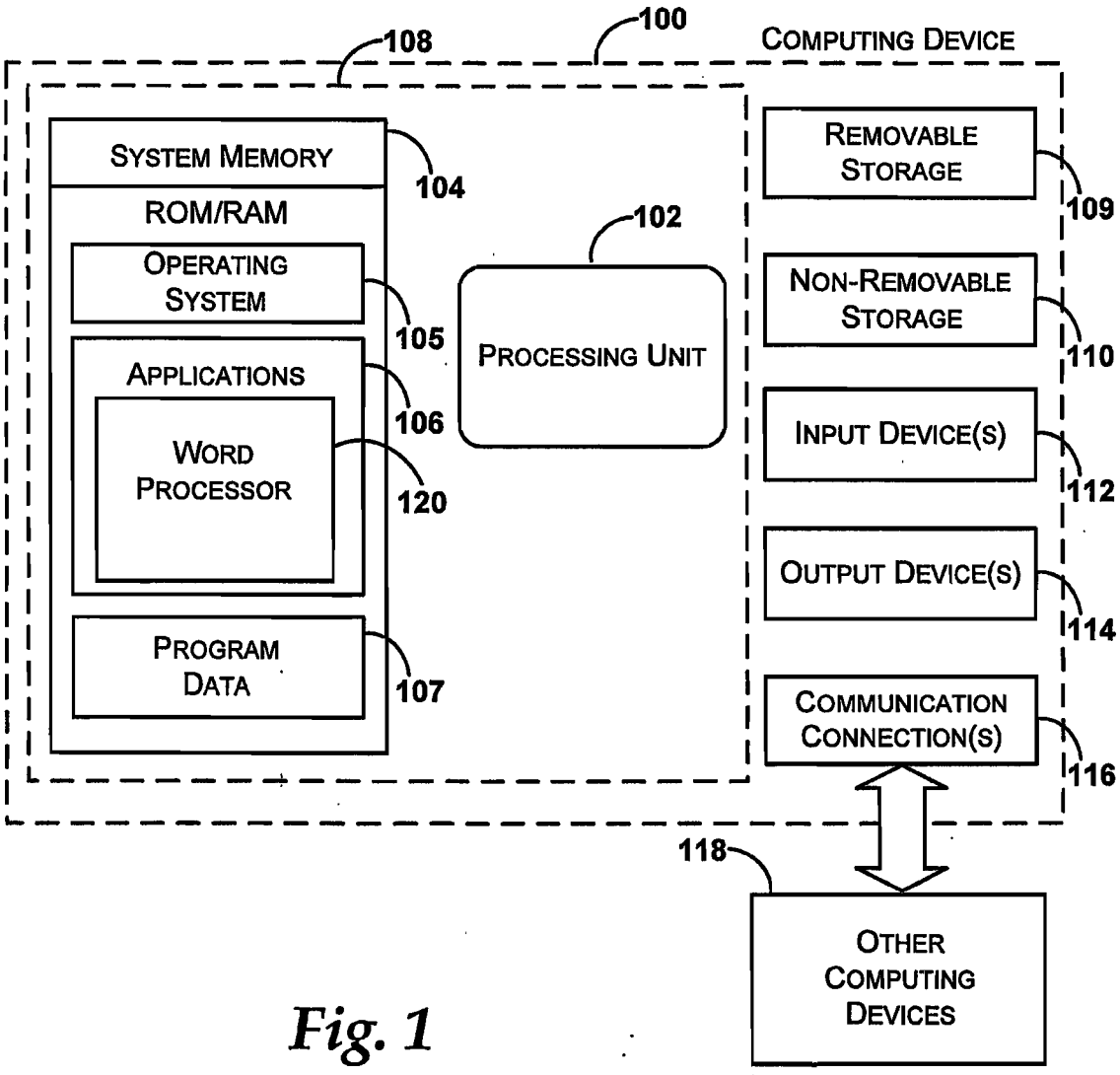

100
COMPUTING DEVICE
108
SYSTEM MEMORY
104
ROM/RAM
OPERATING SYSTEM
105
APPLICATIONS
106
WORD PROCESSOR
120
PROGRAM DATA
107
102
PROCESSING UNIT
REMOVABLE STORAGE
109
NON-REMOVABLE STORAGE
110
INPUT DEVICE(S)
112
OUTPUT DEVICE(S)
114
COMMUNICATION CONNECTION(S)
116
118
OTHER COMPUTING DEVICES

Fig. 1

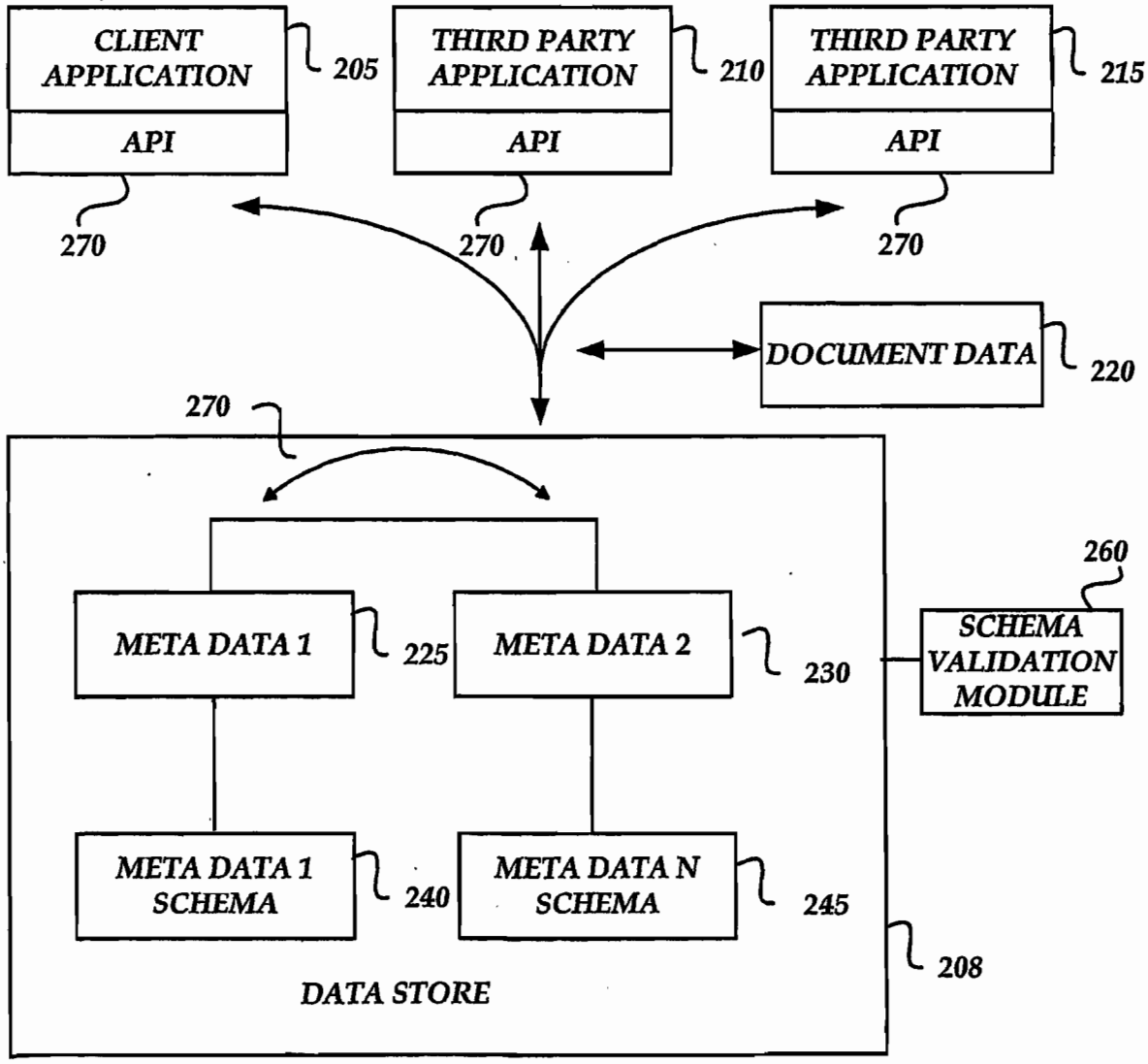

CLIENT APPLICATION
205
API
THIRD PARTY APPLICATION
210
API
THIRD PARTY APPLICATION
215
API
270
270
270
DOCUMENT DATA
220
270
260
META DATA 1
225
META DATA 2
230
SCHEMA VALIDATION MODULE
META DATA 1 SCHEMA
240
META DATA N SCHEMA
245
208
DATA STORE

Fig. 2

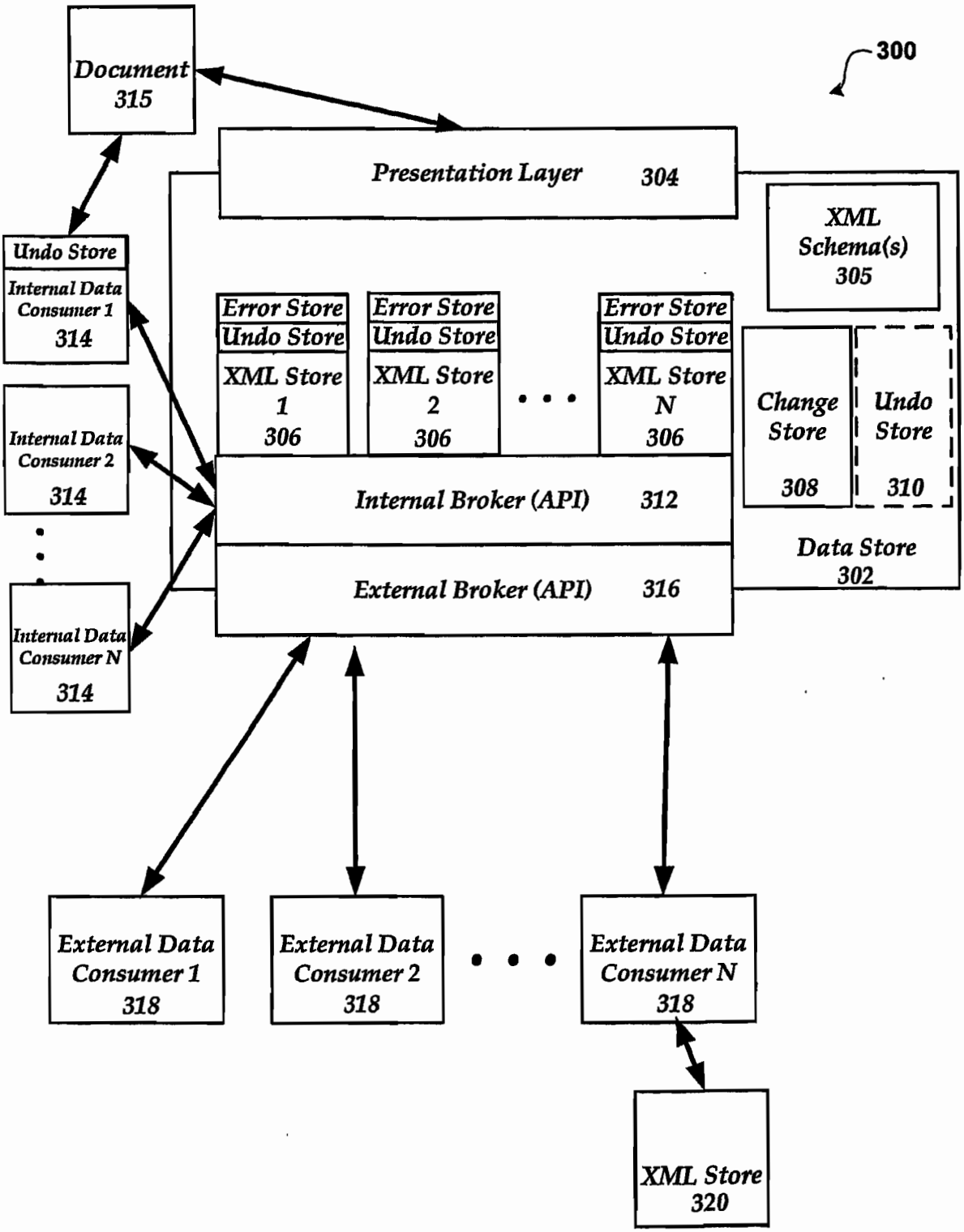

Document
315
300
Presentation Layer
304
XML
Schema(s)
305
Undo Store
Internal Data
Consumer 1
314
Error Store
Undo Store
XML Store
1
306
Error Store
Undo Store
XML Store
2
306
Error Store
Undo Store
XML Store
N
306
Change
Store
308
Undo
Store
310
Internal Data
Consumer 2
314
Internal Broker (API)
312
Data Store
302
External Broker (API)
316
Internal Data
Consumer N
314
External Data
Consumer 1
318
External Data
Consumer 2
318
External Data
Consumer N
318
XML Store
320

Fig. 3

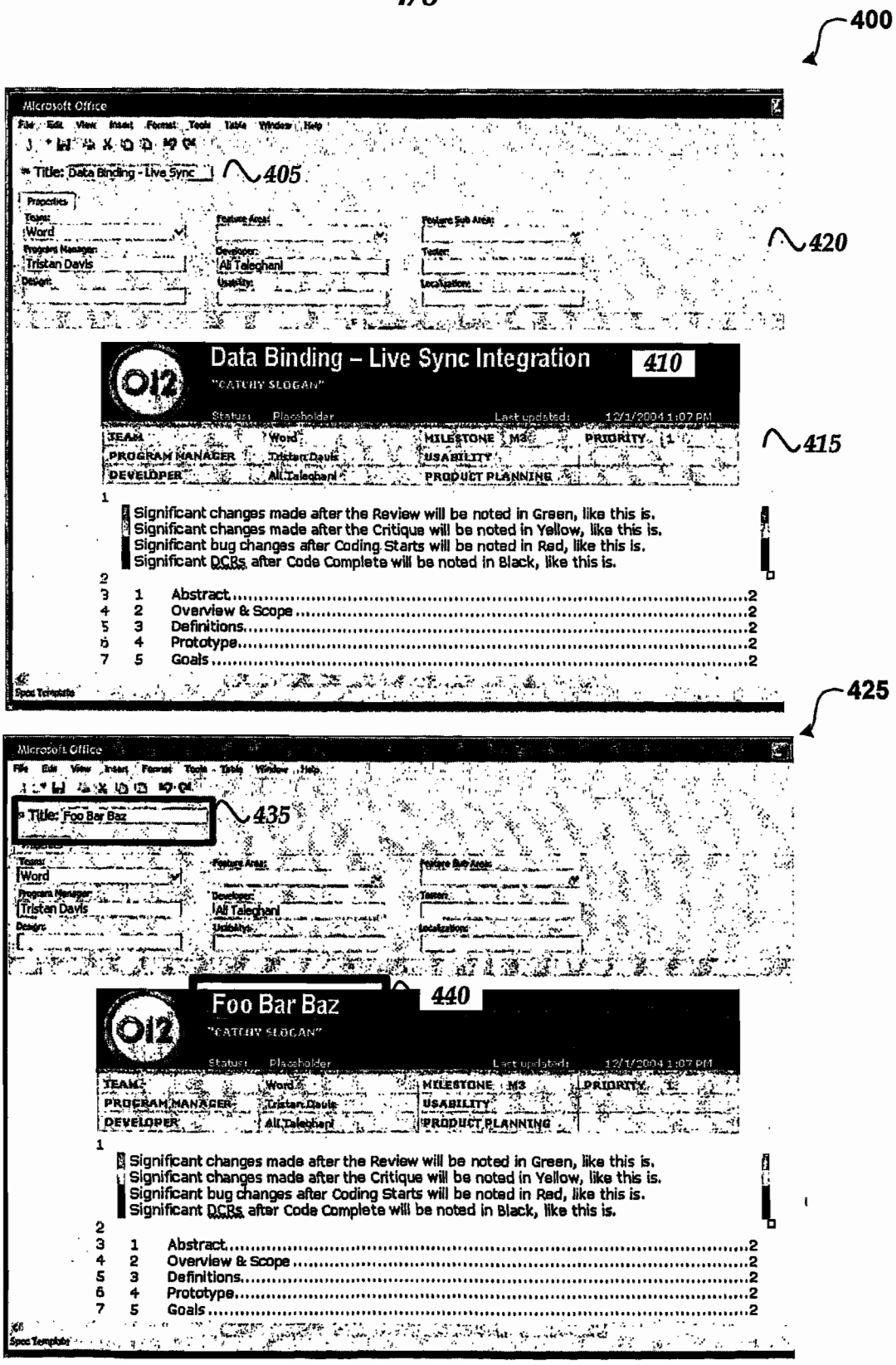

400
Microsoft Office
Title: Data Binding - Live Sync
405
Word
Tristan Davis
Ali Taleghani
420
Data Binding – Live Sync Integration
410
"CATCHY SLOGAN"
Status: Placeholder
Last updated: 12/1/2004 1:07 PM
TEAM Word
MILESTONE M3
PRIORITY 1
PROGRAM MANAGER Tristan Davis
USABILITY
DEVELOPER Ali Taleghani
PRODUCT PLANNING
415
Significant changes made after the Review will be noted in Green, like this is.
Significant changes made after the Critique will be noted in Yellow, like this is.
Significant bug changes after Coding Starts will be noted in Red, like this is.
Significant DCRs after Code Complete will be noted in Black, like this is.
1 Abstract 2
2 Overview & Scope 2
3 Definitions 2
4 Prototype 2
5 Goals 2
425
Microsoft Office
Title: Foo Bar Baz
435
Word
Tristan Davis
Ali Taleghani
Foo Bar Baz
440
"CATCHY SLOGAN"
Status: Placeholder
Last updated: 12/1/2004 1:07 PM
TEAM Word
MILESTONE M3
PRIORITY 1
PROGRAM MANAGER Tristan Davis
USABILITY
DEVELOPER Ali Taleghani
PRODUCT PLANNING
Significant changes made after the Review will be noted in Green, like this is.
Significant changes made after the Critique will be noted in Yellow, like this is.
Significant bug changes after Coding Starts will be noted in Red, like this is.
Significant DCRs after Code Complete will be noted in Black, like this is.
1 Abstract 2
2 Overview & Scope 2
3 Definitions 2
4 Prototype 2
5 Goals 2

Fig. 4

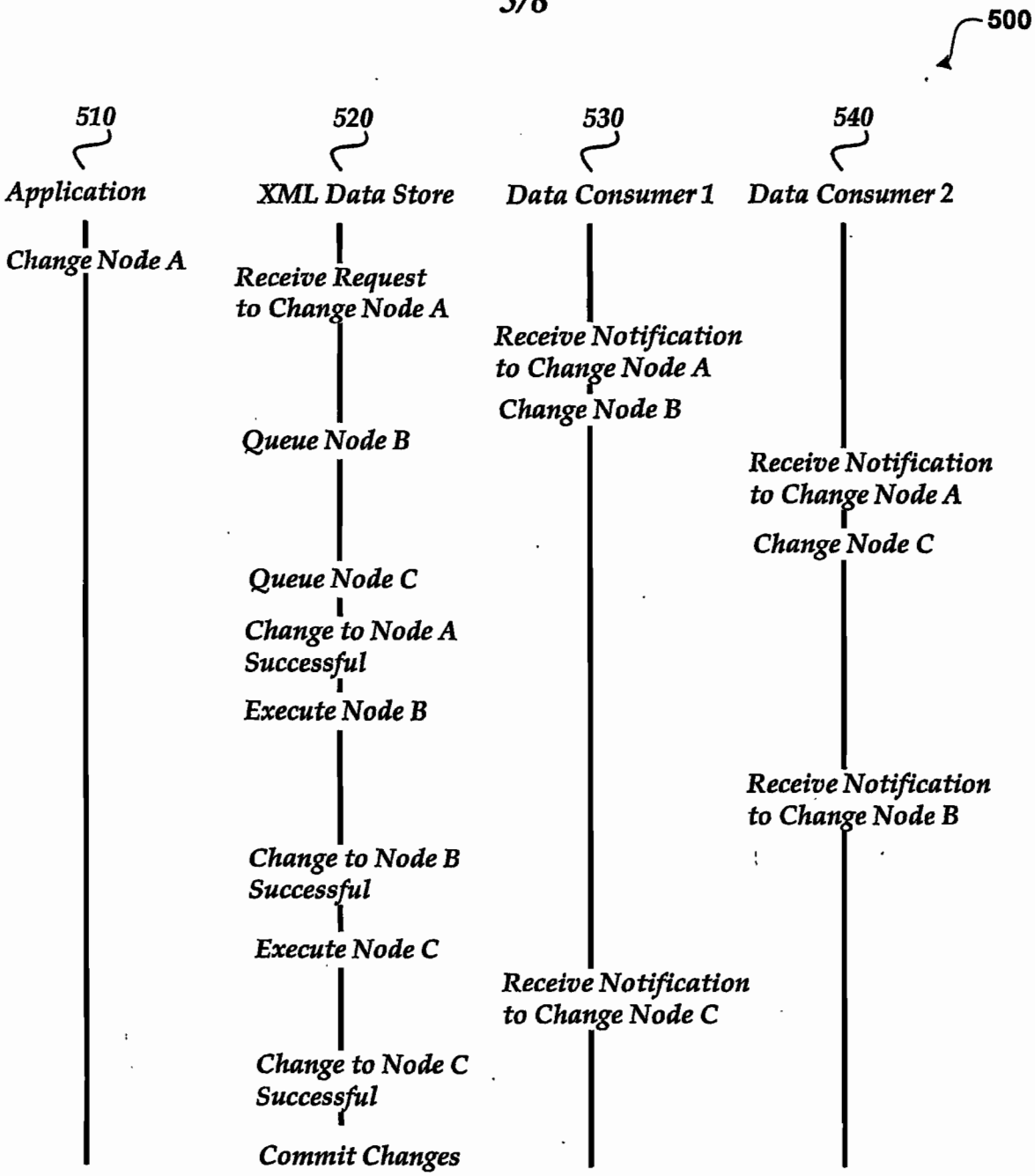

500
510
520
530
540
Application
XML Data Store
Data Consumer 1
Data Consumer 2
Change Node A
Receive Request to Change Node A
Receive Notification to Change Node A
Change Node B
Queue Node B
Receive Notification to Change Node A
Change Node C
Queue Node C
Change to Node A Successful
Execute Node B
Receive Notification to Change Node B
Change to Node B Successful
Execute Node C
Receive Notification to Change Node C
Change to Node C Successful
Commit Changes

Fig. 5

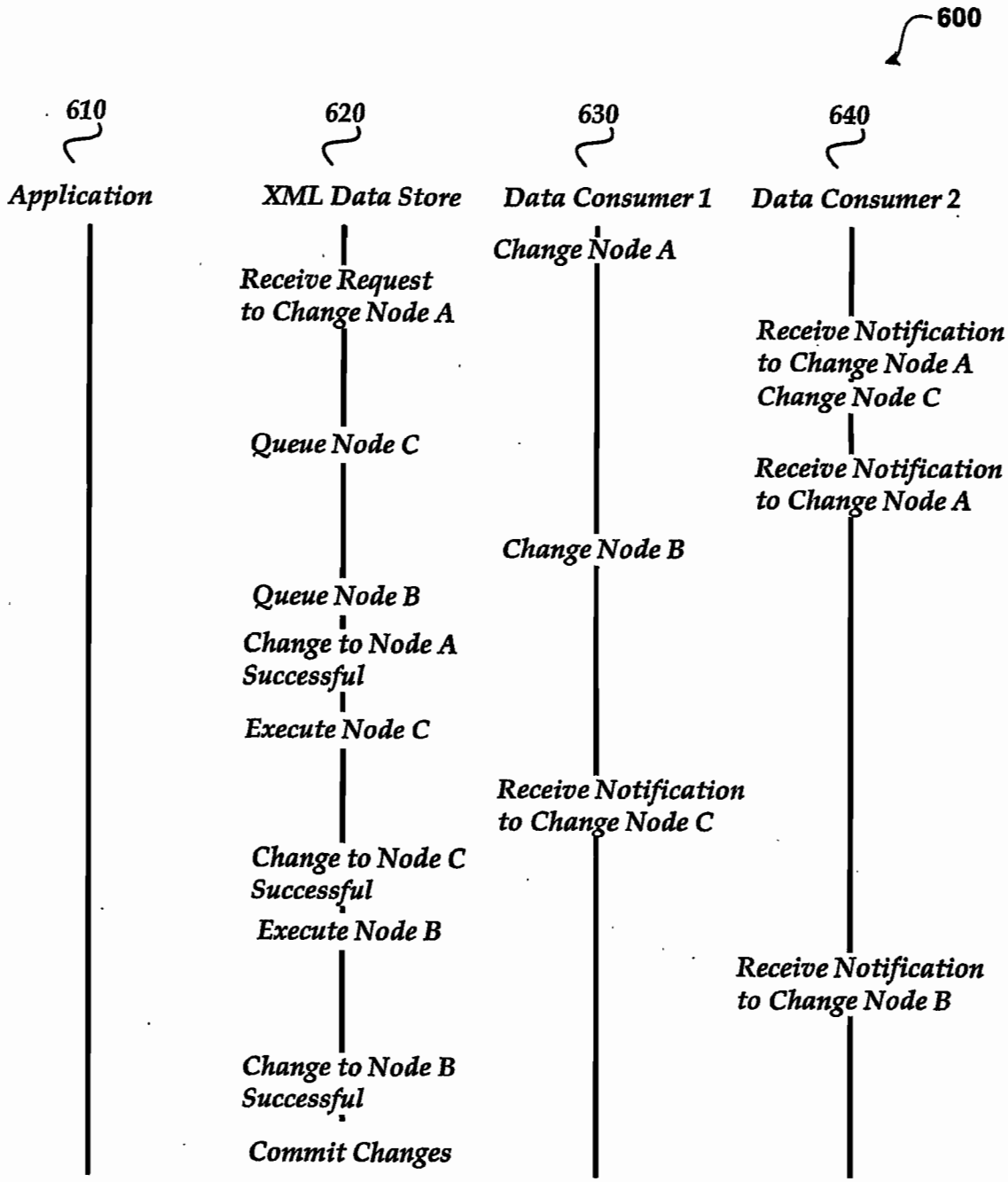

600
610
620
630
640
Application
XML Data Store
Data Consumer 1
Data Consumer 2
Change Node A
Receive Request to Change Node A
Receive Notification to Change Node A
Change Node C
Queue Node C
Receive Notification to Change Node A
Change Node B
Queue Node B
Change to Node A Successful
Execute Node C
Receive Notification to Change Node C
Change to Node C Successful
Execute Node B
Receive Notification to Change Node B
Change to Node B Successful
Commit Changes

Fig. 6

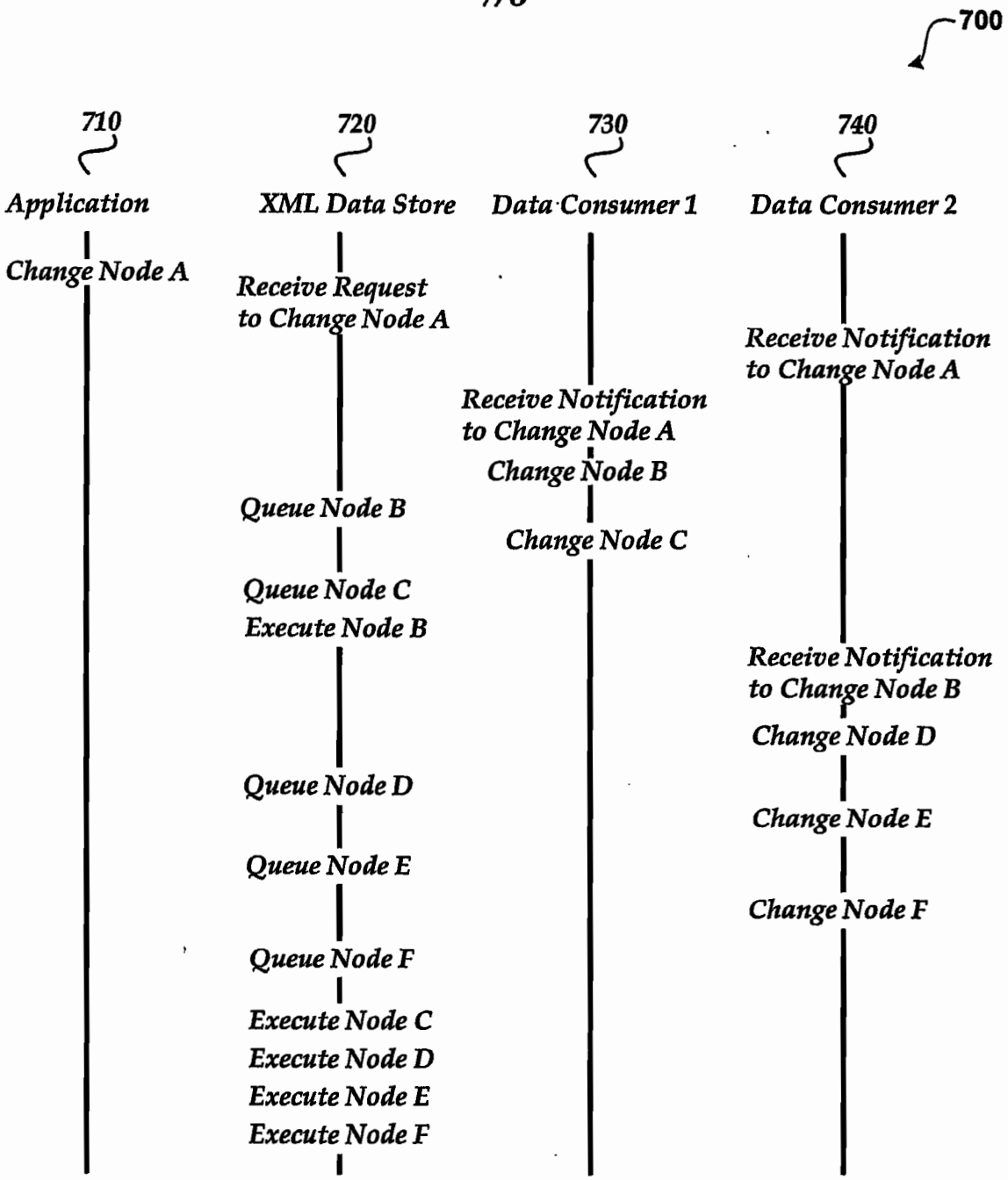

700
710
720
730
740
Application
XML Data Store
Data Consumer 1
Data Consumer 2
Change Node A
Receive Request to Change Node A
Receive Notification to Change Node A
Receive Notification to Change Node A
Change Node B
Queue Node B
Change Node C
Queue Node C
Execute Node B
Receive Notification to Change Node B
Change Node D
Queue Node D
Change Node E
Queue Node E
Change Node F
Queue Node F
Execute Node C
Execute Node D
Execute Node E
Execute Node F

Fig. 7

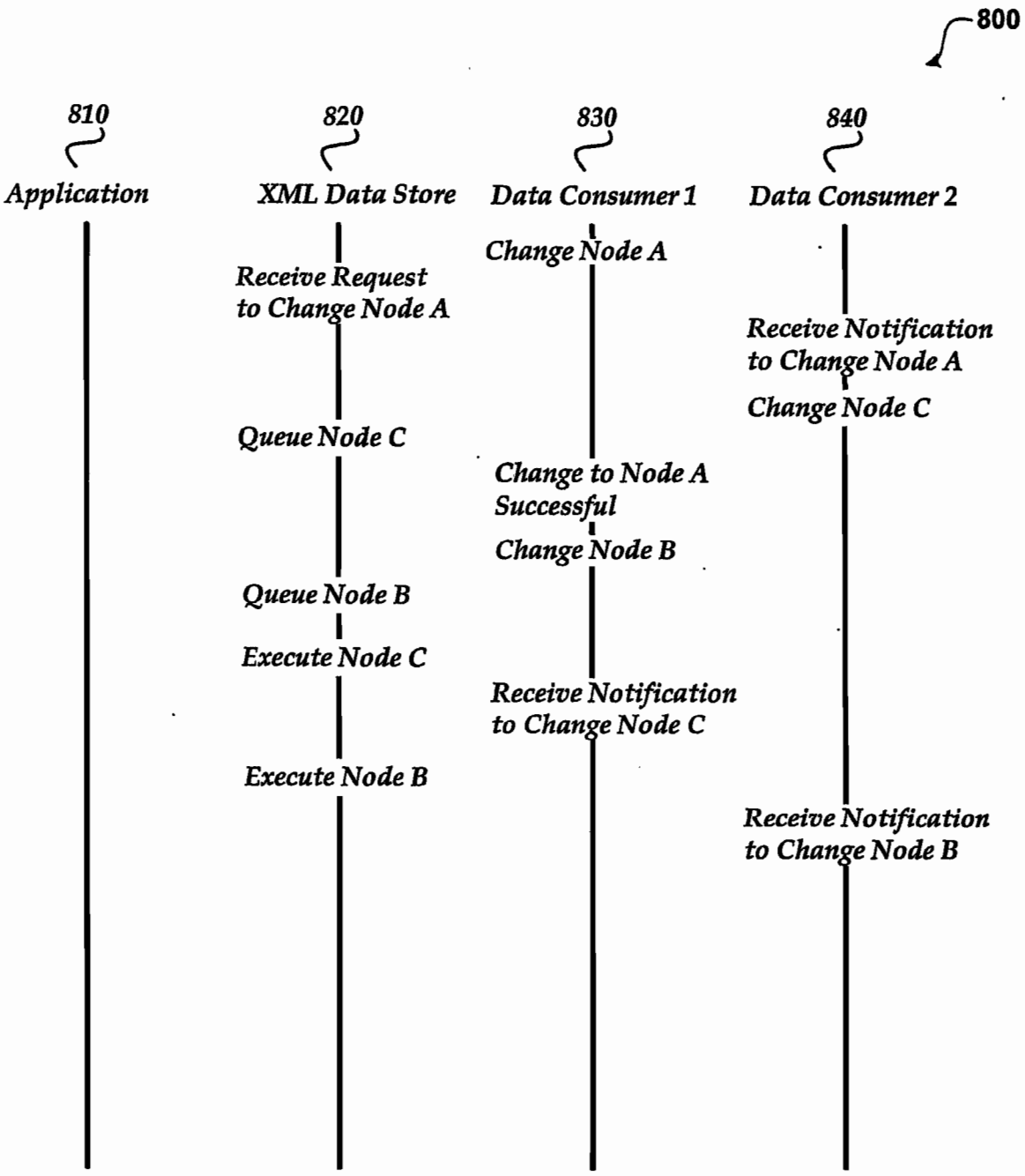

800
810
Application
820
XML Data Store
830
Data Consumer 1
840
Data Consumer 2
Change Node A
Receive Request to Change Node A
Receive Notification to Change Node A
Change Node C
Queue Node C
Change to Node A Successful
Change Node B
Queue Node B
Execute Node C
Receive Notification to Change Node C
Execute Node B
Receive Notification to Change Node B

Fig. 8