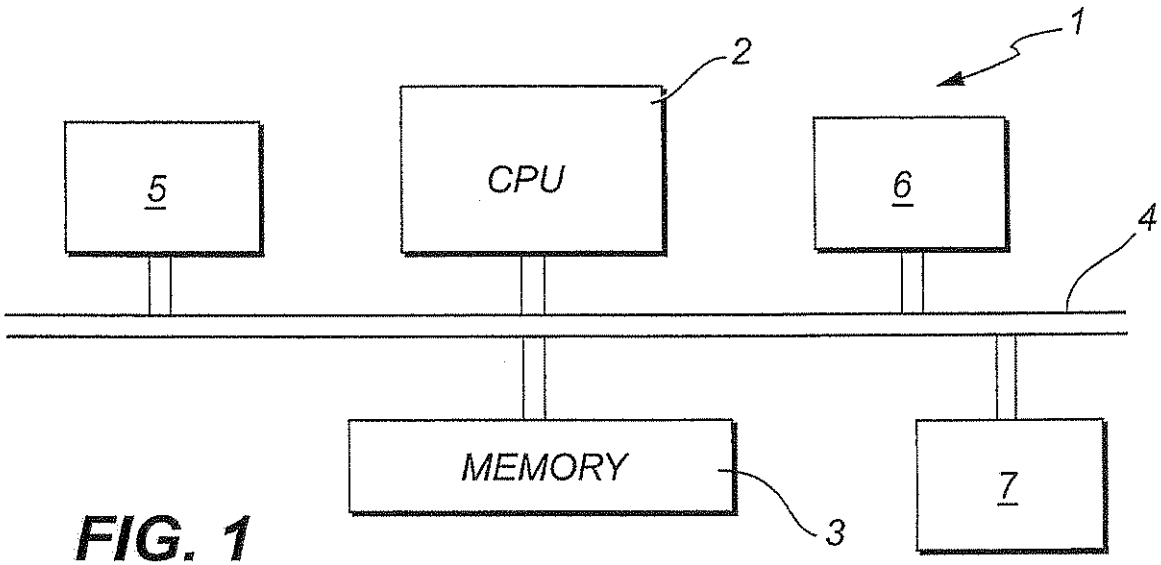


FIG. 1
PRIOR ART

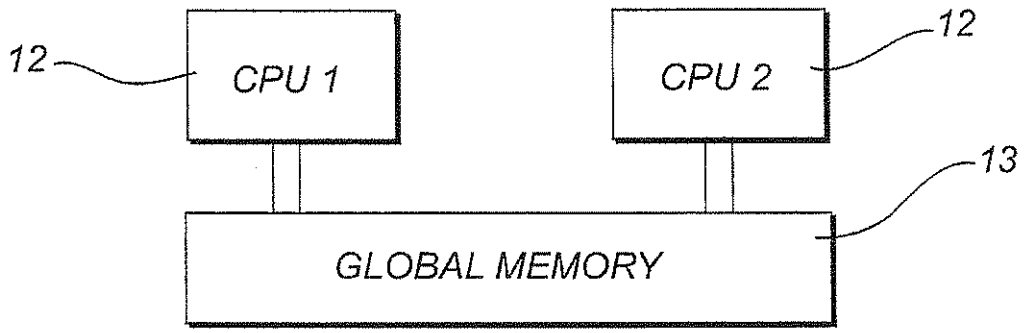


FIG. 2
PRIOR ART

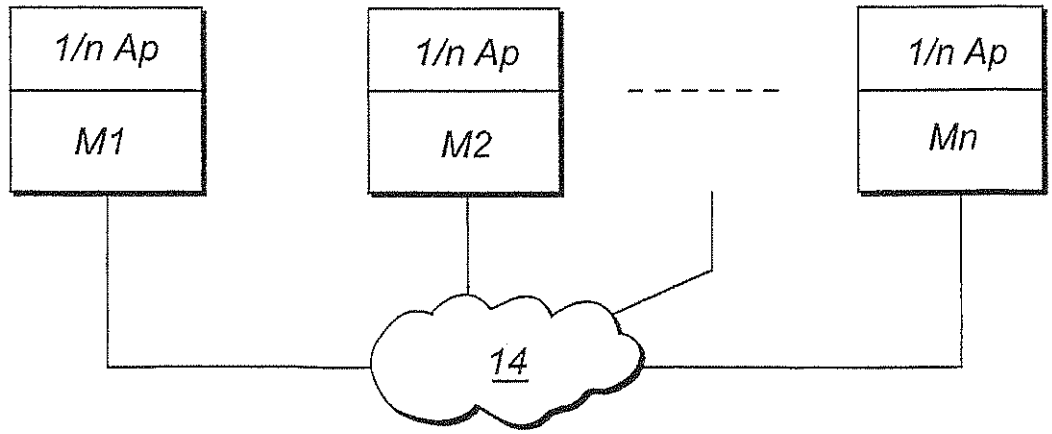


FIG. 3
PRIOR ART

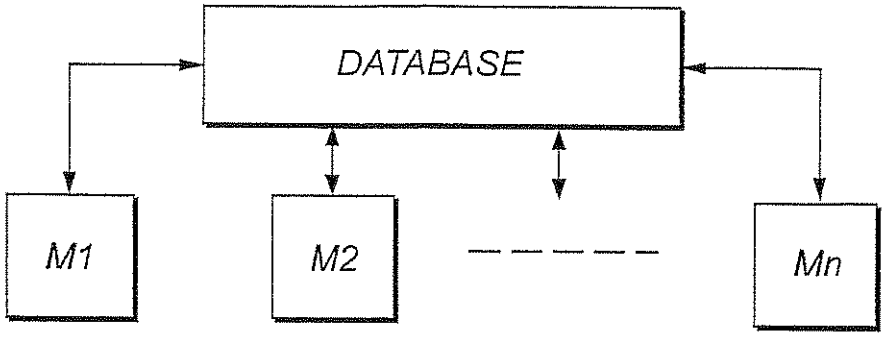


FIG. 4
PRIOR ART

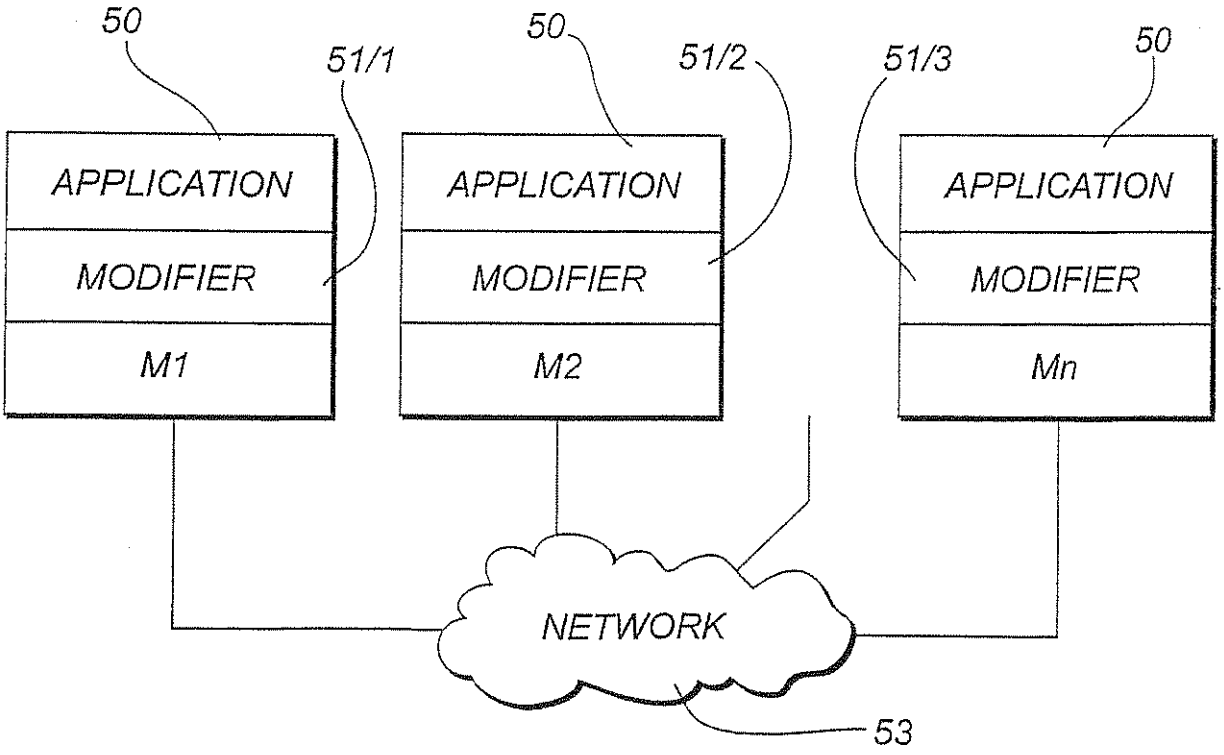


FIG. 5

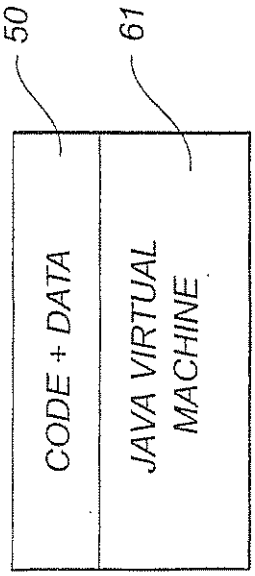


FIG. 6
PRIOR ART

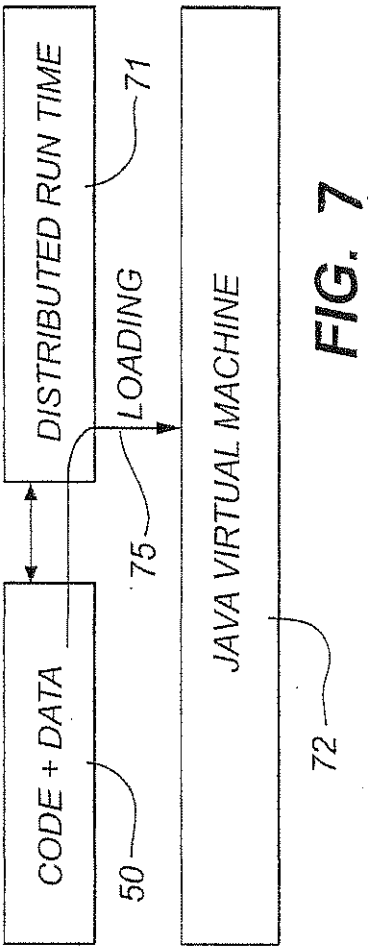


FIG. 7

FIG. 8

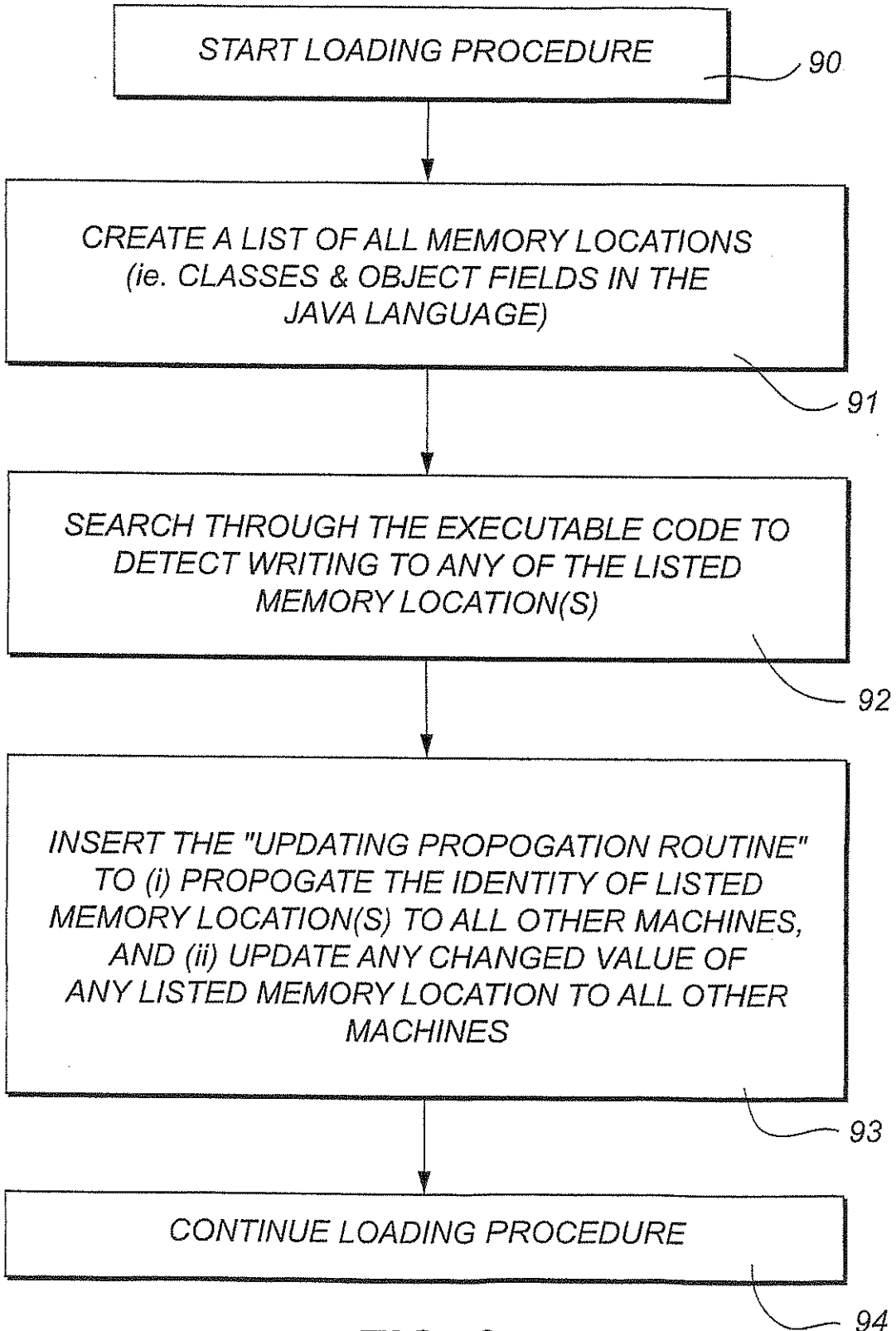

START LOADING PROCEDURE
90
CREATE A LIST OF ALL MEMORY LOCATIONS (ie. CLASSES & OBJECT FIELDS IN THE JAVA LANGUAGE)
91
SEARCH THROUGH THE EXECUTABLE CODE TO DETECT WRITING TO ANY OF THE LISTED MEMORY LOCATION(S)
92
INSERT THE "UPDATING PROPOGATION ROUTINE" TO (i) PROPOGATE THE IDENTITY OF LISTED MEMORY LOCATION(S) TO ALL OTHER MACHINES, AND (ii) UPDATE ANY CHANGED VALUE OF ANY LISTED MEMORY LOCATION TO ALL OTHER MACHINES
93
CONTINUE LOADING PROCEDURE
94

FIG. 9

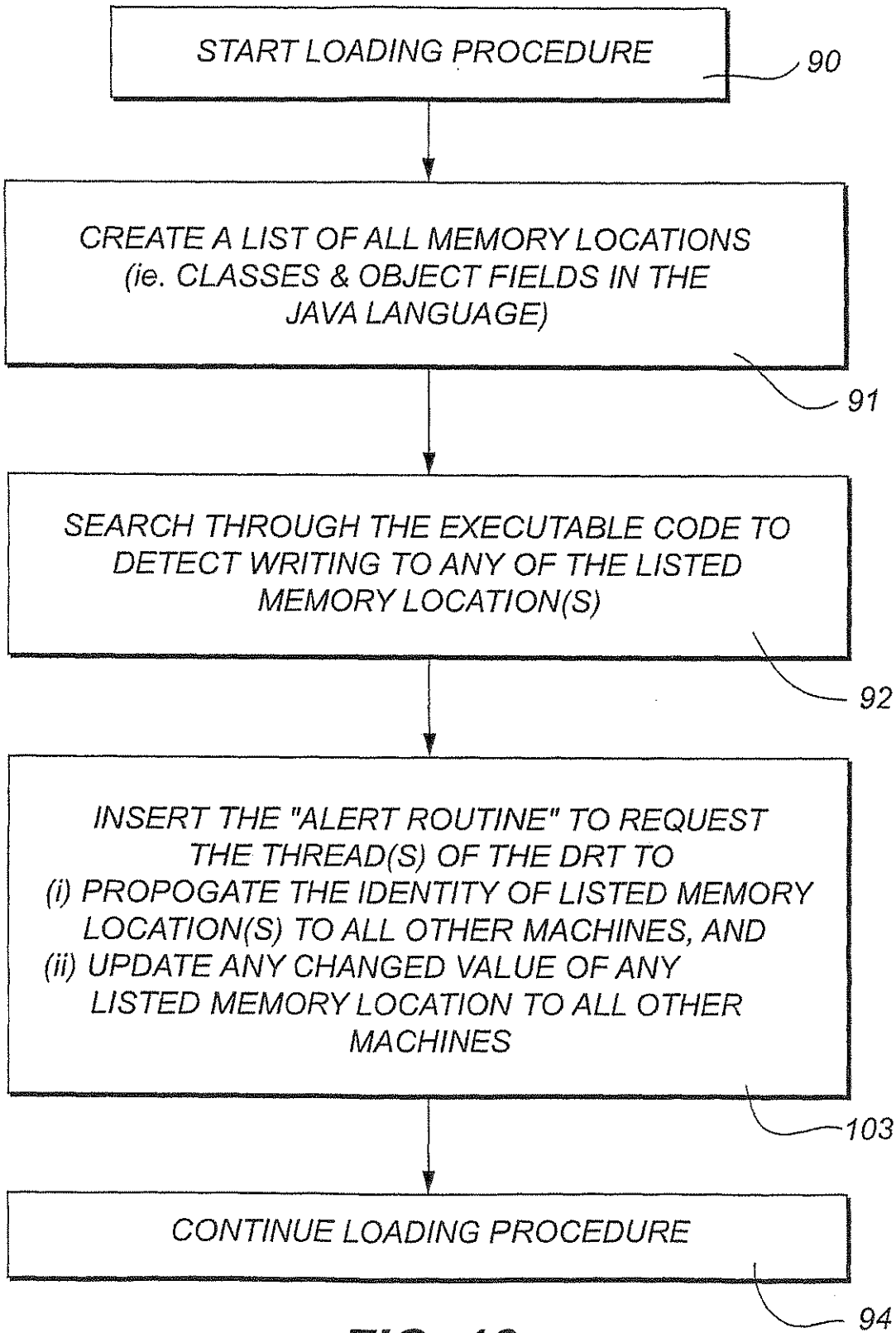

START LOADING PROCEDURE
90
CREATE A LIST OF ALL MEMORY LOCATIONS (ie. CLASSES & OBJECT FIELDS IN THE JAVA LANGUAGE)
91
SEARCH THROUGH THE EXECUTABLE CODE TO DETECT WRITING TO ANY OF THE LISTED MEMORY LOCATION(S)
92
INSERT THE "ALERT ROUTINE" TO REQUEST THE THREAD(S) OF THE DRT TO
(i) PROPOGATE THE IDENTITY OF LISTED MEMORY LOCATION(S) TO ALL OTHER MACHINES, AND
(ii) UPDATE ANY CHANGED VALUE OF ANY LISTED MEMORY LOCATION TO ALL OTHER MACHINES
103
CONTINUE LOADING PROCEDURE
94

FIG. 10

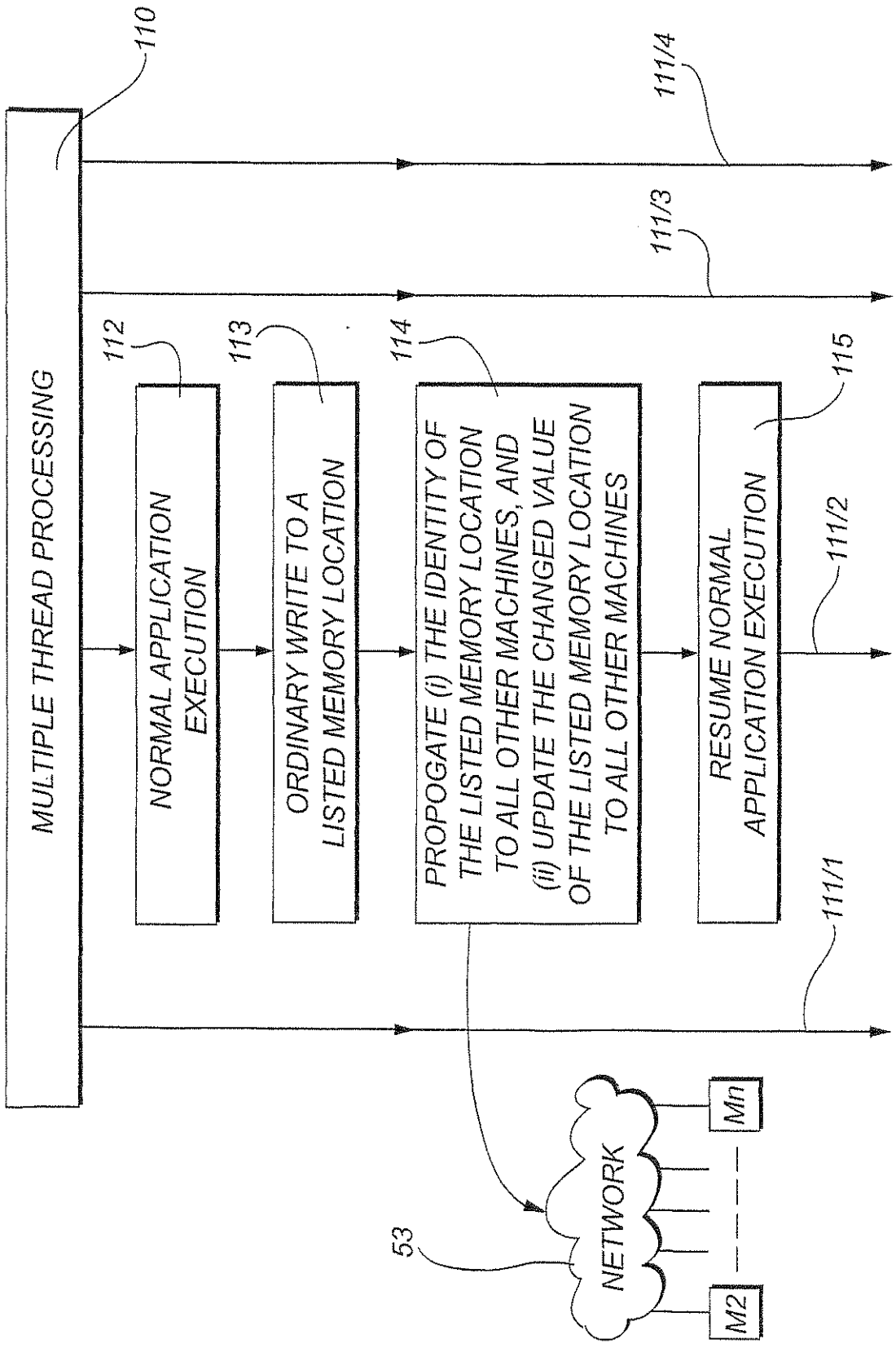

MULTIPLE THREAD PROCESSING
110
NORMAL APPLICATION EXECUTION
112
ORDINARY WRITE TO A LISTED MEMORY LOCATION
113
PROPOGATE (i) THE IDENTITY OF THE LISTED MEMORY LOCATION TO ALL OTHER MACHINES, AND (ii) UPDATE THE CHANGED VALUE OF THE LISTED MEMORY LOCATION TO ALL OTHER MACHINES
114
RESUME NORMAL APPLICATION EXECUTION
115
111/1
111/2
111/3
111/4
NETWORK
53
M2
Mn

FIG.11

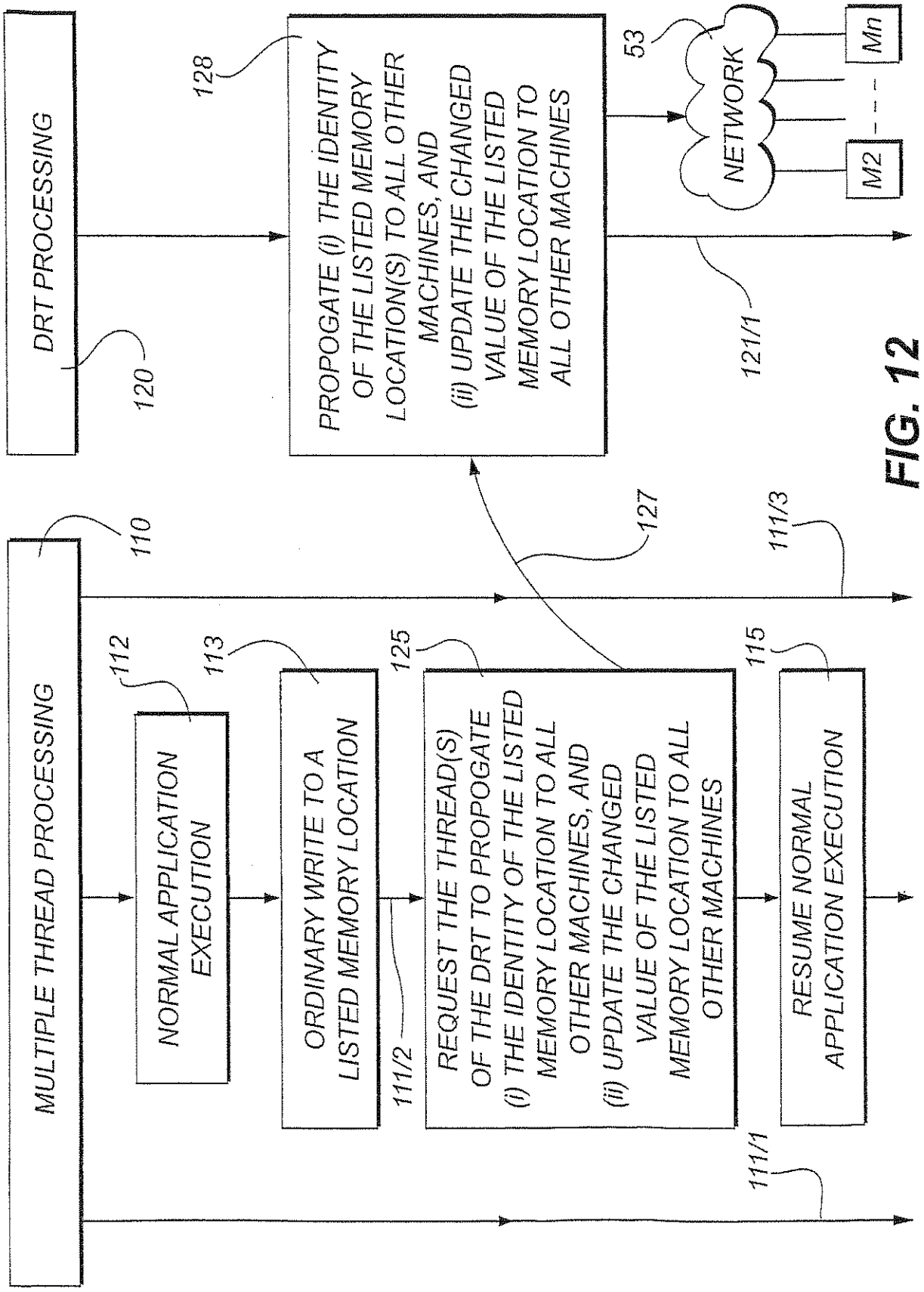

MULTIPLE THREAD PROCESSING
110
NORMAL APPLICATION EXECUTION
112
ORDINARY WRITE TO A LISTED MEMORY LOCATION
113
111/2
REQUEST THE THREAD(S) OF THE DRT TO PROPOGATE (i) THE IDENTITY OF THE LISTED MEMORY LOCATION TO ALL OTHER MACHINES, AND (ii) UPDATE THE CHANGED VALUE OF THE LISTED MEMORY LOCATION TO ALL OTHER MACHINES
125
127
RESUME NORMAL APPLICATION EXECUTION
115
111/1
111/3
DRT PROCESSING
120
PROPOGATE (i) THE IDENTITY OF THE LISTED MEMORY LOCATION(S) TO ALL OTHER MACHINES, AND (ii) UPDATE THE CHANGED VALUE OF THE LISTED MEMORY LOCATION TO ALL OTHER MACHINES
128
121/1
53
NETWORK
M2
Mn

FIG. 12

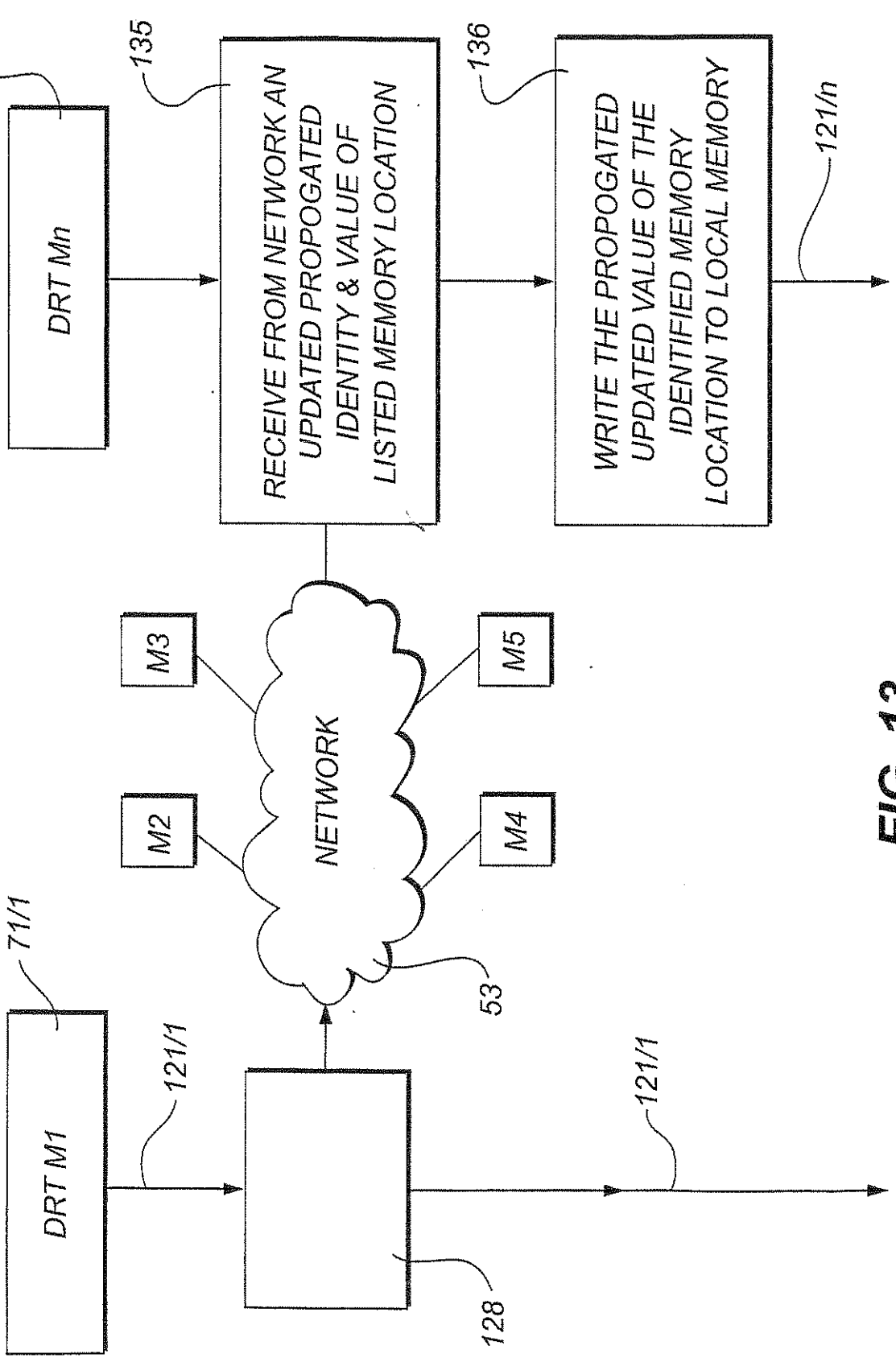

71/n
135
136
DRT Mn
RECEIVE FROM NETWORK AN UPDATED PROPOGATED IDENTITY & VALUE OF LISTED MEMORY LOCATION
WRITE THE PROPOGATED UPDATED VALUE OF THE IDENTIFIED MEMORY LOCATION TO LOCAL MEMORY
121/n
M3
M5
M2
M4
NETWORK
53
71/1
DRT M1
121/1
121/1
128

FIG. 13

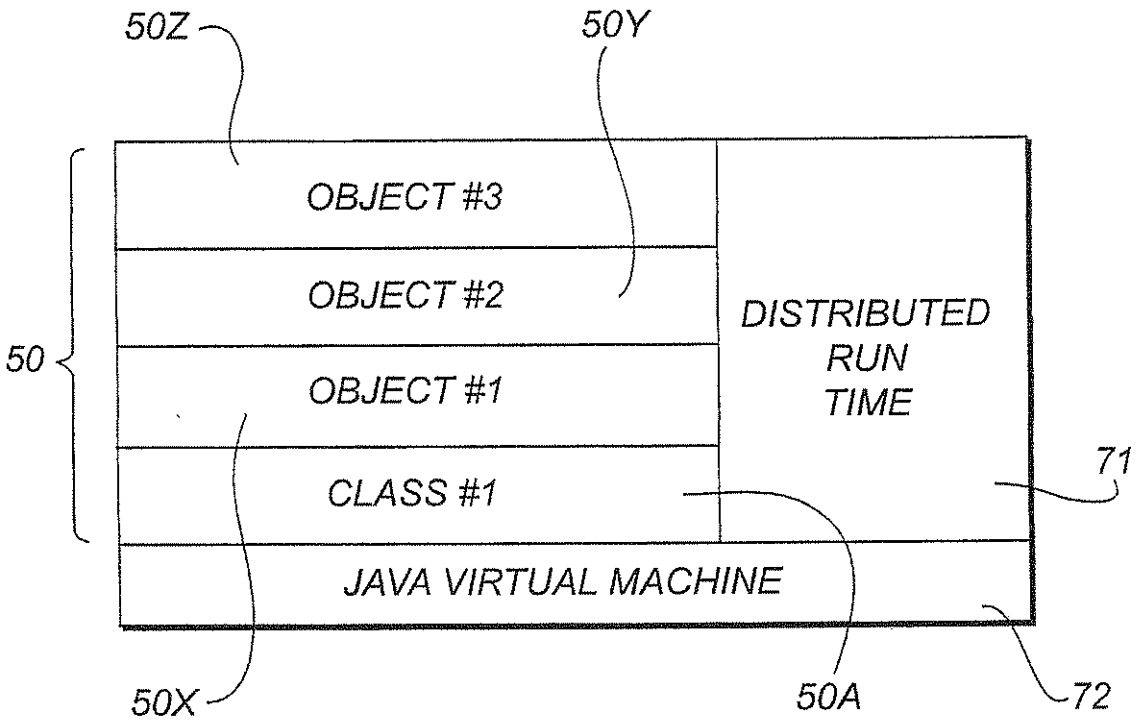


FIG. 14
PRIOR ART

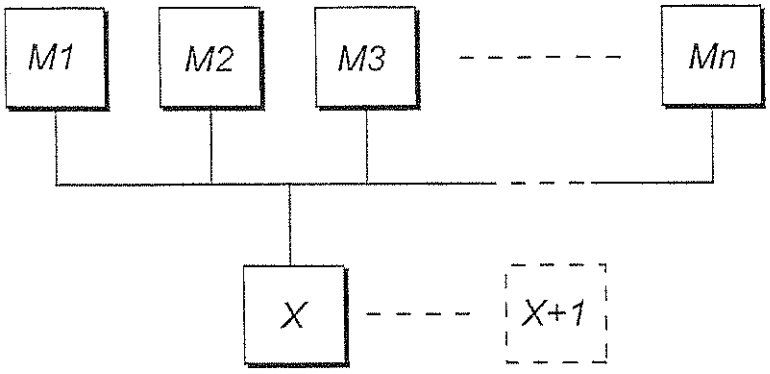

M1
M2
M3
Mn
X
X+1

FIG. 15

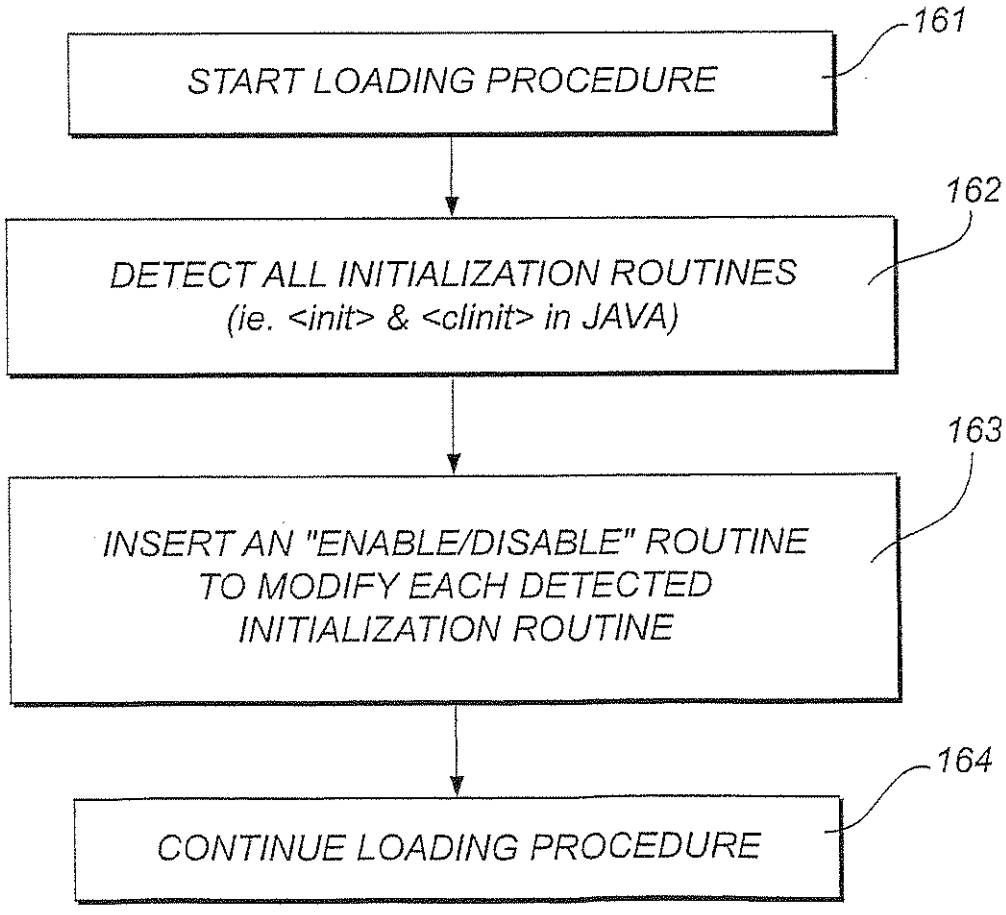

START LOADING PROCEDURE
161
DETECT ALL INITIALIZATION ROUTINES
(ie. <init> & <clinit> in JAVA)
162
INSERT AN "ENABLE/DISABLE" ROUTINE
TO MODIFY EACH DETECTED
INITIALIZATION ROUTINE
163
CONTINUE LOADING PROCEDURE
164

FIG. 16

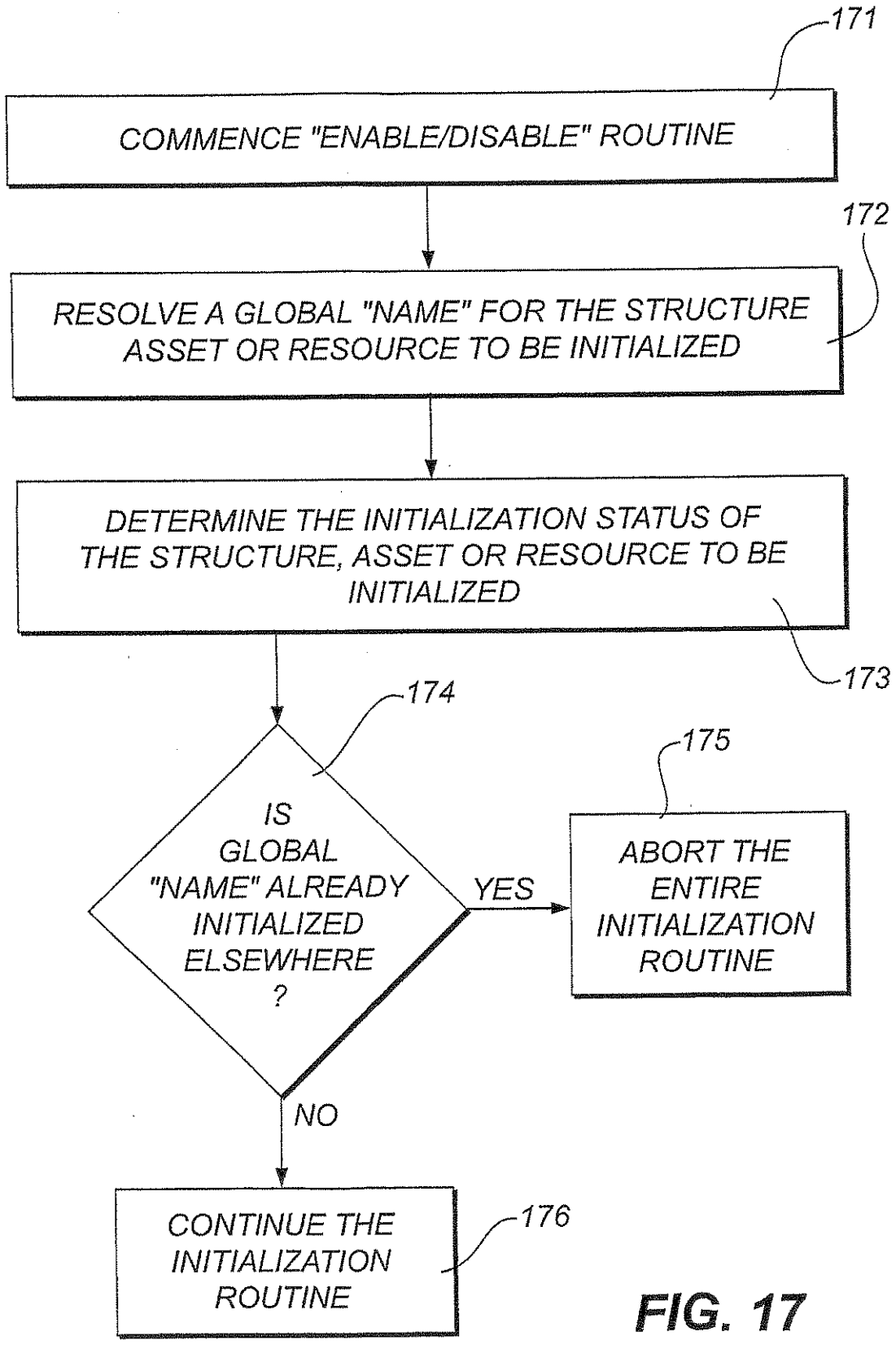

171
COMMENCE "ENABLE/DISABLE" ROUTINE
172
RESOLVE A GLOBAL "NAME" FOR THE STRUCTURE ASSET OR RESOURCE TO BE INITIALIZED
DETERMINE THE INITIALIZATION STATUS OF THE STRUCTURE, ASSET OR RESOURCE TO BE INITIALIZED
173
174
IS GLOBAL "NAME" ALREADY INITIALIZED ELSEWHERE ?
YES
175
ABORT THE ENTIRE INITIALIZATION ROUTINE
NO
CONTINUE THE INITIALIZATION ROUTINE
176

FIG. 17

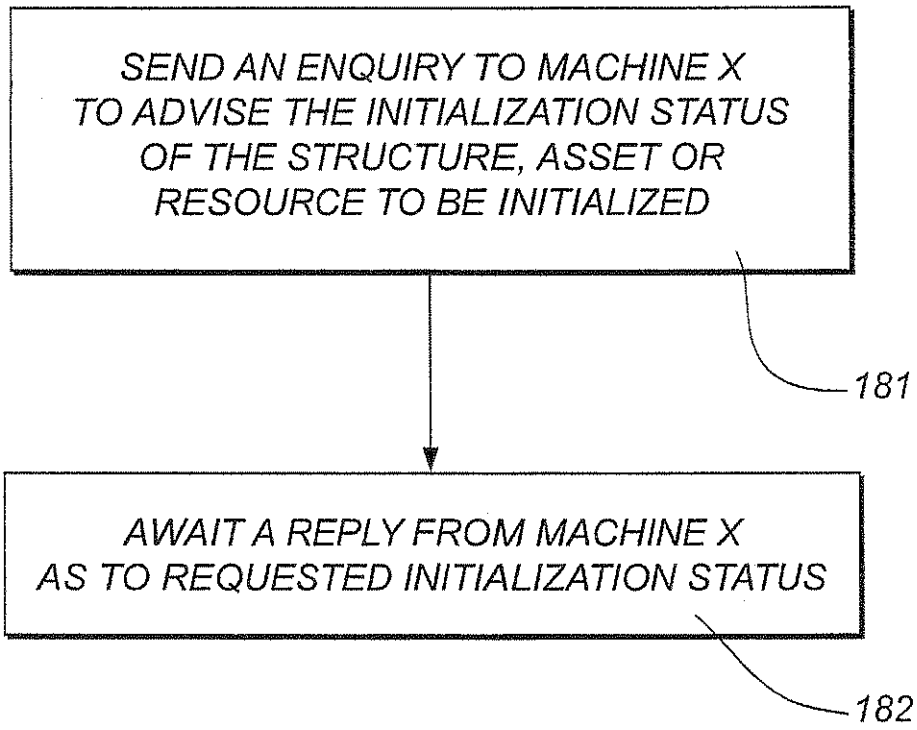

SEND AN ENQUIRY TO MACHINE X TO ADVISE THE INITIALIZATION STATUS OF THE STRUCTURE, ASSET OR RESOURCE TO BE INITIALIZED
181
AWAIT A REPLY FROM MACHINE X AS TO REQUESTED INITIALIZATION STATUS
182

FIG. 18

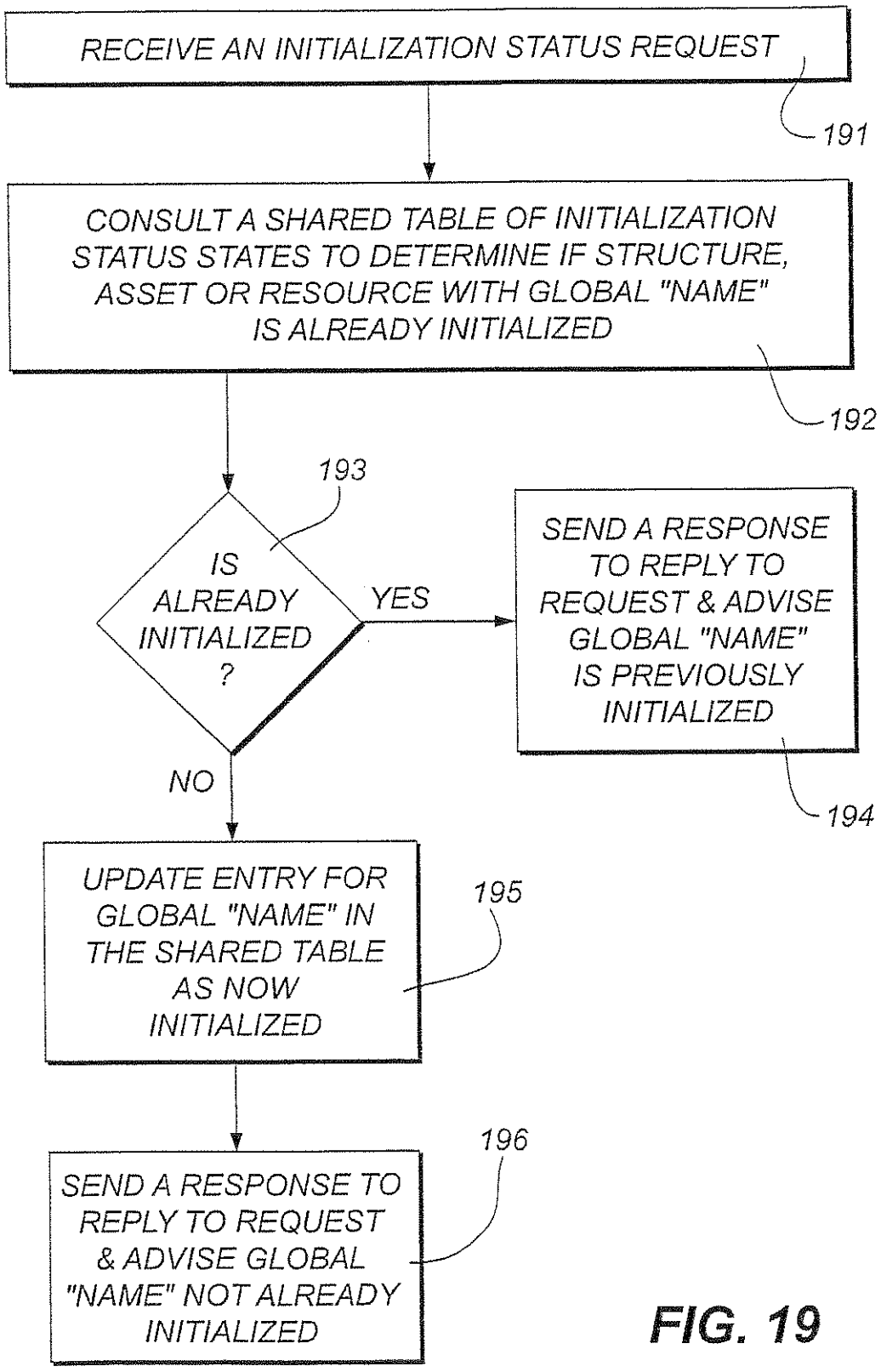

RECEIVE AN INITIALIZATION STATUS REQUEST
191
CONSULT A SHARED TABLE OF INITIALIZATION STATUS STATES TO DETERMINE IF STRUCTURE, ASSET OR RESOURCE WITH GLOBAL "NAME" IS ALREADY INITIALIZED
192
193
IS ALREADY INITIALIZED ?
YES
SEND A RESPONSE TO REPLY TO REQUEST & ADVISE GLOBAL "NAME" IS PREVIOUSLY INITIALIZED
194
NO
UPDATE ENTRY FOR GLOBAL "NAME" IN THE SHARED TABLE AS NOW INITIALIZED
195
SEND A RESPONSE TO REPLY TO REQUEST & ADVISE GLOBAL "NAME" NOT ALREADY INITIALIZED
196

FIG. 19

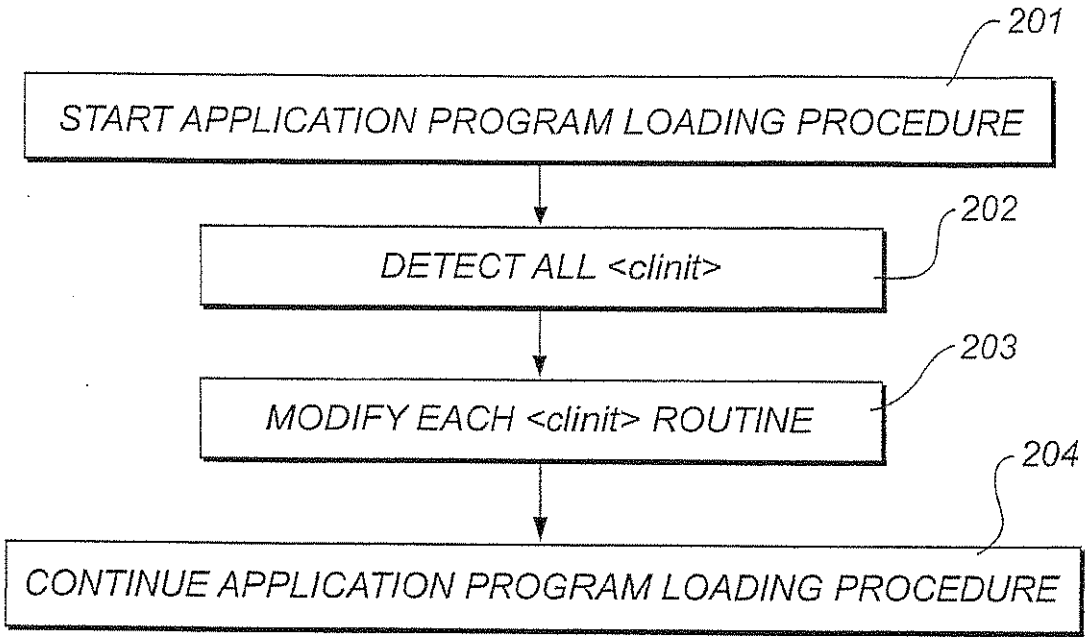

201
START APPLICATION PROGRAM LOADING PROCEDURE
202
DETECT ALL <clinit>
203
MODIFY EACH <clinit> ROUTINE
204
CONTINUE APPLICATION PROGRAM LOADING PROCEDURE

FIG. 20

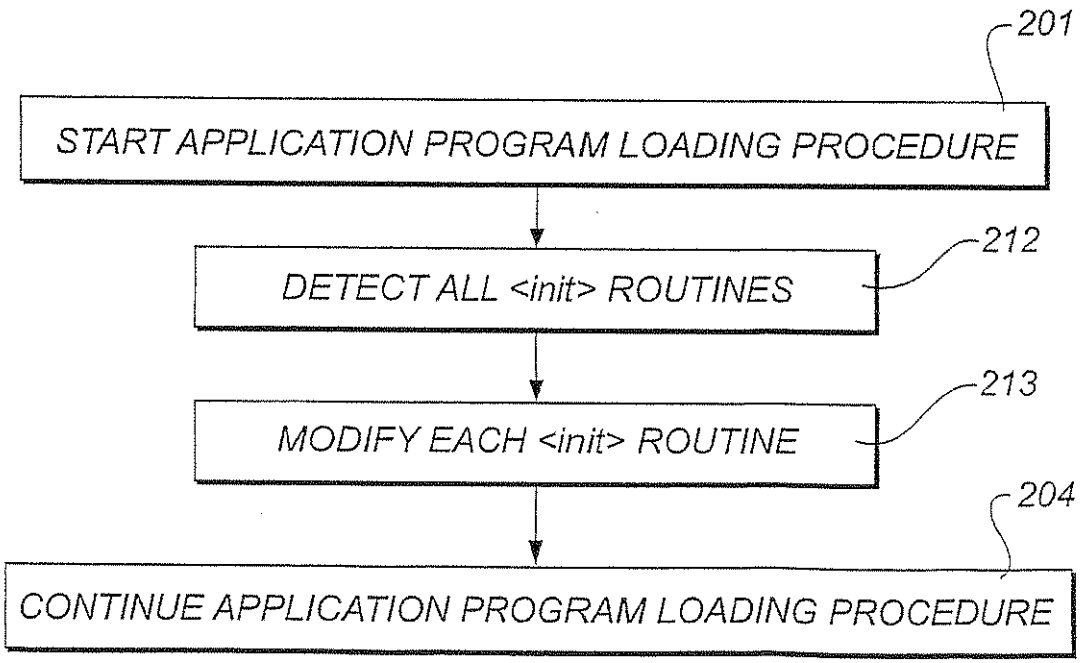

201
START APPLICATION PROGRAM LOADING PROCEDURE
212
DETECT ALL <init> ROUTINES
213
MODIFY EACH <init> ROUTINE
204
CONTINUE APPLICATION PROGRAM LOADING PROCEDURE

FIG. 21

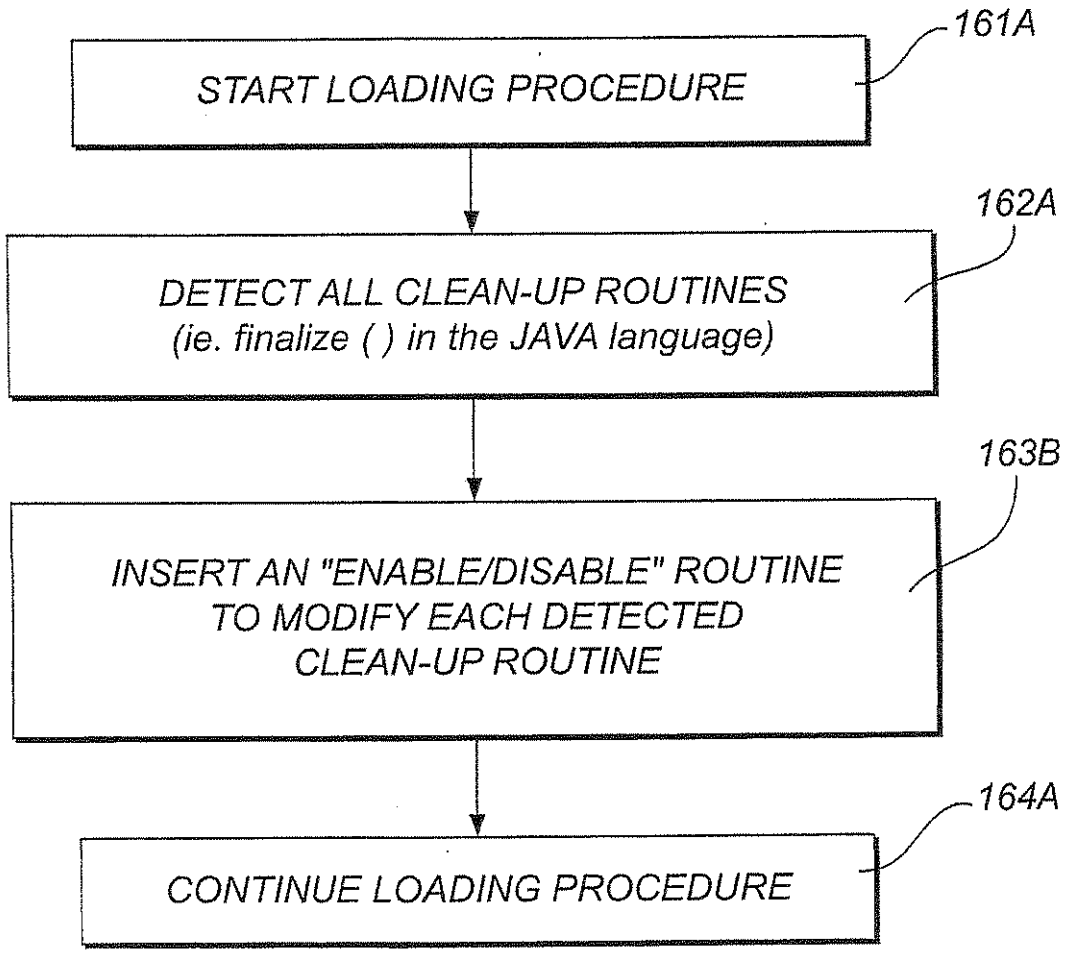

161A
START LOADING PROCEDURE
162A
DETECT ALL CLEAN-UP ROUTINES
(ie. finalize ( ) in the JAVA language)
163B
INSERT AN "ENABLE/DISABLE" ROUTINE
TO MODIFY EACH DETECTED
CLEAN-UP ROUTINE
164A
CONTINUE LOADING PROCEDURE

FIG. 22

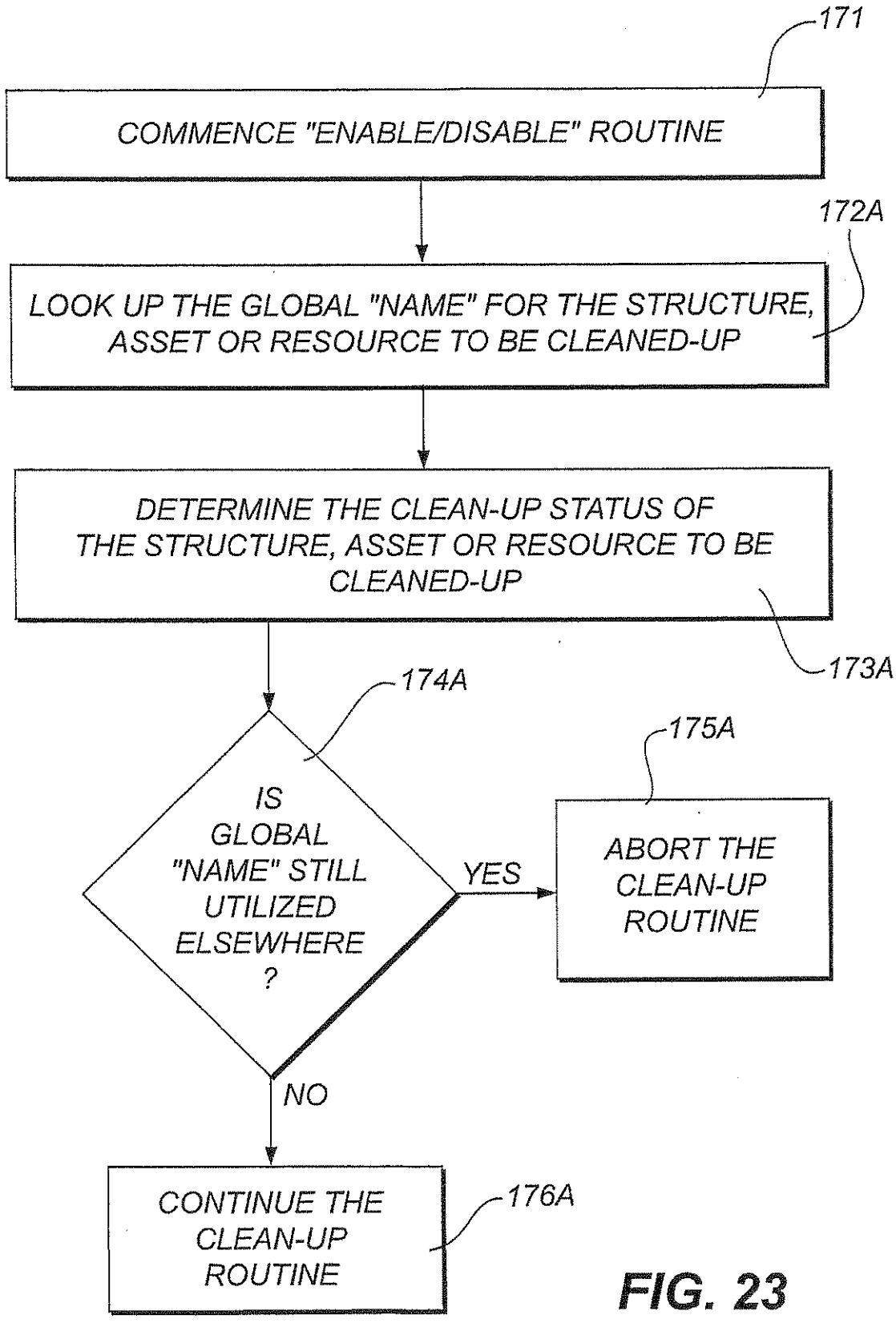

171
COMMENCE "ENABLE/DISABLE" ROUTINE
172A
LOOK UP THE GLOBAL "NAME" FOR THE STRUCTURE, ASSET OR RESOURCE TO BE CLEANED-UP
DETERMINE THE CLEAN-UP STATUS OF THE STRUCTURE, ASSET OR RESOURCE TO BE CLEANED-UP
173A
174A
IS GLOBAL "NAME" STILL UTILIZED ELSEWHERE ?
YES
175A
ABORT THE CLEAN-UP ROUTINE
NO
CONTINUE THE CLEAN-UP ROUTINE
176A

FIG. 23

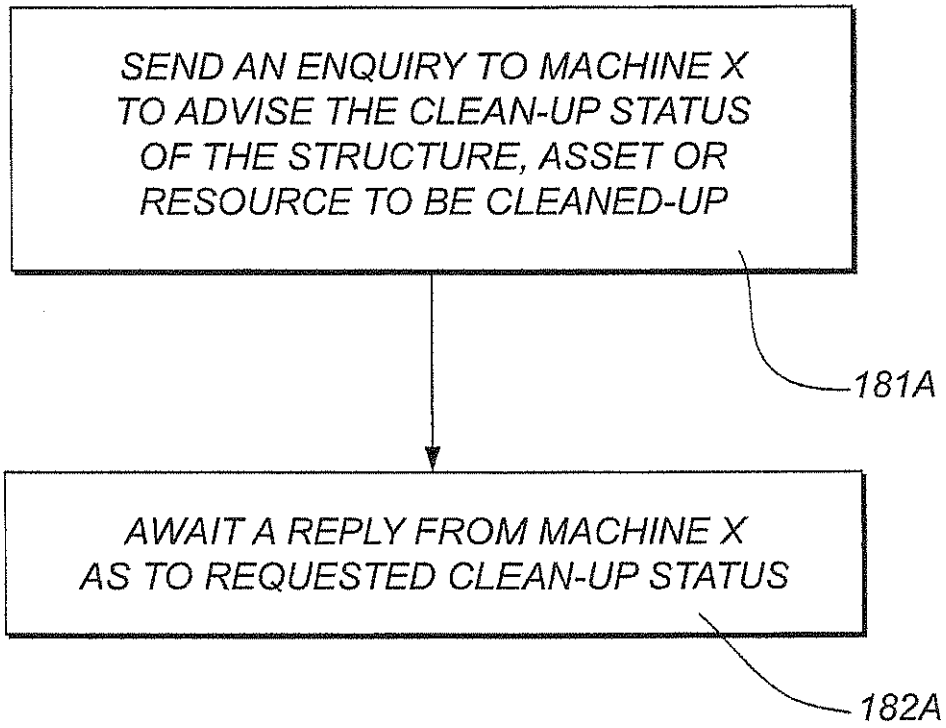

SEND AN ENQUIRY TO MACHINE X TO ADVISE THE CLEAN-UP STATUS OF THE STRUCTURE, ASSET OR RESOURCE TO BE CLEANED-UP
181A
AWAIT A REPLY FROM MACHINE X AS TO REQUESTED CLEAN-UP STATUS
182A

FIG. 24

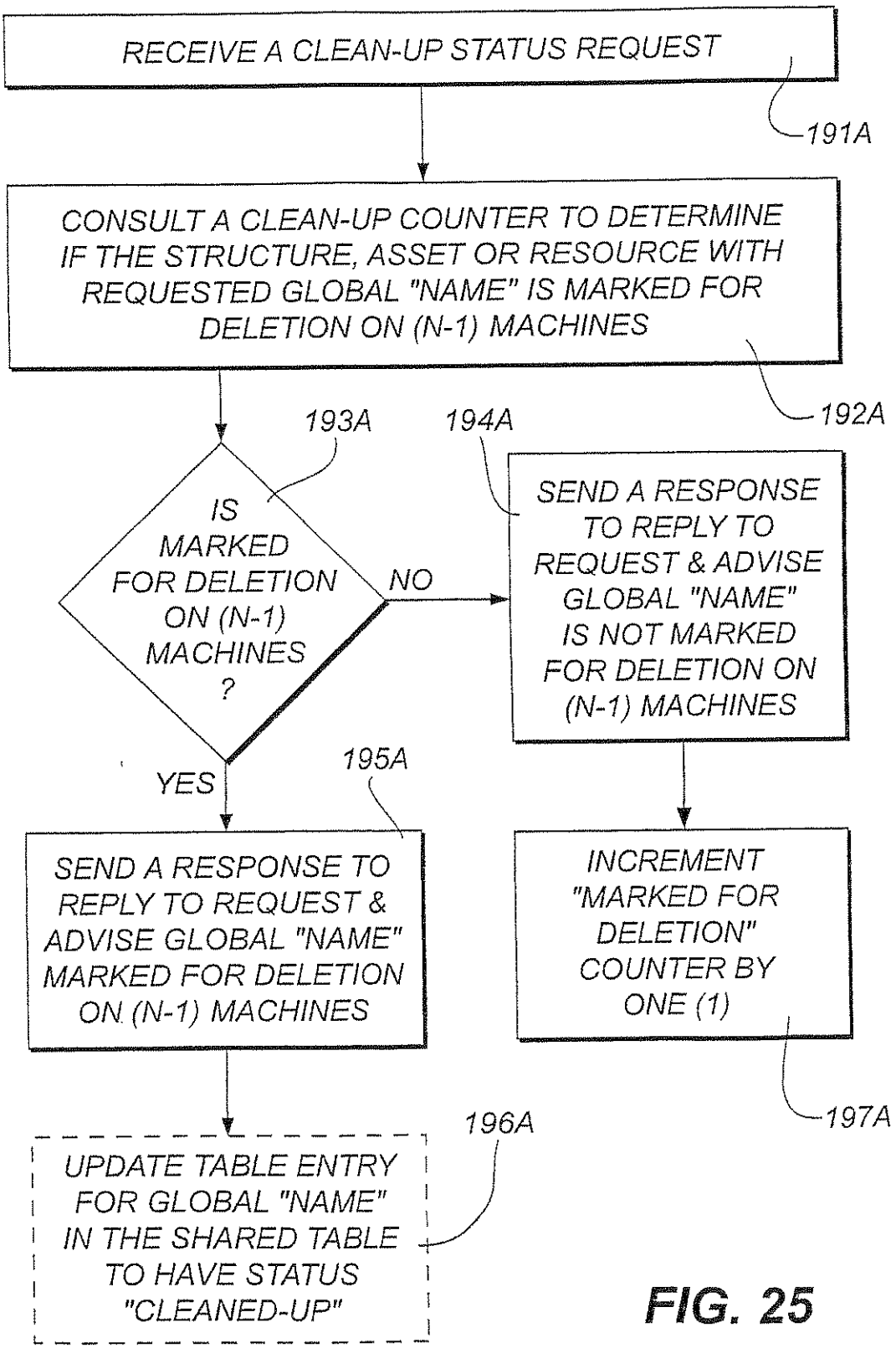

RECEIVE A CLEAN-UP STATUS REQUEST
191A
CONSULT A CLEAN-UP COUNTER TO DETERMINE IF THE STRUCTURE, ASSET OR RESOURCE WITH REQUESTED GLOBAL "NAME" IS MARKED FOR DELETION ON (N-1) MACHINES
192A
193A
IS MARKED FOR DELETION ON (N-1) MACHINES ?
NO
194A
SEND A RESPONSE TO REPLY TO REQUEST & ADVISE GLOBAL "NAME" IS NOT MARKED FOR DELETION ON (N-1) MACHINES
YES
195A
SEND A RESPONSE TO REPLY TO REQUEST & ADVISE GLOBAL "NAME" MARKED FOR DELETION ON (N-1) MACHINES
INCREMENT "MARKED FOR DELETION" COUNTER BY ONE (1)
197A
196A
UPDATE TABLE ENTRY FOR GLOBAL "NAME" IN THE SHARED TABLE TO HAVE STATUS "CLEANED-UP"

FIG. 25

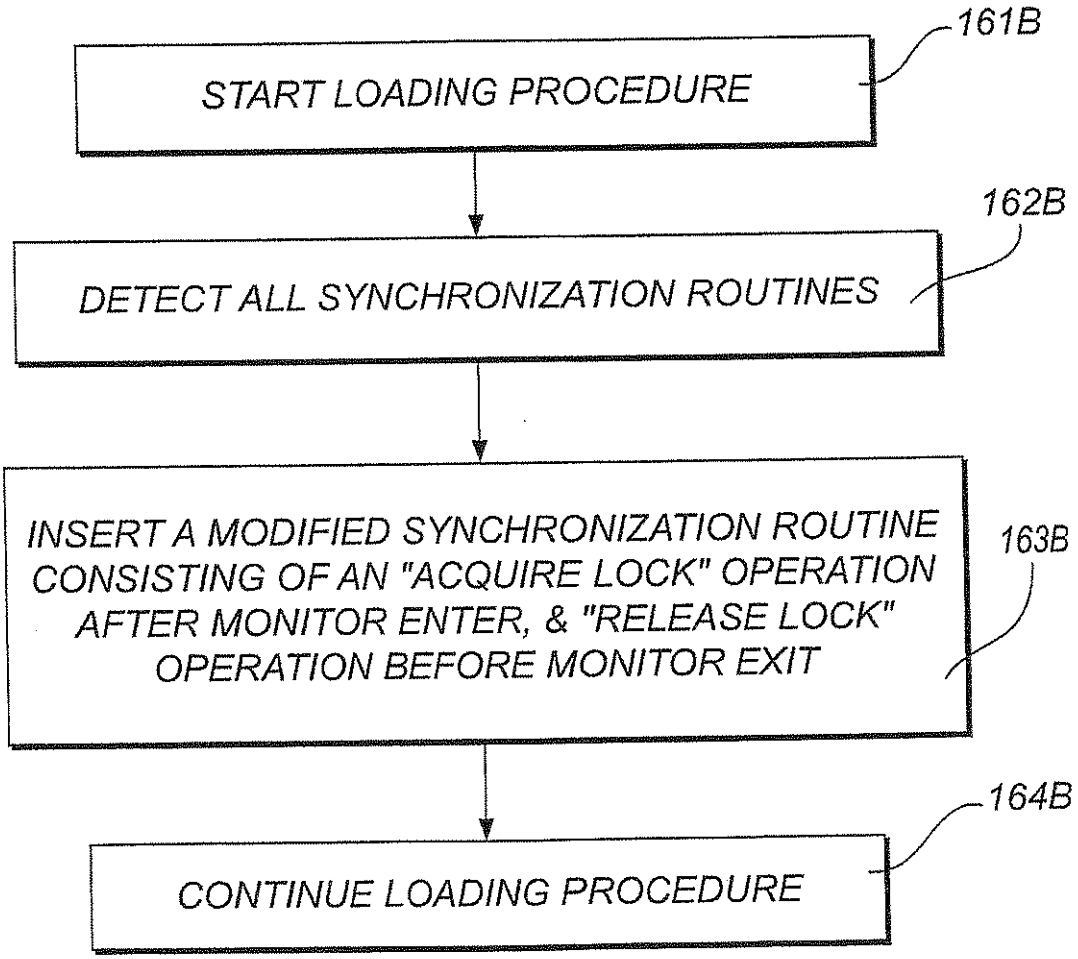

START LOADING PROCEDURE
161B
DETECT ALL SYNCHRONIZATION ROUTINES
162B
INSERT A MODIFIED SYNCHRONIZATION ROUTINE CONSISTING OF AN "ACQUIRE LOCK" OPERATION AFTER MONITOR ENTER, & "RELEASE LOCK" OPERATION BEFORE MONITOR EXIT
163B
CONTINUE LOADING PROCEDURE
164B

FIG. 26

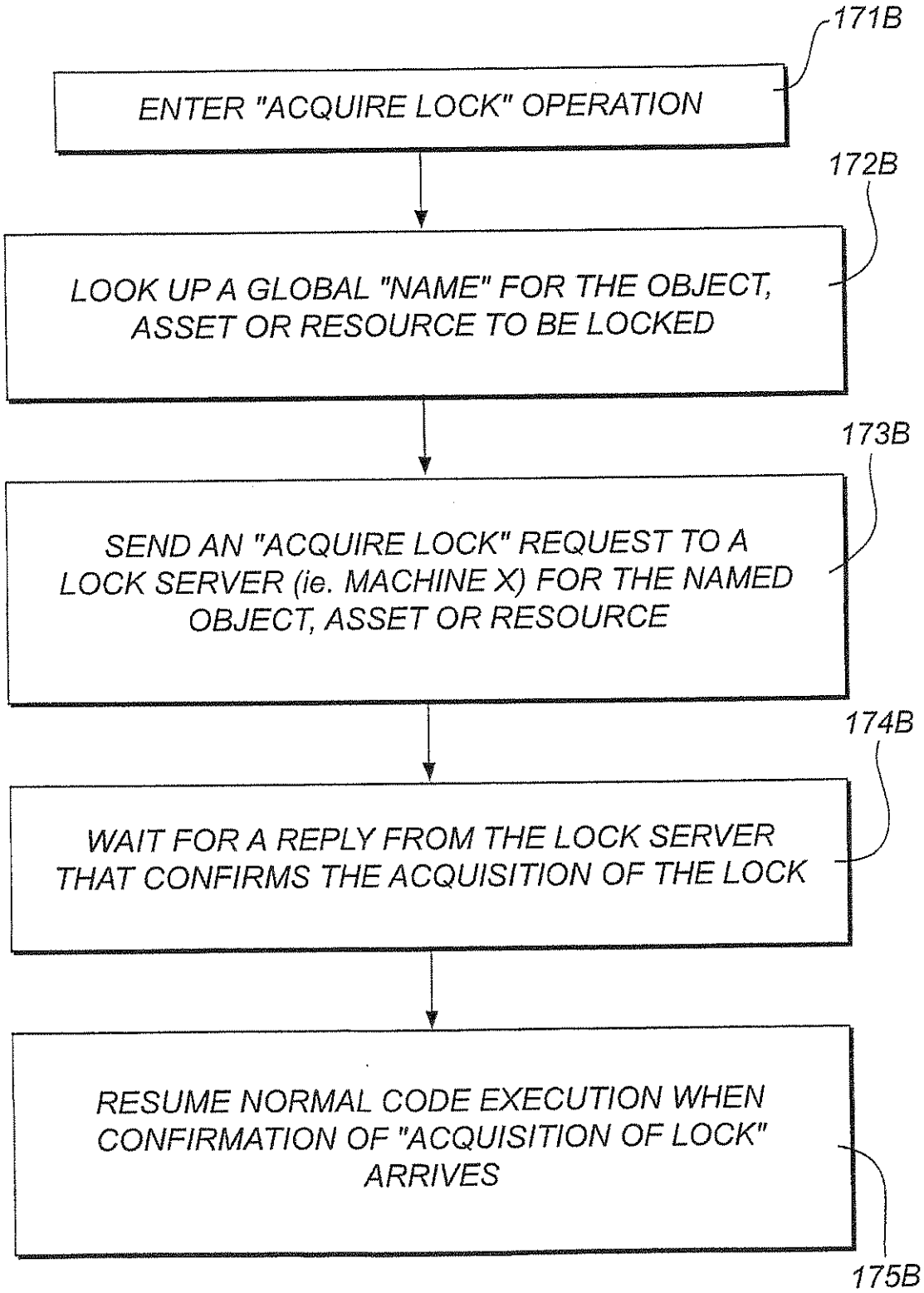

171B
ENTER "ACQUIRE LOCK" OPERATION
172B
LOOK UP A GLOBAL "NAME" FOR THE OBJECT, ASSET OR RESOURCE TO BE LOCKED
173B
SEND AN "ACQUIRE LOCK" REQUEST TO A LOCK SERVER (ie. MACHINE X) FOR THE NAMED OBJECT, ASSET OR RESOURCE
174B
WAIT FOR A REPLY FROM THE LOCK SERVER THAT CONFIRMS THE ACQUISITION OF THE LOCK
RESUME NORMAL CODE EXECUTION WHEN CONFIRMATION OF "ACQUISITION OF LOCK" ARRIVES
175B

FIG. 27

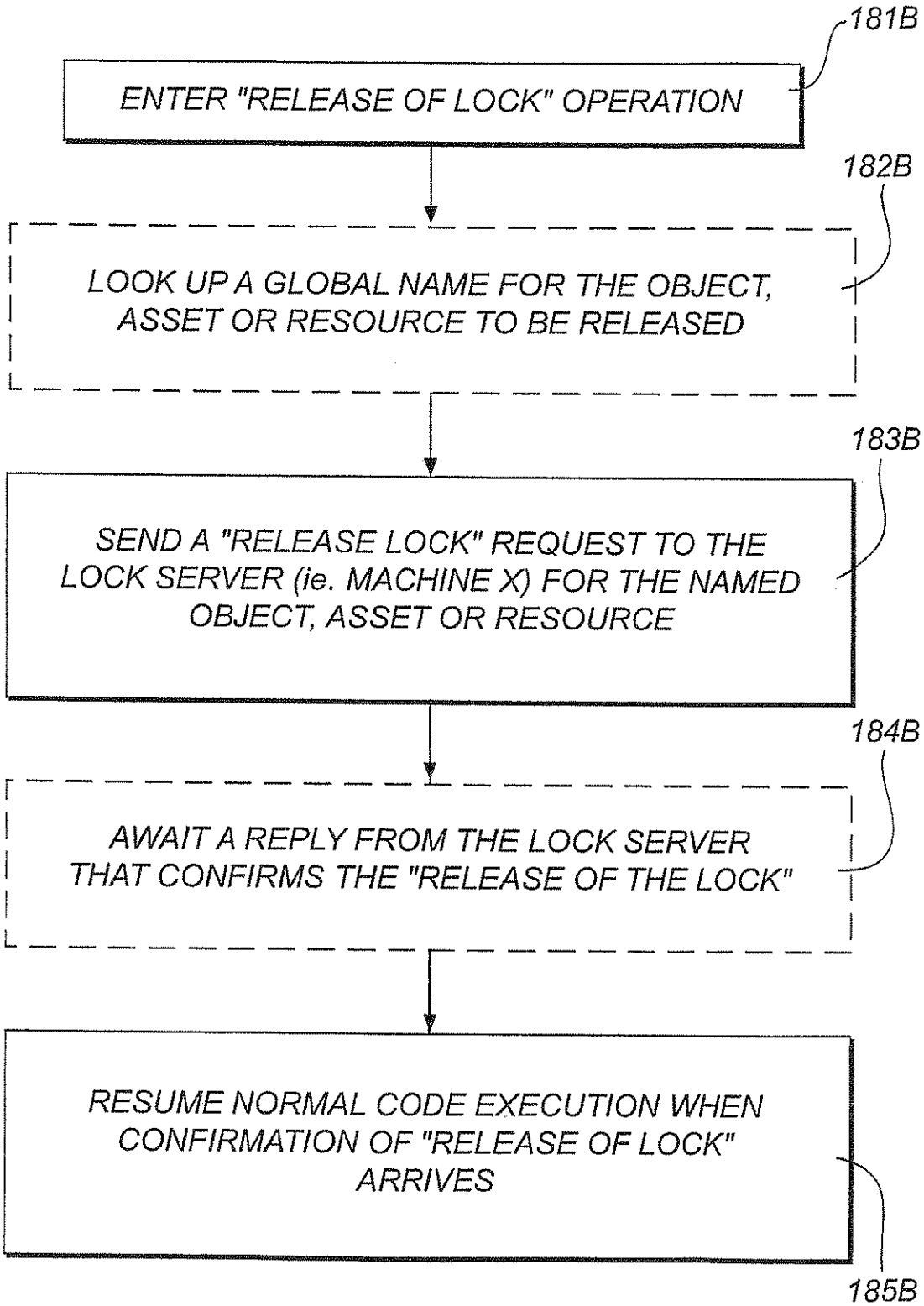

181B
ENTER "RELEASE OF LOCK" OPERATION
182B
LOOK UP A GLOBAL NAME FOR THE OBJECT, ASSET OR RESOURCE TO BE RELEASED
183B
SEND A "RELEASE LOCK" REQUEST TO THE LOCK SERVER (ie. MACHINE X) FOR THE NAMED OBJECT, ASSET OR RESOURCE
184B
AWAIT A REPLY FROM THE LOCK SERVER THAT CONFIRMS THE "RELEASE OF THE LOCK"
RESUME NORMAL CODE EXECUTION WHEN CONFIRMATION OF "RELEASE OF LOCK" ARRIVES
185B

FIG. 28

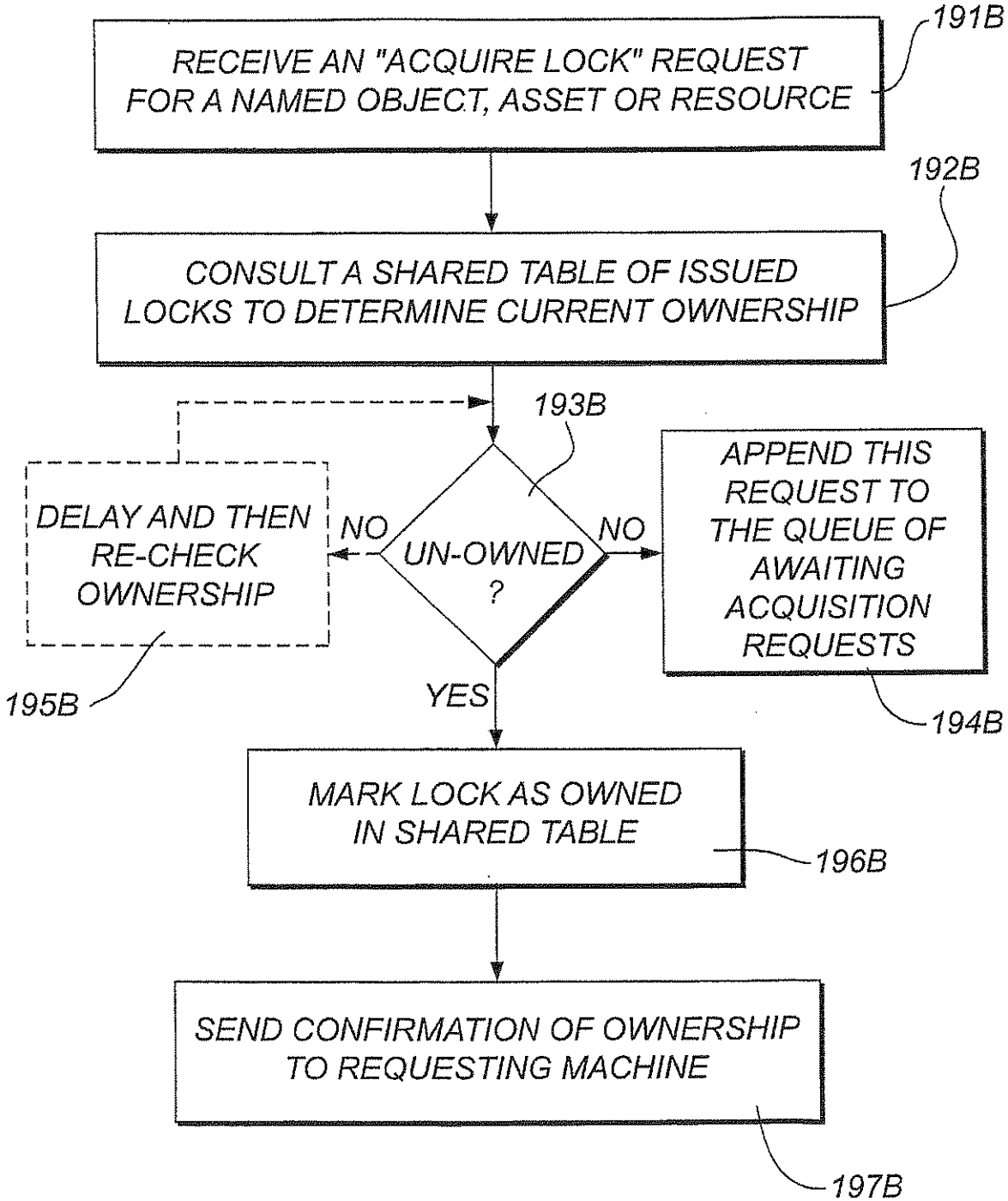

RECEIVE AN "ACQUIRE LOCK" REQUEST FOR A NAMED OBJECT, ASSET OR RESOURCE
191B
CONSULT A SHARED TABLE OF ISSUED LOCKS TO DETERMINE CURRENT OWNERSHIP
192B
193B
UN-OWNED ?
NO
NO
YES
DELAY AND THEN RE-CHECK OWNERSHIP
195B
APPEND THIS REQUEST TO THE QUEUE OF AWAITING ACQUISITION REQUESTS
194B
MARK LOCK AS OWNED IN SHARED TABLE
196B
SEND CONFIRMATION OF OWNERSHIP TO REQUESTING MACHINE
197B

FIG. 29

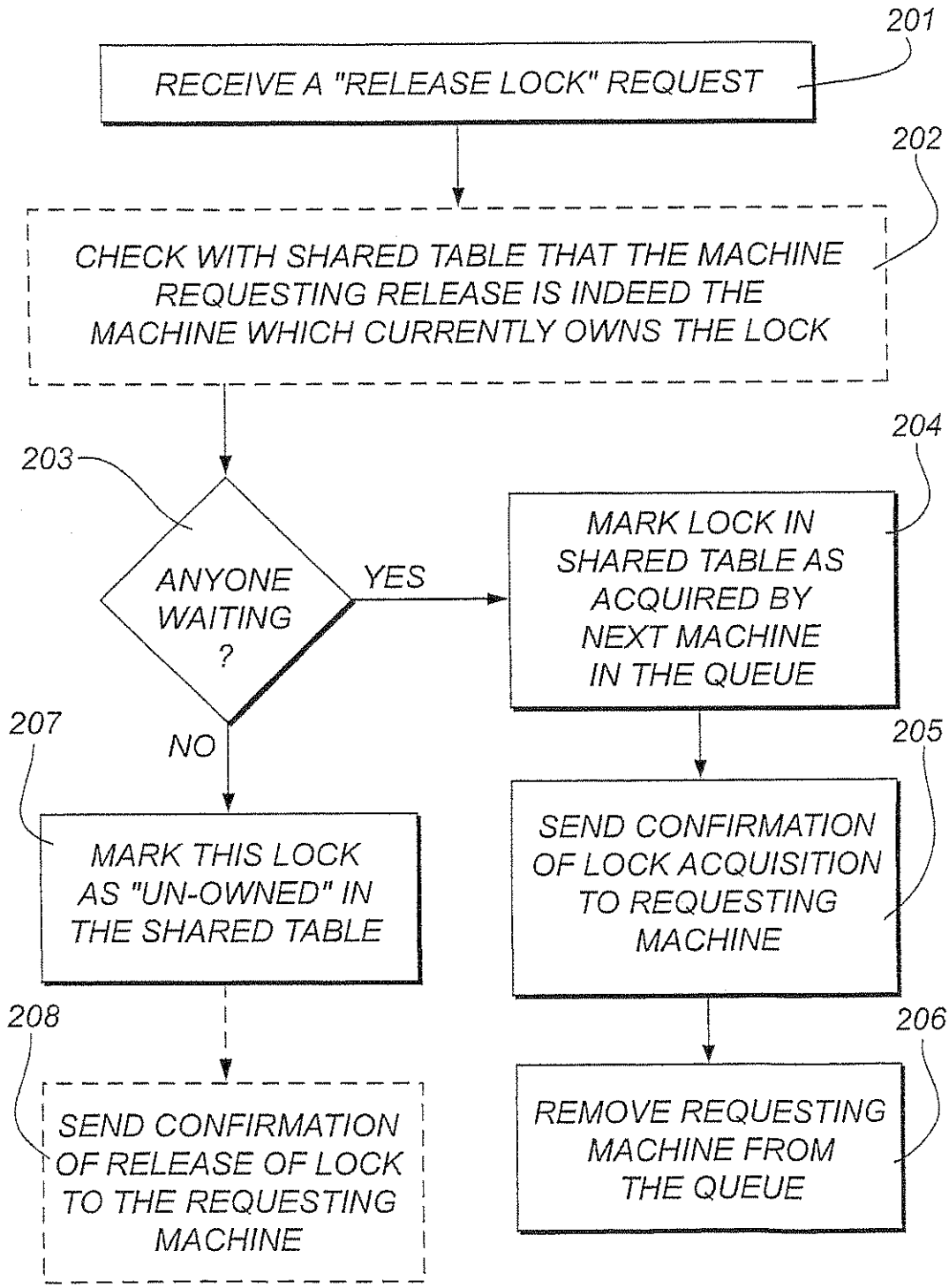

201
RECEIVE A "RELEASE LOCK" REQUEST
202
CHECK WITH SHARED TABLE THAT THE MACHINE REQUESTING RELEASE IS INDEED THE MACHINE WHICH CURRENTLY OWNS THE LOCK
203
ANYONE WAITING ?
YES
NO
204
MARK LOCK IN SHARED TABLE AS ACQUIRED BY NEXT MACHINE IN THE QUEUE
205
SEND CONFIRMATION OF LOCK ACQUISITION TO REQUESTING MACHINE
206
REMOVE REQUESTING MACHINE FROM THE QUEUE
207
MARK THIS LOCK AS "UN-OWNED" IN THE SHARED TABLE
208
SEND CONFIRMATION OF RELEASE OF LOCK TO THE REQUESTING MACHINE

FIG. 30

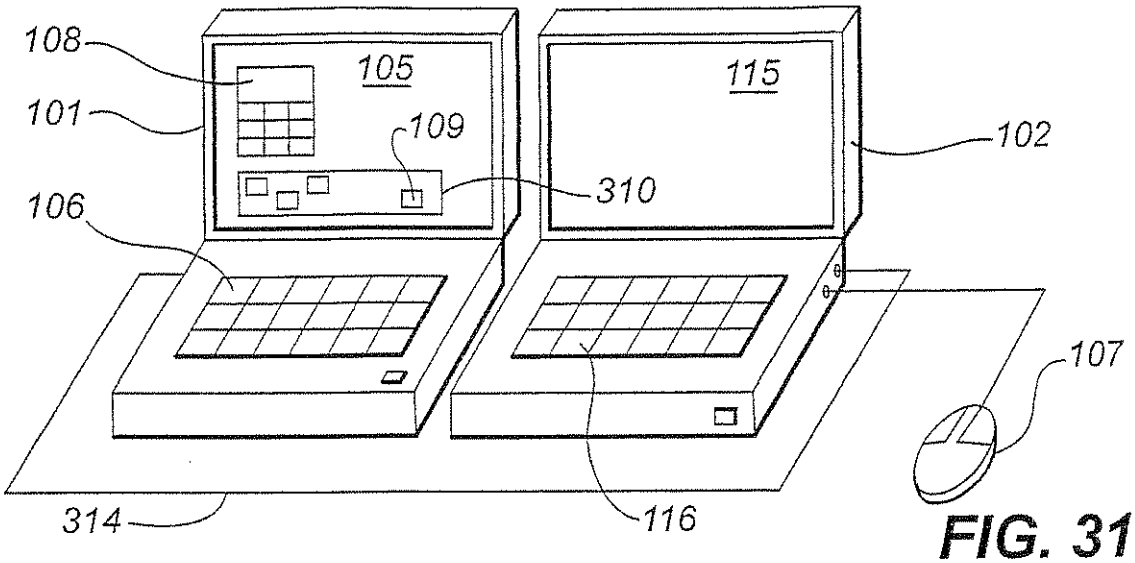


FIG. 31

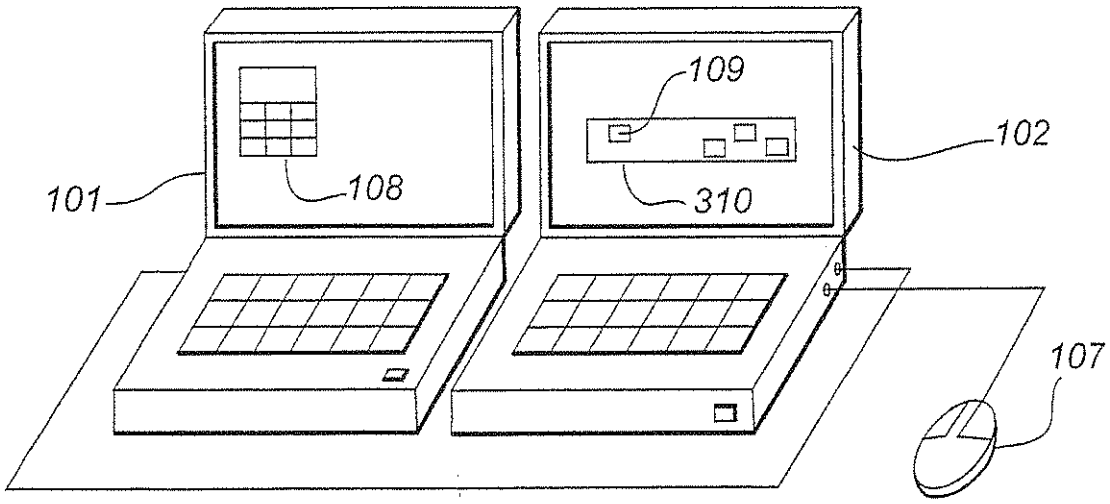


FIG. 32

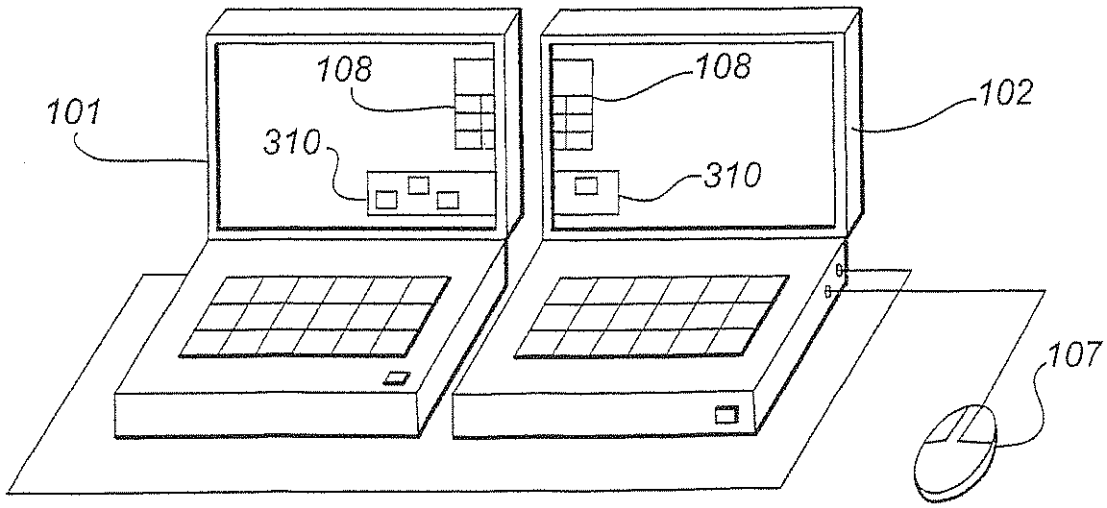


FIG. 33