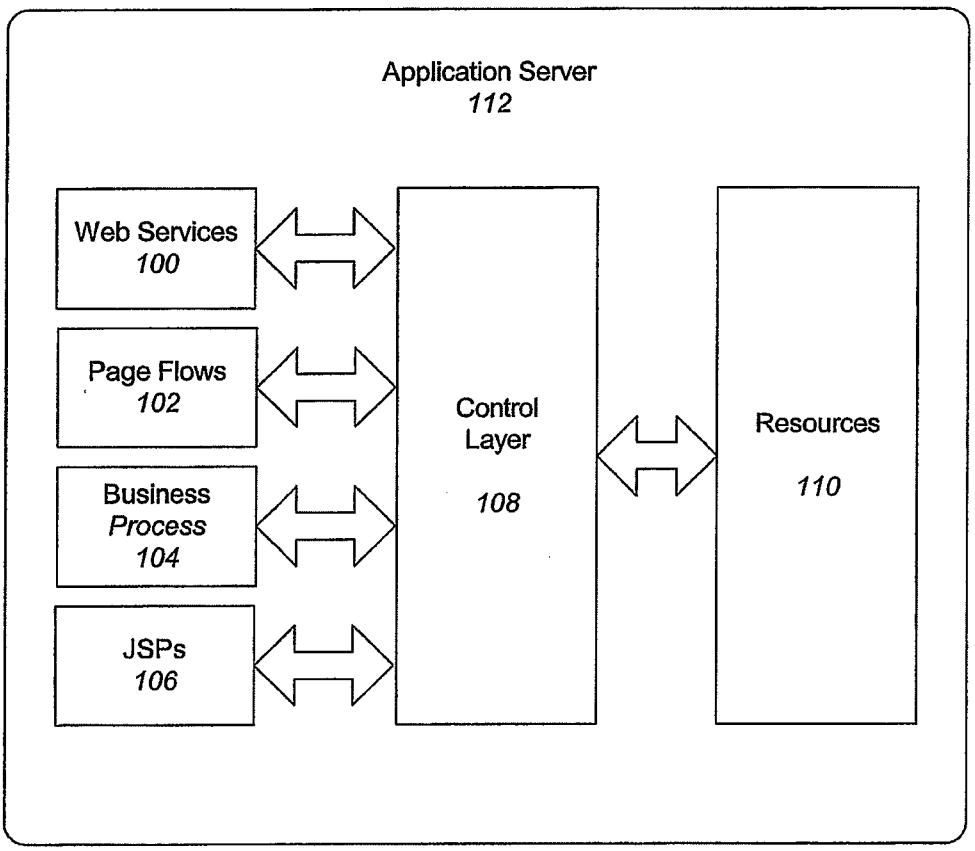

Application Server
112
Web Services
100
Page Flows
102
Business
Process
104
JSPs
106
Control
Layer
108
Resources
110

Figure 1

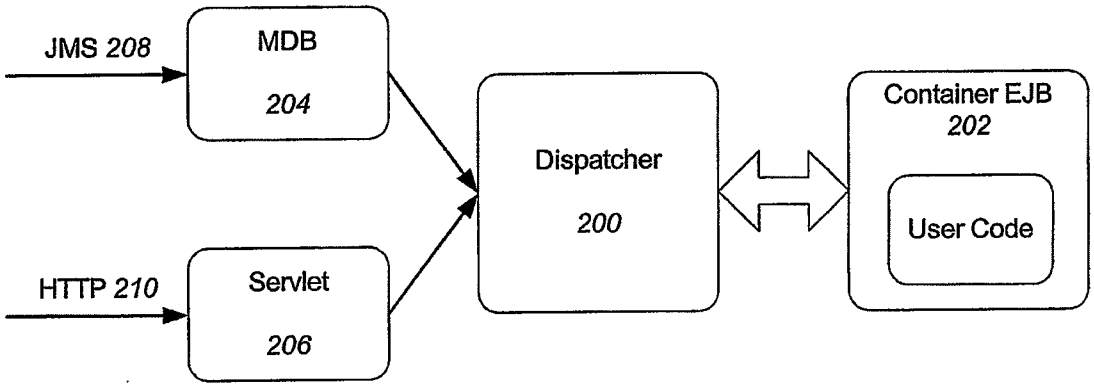

JMS 208
MDB
204
HTTP 210
Servlet
206
Dispatcher
200
Container EJB
202
User Code

Figure 2

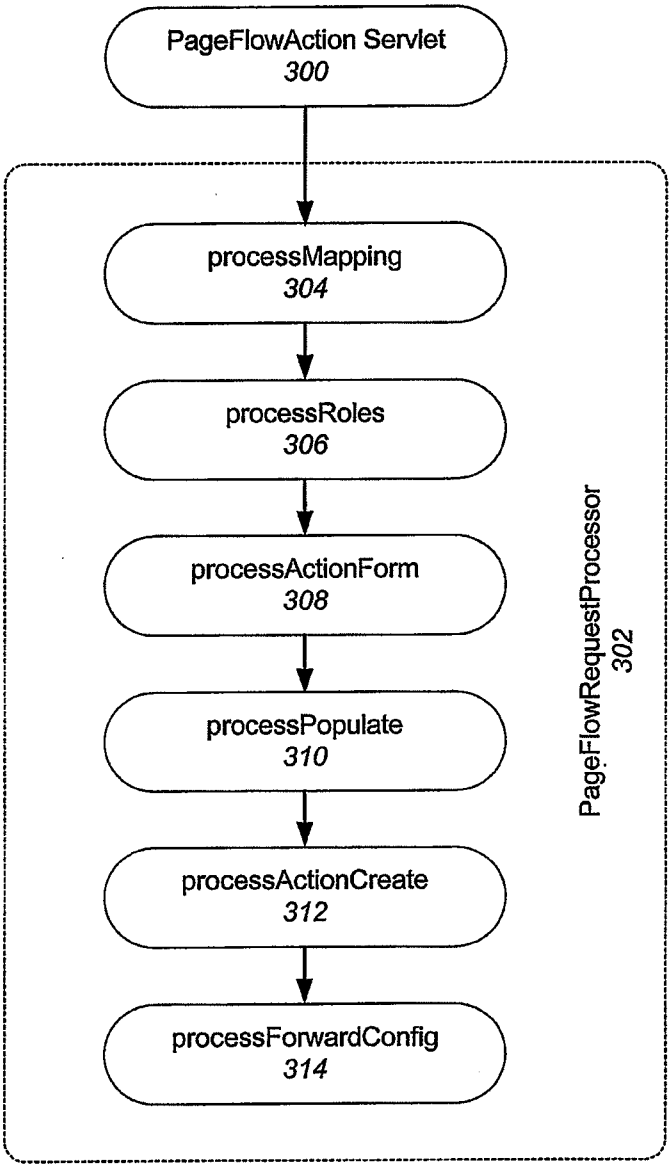

PageFlowAction Servlet
300
processMapping
304
processRoles
306
processActionForm
308
processPopulate
310
processActionCreate
312
processForwardConfig
314
PageFlowRequestProcessor
302

Figure 3

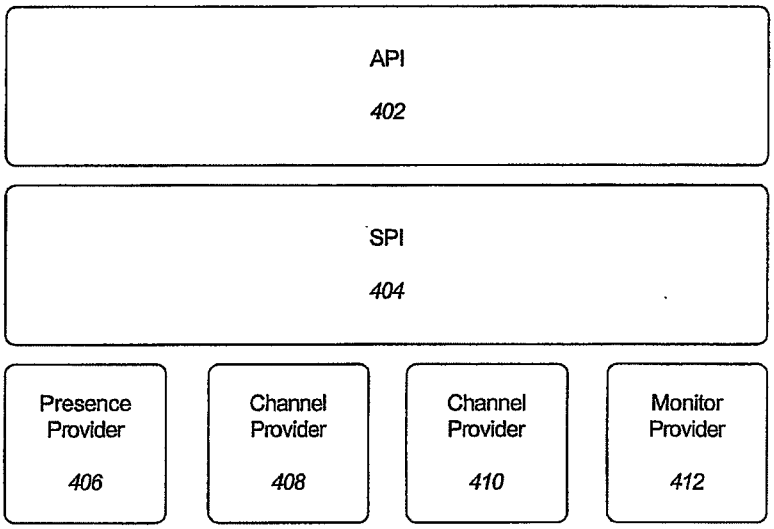

API
402
SPI
404
Presence Provider
406
Channel Provider
408
Channel Provider
410
Monitor Provider
412
400

Figure 4

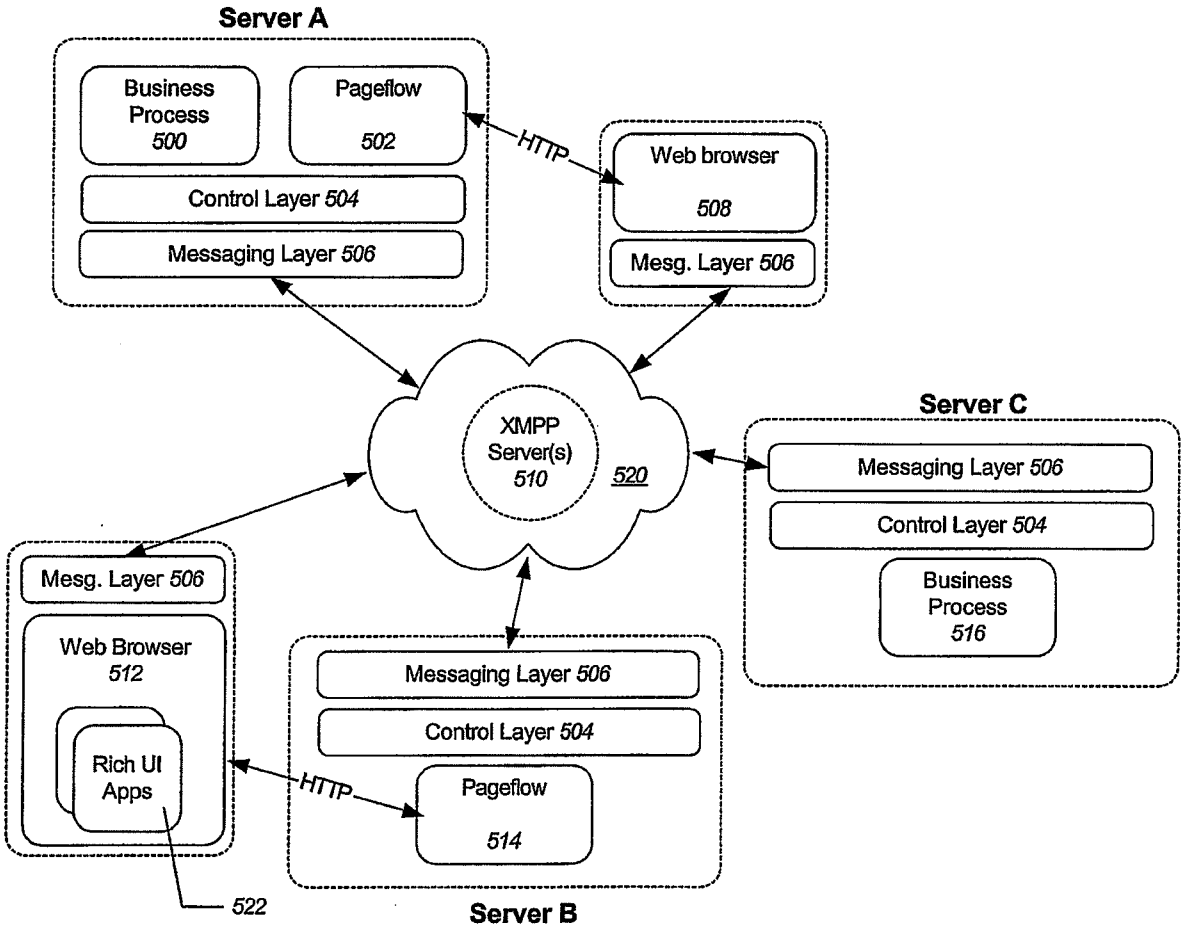

Server A
Business Process 500
Pageflow 502
Control Layer 504
Messaging Layer 506
HTTP
Web browser 508
Mesg. Layer 506
XMPP Server(s) 510
520
Server C
Messaging Layer 506
Control Layer 504
Business Process 516
Mesg. Layer 506
Web Browser 512
Rich UI Apps
522
Messaging Layer 506
Control Layer 504
HTTP
Pageflow 514
Server B

Figure 5

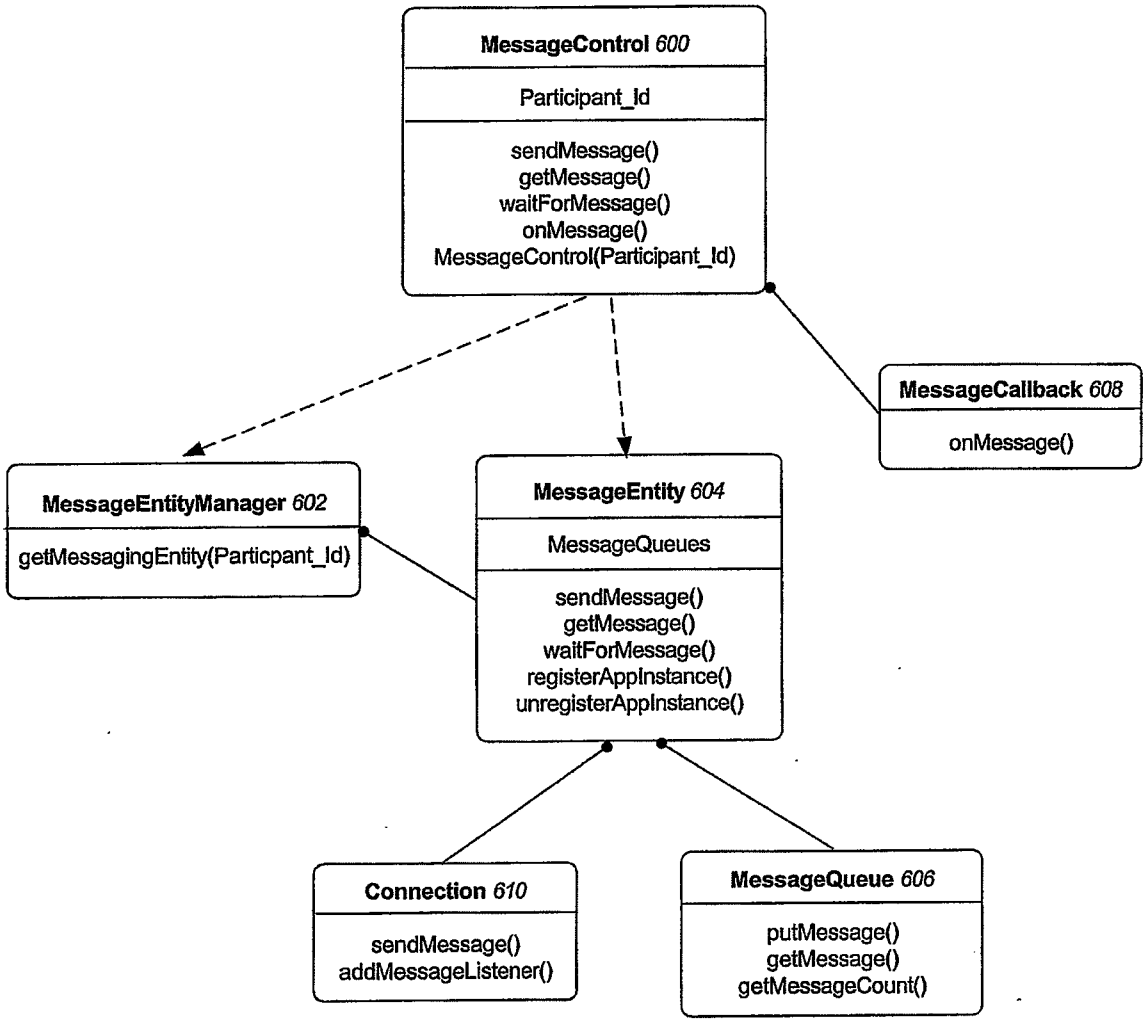

MessageControl 600
Participant_Id
sendMessage()
getMessage()
waitForMessage()
onMessage()
MessageControl(Participant_Id)
MessageCallback 608
onMessage()
MessageEntityManager 602
getMessagingEntity(Particpant_Id)
MessageEntity 604
MessageQueues
sendMessage()
getMessage()
waitForMessage()
registerAppInstance()
unregisterAppInstance()
Connection 610
sendMessage()
addMessageListener()
MessageQueue 606
putMessage()
getMessage()
getMessageCount()

Figure 6

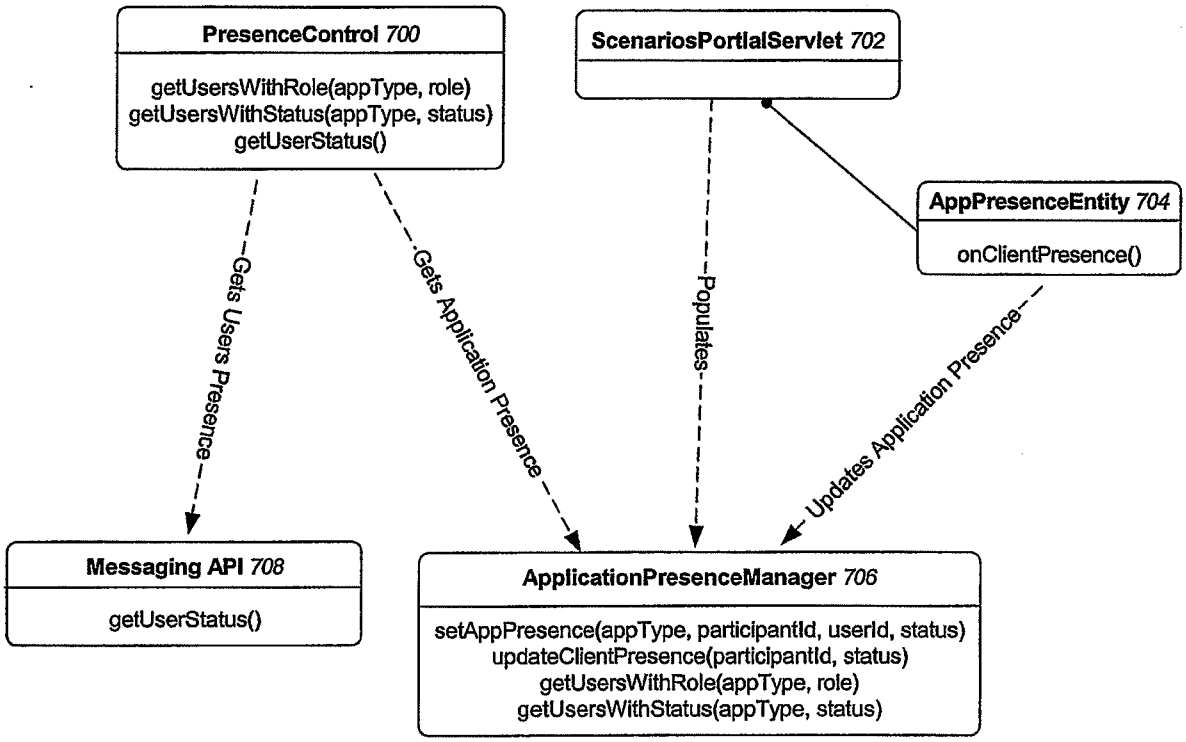

PresenceControl 700
getUsersWithRole(appType, role)
getUsersWithStatus(appType, status)
getUserStatus()
ScenariosPortlalServlet 702
AppPresenceEntity 704
onClientPresence()
Gets Users Presence
Gets Application Presence
Populates
Updates Application Presence
Messaging API 708
getUserStatus()
ApplicationPresenceManager 706
setAppPresence(appType, participantId, userId, status)
updateClientPresence(participantId, status)
getUsersWithRole(appType, role)
getUsersWithStatus(appType, status)

Figure 7

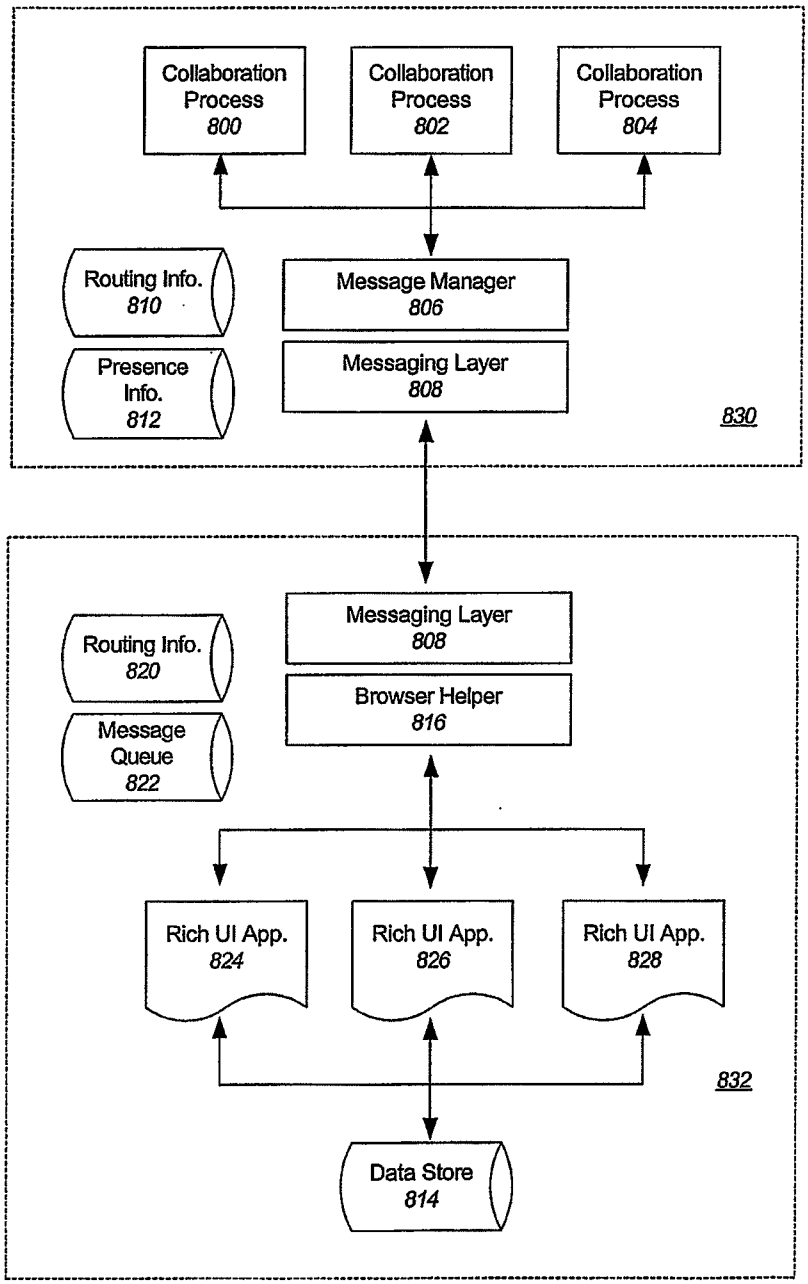

Collaboration Process 800
Collaboration Process 802
Collaboration Process 804
Routing Info. 810
Message Manager 806
Presence Info. 812
Messaging Layer 808
830
Routing Info. 820
Messaging Layer 808
Message Queue 822
Browser Helper 816
Rich UI App. 824
Rich UI App. 826
Rich UI App. 828
832
Data Store 814

Figure 8

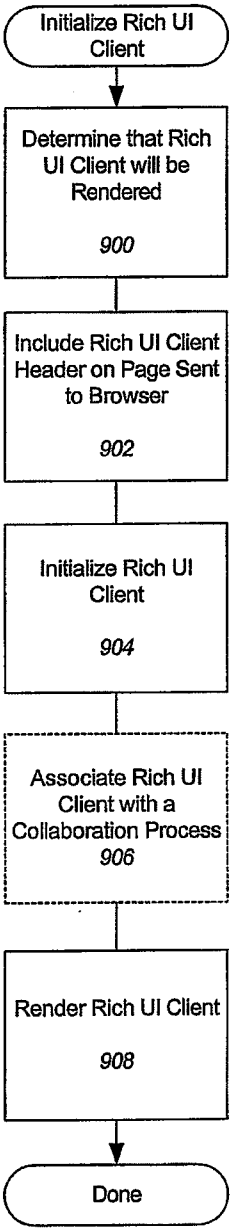

Initialize Rich UI Client
Determine that Rich UI Client will be Rendered
900
Include Rich UI Client Header on Page Sent to Browser
902
Initialize Rich UI Client
904
Associate Rich UI Client with a Collaboration Process
906
Render Rich UI Client
908
Done

Figure 9

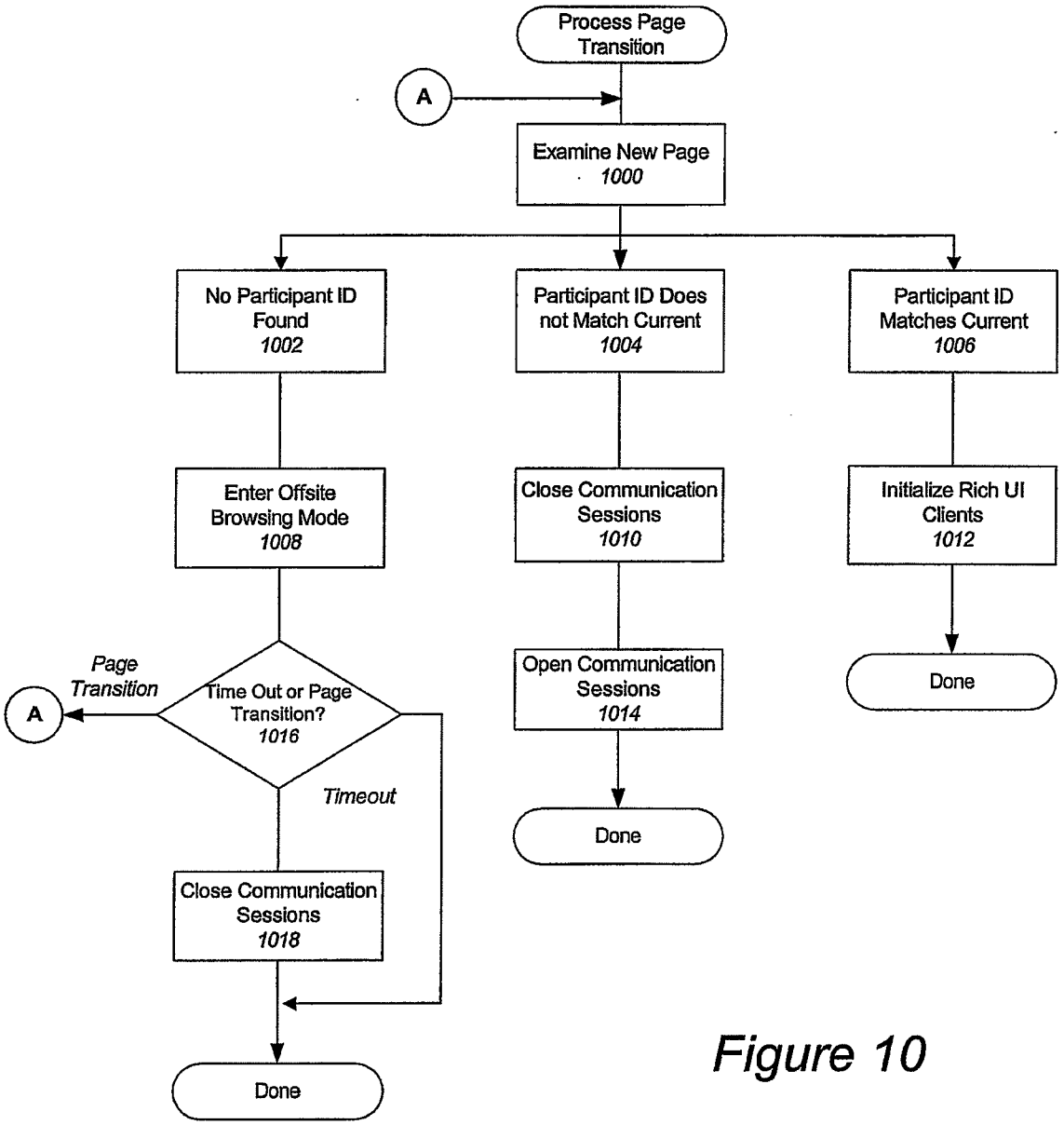

Process Page Transition
A
Examine New Page
1000
No Participant ID Found
1002
Participant ID Does not Match Current
1004
Participant ID Matches Current
1006
Enter Offsite Browsing Mode
1008
Close Communication Sessions
1010
Initialize Rich UI Clients
1012
Page Transition
A
Time Out or Page Transition?
1016
Timeout
Open Communication Sessions
1014
Done
Close Communication Sessions
1018
Done
Done

Figure 10

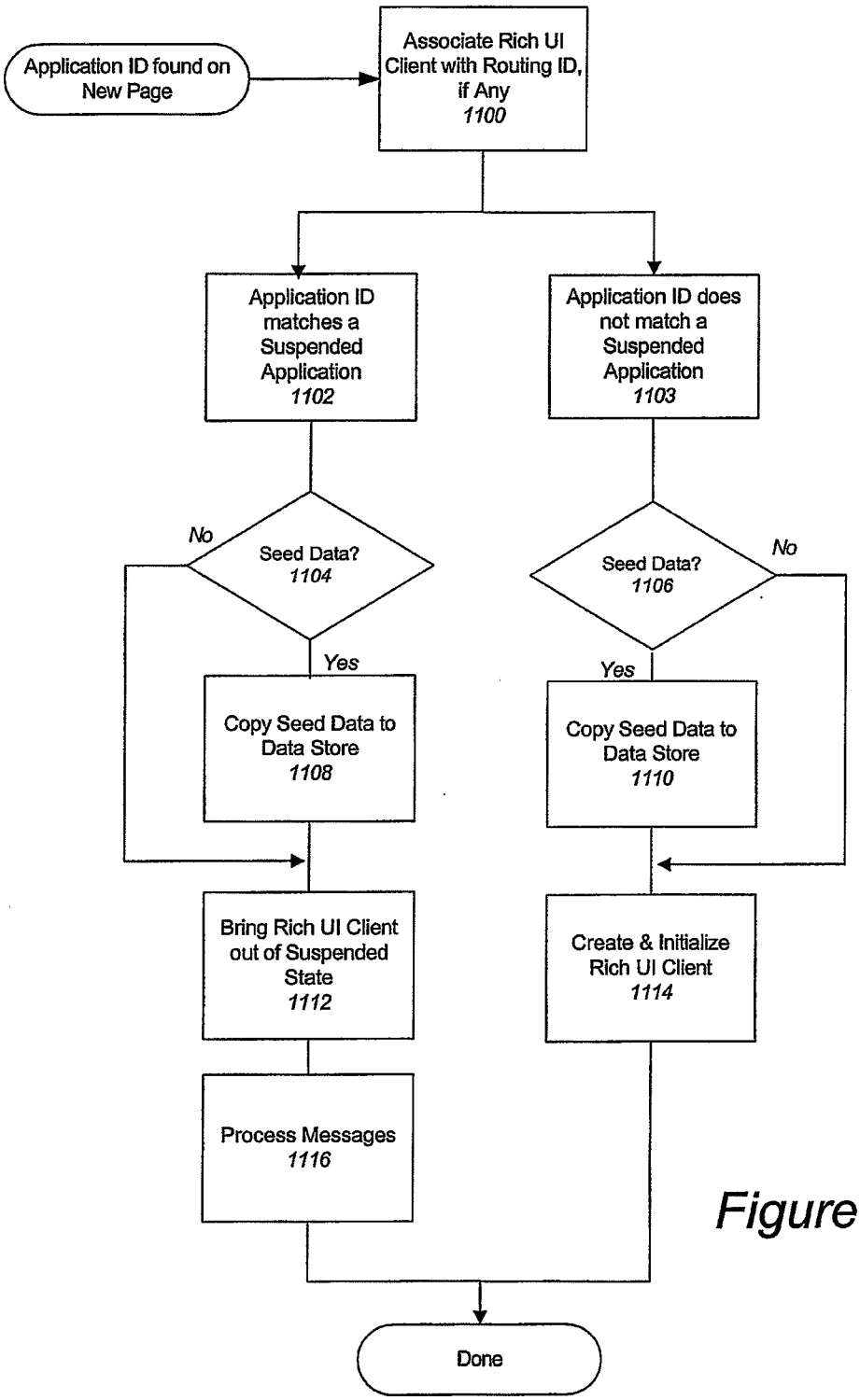

Application ID found on New Page
Associate Rich UI Client with Routing ID, if Any
1100
Application ID matches a Suspended Application
1102
Application ID does not match a Suspended Application
1103
Seed Data?
1104
No
Yes
Seed Data?
1106
No
Yes
Copy Seed Data to Data Store
1108
Copy Seed Data to Data Store
1110
Bring Rich UI Client out of Suspended State
1112
Create & Initialize Rich UI Client
1114
Process Messages
1116
Done

Figure 11

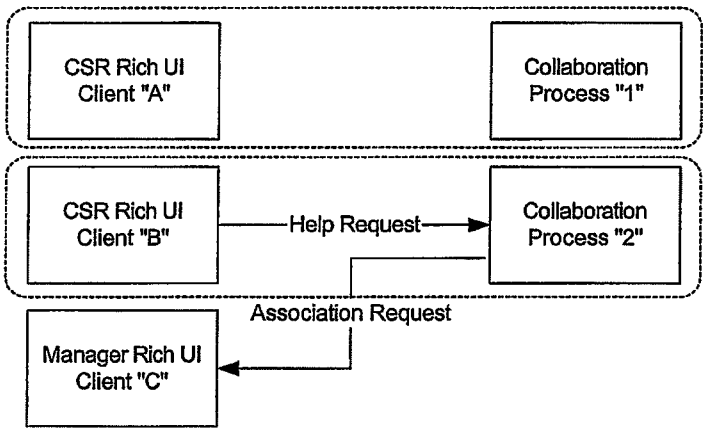


Figure 12a

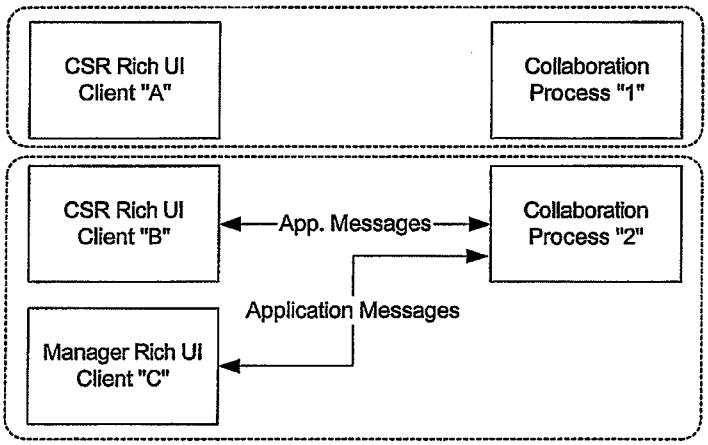


Figure 12b

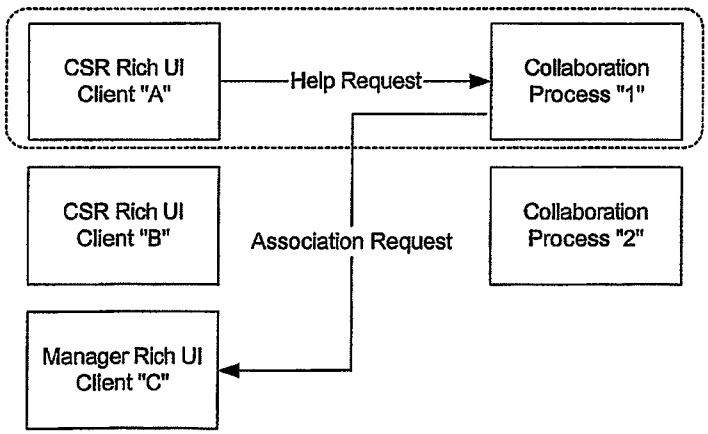


Figure 12c

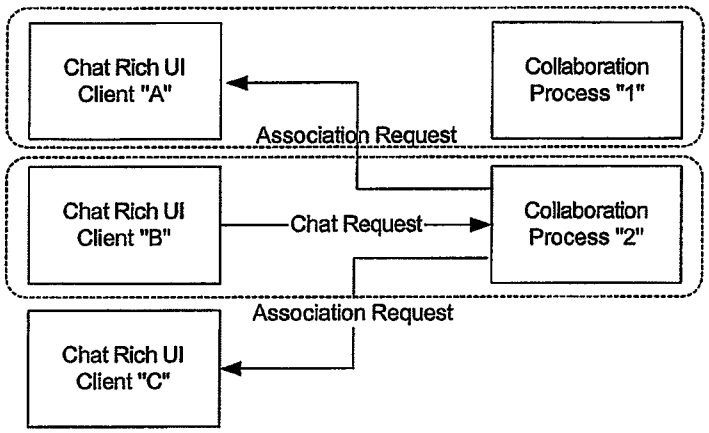


Figure 13a

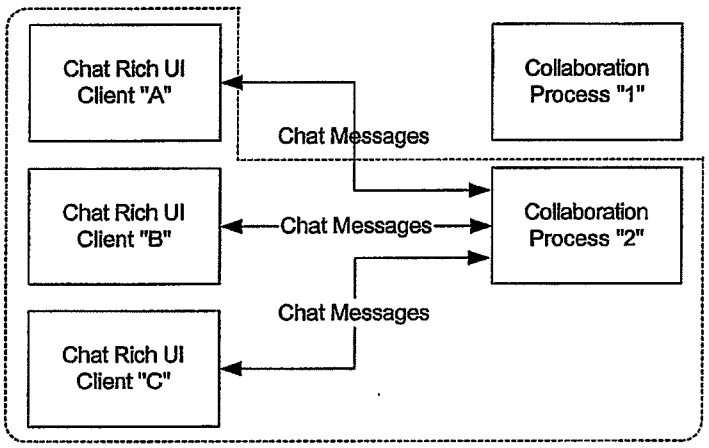


Figure 13b

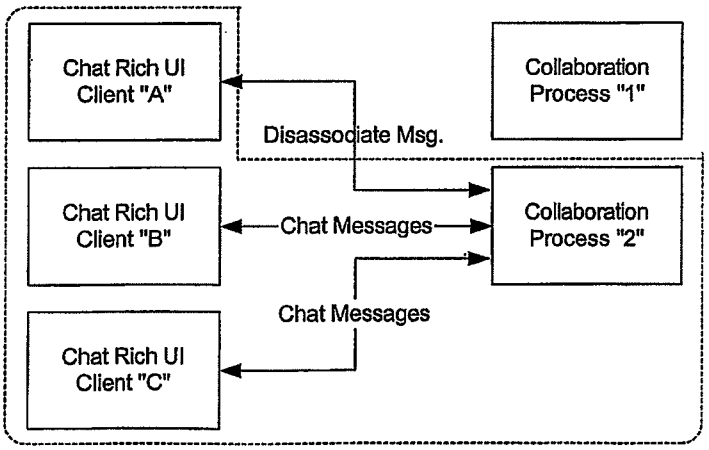


Figure 13c

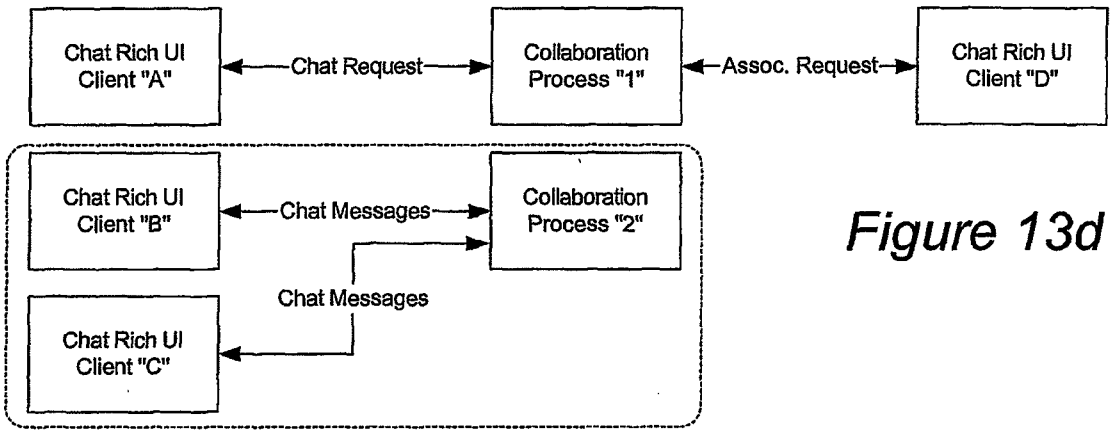


Figure 13d

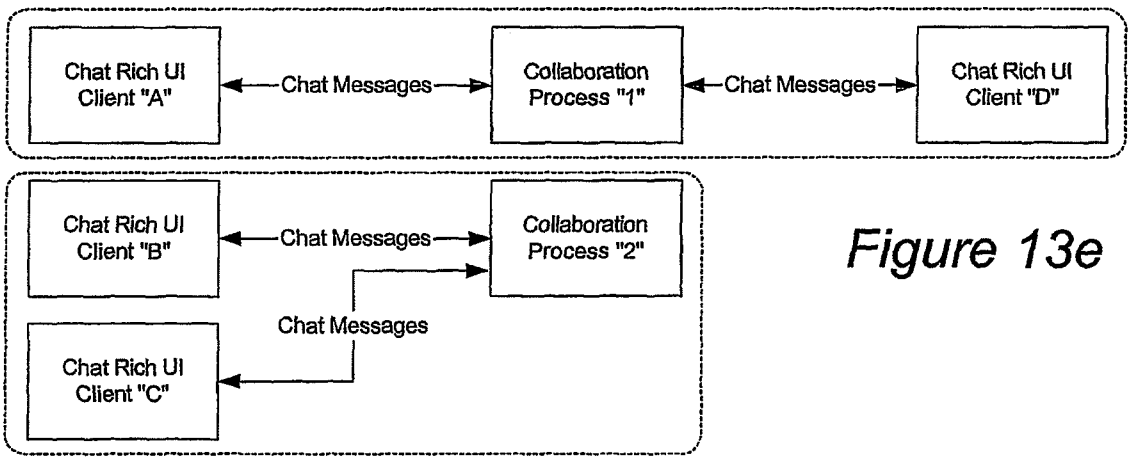


Figure 13e