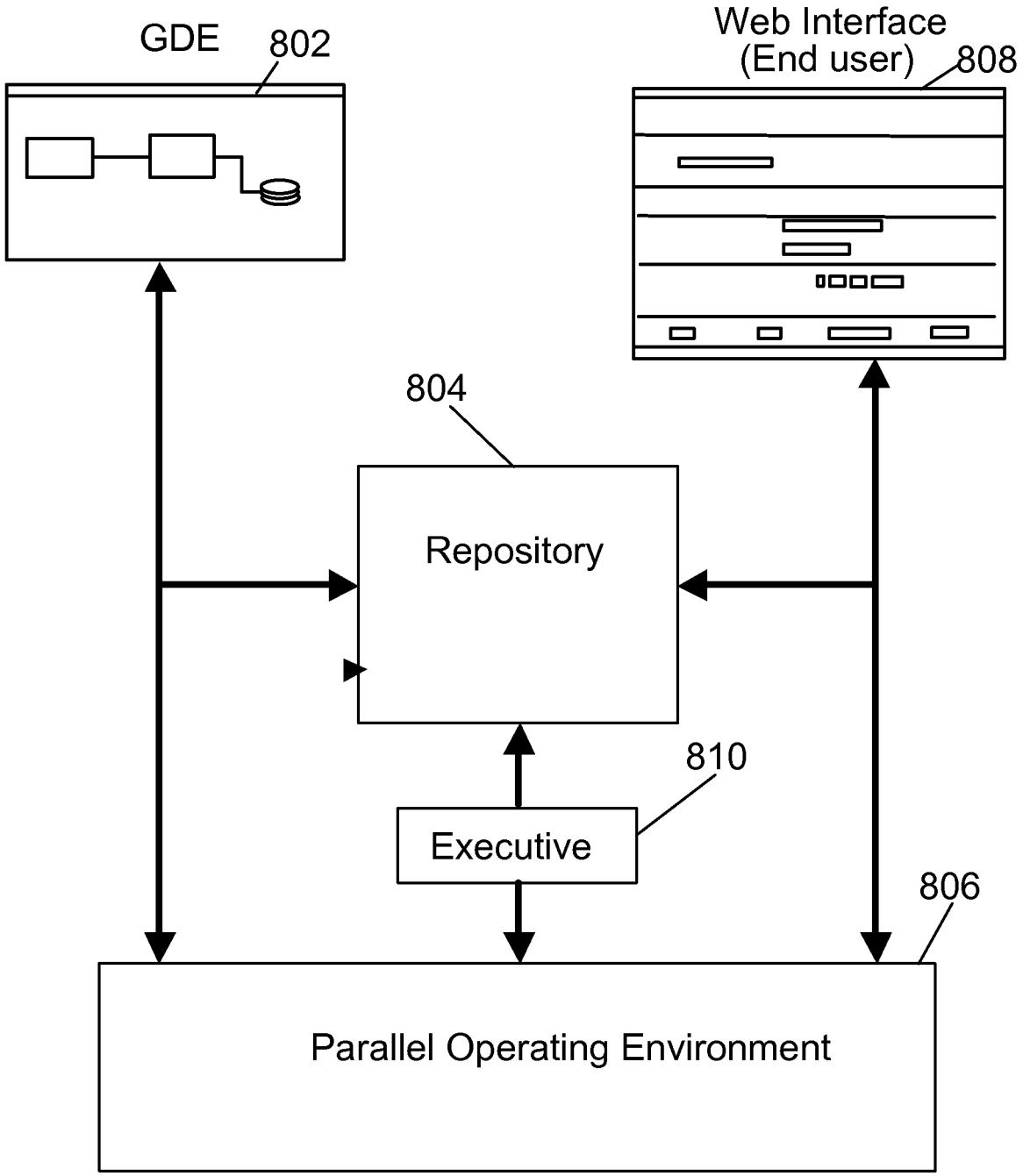
GDE
802
Web Interface
(End user)
808
804
Repository
810
Executive
806
Parallel Operating Environment

FIG. 8A

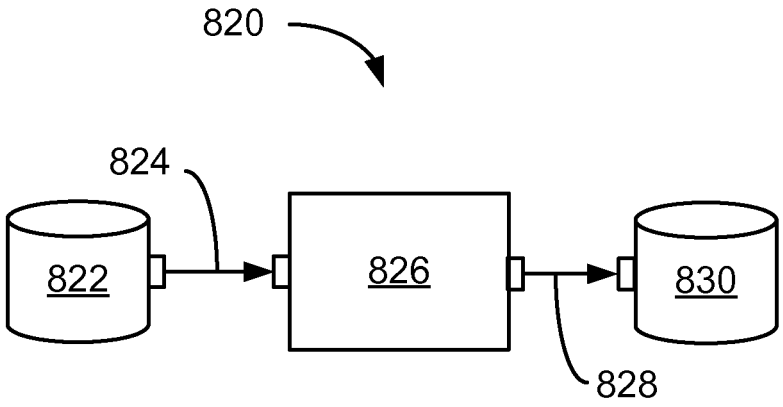
820
824
822
826
830
828

FIG. 8B

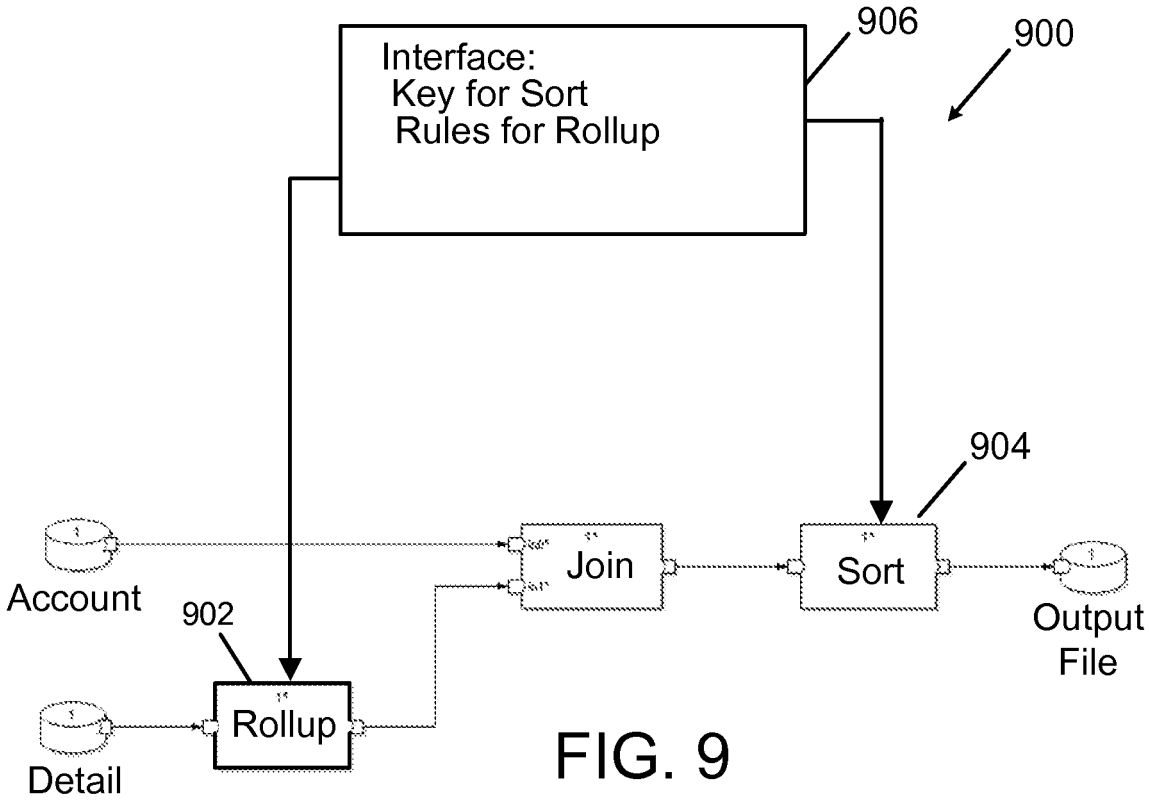
Interface:
Key for Sort
Rules for Rollup
906
900
904
Account
902
Join
Sort
Output
File
Rollup
Detail

FIG. 9

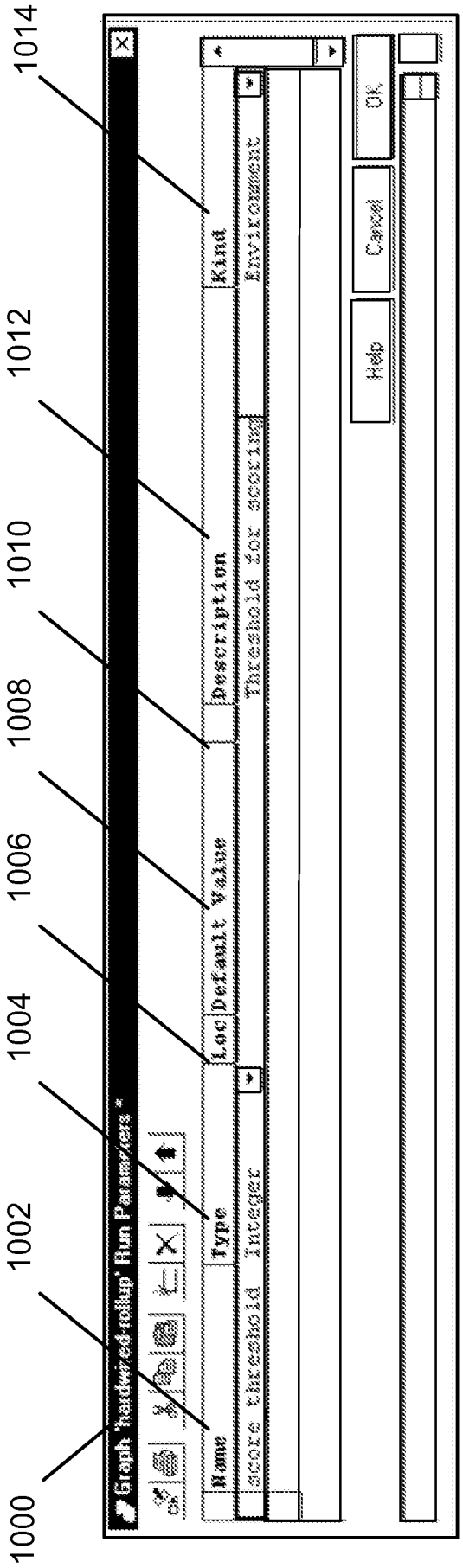
1014
1012
1010
1008
1006
1004
1002
1000
Graph 'hardwired rollup' Run Parameters *
Name
Type
Loc
Default Value
Description
Kind
Environment
score threshold
Integer
Threshold for scoring
OK
Cancel
Help

FIG. 10

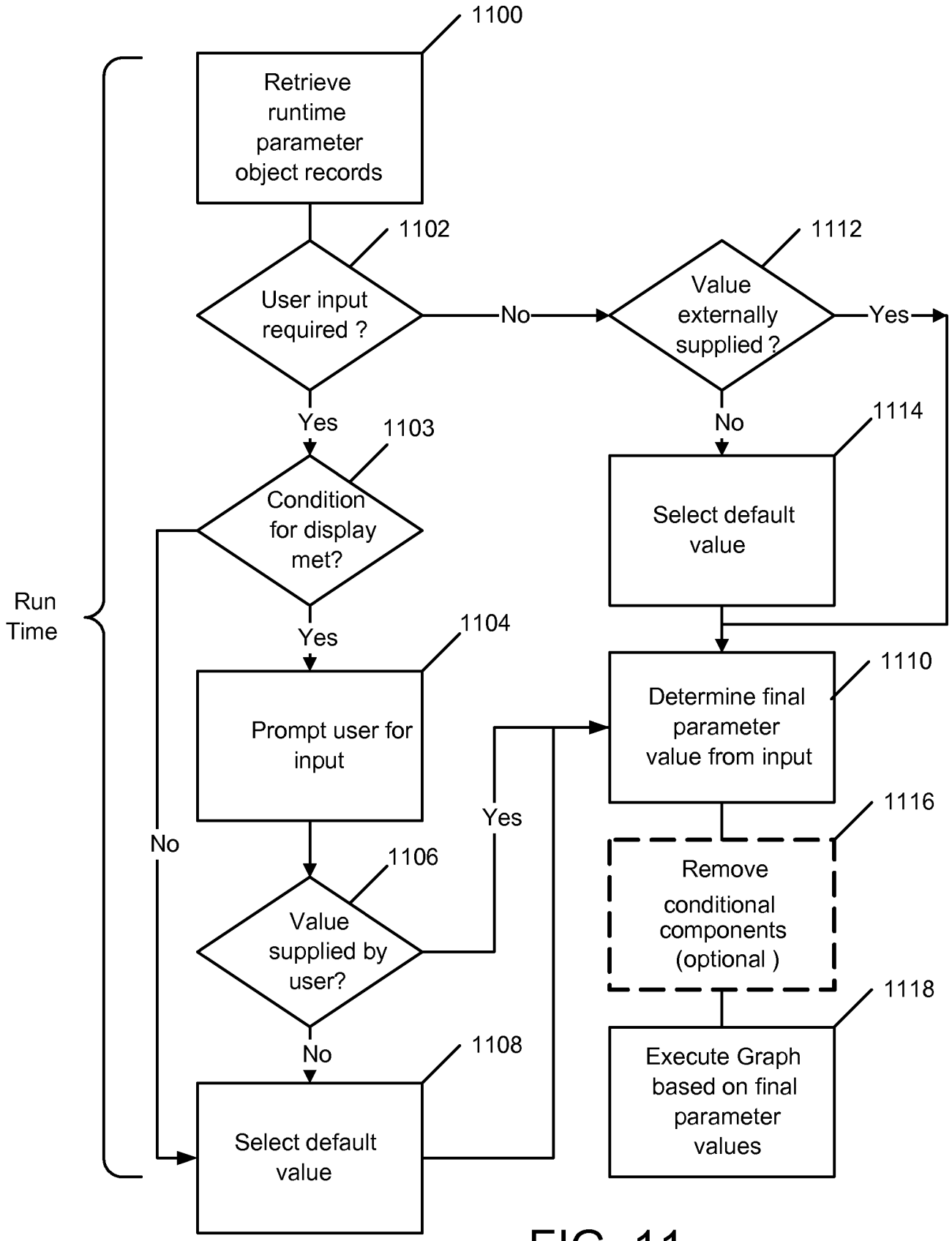
1100
Retrieve runtime parameter object records
1102
User input required ?
No
1112
Value externally supplied ?
Yes
Yes
1103
Condition for display met?
No
1114
Select default value
Run Time
Yes
1104
Prompt user for input
1110
Determine final parameter value from input
Yes
No
1106
Value supplied by user?
1116
Remove conditional components (optional)
No
1108
Select default value
1118
Execute Graph based on final parameter values

FIG. 11

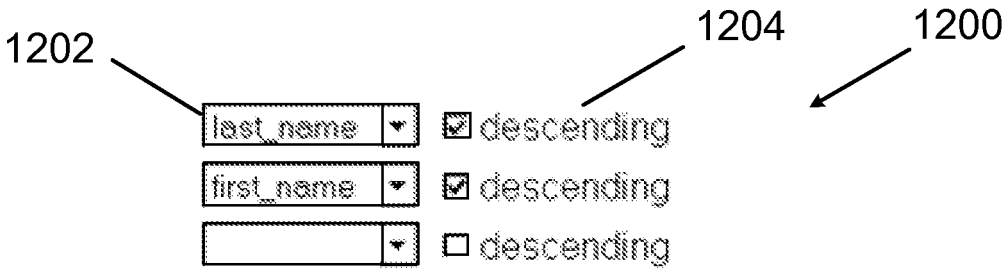
1202
1204
1200
last_name
descending
first_name
descending
descending

FIG. 12

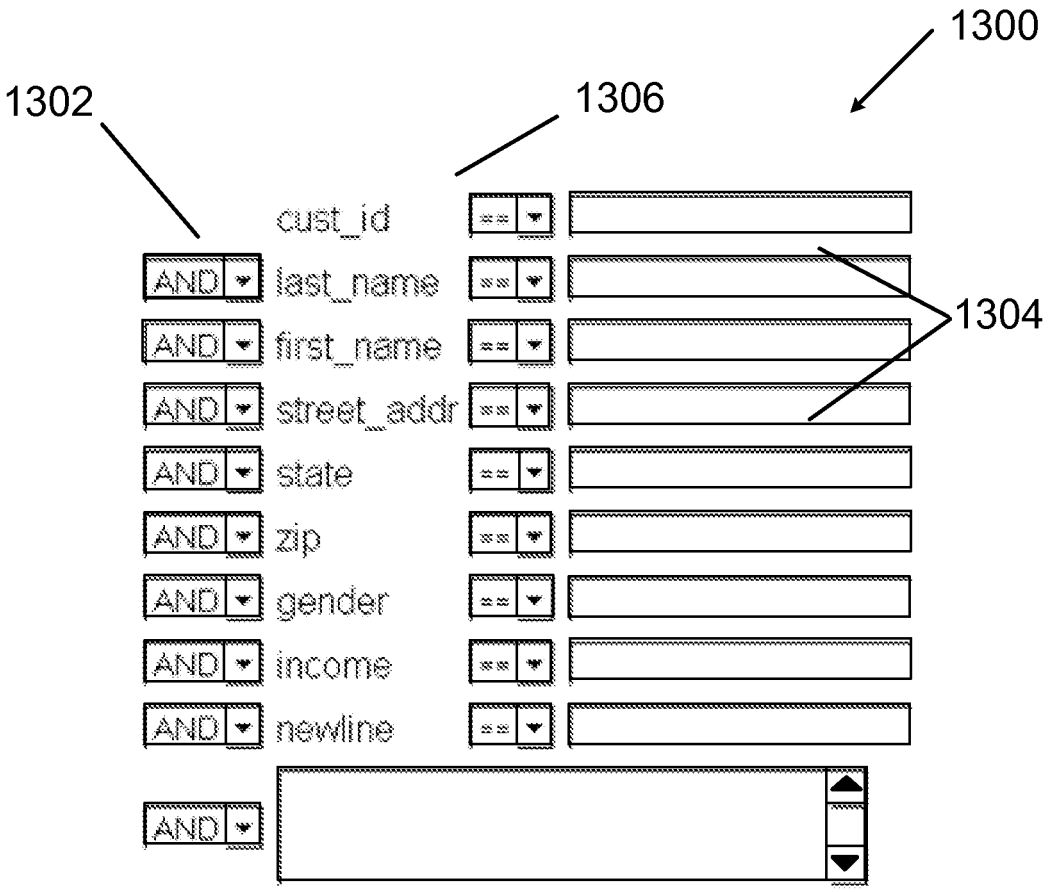
1300
1302
1306
1304
cust_id
AND
last_name
AND
first_name
AND
street_addr
AND
state
AND
zip
AND
gender
AND
income
AND
newline
AND

FIG. 13

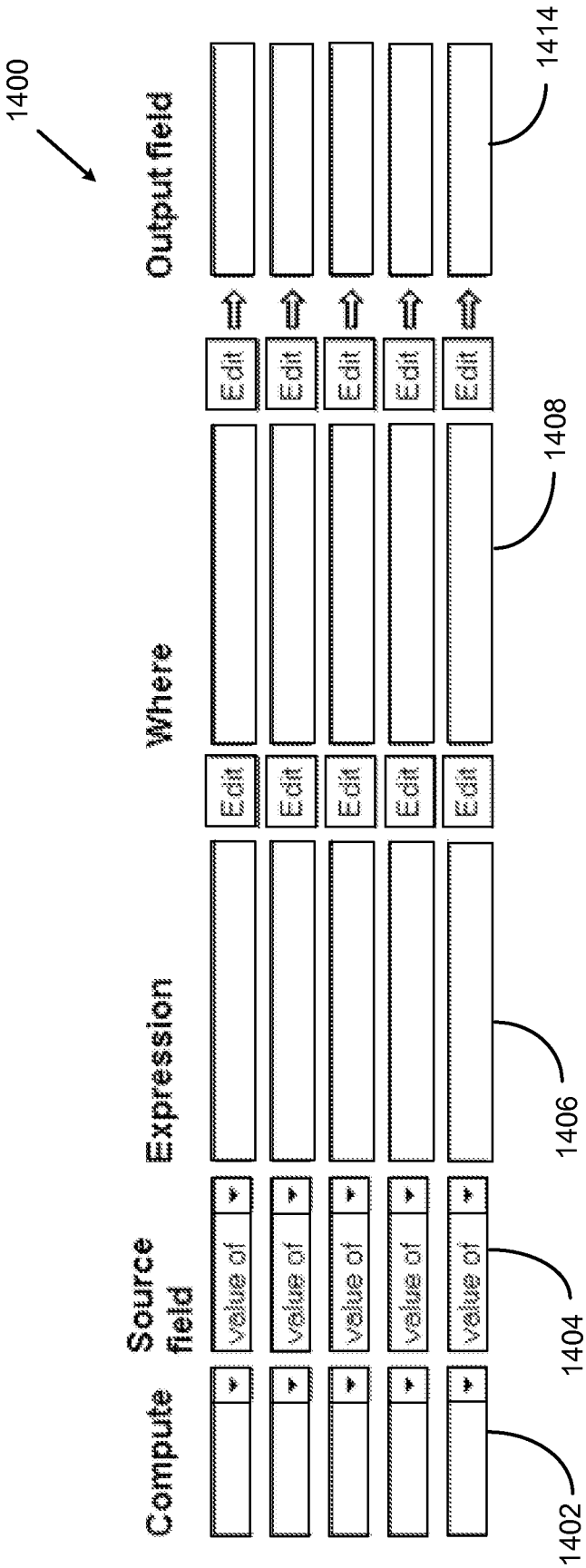
1400
Compute
Source field
Expression
Where
Output field
value of
Edit
1402
1404
1406
1408
1414

FIG. 14

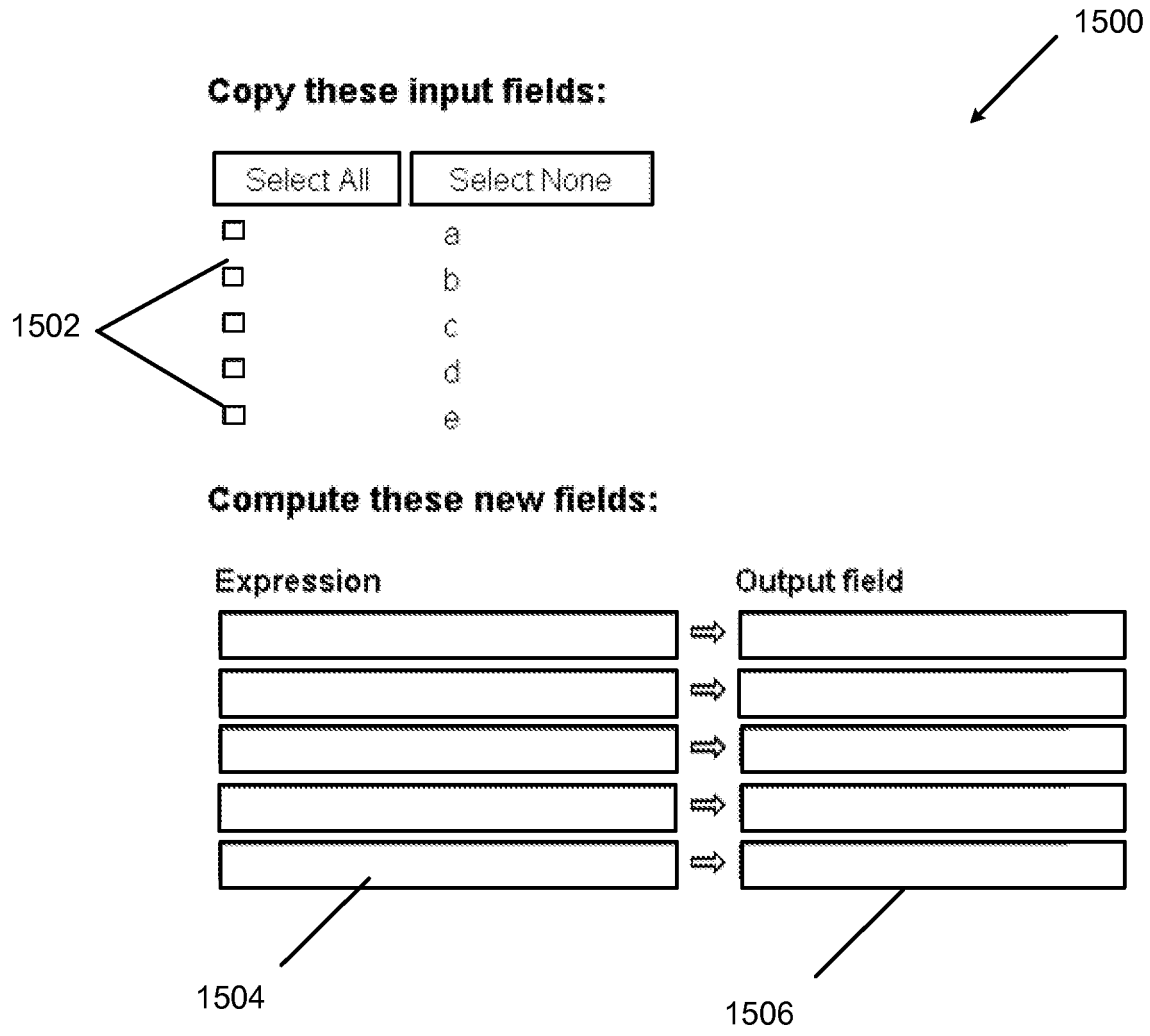
1500
Copy these input fields:
Select All
Select None
a
b
c
d
e
1502
Compute these new fields:
Expression
Output field
1504
1506

FIG. 15

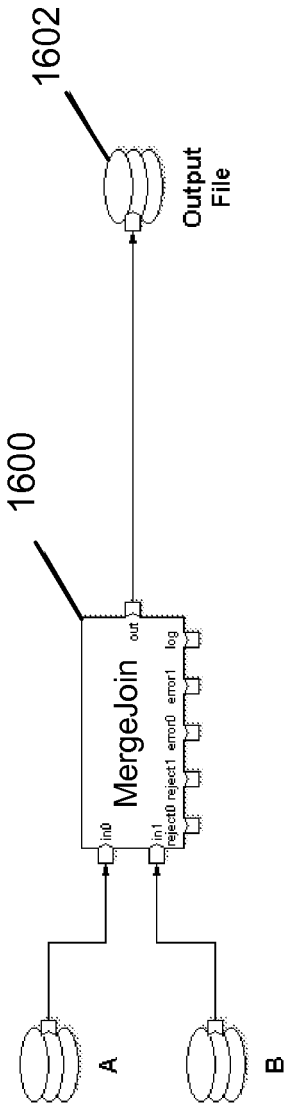

FIG. 16A
Join A and B

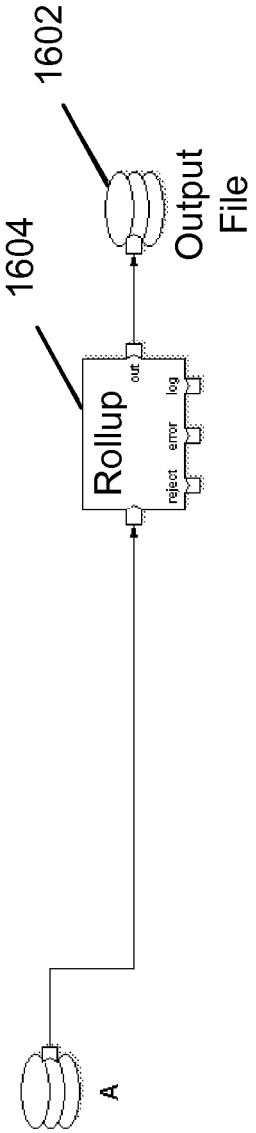

FIG. 16B
Aggregate A

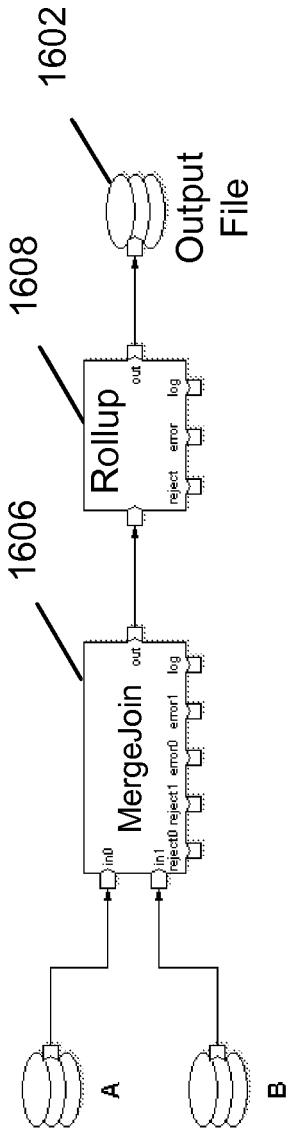

FIG. 16C
Join A & B, then aggregate

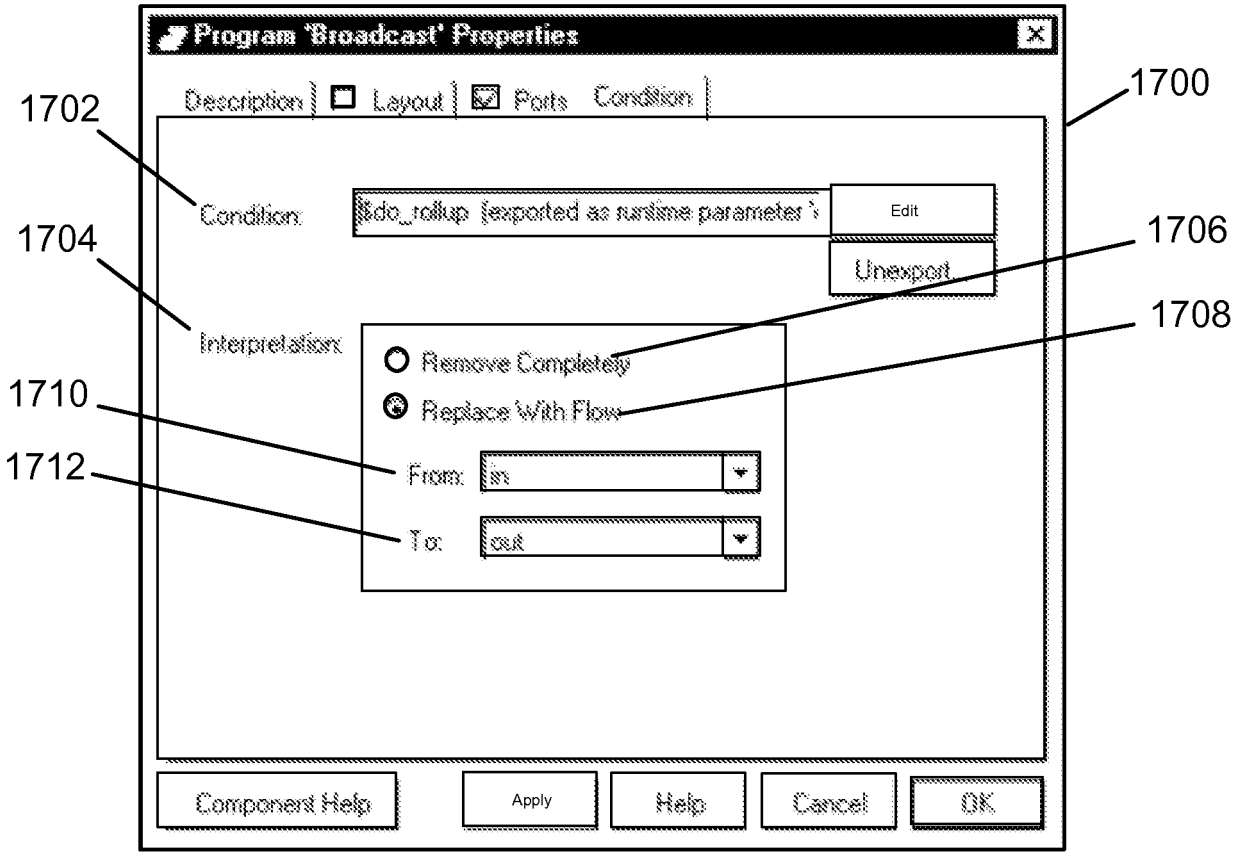
Program 'Broadcast' Properties
Description
Layout
Ports
Condition
Condition:
$do_rollup (exported as runtime parameter)
Edit
Unexport
Interpretation:
Remove Completely
Replace With Flow
From:
in
To:
out
Component Help
Apply
Help
Cancel
OK
1700
1702
1704
1706
1708
1710
1712

FIG. 17

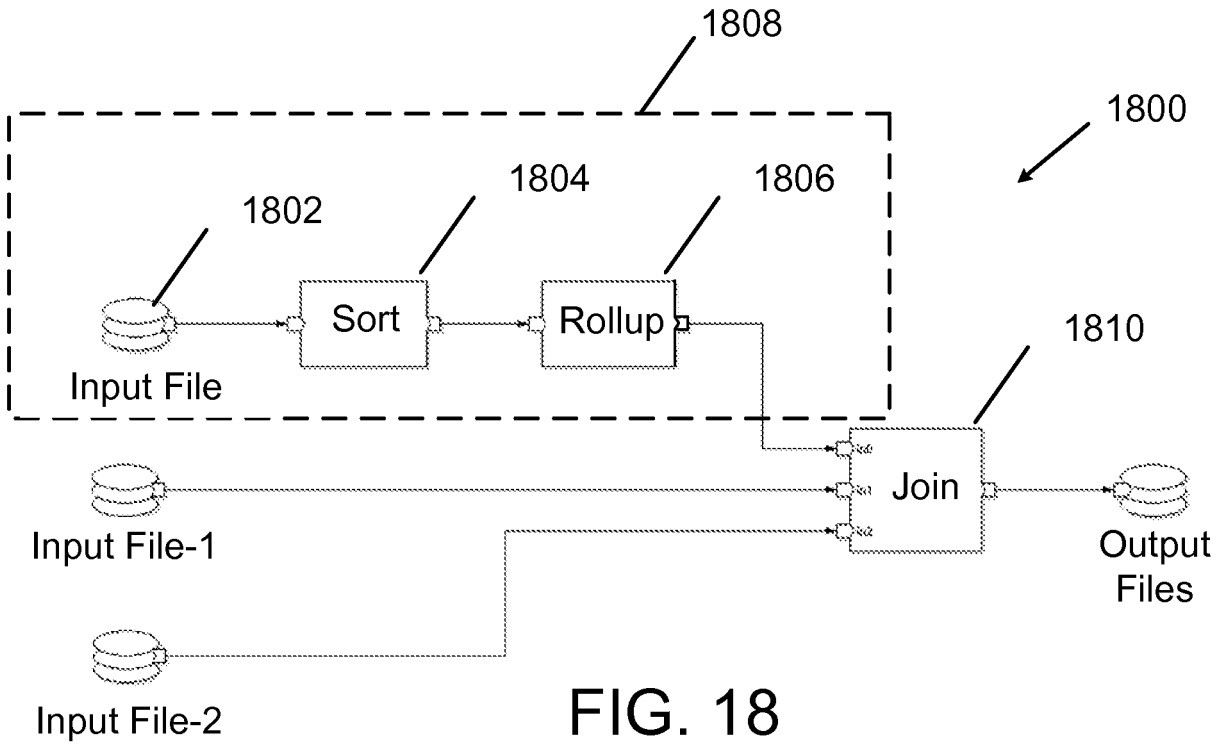
1808
1800
1802
1804
1806
Sort
Rollup
Input File
1810
Join
Input File-1
Output Files
Input File-2

FIG. 18

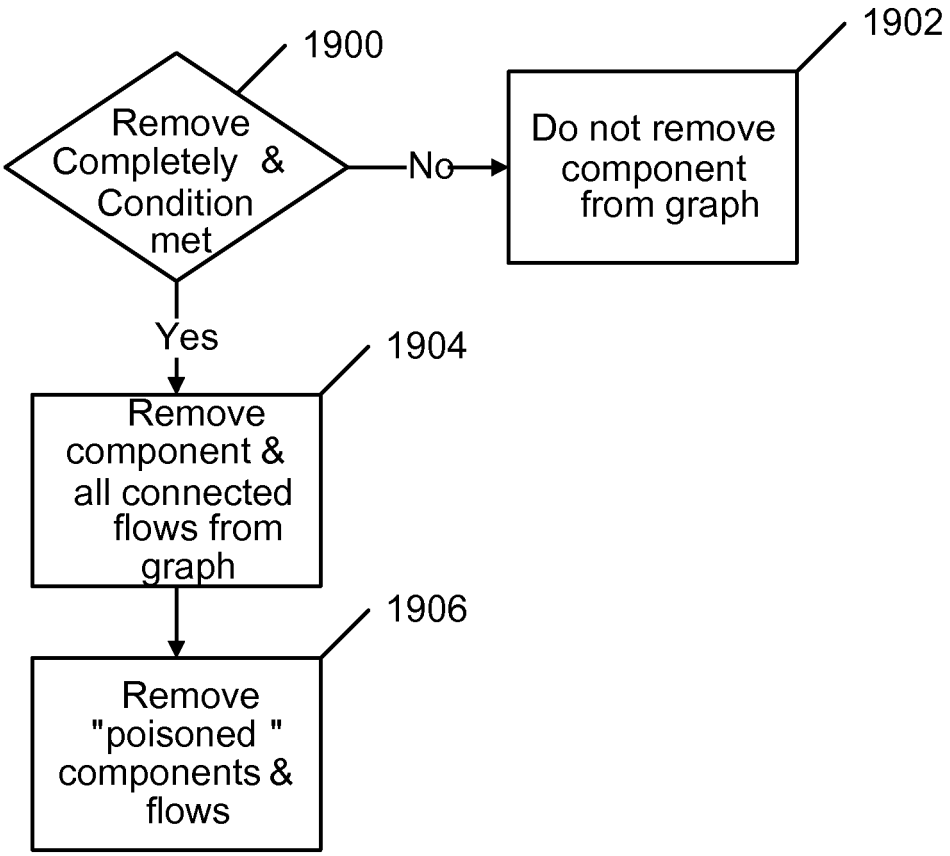
1900
Remove Completely & Condition met
No
1902
Do not remove component from graph
Yes
1904
Remove component & all connected flows from graph
1906
Remove "poisoned " components & flows

FIG. 19

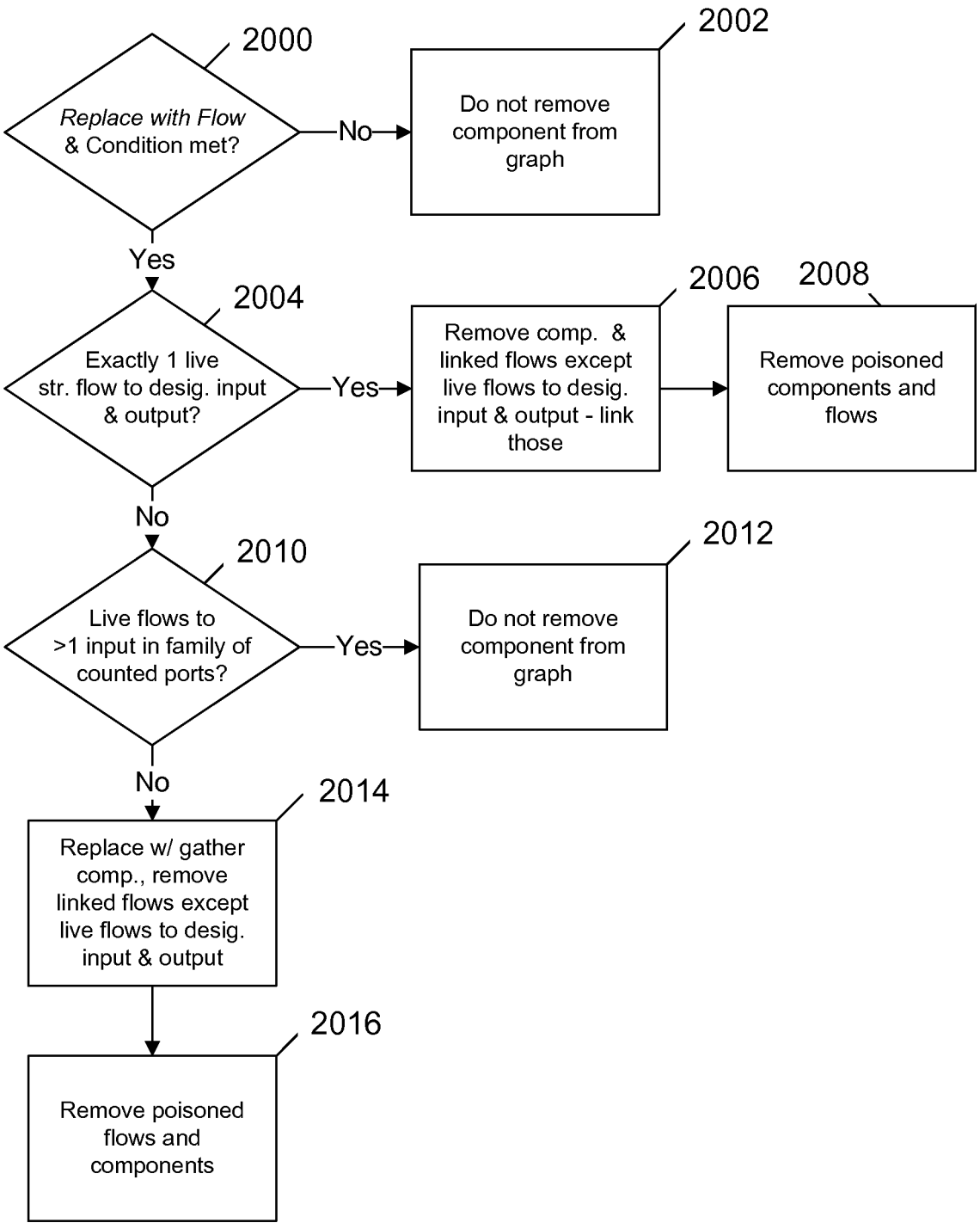
2000
Replace with Flow & Condition met?
No
2002
Do not remove component from graph
Yes
2004
Exactly 1 live str. flow to desig. input & output?
Yes
2006
Remove comp. & linked flows except live flows to desig. input & output - link those
2008
Remove poisoned components and flows
No
2010
Live flows to >1 input in family of counted ports?
Yes
2012
Do not remove component from graph
No
2014
Replace w/ gather comp., remove linked flows except live flows to desig. input & output
2016
Remove poisoned flows and components

FIG. 20

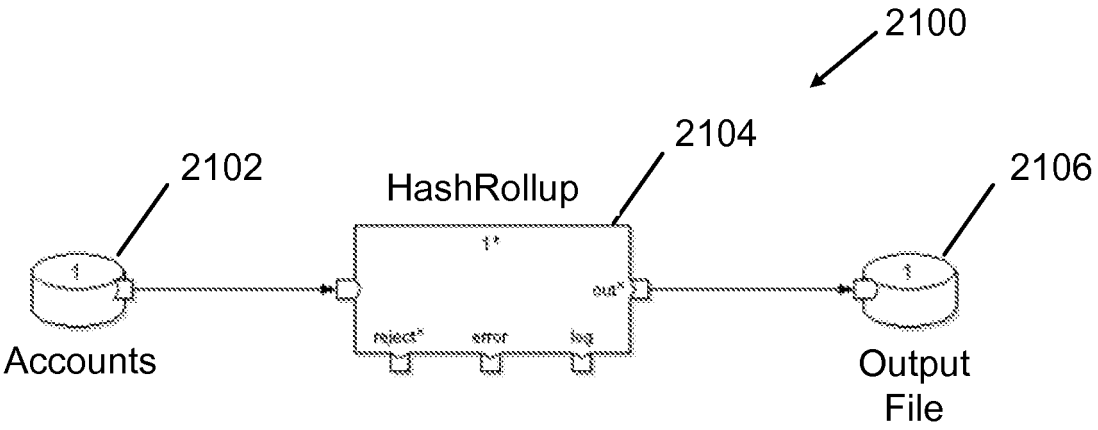
2100
2102
HashRollup
2104
2106
Accounts
Output
File

FIG. 21

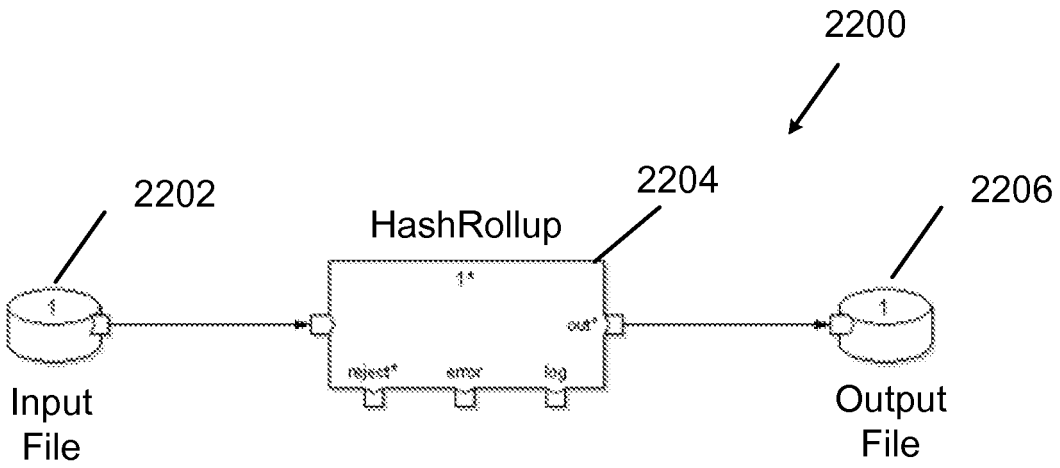
2200
2202
HashRollup
2204
2206
Input
File
Output
File

FIG. 22

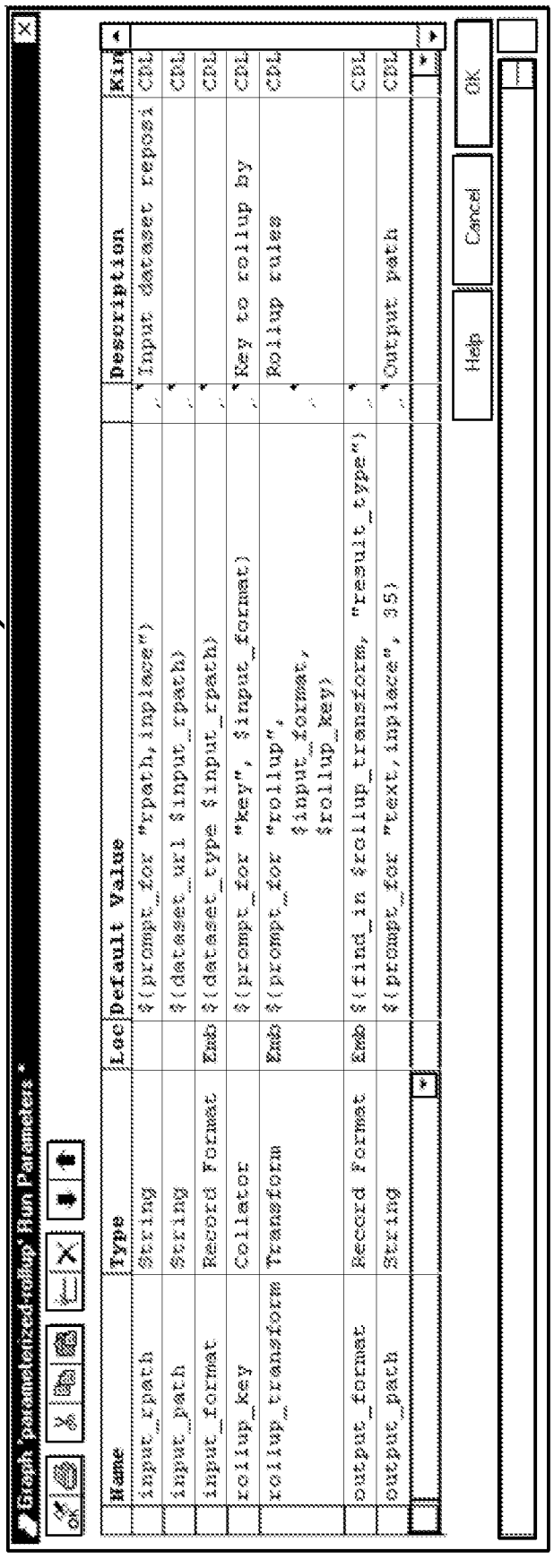
2300
Graph 'parameterized rollup' Run Parameters *
Name
Type
Loc
Default Value
Description
Kin
input_xpath
String
$(prompt_for "xpath,inplace")
Input dataset reposi
CTL
input_path
String
$(dataset_url $input_xpath)
CTL
input_format
Record Format
Emb
$(dataset_type $input_xpath)
CTL
rollup_key
Collator
$(prompt_for "key", $input_format)
Key to rollup by
CTL
rollup_transform
Transform
Emb
$(prompt_for "rollup",
$input_format,
$rollup_key)
Rollup rules
CTL
output_format
Record Format
Emb
$(find_in $rollup_transform, "result_type")
CTL
output_path
String
$(prompt_for "text,inplace", 35)
Output path
CTL
Help
Cancel
OK

FIG. 23

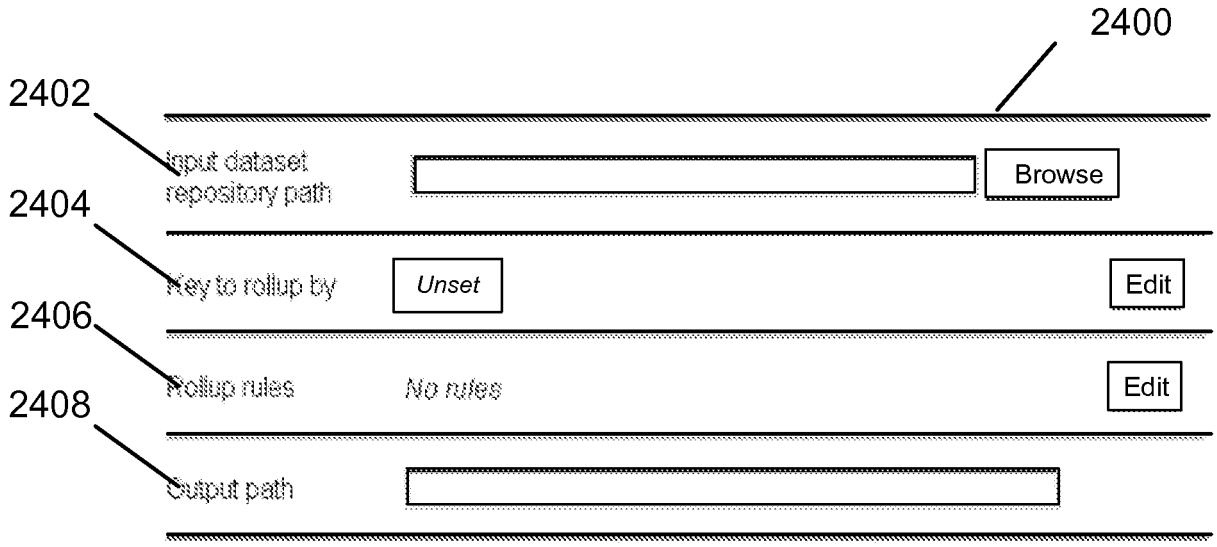

FIG. 24A

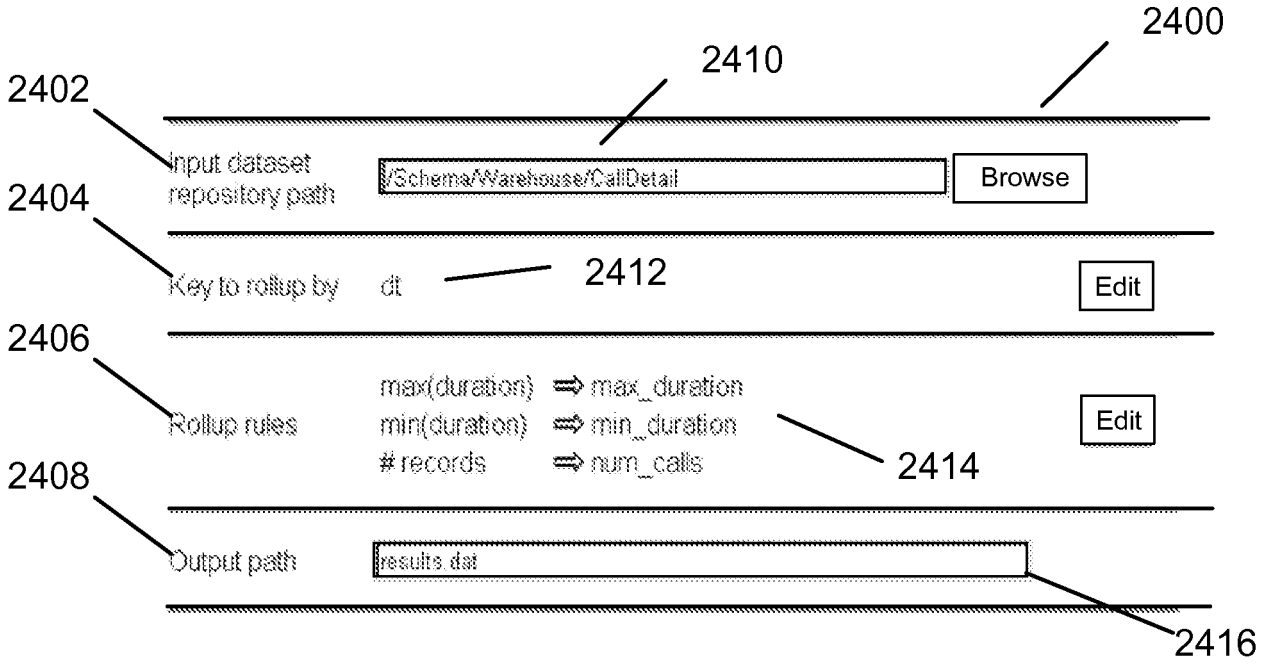

FIG. 24B

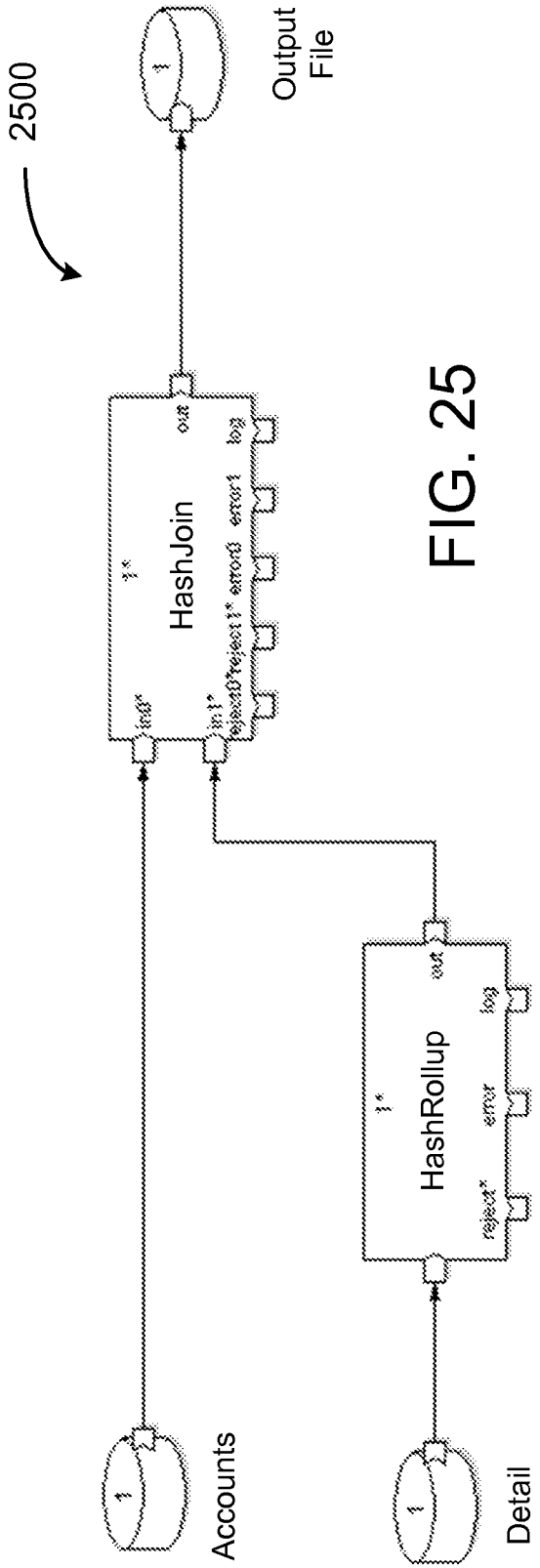

FIG. 25

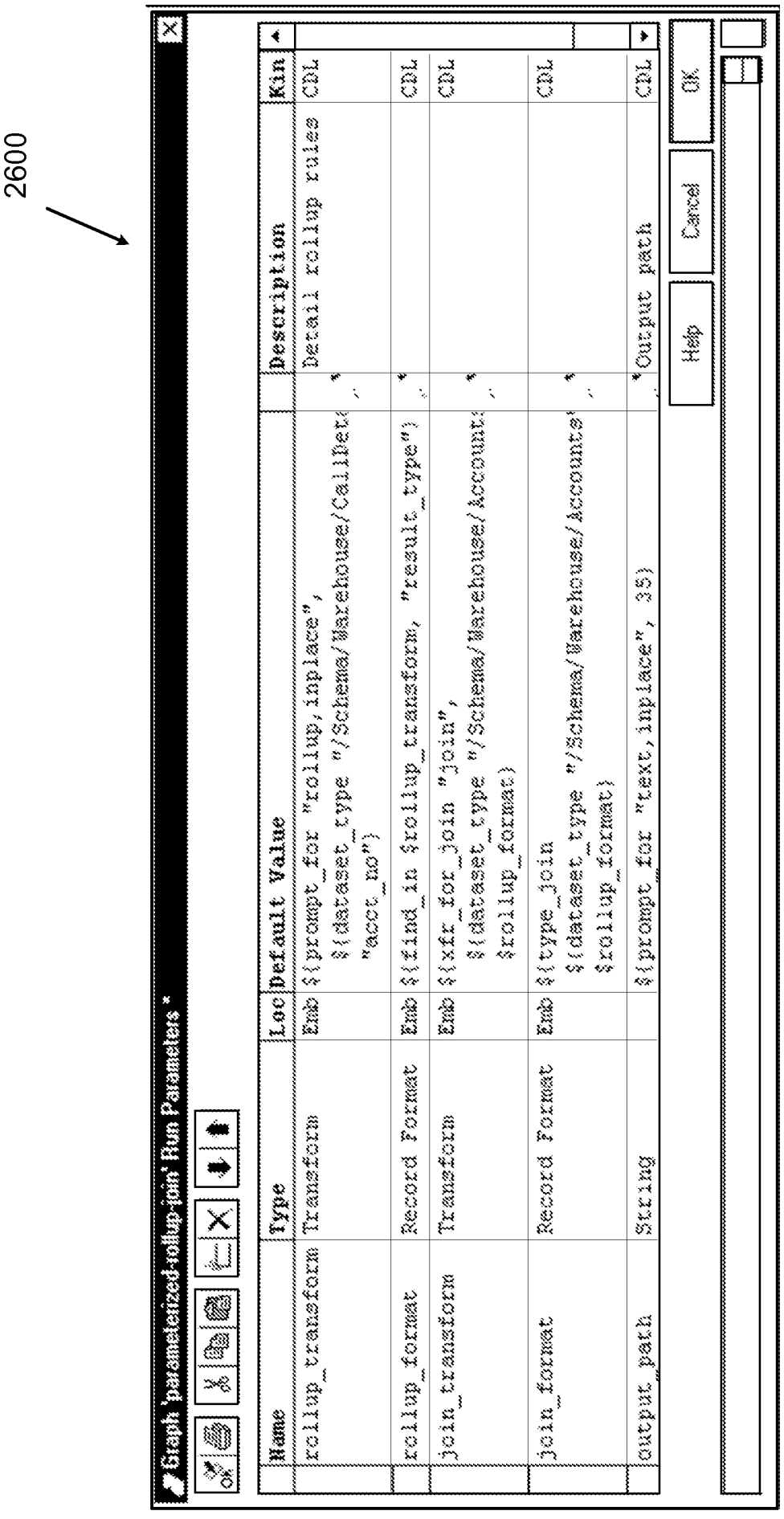
2600
Graph 'parameterized-rollup-join' Run Parameters *
Name
Type
Loc
Default Value
Description
Kin
rollup_transform
Transform
Emb
${prompt_for "rollup,inplace",
${dataset_type "/Schema/Warehouse/CallDet
"acct_no"}
Detail rollup rules
CDL
rollup_format
Record Format
Emb
${find_in $rollup_transform, "result_type"}
CDL
join_transform
Transform
Emb
${xfr_for_join "join",
${dataset_type "/Schema/Warehouse/Account
$rollup_format}
CDL
join_format
Record Format
Emb
${type_join
${dataset_type "/Schema/Warehouse/Accounts
$rollup_format}
CDL
output_path
String
${prompt_for "text,inplace", 35}
Output path
CDL
Help
Cancel
OK

FIG. 26

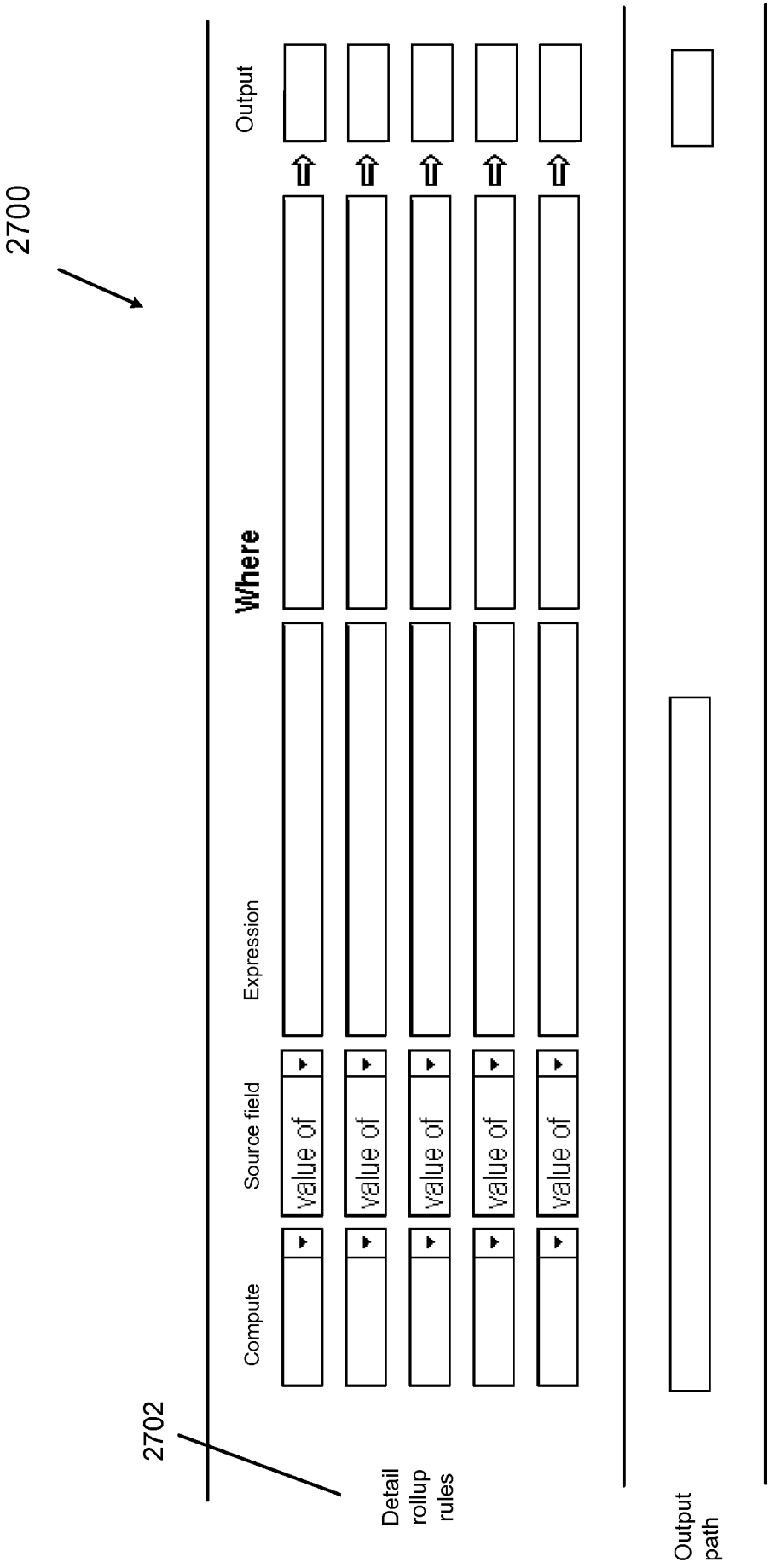
2700
2702
Compute
Source field
Expression
Where
Output
value of
value of
value of
value of
value of
Detail rollup rules
Output path

FIG. 27

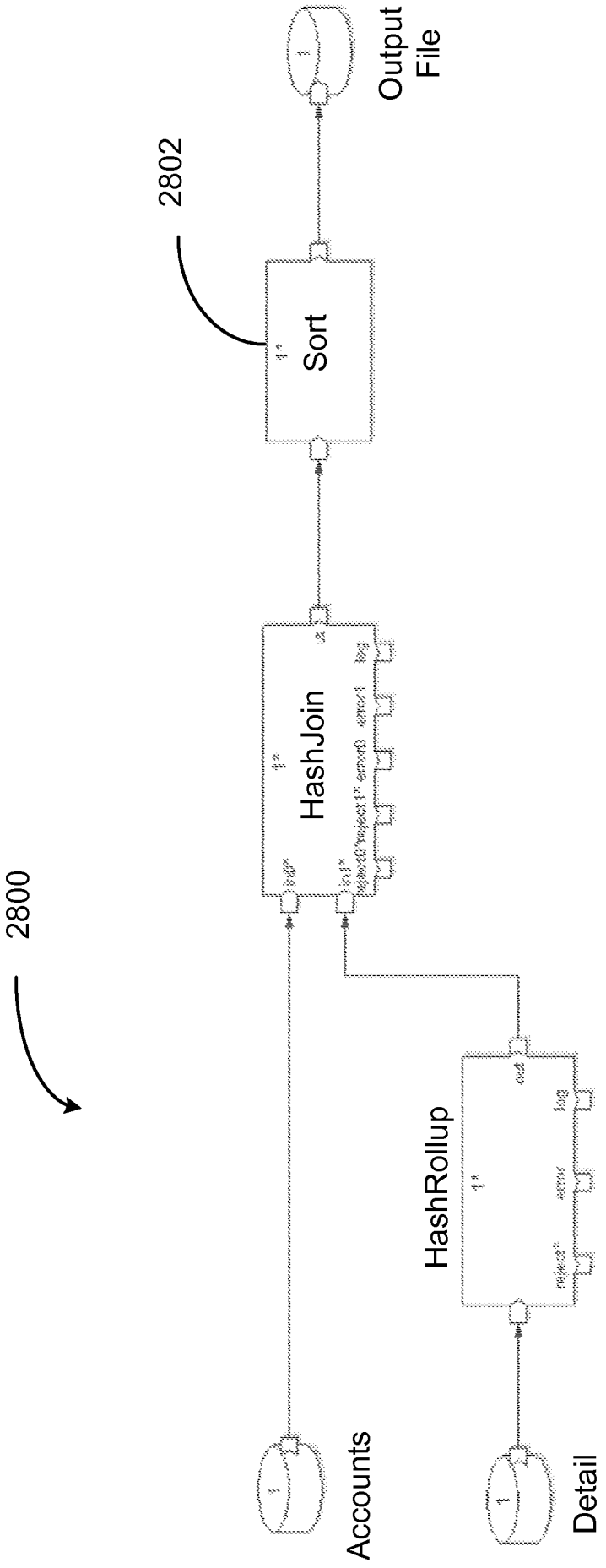
2800
2802
Output File
Sort
HashJoin
HashRollup
Accounts
Detail

FIG. 28

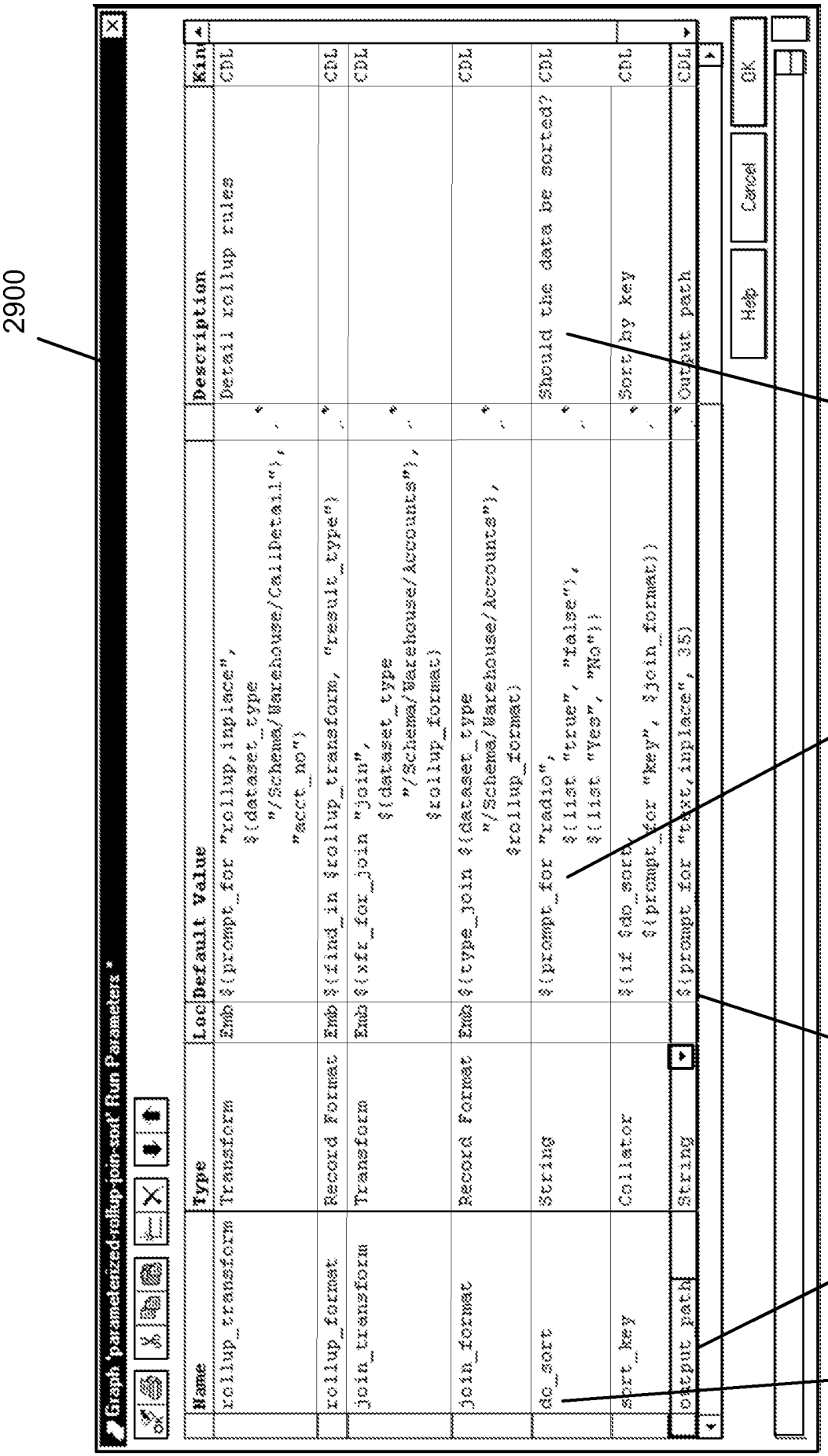
2900
Graph 'parameterized-rollup-join-sort' Run Parameters *
Name
Type
Loc
Default Value
Description
Kin
rollup_transform
Transform
Emb
${prompt_for "rollup,inplace",
${dataset_type
"/Schema/Warehouse/CallDetail"},
"acct_no"}
Detail rollup rules
CDL
rollup_format
Record Format
Emb
${find_in $rollup_transform, "result_type"}
CDL
join_transform
Transform
Emb
${xfr_for_join "join",
${dataset_type
"/Schema/Warehouse/Accounts"},
$rollup_format}
CDL
join_format
Record Format
Emb
${type_join ${dataset_type
"/Schema/Warehouse/Accounts"},
$rollup_format}
CDL
do_sort
String
${prompt_for "radio",
${list "true", "false"},
${list "Yes", "No"}}
Should the data be sorted?
CDL
sort_key
Collator
${if $do_sort
${prompt_for "key", $join_format}}
Sort by key
CDL
output_path
String
${prompt_for "take,inplace", 35}
Output path
CDL
Help
Cancel
OK
2902
2904
2906
2908
2910

FIG. 29

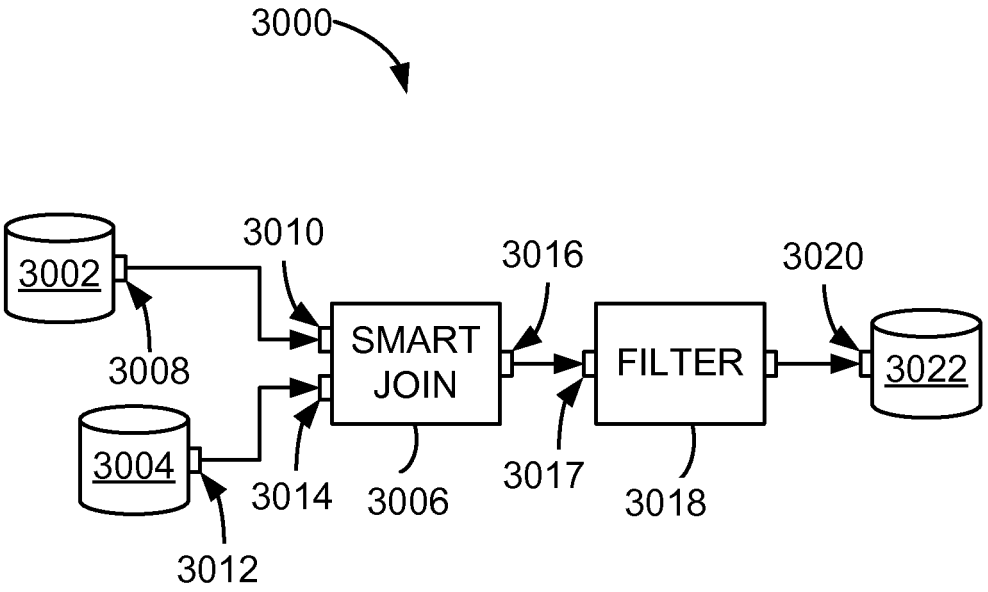

FIG. 30A

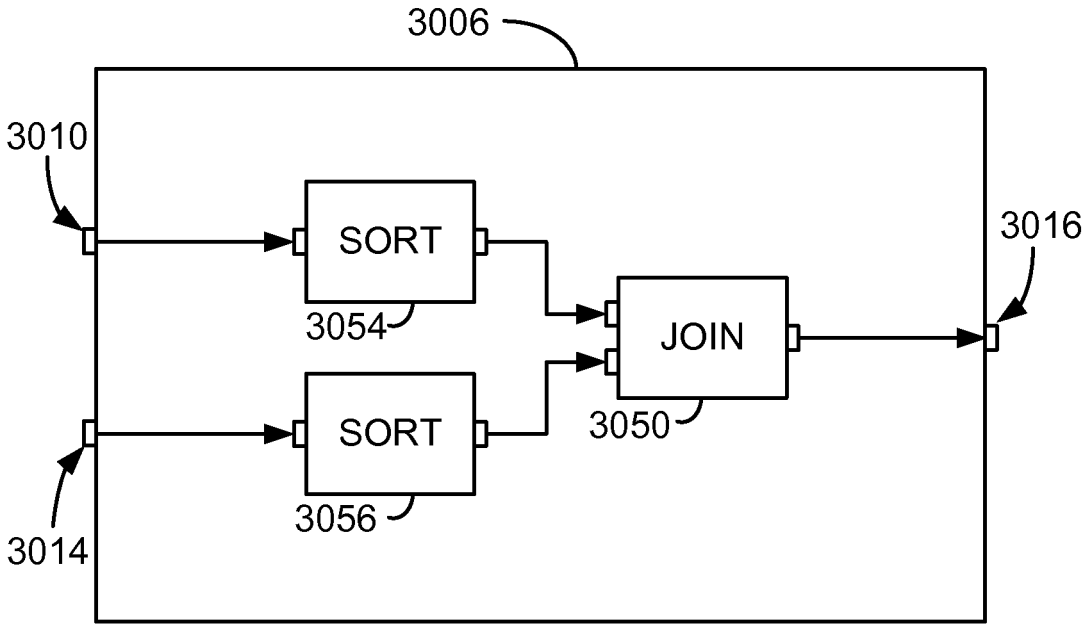

FIG. 30B

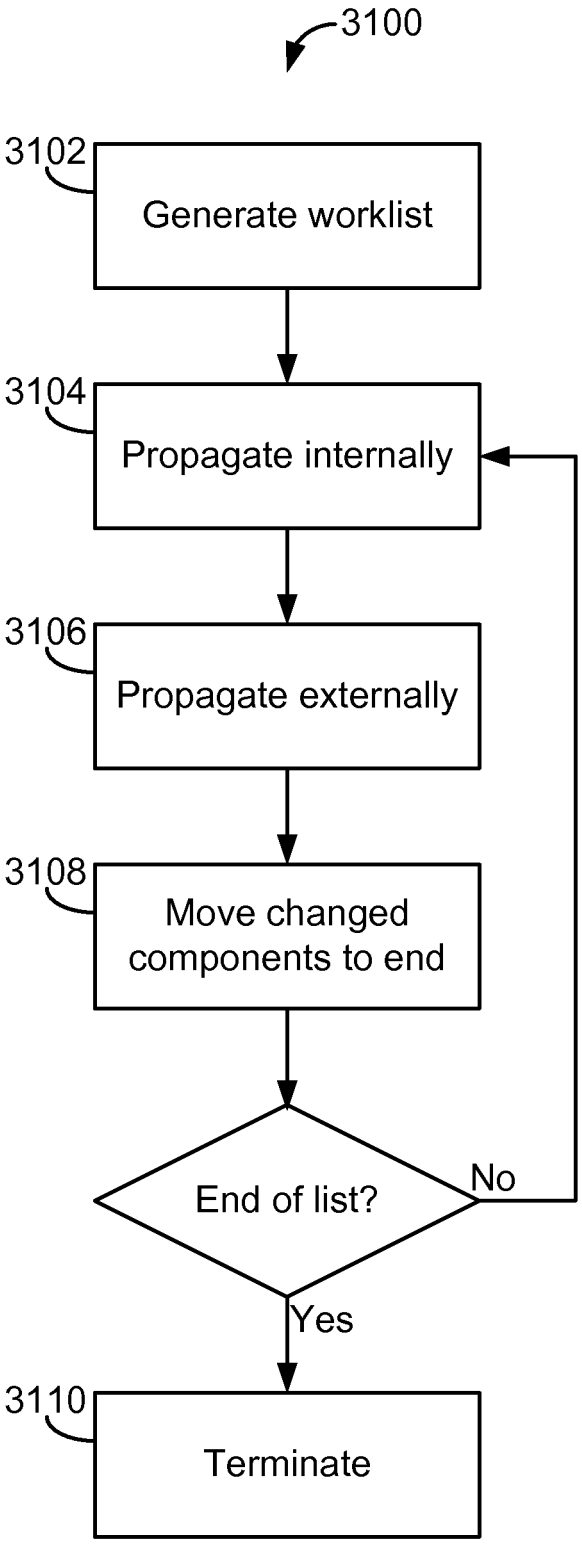
3100
3102
Generate worklist
3104
Propagate internally
3106
Propagate externally
3108
Move changed components to end
End of list?
No
Yes
3110
Terminate

FIG. 31

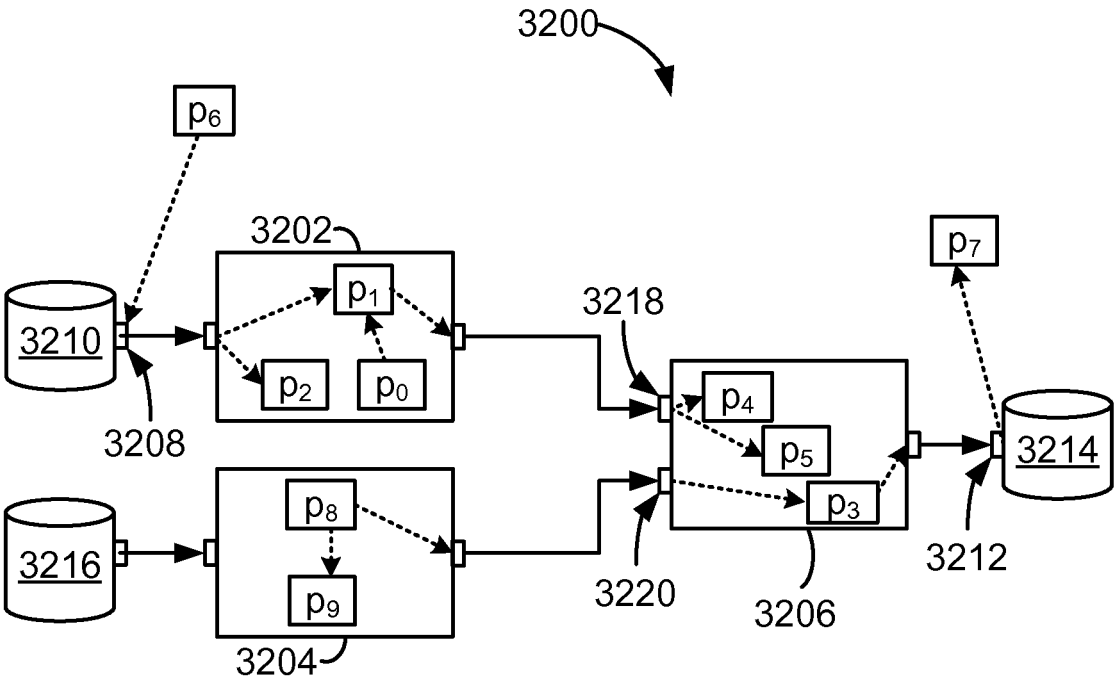

FIG. 32A

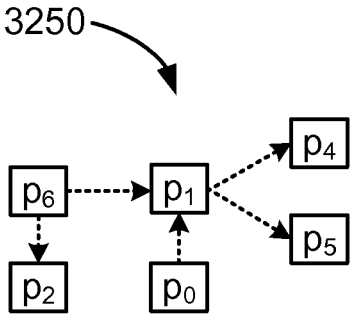

FIG. 32B

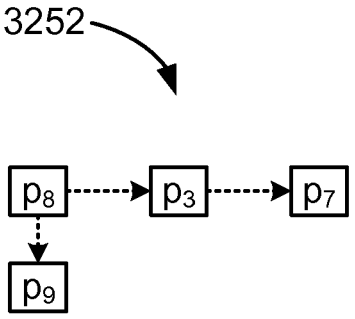

FIG. 32C

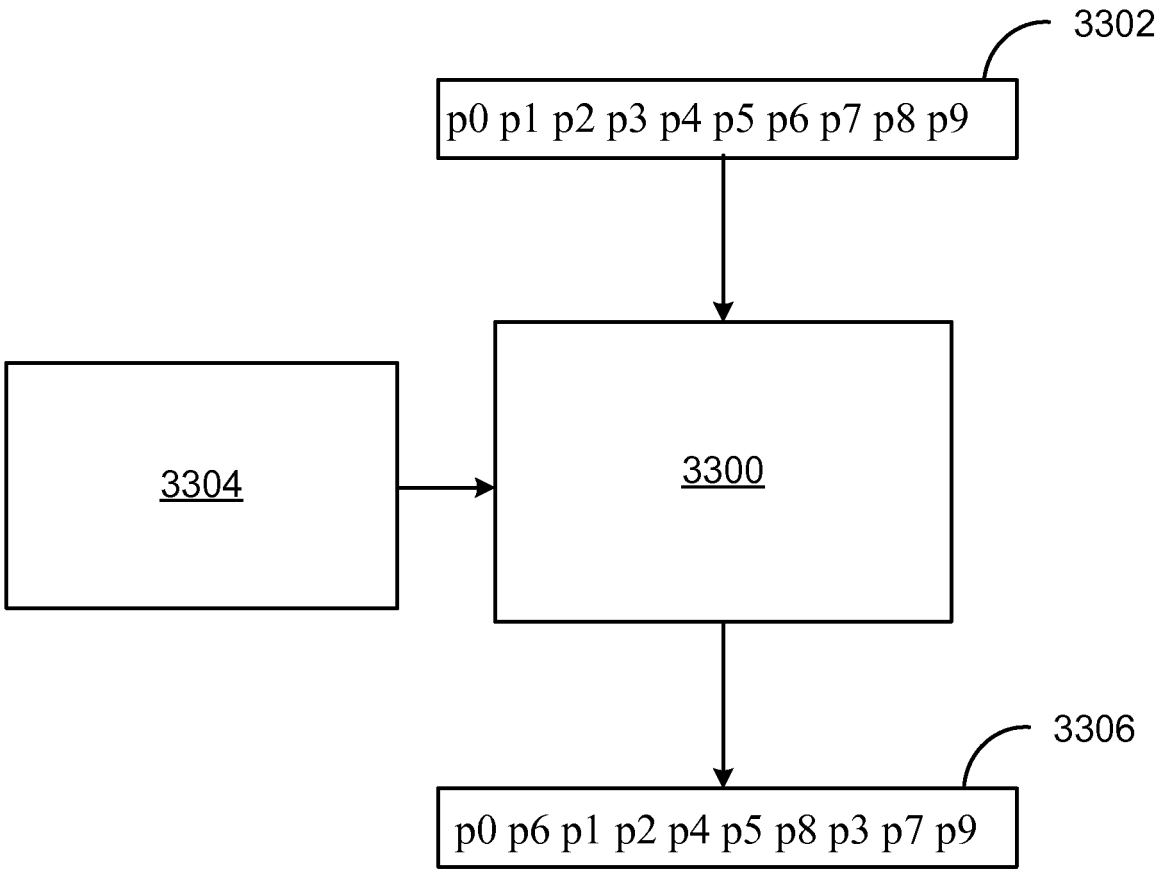
3302
p0 p1 p2 p3 p4 p5 p6 p7 p8 p9
3304
3300
3306
p0 p6 p1 p2 p4 p5 p8 p3 p7 p9

FIG. 33

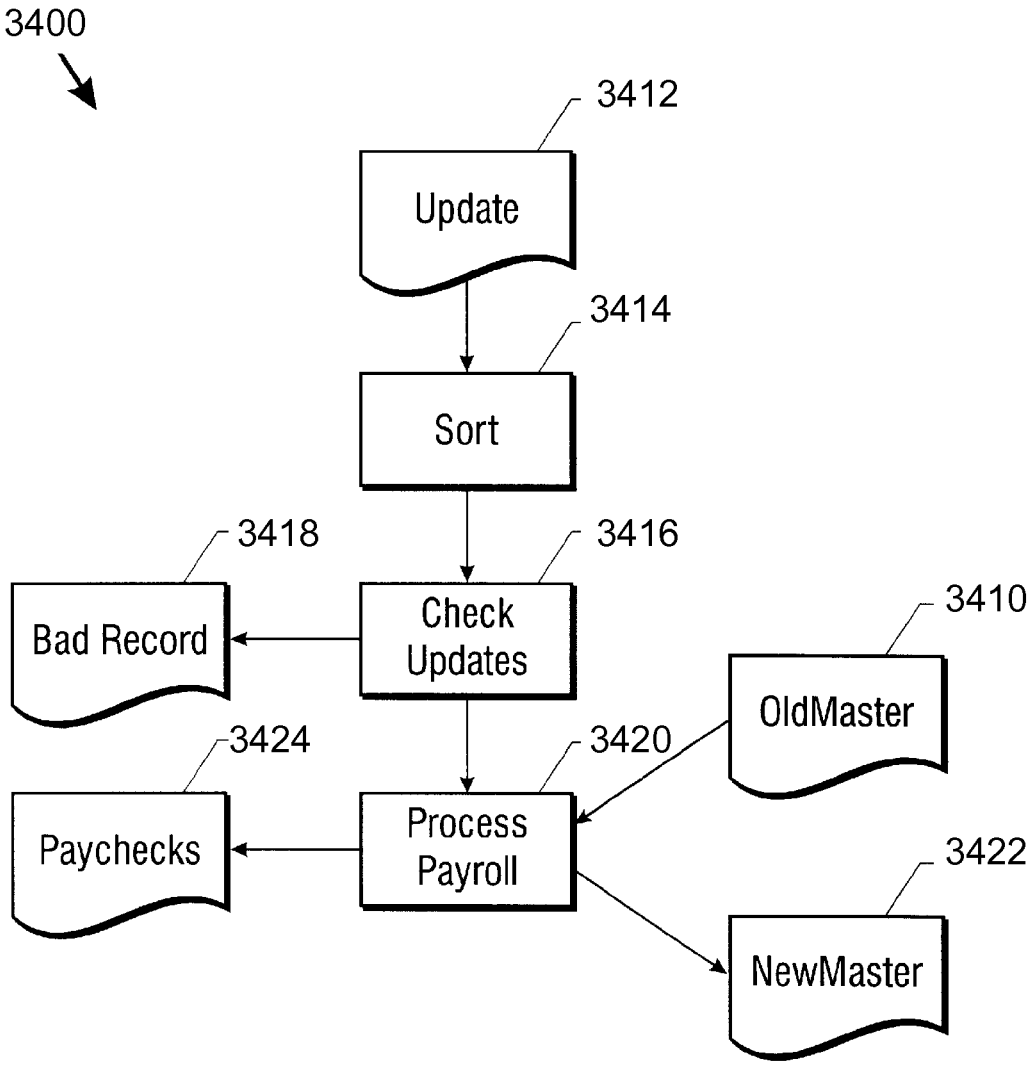
3400
3412
Update
3414
Sort
3418
3416
Bad Record
Check Updates
3410
OldMaster
3424
3420
Paychecks
Process Payroll
3422
NewMaster

FIG. 34A

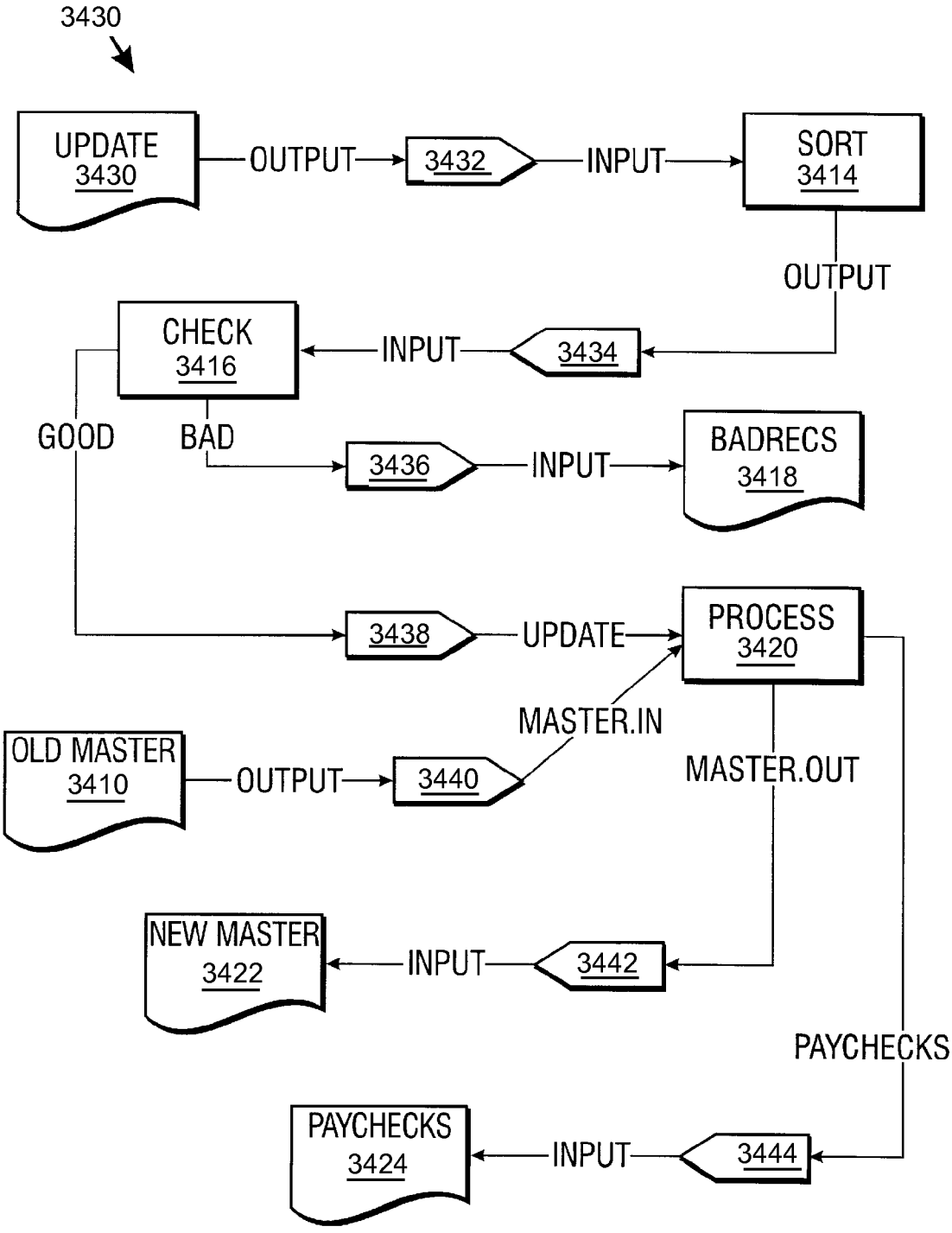
3430
UPDATE
3430
OUTPUT
3432
INPUT
SORT
3414
OUTPUT
CHECK
3416
INPUT
3434
GOOD
BAD
3436
INPUT
BADRECS
3418
3438
UPDATE
PROCESS
3420
MASTER.IN
OLD MASTER
3410
OUTPUT
3440
MASTER.OUT
NEW MASTER
3422
INPUT
3442
PAYCHECKS
PAYCHECKS
3424
INPUT
3444

FIG. 34B

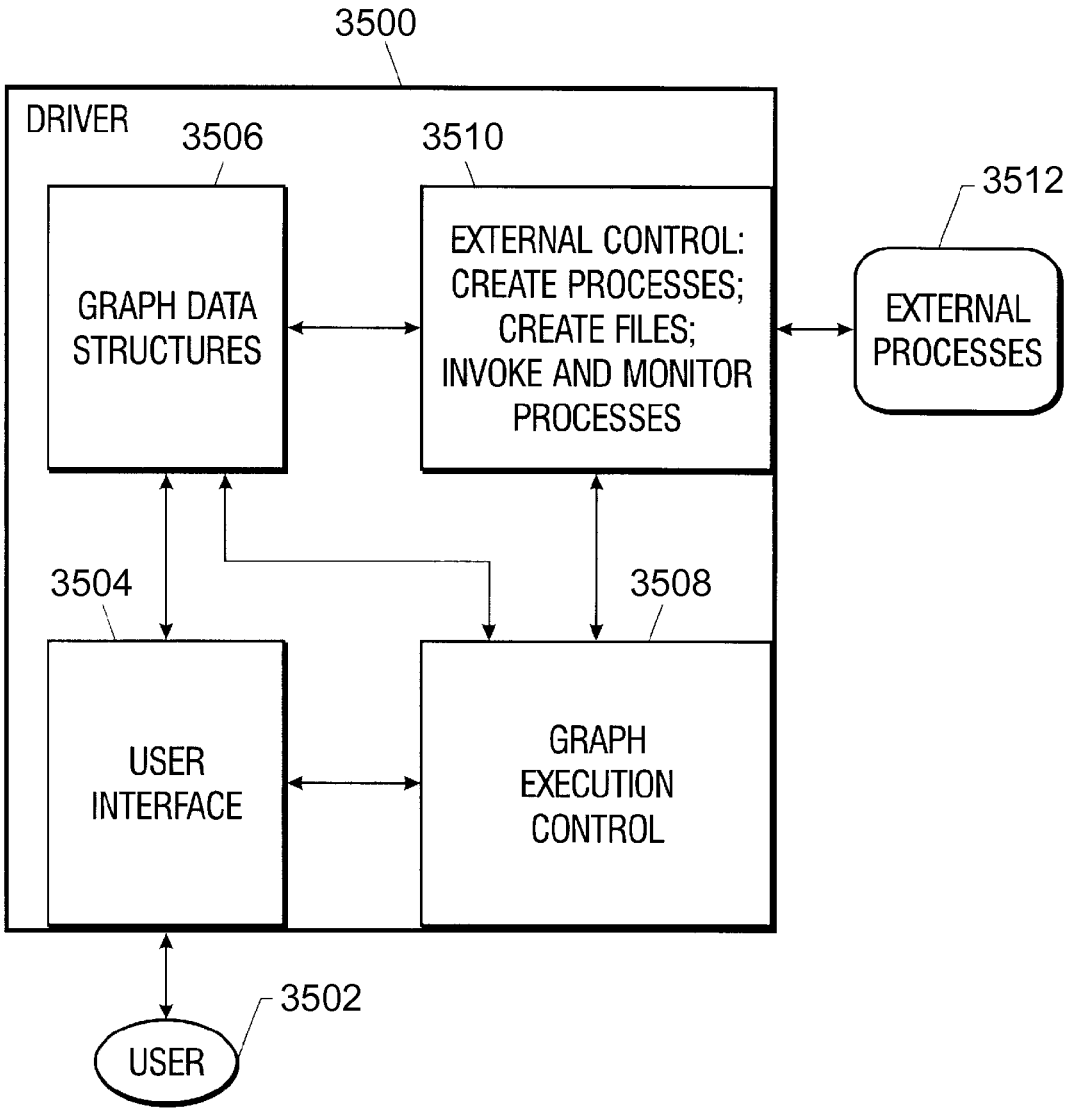
3500
DRIVER
3506
3510
3512
GRAPH DATA STRUCTURES
EXTERNAL CONTROL: CREATE PROCESSES; CREATE FILES; INVOKE AND MONITOR PROCESSES
EXTERNAL PROCESSES
3504
3508
USER INTERFACE
GRAPH EXECUTION CONTROL
3502
USER

FIG. 35

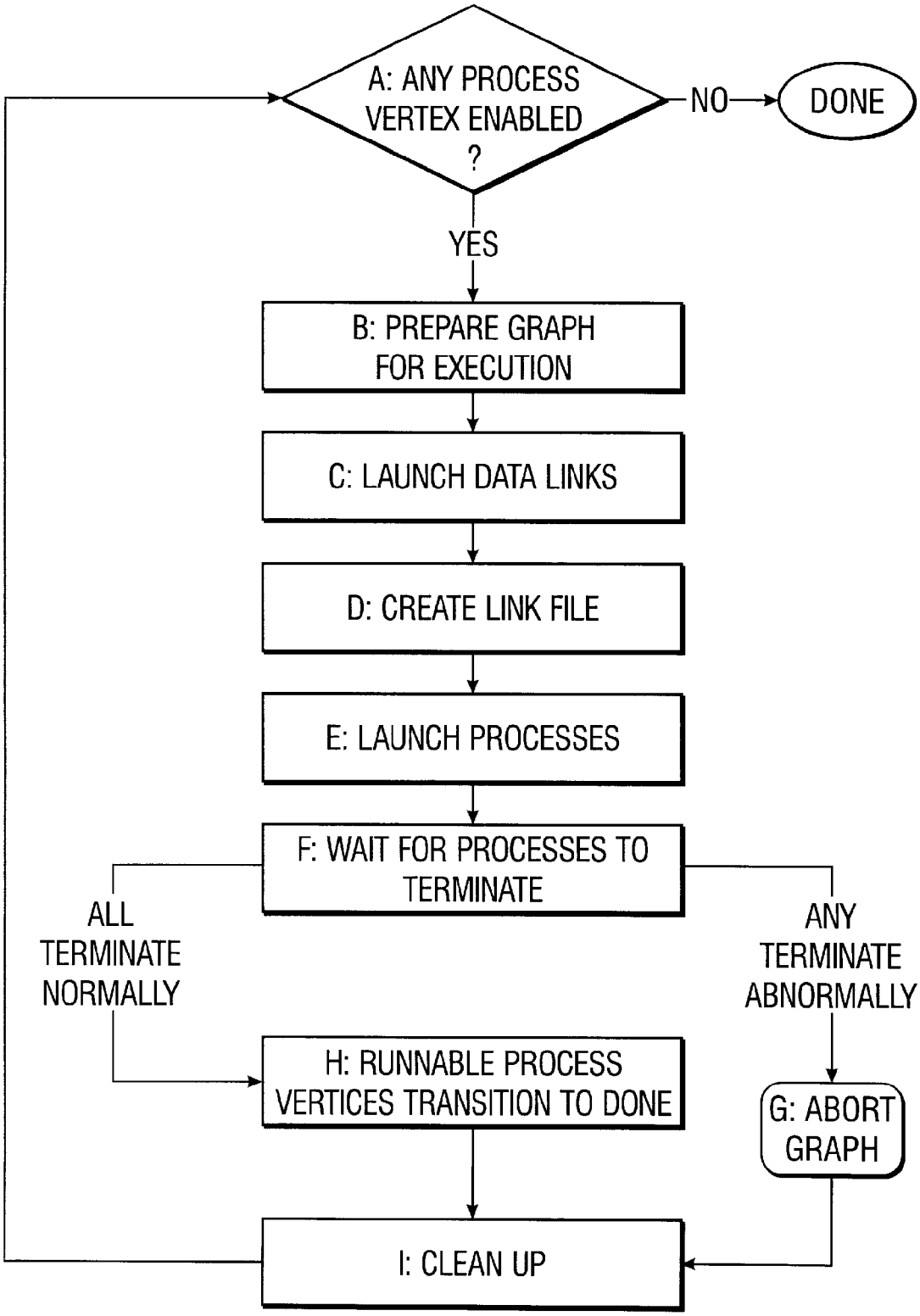
A: ANY PROCESS VERTEX ENABLED ?
NO
DONE
YES
B: PREPARE GRAPH FOR EXECUTION
C: LAUNCH DATA LINKS
D: CREATE LINK FILE
E: LAUNCH PROCESSES
F: WAIT FOR PROCESSES TO TERMINATE
ALL TERMINATE NORMALLY
ANY TERMINATE ABNORMALLY
H: RUNNABLE PROCESS VERTICES TRANSITION TO DONE
G: ABORT GRAPH
I: CLEAN UP

FIG. 36

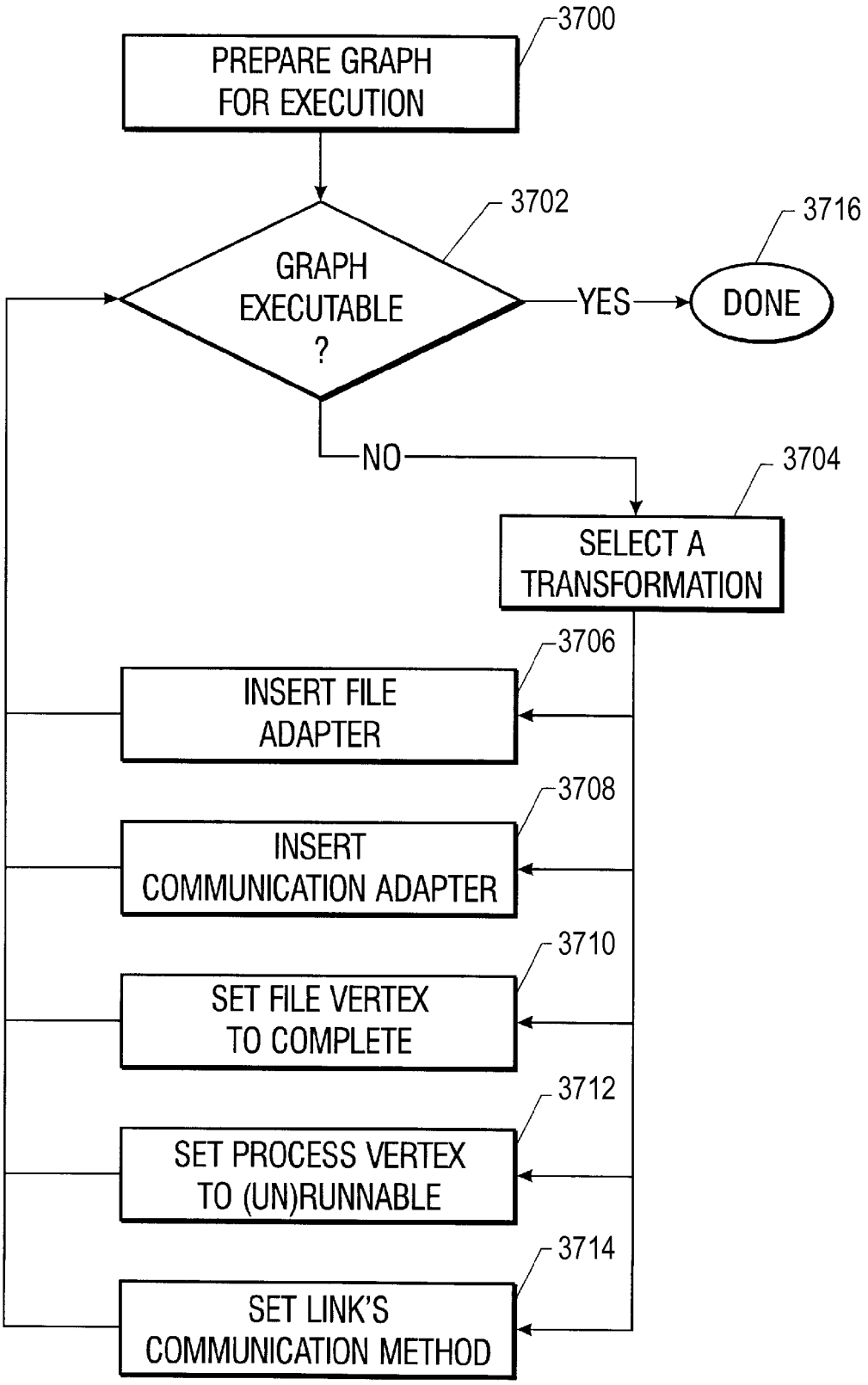
3700
PREPARE GRAPH FOR EXECUTION
3702
GRAPH EXECUTABLE ?
YES
3716
DONE
NO
3704
SELECT A TRANSFORMATION
3706
INSERT FILE ADAPTER
3708
INSERT COMMUNICATION ADAPTER
3710
SET FILE VERTEX TO COMPLETE
3712
SET PROCESS VERTEX TO (UN)RUNNABLE
3714
SET LINK'S COMMUNICATION METHOD

FIG. 37A

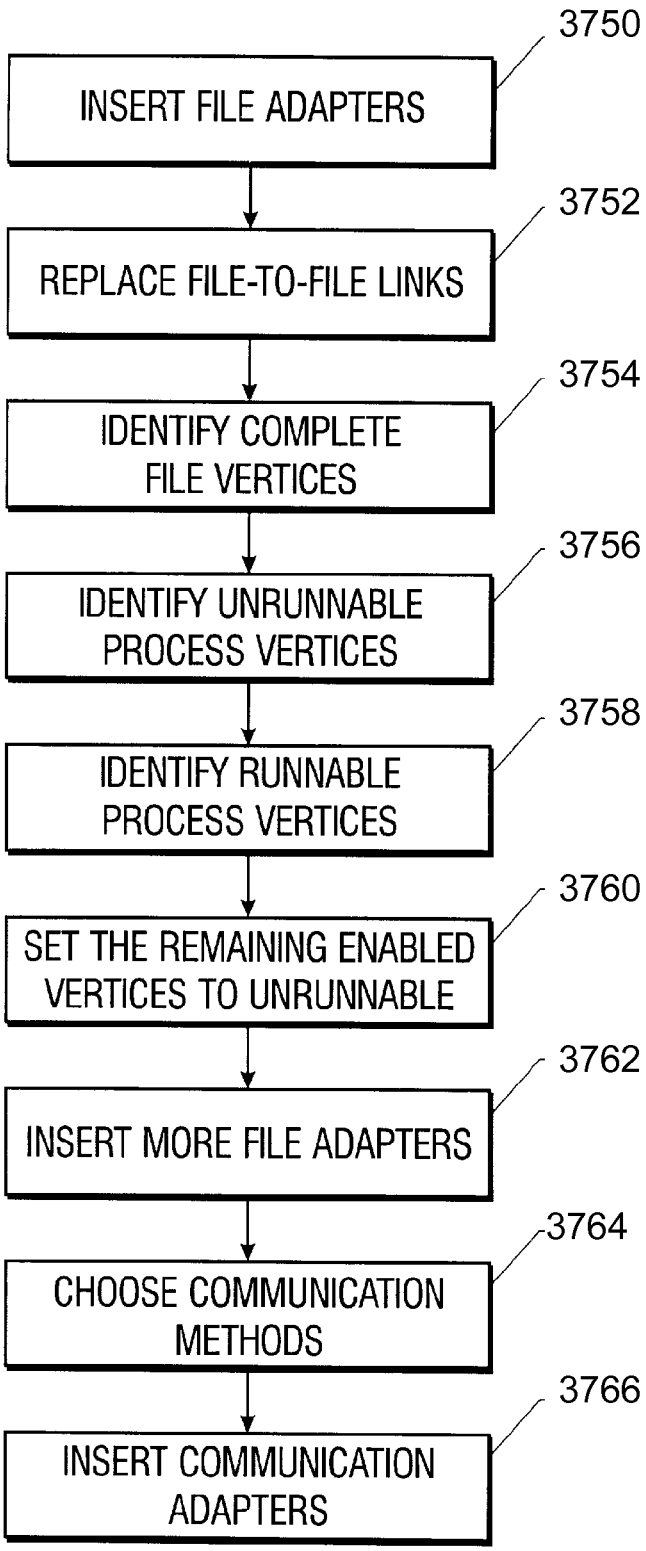
3750
INSERT FILE ADAPTERS
3752
REPLACE FILE-TO-FILE LINKS
3754
IDENTIFY COMPLETE FILE VERTICES
3756
IDENTIFY UNRUNNABLE PROCESS VERTICES
3758
IDENTIFY RUNNABLE PROCESS VERTICES
3760
SET THE REMAINING ENABLED VERTICES TO UNRUNNABLE
3762
INSERT MORE FILE ADAPTERS
3764
CHOOSE COMMUNICATION METHODS
3766
INSERT COMMUNICATION ADAPTERS

FIG. 37B

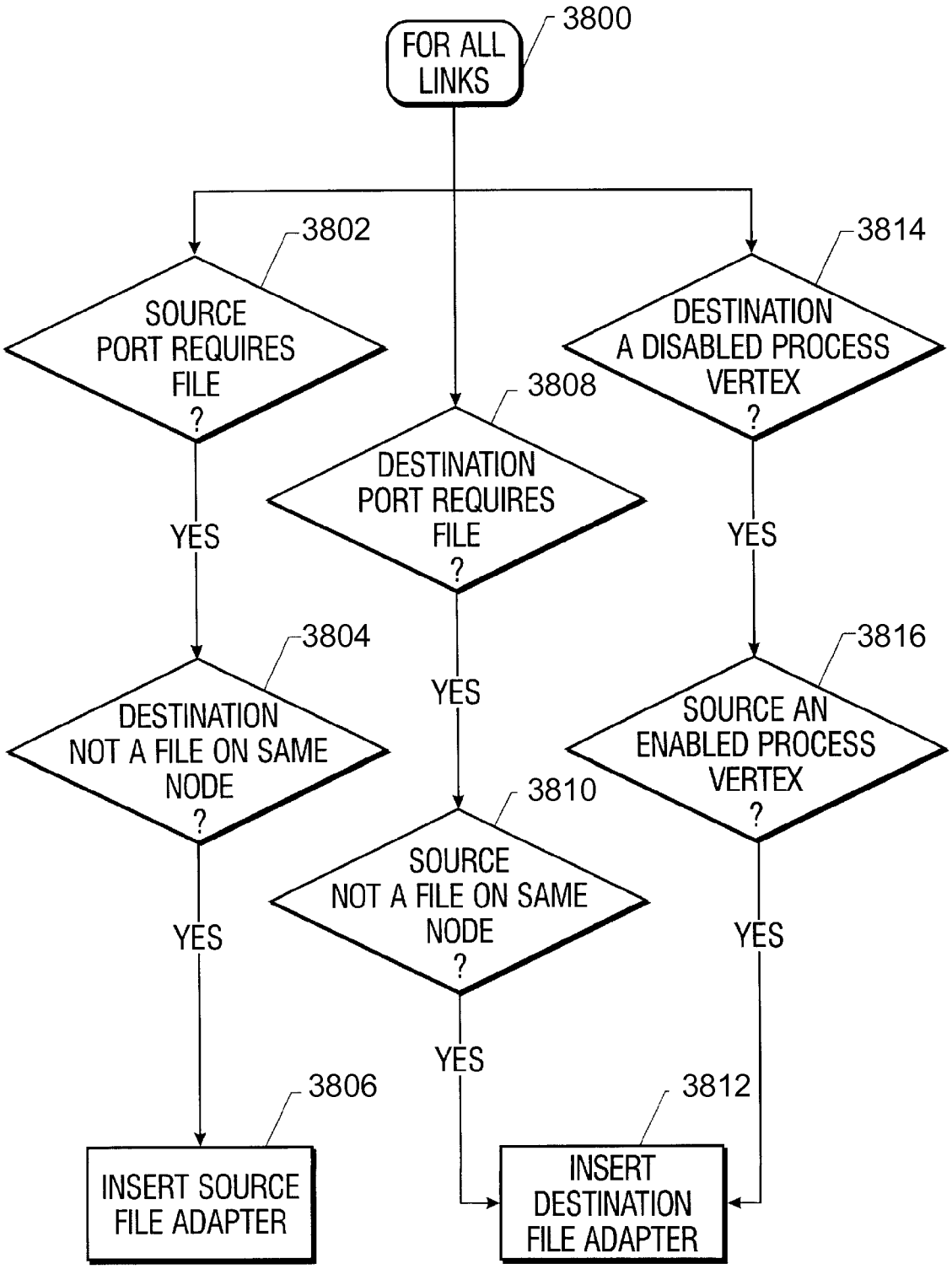
FOR ALL LINKS
3800
SOURCE PORT REQUIRES FILE ?
3802
DESTINATION PORT REQUIRES FILE ?
3808
DESTINATION A DISABLED PROCESS VERTEX ?
3814
YES
YES
YES
DESTINATION NOT A FILE ON SAME NODE ?
3804
SOURCE NOT A FILE ON SAME NODE ?
3810
SOURCE AN ENABLED PROCESS VERTEX ?
3816
YES
YES
YES
INSERT SOURCE FILE ADAPTER
3806
INSERT DESTINATION FILE ADAPTER
3812

FIG. 38A

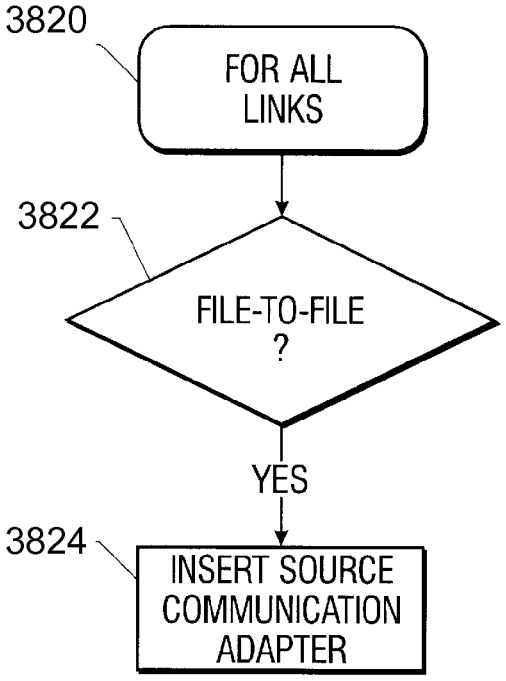

FIG. 38B

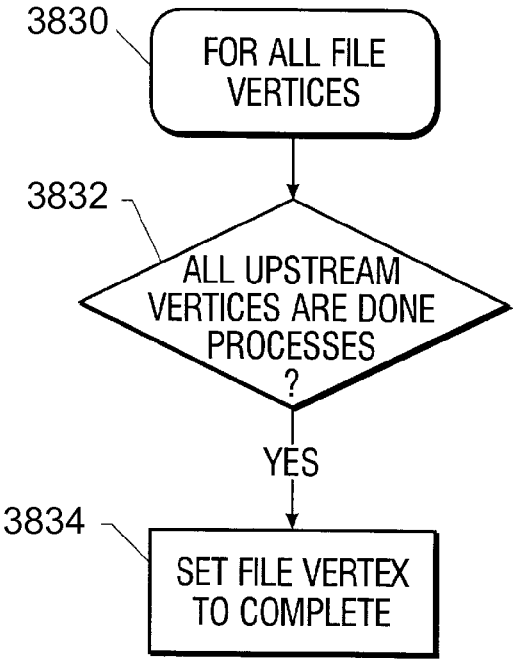

FIG. 38C

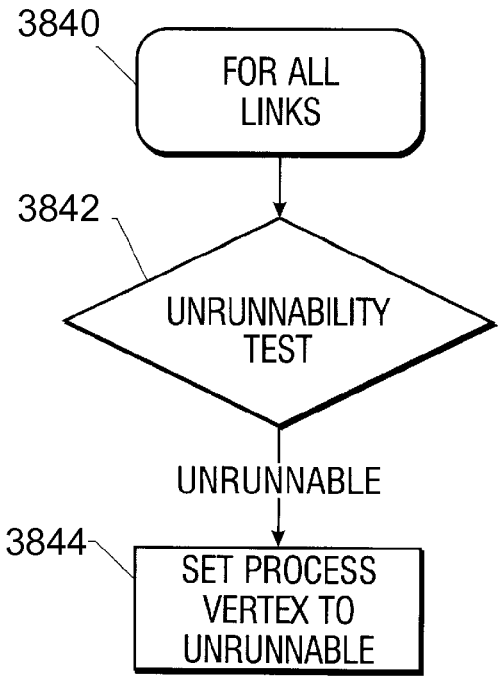

FIG. 38D

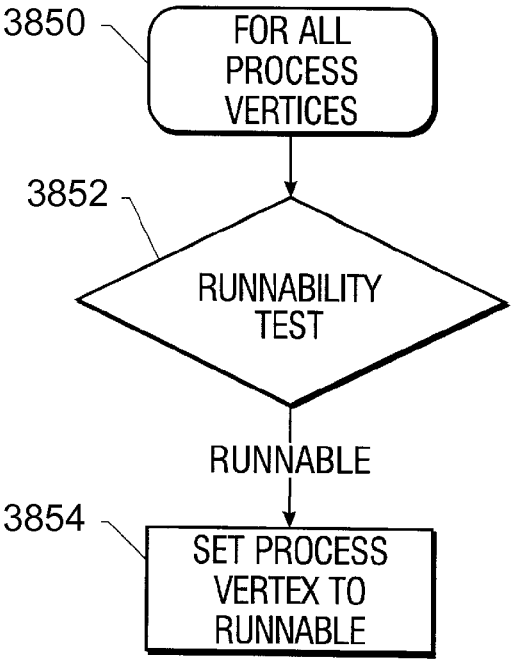

FIG. 38E

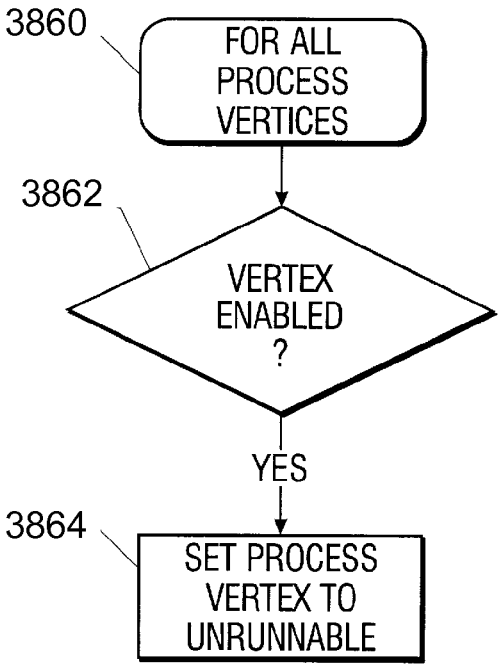

FIG. 38F

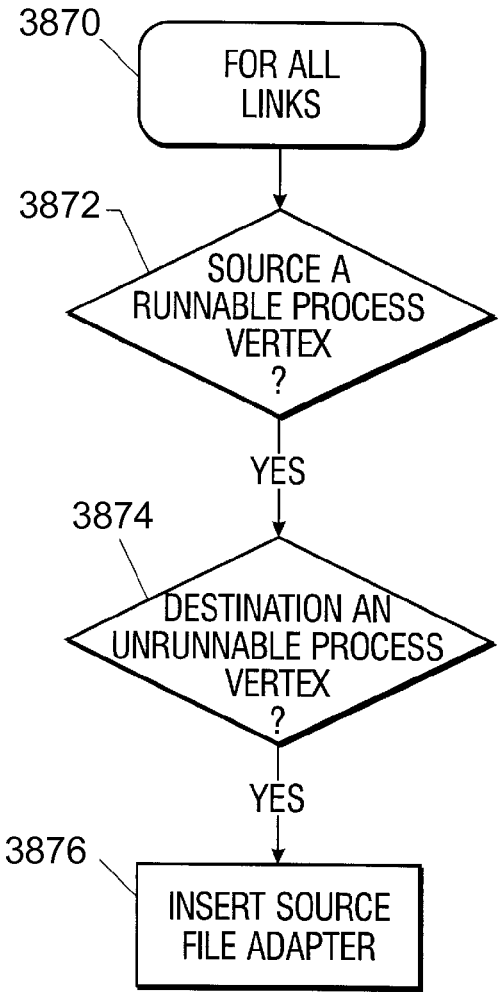

FIG. 38G

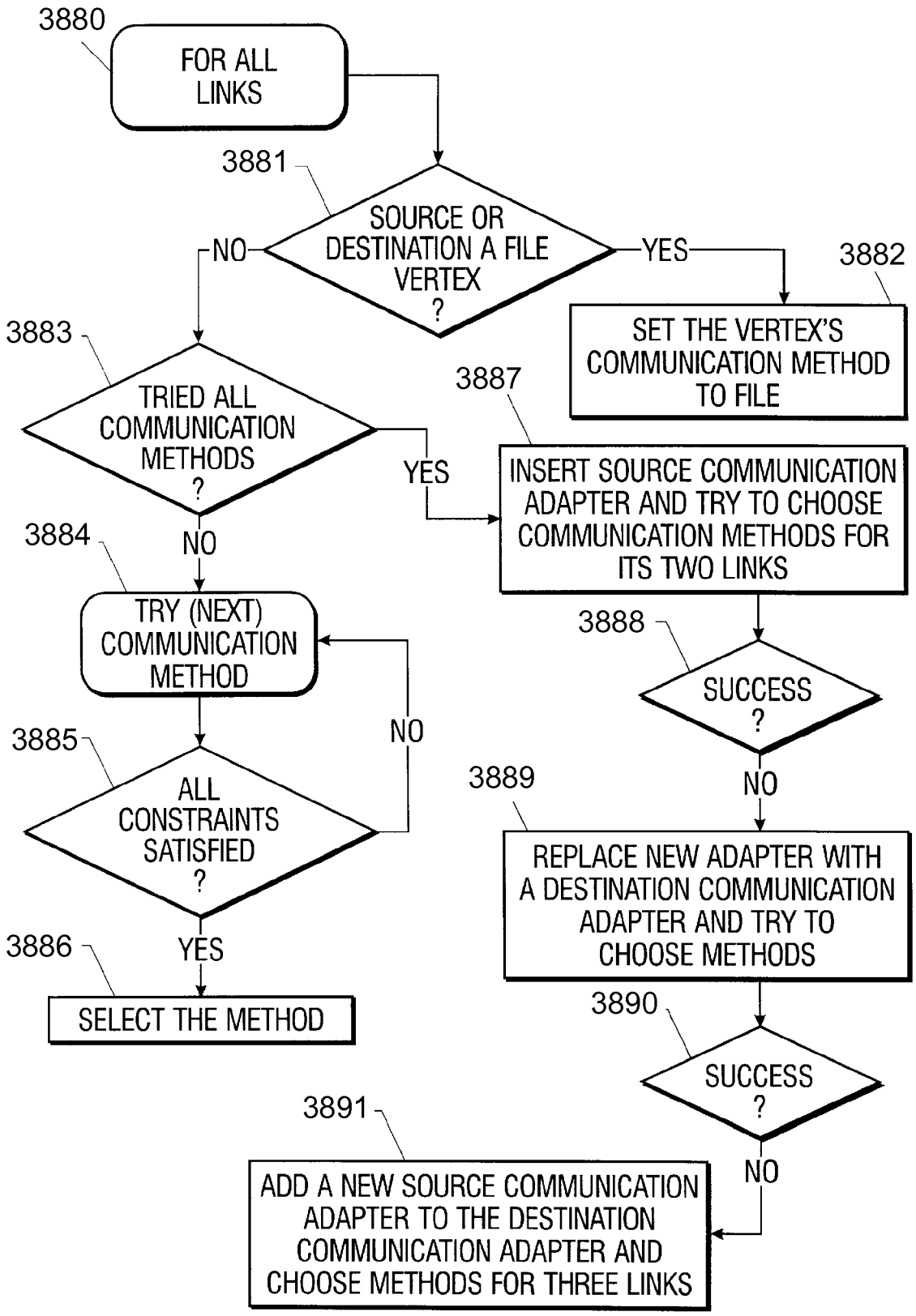
3880
FOR ALL LINKS
3881
SOURCE OR DESTINATION A FILE VERTEX ?
NO
YES
3882
SET THE VERTEX'S COMMUNICATION METHOD TO FILE
3883
TRIED ALL COMMUNICATION METHODS ?
YES
3887
INSERT SOURCE COMMUNICATION ADAPTER AND TRY TO CHOOSE COMMUNICATION METHODS FOR ITS TWO LINKS
NO
3884
TRY (NEXT) COMMUNICATION METHOD
3888
SUCCESS ?
NO
3885
ALL CONSTRAINTS SATISFIED ?
NO
3889
REPLACE NEW ADAPTER WITH A DESTINATION COMMUNICATION ADAPTER AND TRY TO CHOOSE METHODS
YES
3886
SELECT THE METHOD
3890
SUCCESS ?
NO
3891
ADD A NEW SOURCE COMMUNICATION ADAPTER TO THE DESTINATION COMMUNICATION ADAPTER AND CHOOSE METHODS FOR THREE LINKS

FIG. 38H

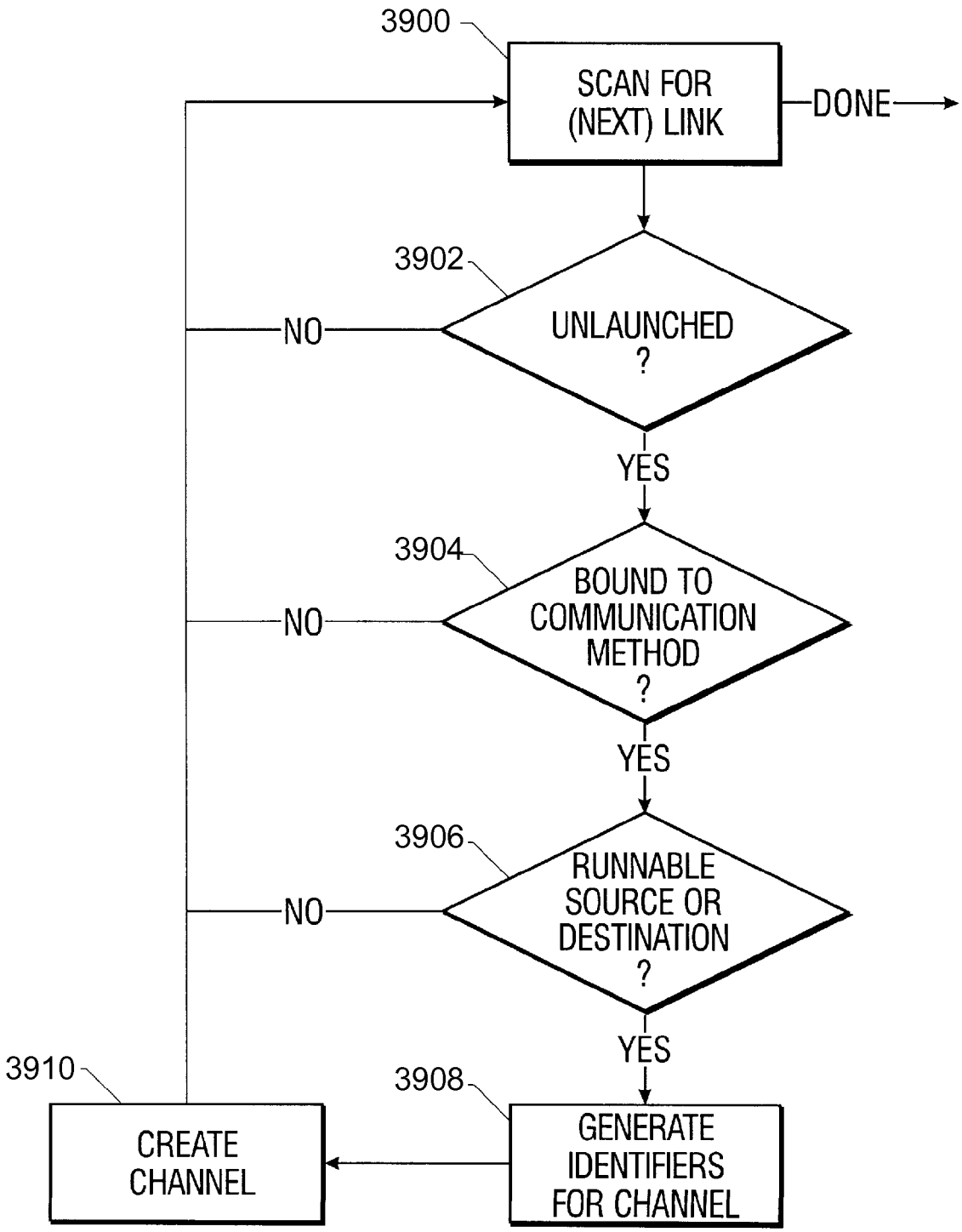
3900
SCAN FOR (NEXT) LINK
DONE
3902
UNLAUNCHED ?
NO
YES
3904
BOUND TO COMMUNICATION METHOD ?
NO
YES
3906
RUNNABLE SOURCE OR DESTINATION ?
NO
YES
3908
GENERATE IDENTIFIERS FOR CHANNEL
3910
CREATE CHANNEL

FIG. 39

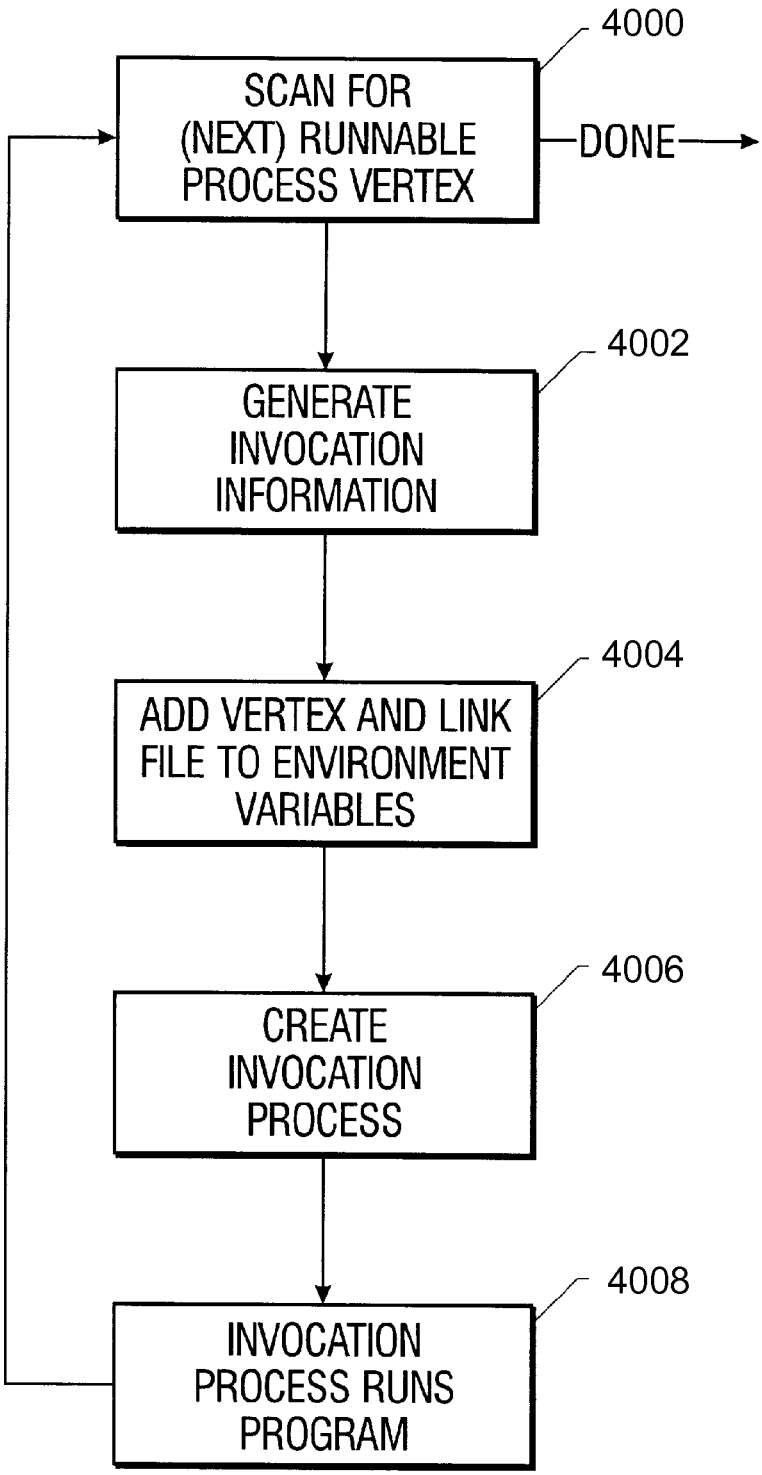
4000
SCAN FOR (NEXT) RUNNABLE PROCESS VERTEX
DONE
4002
GENERATE INVOCATION INFORMATION
4004
ADD VERTEX AND LINK FILE TO ENVIRONMENT VARIABLES
4006
CREATE INVOCATION PROCESS
4008
INVOCATION PROCESS RUNS PROGRAM

FIG. 40

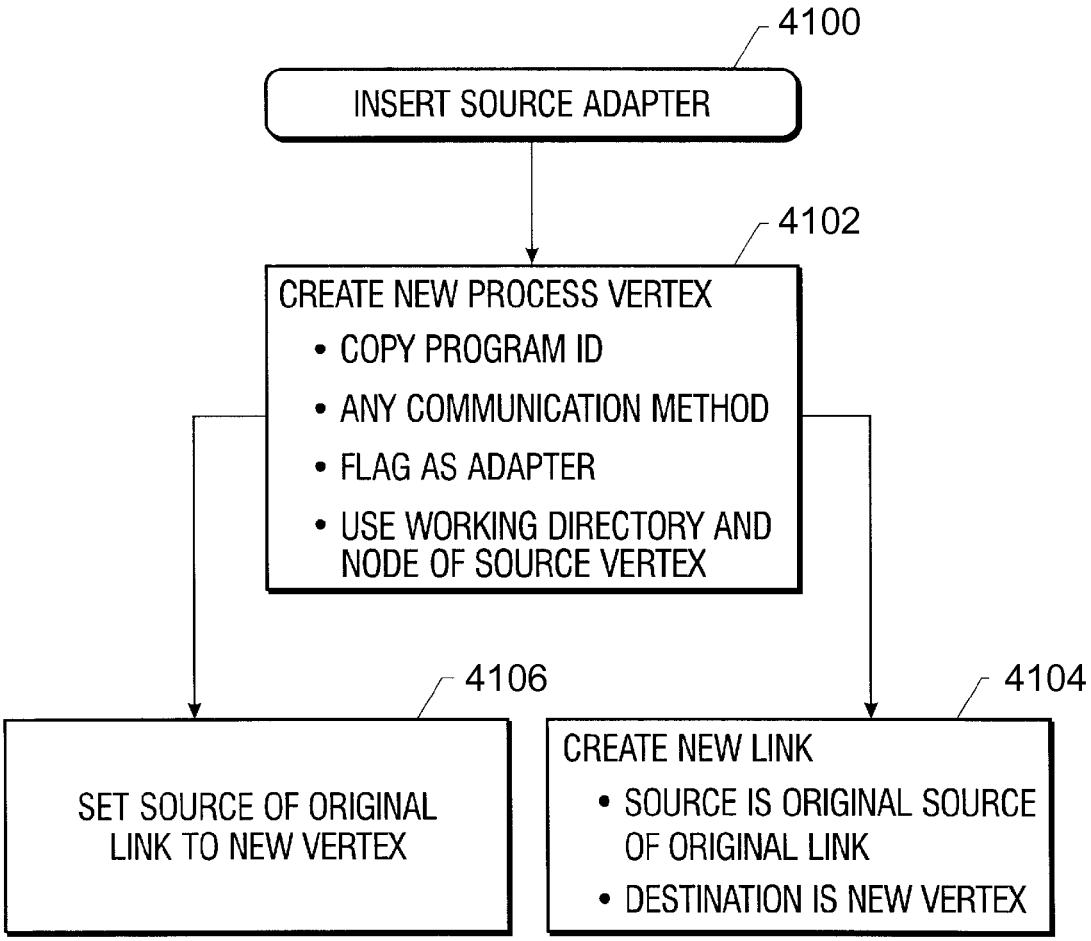
4100
INSERT SOURCE ADAPTER
4102
CREATE NEW PROCESS VERTEX
• COPY PROGRAM ID
• ANY COMMUNICATION METHOD
• FLAG AS ADAPTER
• USE WORKING DIRECTORY AND NODE OF SOURCE VERTEX
4106
SET SOURCE OF ORIGINAL LINK TO NEW VERTEX
4104
CREATE NEW LINK
• SOURCE IS ORIGINAL SOURCE OF ORIGINAL LINK
• DESTINATION IS NEW VERTEX

FIG. 41

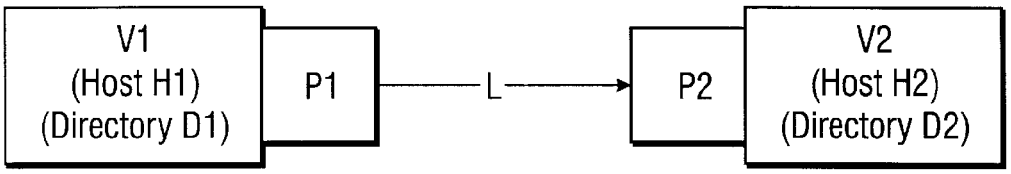

FIG. 42A

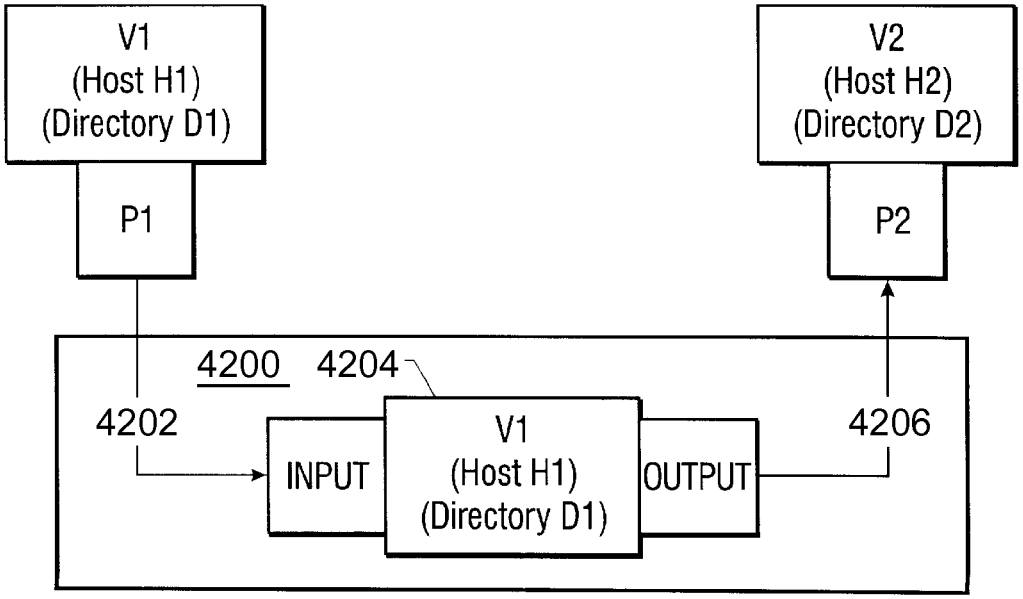

FIG. 42B

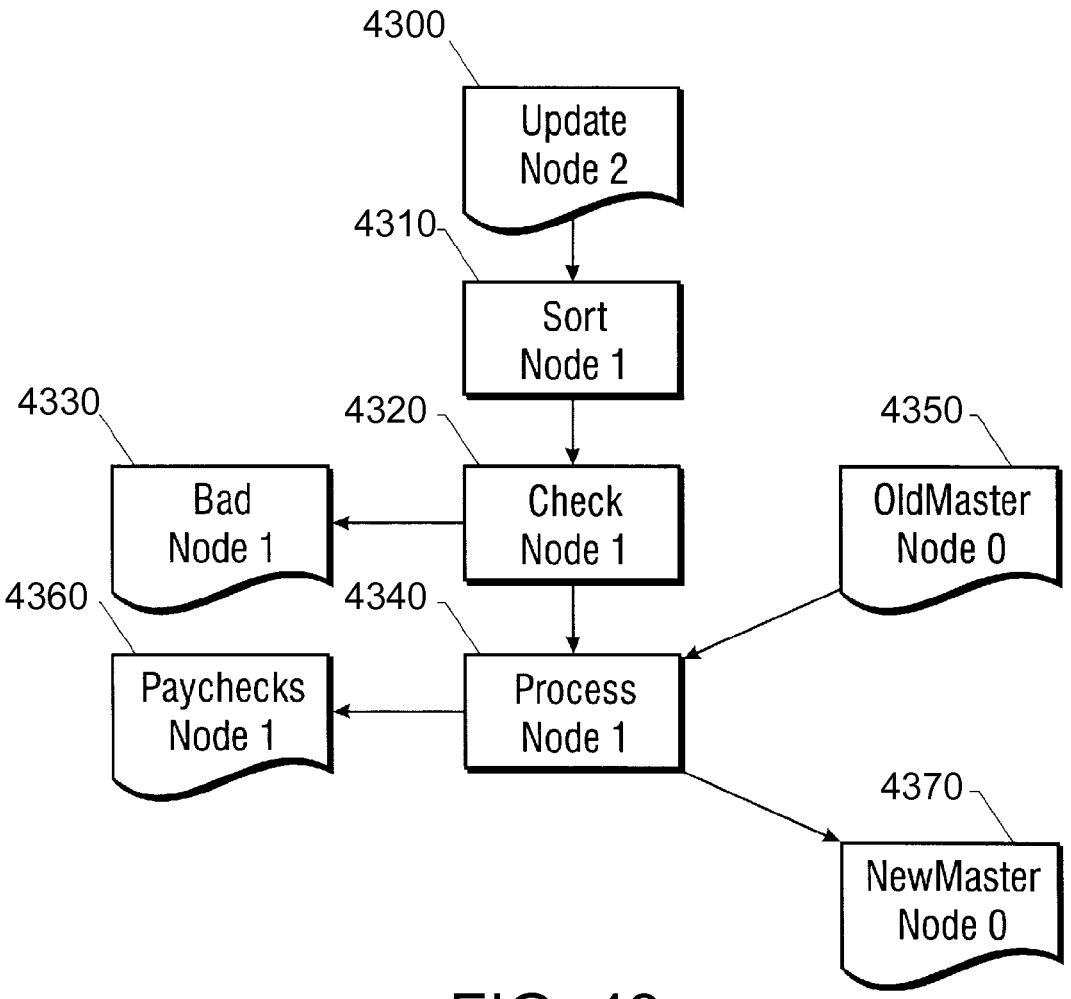
4300
Update
Node 2
4310
Sort
Node 1
4330
4320
4350
Bad
Node 1
Check
Node 1
OldMaster
Node 0
4360
4340
Paychecks
Node 1
Process
Node 1
4370
NewMaster
Node 0

FIG. 43

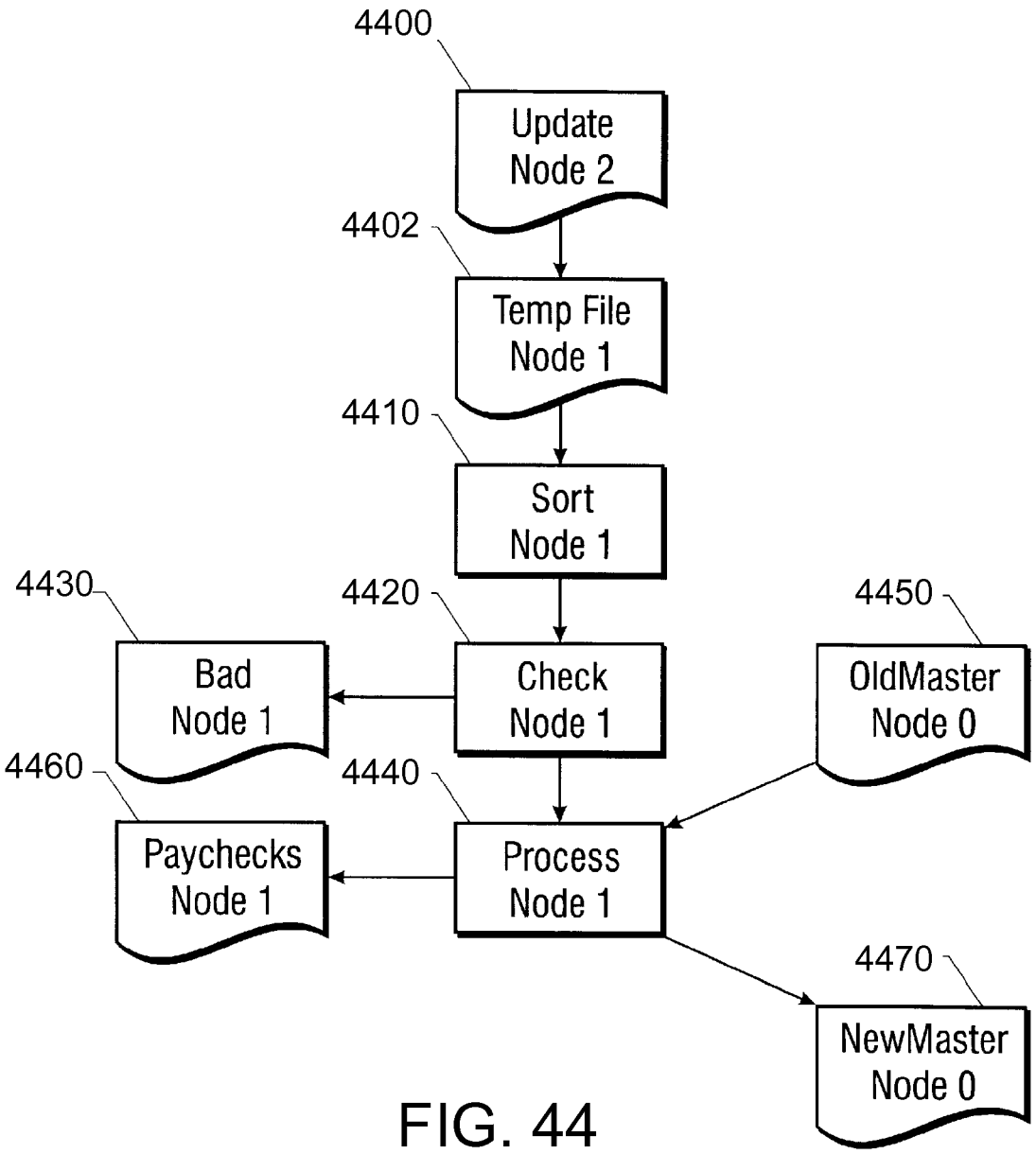
4400
Update
Node 2
4402
Temp File
Node 1
4410
Sort
Node 1
4430
4420
4450
Bad
Node 1
Check
Node 1
OldMaster
Node 0
4460
4440
Paychecks
Node 1
Process
Node 1
4470
NewMaster
Node 0

FIG. 44

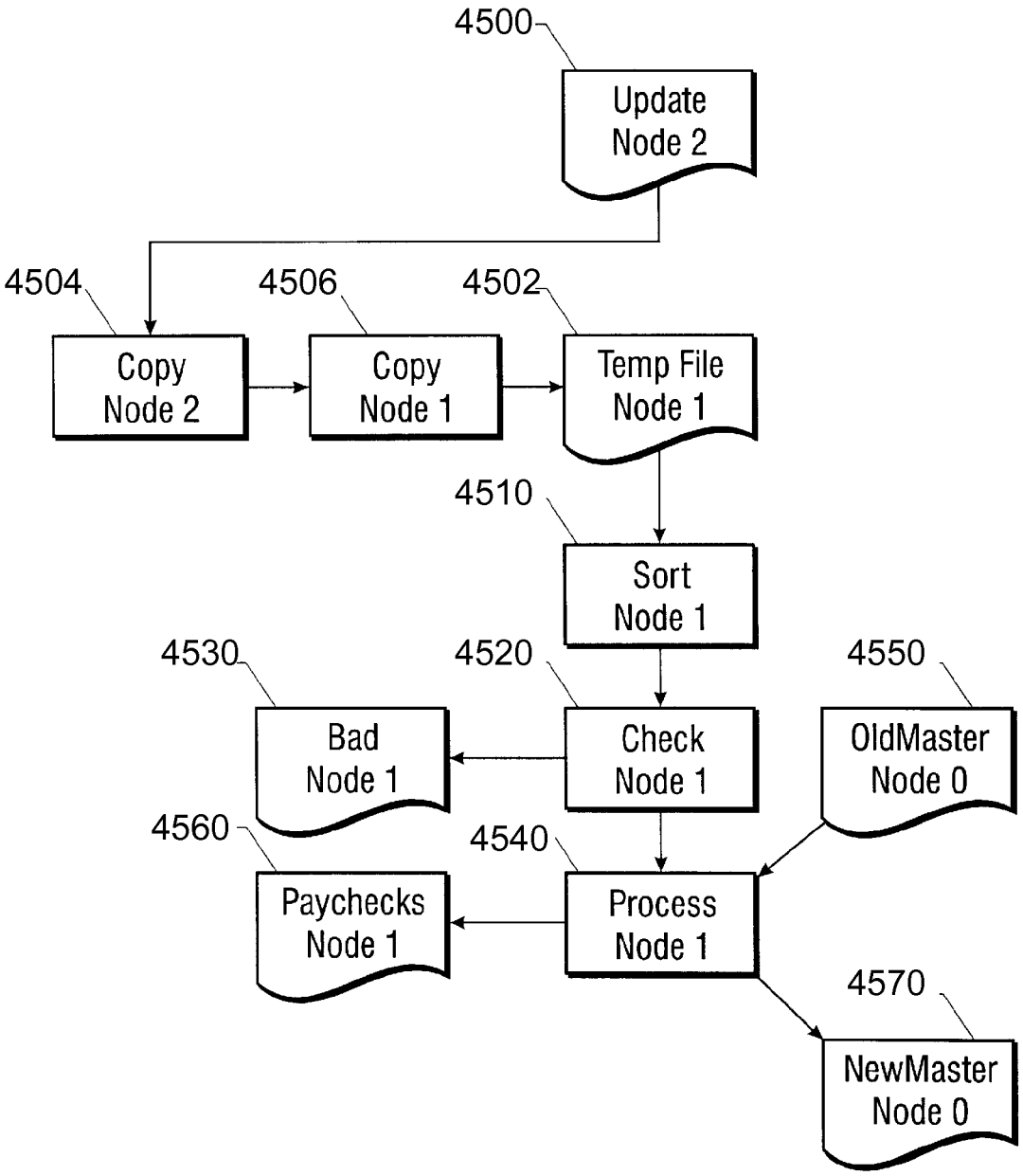
4500
Update Node 2
4504
4506
4502
Copy Node 2
Copy Node 1
Temp File Node 1
4510
Sort Node 1
4530
4520
4550
Bad Node 1
Check Node 1
OldMaster Node 0
4560
4540
Paychecks Node 1
Process Node 1
4570
NewMaster Node 0

FIG. 45

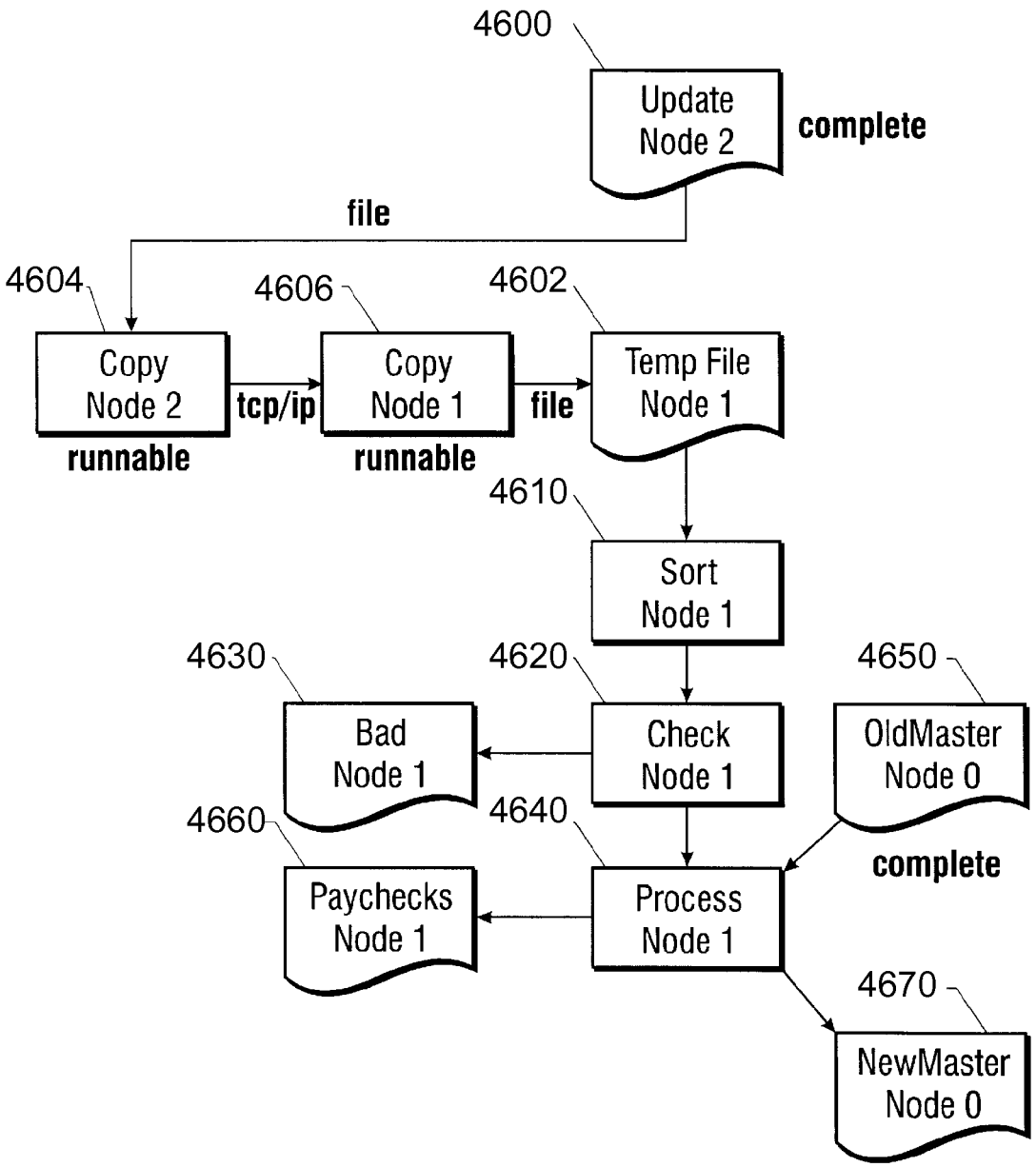
4600
Update Node 2
complete
file
4604
4606
4602
Copy Node 2
tcp/ip
Copy Node 1
file
Temp File Node 1
runnable
runnable
4610
Sort Node 1
4630
4620
4650
Bad Node 1
Check Node 1
OldMaster Node 0
4660
4640
complete
Paychecks Node 1
Process Node 1
4670
NewMaster Node 0

FIG. 46

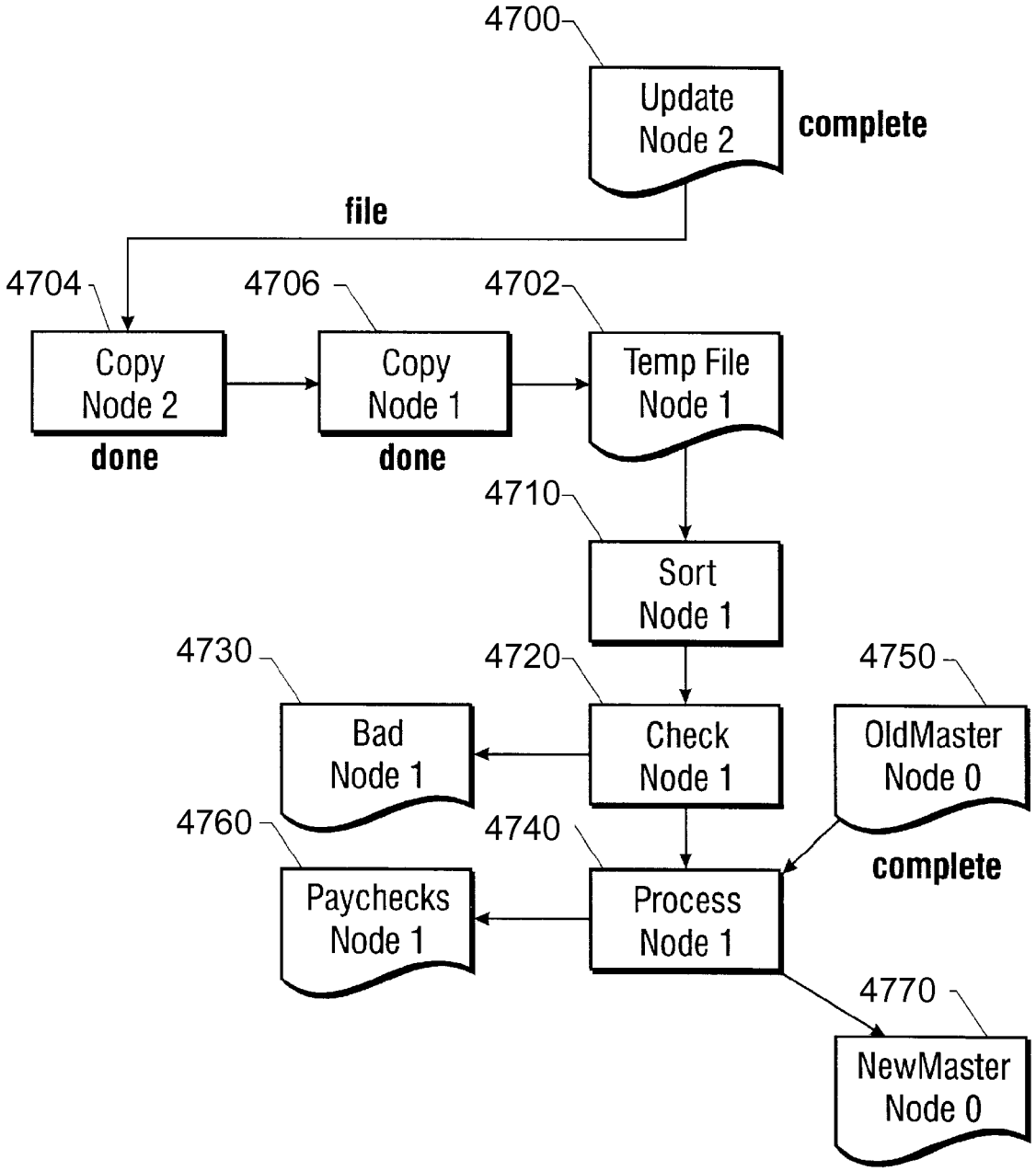
4700
Update Node 2
complete
file
4704
4706
4702
Copy Node 2
done
Copy Node 1
done
Temp File Node 1
4710
Sort Node 1
4730
4720
4750
Bad Node 1
Check Node 1
OldMaster Node 0
4760
4740
complete
Paychecks Node 1
Process Node 1
4770
NewMaster Node 0

FIG. 47

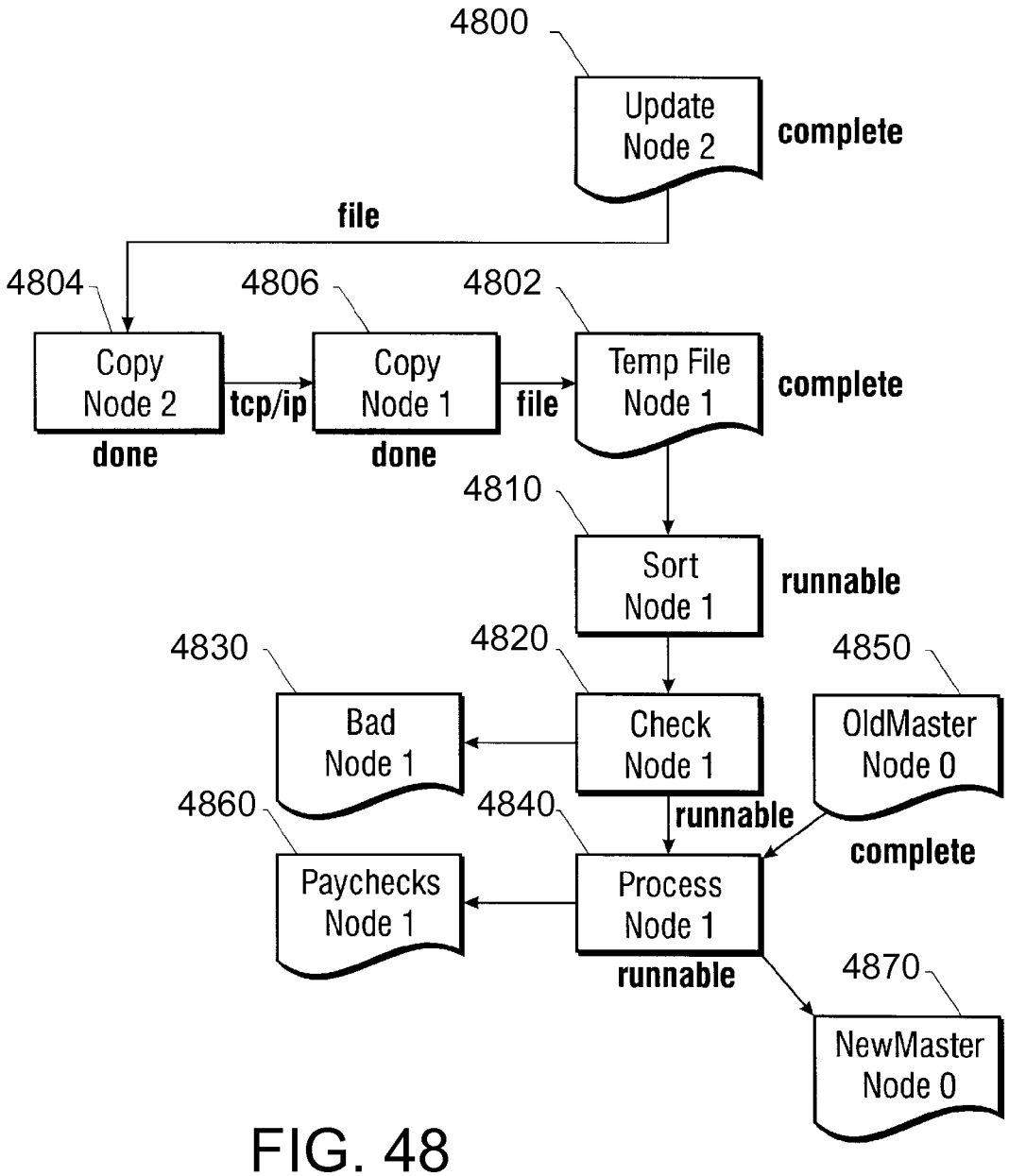
4800
Update Node 2
complete
file
4804
4806
4802
Copy Node 2
tcp/ip
Copy Node 1
file
Temp File Node 1
complete
done
done
4810
Sort Node 1
runnable
4830
4820
4850
Bad Node 1
Check Node 1
OldMaster Node 0
4860
4840
runnable
complete
Paychecks Node 1
Process Node 1
runnable
4870
NewMaster Node 0

FIG. 48

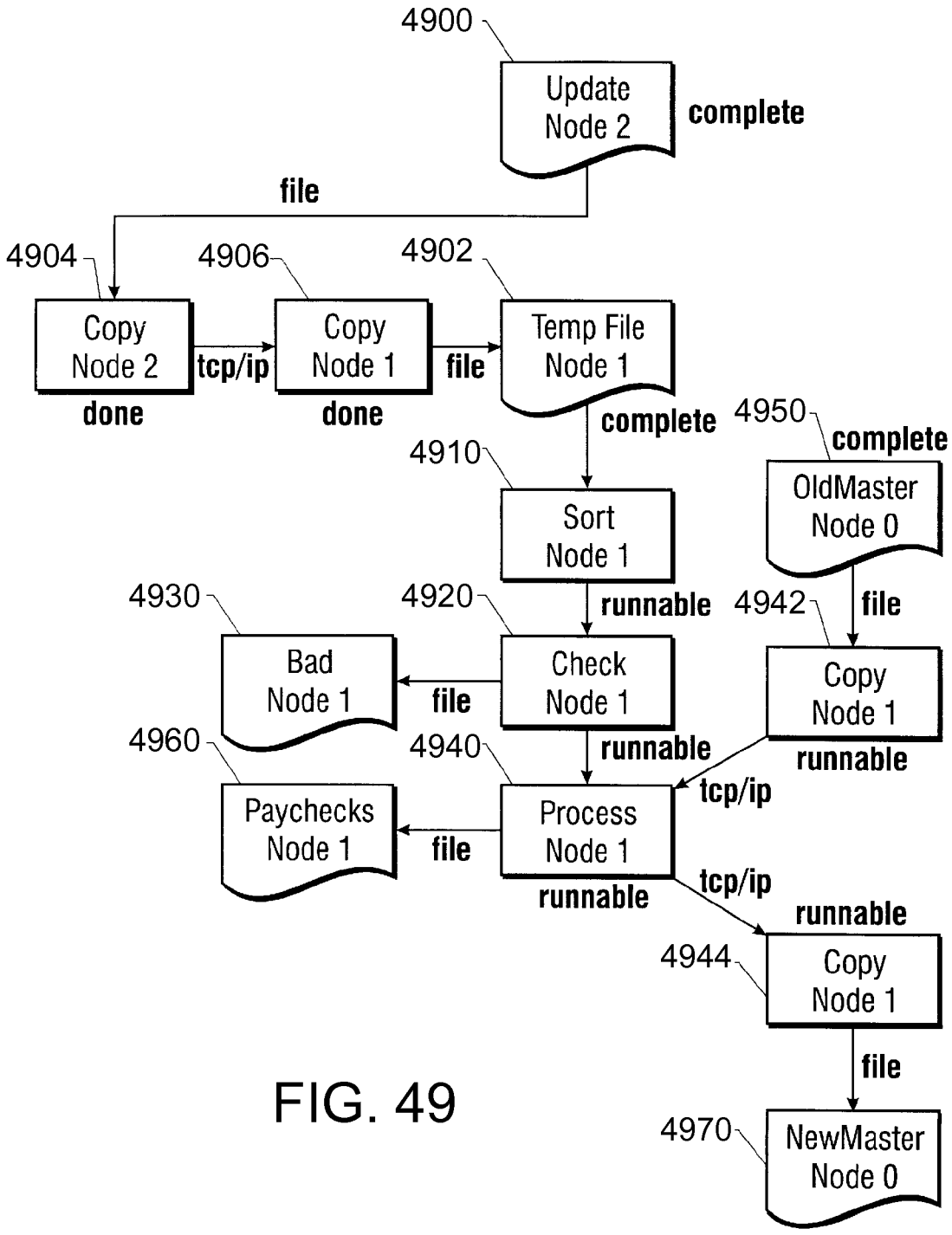
4900
Update Node 2
complete
file
4904
4906
4902
Copy Node 2
tcp/ip
Copy Node 1
file
Temp File Node 1
done
done
complete
4950
complete
4910
OldMaster Node 0
Sort Node 1
runnable
4930
4920
4942
file
Bad Node 1
file
Check Node 1
Copy Node 1
4960
4940
runnable
runnable
tcp/ip
Paychecks Node 1
file
Process Node 1
runnable
tcp/ip
runnable
4944
Copy Node 1
file
4970
NewMaster Node 0

FIG. 49