



(12) 发明专利申请

(10) 申请公布号 CN 101932999 A

(43) 申请公布日 2010. 12. 29

(21) 申请号 200880126061. 5

代理人 叶勇

(22) 申请日 2008. 12. 17

(51) Int. Cl.

(30) 优先权数据

G06F 9/44 (2006. 01)

61/015, 617 2007. 12. 20 US

(85) PCT申请进入国家阶段日

2010. 08. 02

(86) PCT申请的申请数据

PCT/US2008/087144 2008. 12. 17

(87) PCT申请的公布数据

W02009/082653 EN 2009. 07. 02

(71) 申请人 汇丰技术股份有限公司

地址 美国新泽西

(72) 发明人 M·海霍 D·阿维安 D·芬恩

(74) 专利代理机构 中国国际贸易促进委员会专

利商标事务所 11038

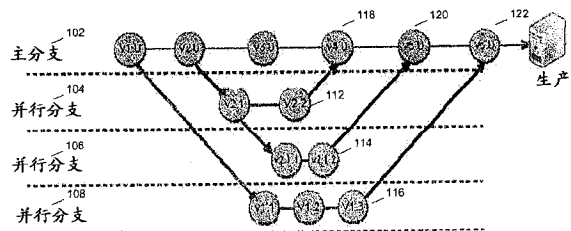
权利要求书 5 页 说明书 23 页 附图 39 页

(54) 发明名称

用于并行开发和部署项目的自动方法和系统

(57) 摘要

本发明的实施例涉及用于管理项目的并行开发的系统和方法。本发明的一个实施例是通过图形用户界面自动管理应用的并行开发中的处理的方法。该图形用户界面管理自动处理,所述自动处理能够完成其任务而不需要进一步的用户交互。由图形用户界面管理的自动处理包括提供第一开发环境和第二开发环境。所述处理还包括将应用的第二发行部署到第二开发环境,以及将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容。在已将发行合并在一起之后,处理自动地将应用的第三发行部署到生产环境以便由用户使用。



1. 一种通过客户端计算机系统上的图形用户界面自动管理应用的并行开发中的处理的方法,所述图形用户界面被用于管理在服务器上执行的自动处理,其中所述自动处理完成其任务而不需要进一步的用户交互,所述自动处理包括:

提供第一开发环境,第一开发环境至少包括执行应用平台的第一服务器,源代码控制系统和内容管理系统;

使用第一开发环境根据具有第一源代码和第一内容的应用的第一发行,产生应用的第二发行,其中第二发行包括应用的第二源代码和第二内容;

提供第二开发环境,第二开发环境至少包括执行应用平台的第二服务器,源代码控制系统和内容管理系统;

将第二发行部署到第二开发环境;

将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容内,以便产生包括第三源代码和第三内容的应用的第三发行;和

将应用的第三发行部署到至少一个生产环境服务器,以便由应用的用户使用,其中所述部署包括 (a) 在执行生产应用平台的生产环境服务器上安装第三源代码的可执行版本,和 (b) 将第三内容安装到生产环境内容管理系统。

2. 如权利要求 1 所述的方法,其中应用的第一发行是生产流。

3. 如权利要求 1 所述的方法,其中由非技术事务用户通过图形用户界面执行将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容,以便产生包括第三源代码和第三内容的应用的第三发行,所述图形用户界面包括用于管理任务和用于检查完成的任务的结果的界面。

4. 如权利要求 1 所述的方法,其中所述源代码控制系统和内容管理系统具有内部分支机制。

5. 如权利要求 4 所述的方法,其中应用的第一发行、第二发行和第三发行中的每一个是源代码控制系统和内容管理系统中的不同版本。

6. 如权利要求 1 所述的方法,其中提供第二应用平台包括:

运行一个或多个脚本,以便在第二应用平台上安装源代码和内容。

7. 如权利要求 6 所述的方法,还包括:

在运行所述一个或多个脚本之前锁定第二应用平台;和

在运行所述一个或多个脚本之后解锁第二应用平台。

8. 如权利要求 7 所述的方法,其中提供还包括:

向中央记录组件记录由所述一个或多个脚本执行的任务,并且记录由所述一个或多个脚本执行的任务的结果。

9. 如权利要求 1 所述的方法,还包括:

在将对第二发行的第二源代码和第二内容的修改合并到第一发行内之前,产生第一发行的备份。

10. 如权利要求 9 所述的方法,还包括:

使用产生的备份回滚对修改的合并。

11. 如权利要求 1 所述的方法,还包括:

将应用的第三发行部署到预生产环境;和

在将第三发行部署到生产环境之前,在预生产环境中测试第三发行。

12. 如权利要求 11 所述的方法,还包括:

基于第二开发环境产生基线,其中所述基线包括第二开发环境内的源代码和内容文件的状态;

将应用的第二发行与基线进行比较;和

响应所述比较,将应用的第二发行部署到第二开发环境。

13. 如权利要求 12 所述的方法,其中响应所述比较将应用的第二发行部署到第二开发环境包括:部署被修改的源代码文件。

14. 如权利要求 1 所述的方法,还包括:

在将第二发行部署到第二开发环境之前,产生第二发行的至少一个源代码文件和一个内容文件的唯一标识,并且当将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容以便产生应用的第三发行时,使用所述唯一标识。

15. 如权利要求 1 所述的方法,还包括:

由用户检查将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容的记录,以便确定在合并处理过程中是否发生了任何错误。

16. 如权利要求 1 所述的方法,还包括:

在有故障的情况下,使用户的生产环境转换到备份的生产环境,并且使用存储区域网络同步生产环境和备份生产环境。

17. 如权利要求 1 所述的方法,其中将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容以便产生第三发行包括以下至少一个:增加源代码文件、删除源代码文件、忽略对源代码文件的改变、覆盖源代码文件和仅包括部分源代码文件。

18. 如权利要求 1 所述的方法,还包括:在第一发行和第二发行之间比较对源代码和内容的改变。

19. 如权利要求 18 所述的方法,其中在第一发行和第二发行之间比较对源代码和内容的改变包括比较源代码列表,并且比较内容文件的呈现显示。

20. 如权利要求 19 所述的方法,其中内容文件的呈现显示被选择性地突出显示以便进行视觉比较。

21. 如权利要求 1 所述的方法,其中将应用的第三发行部署到生产环境包括:

在部署第三发行之前,自动停止生产环境上的至少一个应用的执行;和

在将第三发行部署到生产环境之后,自动开始应用的第三发行的执行。

22. 如权利要求 1 所述的方法,其中执行应用平台的每个服务器、源代码控制系统和内容控制系统是单独并且不同的服务器。

23. 如权利要求 1 所述的方法,其中将第二发行部署到第二开发环境包括将第二源代码的可执行版本安装到第二开发环境的应用平台,以及将第二内容安装到第二开发环境的内容管理服务器。

24. 如权利要求 1 所述的方法,其中客户端计算机系统上的图形用户界面向运行在部署和建立服务器上的相应管理应用发送命令,并且其中部署和建立服务器被通过电子网络耦接到生产环境,并且建立第三源代码的可执行版本,以便部署到生产环境应用平台服务

器上,并且其中部署建立服务器将第三内容安装在生产环境的内容管理服务器。

25. 一种通过图形用户界面自动管理应用的并行开发中的处理的系统,所述图形用户界面被用于管理执行自动处理的组件,其中所述组件完成其任务而不需要进一步的用户交互,所述组件包括:

提供组件,用于提供第一开发环境,第一开发环境至少包括第一应用平台,源代码控制系统和内容管理系统,其中所述提供组件

(i) 使用第一开发环境根据具有第一源代码和第一内容的应用的第一发行,产生应用的第二发行,其中第二发行包括应用的第二源代码和第二内容;

(ii) 提供第二开发环境,第二开发环境至少包括第二应用平台,源代码控制系统和内容管理系统;

部署组件,用于将第二发行部署到第二开发环境;

合并组件,用于将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容内,以便产生包括第三源代码和第三内容的应用的第三发行;和

所述部署组件将应用的第三发行部署到生产环境,以便由应用的用户使用,其中所述部署包括在生产环境的另一个应用平台上安装 (a) 第三源代码的可执行版本,和 (b) 第三内容。

26. 如权利要求 25 所述的系统,其中应用的第一发行是生产流。

27. 如权利要求 25 所述的系统,其中由非技术事务用户通过图形用户界面执行将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容,以便产生包括第三源代码和第三内容的应用的第三发行。

28. 如权利要求 25 所述的系统,其中所述源代码控制系统和内容管理系统具有内部分支机制。

29. 如权利要求 28 所述的系统,其中应用的第一发行、第二发行和第三发行中的每一个是源代码控制系统和内容管理系统中的不同版本。

30. 如权利要求 25 所述的系统,其中提供第二应用平台包括:

运行一个或多个脚本,以便在第二应用平台上安装源代码和内容。

31. 如权利要求 25 所述的系统,还包括:

在将对第二发行的第二源代码和第二内容的修改合并到第一发行内之前,产生第一发行的备份。

32. 如权利要求 25 所述的系统,还包括:

将应用的第三发行部署到预生产环境;和

在将第三发行部署到生产环境之前,在预生产环境中测试第三发行。

33. 如权利要求 32 所述的系统,还包括:

基于第二开发环境产生基线,其中所述基线包括第二开发环境内的源代码和内容文件的状态;

将应用的第二发行与基线进行比较;和

响应所述比较,将应用的第二发行部署到第二开发环境。

34. 如权利要求 26 所述的系统,还包括:

在将第二发行部署到第二开发环境之前,产生第二发行的至少一个源代码文件和一个

内容文件的唯一标识,并且当将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容以便产生应用的第三发行时,使用所述唯一标识。

35. 如权利要求 26 所述的系统,还包括:

由用户检查将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容的记录,以便确定在合并处理过程中是否发生了任何错误。

36. 如权利要求 26 所述的系统,还包括:

在有故障的情况下,将用户的生产环境转换到备份的生产环境,并且使用存储区域网络同步生产环境和备份生产环境。

37. 如权利要求 26 所述的系统,其中将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容以便产生第三发行包括以下至少一个:增加源代码文件、删除源代码文件、忽略对源代码文件的改变、覆盖源代码文件和仅包括部分源代码文件。

38. 如权利要求 26 所述的系统,还包括:在第一发行和第二发行之间比较对源代码和内容的改变。

39. 如权利要求 38 所述的系统,其中在第一发行和第二发行之间比较对源代码和内容的改变包括比较源代码列表,并且比较内容文件的呈现显示。

40. 一种通过分开的计算机系统生产环境管理和协调计算机应用的并行开发的计算机应用开发系统,所述计算机系统包括:

第一开发环境,第一开发环境包括提供至少第一应用平台、源代码控制系统和内容管理系统的第一提供系统,其中第一提供系统使用第一开发环境根据具有第一源代码和第一内容的应用的第二发行,产生应用的第二发行,其中第二发行包括应用的第二源代码和第二内容;

第二开发环境,第二开发环境包括提供至少第二应用平台、源代码控制系统和内容管理系统的第二提供系统;

合并系统组件,用于将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容内,以便产生包括第三源代码和第三内容的应用的第三发行;和

部署系统组件,部署系统组件将第二发行部署到第二开发环境,并且将应用的第三发行部署到生产环境,以便由应用的用户使用,其中第三发行的部署包括在生产环境的另一个应用平台上安装 (a) 第三源代码的可执行版本,和 (b) 第三内容。

41. 如权利要求 40 所述的方法,其中第一和第二开发环境在地理上是不同的。

42. 如权利要求 40 所述的方法,其中第一开发环境用于开发第一发行,并且第二开发环境用于开发第二发行,其中第一和第二发行表示应用的不同版本,并且其中第一发行是应用的主发行,对第二发行的修改被合并到第一发行内。

43. 一种通过图形用户界面自动管理应用的并行开发中的处理的指令的计算机可读介质,所述图形用户界面被用于管理自动处理,其中所述自动处理完成其任务而不需要进一步的用户交互,所述自动处理的指令包括:

提供第一开发环境,第一开发环境至少包括第一应用平台、源代码控制系统和内容管理系统;

使用第一开发环境根据具有第一源代码和第一内容的应用的第二发行,产生应用的第

二发行,其中第二发行包括应用的第二源代码和第二内容;

提供第二开发环境,第二开发环境至少包括第二应用平台、源代码控制系统和内容管理系统;

将第二发行部署到第二开发环境;

将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容内,以便产生包括第三源代码和第三内容的应用的第三发行;和

将应用的第三发行部署到生产环境以便由应用的用户使用,其中所述部署包括在生产环境的另一个应用平台上安装 (a) 第三源代码的可执行版本,和 (b) 第三内容。

44. 如权利要求 43 所述的方法,其中由非技术事务用户通过图形用户界面执行将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容,以便产生包括第三源代码和第三内容的应用的第三发行,所述图形用户界面包括用于管理任务和用于检查完成的的任务的结果的界面。

45. 如权利要求 43 所述的方法,其中应用的第一发行、第二发行和第三发行中的每一个是源代码控制系统和内容管理系统中的不同版本。

46. 如权利要求 43 所述的方法,其中提供第二应用平台包括:

运行一个或多个脚本,以便在第二应用平台上安装源代码和内容。

47. 如权利要求 43 所述的方法,还包括:

将应用的第三发行部署到预生产环境;和

在将第三发行部署到生产环境之前,在预生产环境中测试第三发行。

48. 如权利要求 43 所述的方法,还包括:

基于第二开发环境产生基线,其中所述基线包括第二开发环境内的源代码和内容文件的状态;

将应用的第二发行与基线进行比较;和

响应所述比较,将应用的第二发行部署到第二开发环境。

用于并行开发和部署项目的自动方法和系统

[0001] 与相关申请的交叉引用

[0002] 本申请按照 35 U.S.C. 119(e) 要求享有题目为“METHODS AND SYSTEMS FOR MANAGING PROJECTS IN PARALLEL IN AN AUTOMATED MANNER”的提交于 2007 年 12 月 20 日的美国临时专利申请 No. 61/015, 617 的优先权, 通过引用将其完整结合在此。

技术领域

[0003] 本发明一般地涉及应用的并行开发, 更具体地涉及自动进行整个开发、测试和部署处理中的并行开发处理。

背景技术

[0004] 以健壮的方式开发和部署由用户访问的应用, 尤其是网络应用, 需要大量处理。对于网络应用, 这些处理包括管理源代码、测试代码和将代码随同内容部署到生产系统, 用户可以在生产系统访问代码。这些处理可能花费大量时间 (并且易于出错), 意味着直到其它改变已被测试和执行为止, 某些改变不能被测试或执行。这需要放慢应用的整个开发。

[0005] IT 开发者使用源代码控制工具例如 MKS 源代码控制系统执行并行开发任务。通过允许多个开发路径或相同文件的多个版本存在于相同位置, MKS 提供对并行开发的支持。然而, 并行管理源代码仅是开发、测试和部署健壮应用尤其是网络应用的处理的一部分。除了源代码之外, 在网络应用的整个生命期中, 对网络应用基础页面定义、URL、元数据、内容、图形、文档和资源的改变可能需要被并行操作或测试。

[0006] 通过使用并行开发处理, 对应用的多个改变和添加可被在大体相同的时间执行。一般地, 并行开发和并行管理资源的问题可被基本描述为同时修改相同资源或制品的集合。由于如果存在交叉或资源间的相关性和引用, 资源可能被间接改变的事实, 这可能是复杂的。支持平台的底层技术进一步复杂化了并行管理事务和 IT 资源 (页面、内容和 portlet 应用或事务功能) 的基本能力。例如, IBM Portal Server 不提供能够实现并行资源开发的自动框架。

[0007] 一般地, 并行开发需要能够同时开发、虚拟化、测试和部署相同资源或资源或集合; 进行并行修改, 对一个资源的改变对该资源的其它版本没有任何影响。

[0008] 对此而言, 将并行开发延及整个应用开发处理的方法曾是特别的、手动的或用户劳动密集的, 并且是易于出错的。需要整个系统和处理, 优选自动执行应用的并行开发。

发明内容

[0009] 本发明的实施例涉及用于在整个开发、测试和部署处理中自动进行并行开发处理的系统。本发明给非技术用户提供了初始化并行流; 将资源从并行流提升到生产流; 将最新资源从生产流带入并行流, 以及在流之间进行比较的自动框架。为了支持资源的并行开发, 本发明的实施例给用户提供了创建被称为生产流的主干和被称为并行流的分支, 同时成功地保持并行开发环境的能力。本发明的实施例可被安装在一个或多个服务器上, 包括

特别用于支持应用平台、源代码控制系统或内容管理系统的不同服务器。本发明的替换实施例利用分开的创建和部署服务器,以便创建可执行源代码,以及将其部署到构成开发环境的一个或多个服务器。

[0010] 本发明的替换实施例利用分支系统,以便允许应用的并行开发,包括源代码和内容。用户可以工作于被称为生产流的主干和被称为并行流的分支。通过使用分支策略,可以与资源的其它版本相隔离地修改和管理相同资源(例如,网页的源代码文件或图像)的多个版本。本发明的实施例在包括测试环境、登台服务器、生产服务器、应用平台和内容管理系统的开发处理的多个组件上应用这种分支策略。这些系统一般被称为事务开发环境(即, BDE 或 eBDE)。不论并行流的数目如何,优选地存在可被部署到生产环境并且被终端用户或顾客看到的资源的一个“活动的”版本。这意味着,如果必要,所有并行流最终必须被合并或整合到生产流中,从而资源可被发行到生产环境。虽然在技术上更具挑战性,本发明的实施例确保在任意时刻和在永久基础上,“生产”流相应于“活动的”流,其操作地进行哪个流相应于“活动的”流的猜测工作,并且确保在“设计时”而不是在“运行时”发现任何资源冲突。

[0011] 本发明的替换实施例管理资源到生产环境的部署,包括以自动方式将多个开发的并行流合并到生产流中。这些资源包括在 TeamSite 中管理的内容片段、网口资源(例如,页面定义、URL 映像和 portlet 应用)、分配给 portlet 应用的顾客验证模块(CAM)资源和安装在网口服务器环境内的共享应用组件中的一个或多个。本发明的实施例提供了创建新流,并且在流之间移动资源以便能够进行单个资源和资源集合的并行开发的方法,这些资源然后可被移动到将被部署到生产“活动的”环境中的生产流内。本发明的实施例提供了创建新流,并且在流之间移动资源以便能够进行单个资源和资源集合的并行开发的方法,这些资源然后可被移动到将被部署到生产环境中的生产流内。其它自动操作可以包括初始化新的并行流,获得最新的流以便从其开始工作,比较流之间的差异(和流内的各个资源)、管理测试和流的用户、以及执行合并改变的回滚。

[0012] 本发明的替换实施例包括连接到开发、测试、预生产和生产环境的基于 web 的系统。可以使用 web 界面进行通过这些系统对应用的管理。该 Web 界面还包括用于协作决策制定的能力,以及多个体参与的能力,包括批准或不批准某些动作,以及增加/修改/删除资源。通过在将资源部署到开发的下一个阶段之前各个地评审资源,该 Web 界面还提供详细地视觉管理合并处理的能力。这可以包括合并来自资源的并行版本的改变。

[0013] 使用这种自动系统,从公共开始点对多个特征可以有利地执行并行开发,然后这些特征可被一起组合到最终的生产流中。可替换地,一个分支可以首先进入生产,并且改变可被在以后结合到该生产分支内。还可以按照非技术非 IT 用户的指示有利地自动地管理和执行将并行流之一部署到生产环境。

[0014] 更具体地,本发明的某些实施例涉及用于管理项目的并行开发的系统和方法。本发明的一个实施例是一种使用图形用户界面,在应用的并行开发中自动管理处理的方法。该图形用户界面管理能够完成其任务而不需要用户干预的自动处理。以图形用户界面管理的自动处理包括提供至少包括第一应用平台、源代码控制系统和内容管理系统的的第一开发环境。所述处理还包括使用第一开发环境,从具有第一源代码和第一内容的应用的第一发行产生应用的第二发行,其中第二发行包括应用的第二源代码和第二内容。所述处理还包

括提供至少包括第二应用平台、源代码控制系统和内容管理系统的第二开发环境。所述处理还包括将第二发行部署到第二开发环境,并且将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容内,以便产生包括第三源代码和第三内容的应用的第三发行。所述处理还包括在已经将发行合并在一起之后,将应用的第三发行部署到生产环境,以便由应用的用户使用。所述部署包括将第三源代码的可执行版本和第三内容安装到生产环境的另一个应用平台。

[0015] 本发明的替换实施例还包括,由非技术的事务用户通过图形用户界面执行将对第二发行的第二源代码和第二内容的修改合并到第一发行的第一源代码和第一内容中,以便产生包括第三源代码和第三内容的应用的第三发行。本发明的实施例还包括,提供第二应用平台包括运行一个或多个脚本,以便在第二应用平台上安装源代码和内容。本发明的实施例还包括在将第三发行部署到生产环境中之前,将应用的第三发行部署到预生产环境,并且在预生产环境中测试第三发行。另外,这可以包括基于第二开发环境产生基线,其中该基线包括第二开发环境内的源代码和内容文件的状态;将应用的第二发行与基线进行比较;以及响应该比较,将应用的第二发行部署到第二开发环境。

[0016] 本发明的替换实施例还包括用于合并第一发行和第二发行之间的对源代码和内容的改变的图形比较。本发明的实施例还包括比较内容文件的呈现显示。本发明的实施例还包括为了进行视觉比较,选择性地突出显示内容文件的呈现显示。

附图说明

[0017] 结合如下附图考虑,并参考下面对本发明的详细描述,可以更全面地明了本发明的各种目的、特征和优点,在附图中类似的附图标记指示类似的元件。

[0018] 图 1 是示出了如何能够将开发处理的不同并行分支部署到生产系统的一种方式

的图;

[0019] 图 2 是示出了生产流和并行流的并行开发中所涉及的各种系统和实体的系统图;

[0020] 图 3 是示出了并行开发处理中所涉及的各种系统和实体以及其唯一制品 ID 的系

统图;

[0021] 图 4 是示出了如何能够使用环境配置器提供新的并行 BDE 环境的系统图;

[0022] 图 5 是示出了并行开发处理中所涉及的系统之间的关系

的系统图;

[0023] 图 6 是示出了并行开发处理中所涉及的系统之间的关系的更详细的系统图;

[0024] 图 7 是示出了第一发行的并行开发处理内的环境之间的信息流动的系统图;

[0025] 图 8 是示出了所述环境内的相应工作区域的系统图;

[0026] 图 9 是示出了第二发行的并行开发处理内的环境之间的信息流动的系统图;

[0027] 图 10 是示出了所述环境内的相应工作区域的系统图;

[0028] 图 11 是示出了第二发行的并行开发处理内的环境之间的信息流动的系统图;

[0029] 图 12 是示出了所述环境内的相应工作区域的系统图;

[0030] 图 13 是示出了 MPP 的部署组件的设计的系统图;

[0031] 图 14 是示出了 MPP 的安装组件的设计的系统图;

[0032] 图 15 是示出了 BDE 的实现

的系统图;

[0033] 图 16 是示出了 BDE 环境的进一步细节的系统图;

- [0034] 图 17 是示出了 MPP 内的系统之间的信息流动的系统图；
- [0035] 图 18 是示出了系统之间的目录信息的传递的系统图；
- [0036] 图 19 是示出了系统之间的通信的进一步细节的系统图；
- [0037] 图 20 是示出了能够用于 MPP 的目录系统的细节的系统图；
- [0038] 图 21 是示出了在并行 BDE 系统之间目录信息共享的系统图；
- [0039] 图 22 是示出了在 BDE 环境之间目录信息的传递的系统图；
- [0040] 图 23 是示出了完成的任务和其它系统消息的记录的到 MPP 的用户界面；
- [0041] 图 24 是示出了在并行开发和部署多个 web 应用中涉及的系统的细节的系统图；
- [0042] 图 25 是示出了并行流部署中所涉及的系统之间的链接的系统图；
- [0043] 图 26 是示出了 BDE 环境和部署系统中所涉及的系统之间的链接的系统图；
- [0044] 图 27 是示出了 MPP 的实现的细节的系统图；
- [0045] 图 28 是示出了 MPP 的灾难恢复功能的细节的系统图；
- [0046] 图 29 是示出了 MPP 的灾难恢复功能的进一步细节的系统图；
- [0047] 图 30 是示出了 MPP 的灾难恢复功能的进一步细节的系统图；
- [0048] 图 31 是示出了 MPP 的故障恢复功能中所涉及的组件的系统图；
- [0049] 图 32 是示出了部署并行流中所涉及的系统的组件的系统图；
- [0050] 图 33 是示出了用于创建新的并行流的整体处理的流程图；
- [0051] 图 34 是示出了用于初始化新的并行流的处理的流程图；
- [0052] 图 35 是示出了用于获得 MPP 的最新处理的处理的流程图；
- [0053] 图 36 是示出了初始化处理的细节的流程图；
- [0054] 图 37 是示出了部署处理的细节的流程图；
- [0055] 图 38 是示出了提升处理的细节的流程图；
- [0056] 图 39 是示出了部署处理和随时间的信息流动的系统图；
- [0057] 图 40 是示出了获得最新处理和信息随时间的流动的系统图；
- [0058] 图 41 是示出了可以如何配置不同安全组的 MPP 的用户界面；
- [0059] 图 42 是示出了获得最新处理的配置的 MPP 的用户界面；
- [0060] 图 43 是示出了系统的总体设计的 MPP 的用户界面；
- [0061] 图 44 是示出了代码部署工作流的开始的 MPP 的用户界面；
- [0062] 图 45 是示出了用于部署处理的配置设置的 MPP 的用户界面；
- [0063] 图 46 是示出了获得最新处理的一部分的 MPP 的用户界面；
- [0064] 图 47 是示出了部署处理的一部分的 MPP 的用户界面；
- [0065] 图 48 是示出了部署处理的结果的 MPP 的用户界面；
- [0066] 图 49 是示出了将代码部署到生产的 MPP 的用户界面；
- [0067] 图 50 是示出了将代码部署到生产的进一步细节的 MPP 的用户界面；
- [0068] 图 51 是示出了将代码部署到生产的进一步细节的 MPP 的用户界面；
- [0069] 图 52 是示出了部署处理的结果的 MPP 的用户界面；
- [0070] 图 53 是示出了如何比较项目的生产和并行版本的 MPP 的用户界面；
- [0071] 图 54 是示出了如何比较 web 页面布局的生产和并行版本的 MPP 的用户界面；
- [0072] 图 55 是示出了如何比较 web 页面的生产和并行版本的 MPP 的用户界面；

[0073] 图 56 是示出了合并处理的概要的 MPP 的用户界面 ;和

[0074] 图 57 是示出了如何比较代码的生产和并行版本的 MPP 的用户界面。

具体实施方式

[0075] 在详细解释本发明的至少一个实施例之前,应当理解本发明不限于其对在下述的描述中提出或在附图中示出的组件的构造和布置的细节的应用。本发明能够具有其它实施例,并能够以各种方式实施和执行。另外,还应当理解,此处采用的措词和术语是出于描述的目的,并且不应被认为是限制。

[0076] 从而,本领域的技术人员将明了,本公开所基于的概念可被容易地用作设计用于执行本发明的若干目的其它结构、方法和系统的基础。因此重要的是,本发明被认为包括此处描述的内容的等同构造,而不脱离本发明的精神和范围。

[0077] 另外,作为一个实施例的一部分示出或描述的特征可被用于其它实施例,以便实现其它实施例。另外,某些特征可被与执行相同或类似功能的未提及的类似设备或特征互换。因此,旨在将这些修改和变体包括在本发明的整体内。

[0078] 本发明的实施例涉及用于在整个开发、测试和部署处理中自动进行并行开发处理的系统。本发明给非技术用户提供了初始化并行流 ;将资源从并行流提升到生产流 ;将最新资源从生产流带入并行流,以及在流之间进行比较的自动框架。为了支持资源的并行开发,本发明的实施例给用户提供了创建被称为生产流的主干和被称为并行流的分支,同时成功地保持并行开发环境的能力。

[0079] 本发明的实施例利用分支系统,以便允许应用的并行开发,包括源代码和内容。用户可以工作于被称为生产流的主干和被称为并行流的分支。通过使用分支策略,可以与资源的其它版本相隔离地修改和管理相同资源(例如,网页的源代码文件或图像)的多个版本。本发明的实施例在包括测试环境、登台服务器、生产服务器、应用平台和内容管理系统的多个开发处理组成部分上应用这种分支策略。这些系统一般被称为事务开发环境(即, BDE 或 eBDE)。不论并行流的数目如何,优选地存在可被部署到生产环境并且被终端用户或顾客看到的资源的一个“活动的”版本。这意味着,如果必要,所有并行流最终必须被合并或整合到生产流内,从而资源可被发行到生产环境。虽然在技术上更具挑战性,本发明的实施例确保在任意时刻和在永久基础上,“生产”流相应于“活动的”流,其操作地进行哪个流相应于“活动的”流的猜测工作,并且确保在“设计时”而不是在“运行时”发现任何资源冲突。

[0080] 本发明的替换实施例管理资源到生产环境的部署,包括以自动方式将开发的并行流合并到生产流中。这些资源包括在 TeamSite 中管理的内容片段、网口资源(例如,页面定义、URL 映像和 portlet 应用)、分配给 portlet 应用的顾客验证模块(CAM)资源和安装在网口服务器环境内的共享应用组件中的一个或多个。本发明的实施例提供了创建新流,并且在流之间移动资源,以便能够进行单个资源和资源集合的并行开发的方法,这些资源然后可被移动到将被部署到生产“活动的”环境中的生产流内。本发明的实施例提供了创建新流,并且在流之间移动资源以便能够进行单个资源和资源集合的并行开发的方法,这些资源然后可被移动到将被部署到生产环境中的生产流内。其它自动操作可以包括初始化新的并行流,获得最新的流以便从其开始工作,比较流之间的差异(和流内的各个资源)、

管理测试和流的用户、以及执行合并改变的回滚。

[0081] 本发明的实施例包括连接到开发、测试、预生产和生产环境的基于 web 的系统。可以使用 web 界面进行通过这些系统对应用的管理。该 Web 界面还包括用于协作决策制定的能力,以及多个体参与的能力,包括批准或不批准某些动作,以及增加/修改/删除资源。通过在将资源部署到开发的下一个阶段之前各个地评审资源,该 Web 界面还提供详细地视觉管理合并处理的能力。这可以包括合并来自资源的并行版本的改变。

[0082] 使用这种自动系统,可以从公共开始点对多个特征有利地并行执行开发,然后这些特征可被一起组合到最终的生产流中。可替换地,一个分支可以首先进入生产,并且改变可被在以后结合到该生产分支内。还可以按照非技术非 IT 用户的指示,有利地自动管理和执行将并行流之一部署到生产环境。

[0083] 本发明的实施例包括与开发系统的组件接口的硬件和软件层,开发系统的组件包括内容管理系统(例如,Interwoven 的 TeamSite)、源代码管理系统和应用平台。硬件和软件层包括可被用于控制其操作的图形用户界面(可以包括非图形用户界面元素)。创建具有自己的同步策略开发环境(SDE)集合的并行 BDE,在该 SDE 集合上部署代码,并且在其中可以进行各种测试,诸如, SAT、UAT 和 Pre-Prod 测试。

[0084] 在本发明的至少一个实施例中,并行管理项目包括初始化处理,获得最新处理以及提升处理。使用并行开发流的处理以初始化并行流开始。这是并行流的 BDE 环境被配置、建立和安装生产流 BDE 的内容的处理。一旦已经初始化了并行流 BDE,它基本上是生产流的克隆,或大体类似于生产流。这允许从最后的当前版本对代码、内容和配置进行并行管理。

[0085] 为了并行管理已有的网口资源,这些网口资源优选地是相同的,并且这意味着在不同的并行流上,这些资源的内部唯一标识符必须相同。然而重要的是,当在并行流或生产流中创建新资源时,它们必须被给予不与已经在其它流上创建的资源相冲突的新的唯一标识符。这可被通过一个定位脚本进行。本发明的实施例使得非技术用户能够初始化并行流并且并行工作,同时以自动方式保持已有资源的唯一标识符或在必要时创建新的唯一标识符,而不用实际看到这些唯一 ID 是什么。

[0086] 为了实际开始并行开发流的工作,使用获得最新 workflow,以便将资源生产流移动到并行流,或从一个并行流移动到另一个并行流。作为获得最新 workflow 的一部分,对目标环境/并行流工作区域(其包含内容,包括具有版本的网口资源)与基线进行详细比较。基线还优选地包括该环境上已经安装的应用和其相关联的版本的列表。一旦已经执行了比较,该结果优选地被显示给 BDE Staff 用户(仍然在该 workflow 的上下文内),接着该用户可以对他们希望被同步的资源作出决定。这允许并行开发所有或仅仅部分的应用。

[0087] 使用提升处理将资源从并行流移动到生产流。生产流优选地处于这样的环境,该环境被用于将这些资源部署到预生产和生产环境中。提升处理对已经在并行流中做出的改变和其将被合并到的生产流(或其它并行流)进行比较。这些改变优选地被通过图形用户界面显示给非技术用户,从而 BDE 用户可以容易地接受或拒绝将被合并到新流内的改变。在合并流已被创建之后,它可被与必要资源一起可选择地自动部署到生产环境,以便被终端用户访问。

[0088] 并行开发系统的其它功能可以包括高级记录功能、灾难恢复保护和与适合于包括并行流的开发处理的 LDAP 系统的集成。

[0089] 本发明的实施例包括在计算机系统上进行实现。该计算机系统包括用于传递信息的总线或其它通信机制,以及与总线耦接的用于处理信息的处理器。该计算机系统还包括用于存储信息和将被处理器执行的指令的耦接到总线的主存储器,诸如随机访问存储器(RAM)或其它动态存储设备。在将被处理器执行的指令的执行过程中,主存储器还可被用于存储临时变量或其它中间信息。该计算机系统还包括用于存储用于处理器的静态信息和指令的耦接到总线的只读存储器(ROM)或其它静态存储设备。存储设备,诸如磁盘或光盘,被提供并且耦接到总线以便存储信息和指令。

[0090] 该计算机系统可被通过总线耦接到显示器,诸如阴极射线管(CRT),以便向计算机用户显示信息。包括字母数字和其它按键的输入设备被耦接到总线,以便向处理器传递信息和命令选择。另一种类型的用户输入是光标控制器,诸如鼠标、轨迹球或用于向处理器传递方向信息和命令选择,并且用于控制显示器上的光标移动的光标方向键。这种输入设备通常具有两个轴,即,第一轴(例如,x)和第二轴(例如,y)上的两个自由度,其允许设备指定平面内的位置。

[0091] 本发明涉及用于并行管理项目的计算机系统的使用。根据本发明的一个实施例,响应处理器执行包含在主存储器内的一个或多个指令的一个或多个序列,由计算机系统提供并行管理项目的系统。这些指令可被从另一个计算机可读介质,诸如,存储设备读入主存储器。包含在主存储器内的指令序列的执行使得处理器执行此处描述的处理步骤。还可以采用多处理布置内的一个或多个处理器,以便执行包含在主存储器中的指令序列。在替换实施例中,可以取代或结合软件指令使用硬布线电路来实现本发明。因此,本发明的实施例不限于硬件电路和软件的任意特定组合。

[0092] 此处使用的术语“计算机可读介质”指参与向处理器提供指令以便执行的任意介质。这种介质可以采取许多形式,包括但不限于非易失介质、易失介质和传输介质。非易失介质包括例如光盘或磁盘,诸如存储设备。易失介质包括动态存储器,诸如主存储器。传输介质包括同轴电缆、铜线和光纤,包含包括总线的线路。传输介质还可以采取声波或光波的形式,诸如在无线电波和红外线数据传输过程中产生的波。

[0093] 计算机可读介质的常见形式包括例如软盘、软磁盘、硬盘、磁带或任意其它磁介质、CD-ROM、任意其它光学介质、穿孔卡、纸带、具有孔图案的任意其它物理介质、RAM、PROM和FPROM、FLASH-EPROM、任意其它存储器芯片或盒、后文描述的载体、或计算机可以读取的任意其它介质。

[0094] 在向处理器传送一个或多个指令的一个或多个序列以便执行时,可以涉及计算机可读介质的各种形式。例如,指令最初可被承载在远程计算机的磁盘上。远程计算机可将指令加载到其动态存储器中,并且使用调制解调器在电话线上发送该指令。位于该计算机系统的调制解调器可以在电话线上接收数据,并且使用红外线发射器将数据转换为红外线信号。耦接到总线的红外线检测器可以接收红外线信号承载的数据,并且将数据放置在总线上。总线将数据传送到主存储器,处理器从主存储器检索并且执行指令。由主存储器接收的指令可被可选择地在处理器执行之前或之后存储在存储设备上。

[0095] 该计算机系统还包括耦接到总线的通信接口。该通信接口提供耦接到被连接到局域网的网络链路的双向数据通信。例如,该通信接口可以是综合业务数字网络(ISDN)卡或调制解调器,以便提供到相应类型的电话线的数据通信连接。作为另一个例子,该通信接口

可以是局域网 (LAN) 卡,以提供到兼容的 LAN 的数据通信连接。还可以实现无线链路。在任意这种实现中,通信接口发送和接收承载表示各种类型信息的数字数据流的电、电磁或光信号。

[0096] 网络链路通常通过一个或多个网络向其它数据设备提供数据通信。例如,网络链路可以通过局域网提供到由因特网服务提供商 (ISP) 运营的主计算机或数据装置的连接。ISP 又通过现在通常被称为“因特网”的世界范围的包数据通信网络提供数据通信服务。局域网和因特网两者使用承载数字数据流的电、电磁或光学信号。承载到和来自计算机系统的数字数据的通过各种网络的信号以及网络链路上的并且通过通信接口的信号是传输信息的载波的示例形式。

[0097] 计算机系统可以通过网络、网络链路和通信接口发送消息和接收数据,包括程序代码。在因特网的例子中,服务器可以通过因特网、ISP、局域网和通信接口传输应用程序的请求代码。根据本发明,为此处描述的并行管理项目提供一个这种下载的应用。

[0098] 图 1 是示出了开发处理的不同并行分支可被部署到生产系统的一种方式。该图包括主生产分支 102,以及开发的三个并行分支(流)104、106 和 108。每个流包括最终到达主生产分支 102 的多个版本(例如,118、120 和 122)。

[0099] MPP 系统使用分支策略,以便允许同时开发多个流。主生产分支 102 是活动的,或向用户发行的生产分支。其它分支可以同时工作,但是在进入将被终端用户使用的生产环境之前与主分支合并。

[0100] 并行开发分支 104、106 和 108 表示主分支的其它开发。为了组织这些多个分支,分配示出它们被从主分支(或父分支)中的何处得出的版本号,并且为该分支内的版本分配子版本号。在图 1 的示例图中,并行分支 104 的版本 2.2 112 被合并到生产分支内作为版本 4.0 118,而并行分支 106 的版本 2.1.2 114 被合并到生产分支的版本 5.0 120 内,并且而并行分支 108 的版本 1.3 被合并到生产分支的版本 6.0 122 内。这显示了可以从生产分支内的最初版本,从生产分支内的后续版本(并行分支 104),并且还从并行分支内的版本(并行分支 106)得出并行开发分支。

[0101] 生产分支和并行分支中的每一个都是正被开发的整个项目的一个分支版本。这包括被在内容管理系统(例如,TeamSite)内管理的内容片段、网口资源(例如,页面定义、URL 映像)、portlet 应用(例如,较大的应用)、CAM 资源(被分配给 portlet 应用)以及共享的应用组件(安装在网口服务器环境内)的分支版本。这允许 MPP 系统被以内聚性和自动的方式用于整个项目,而不是仅仅孤立地用于代码、html 或内容。

[0102] 图 2 是示出了具有生产流和并行流的并行开发中所涉及的各种系统和实体的系统图。该图示出了具有事务开发环境 (BDE) 206、同步策略开发环境 (SDE)、预生产环境 210 以及生产环境 210 的生产流 202。类似地,存在具有 BDE220 和同步 SDE222 的并行流 224。这两个环境通过 MKS214 和非同步 SDE216 连接。平台组件,诸如内容仓库、Interwoven TeamSite 和 IBM Portal Server 组件不明确地支持资源分支,每个组件在任意一个时刻仅可以具有资源的一个活动的版本。因此,本发明的实施例使用基于硬件和环境的解决方案,而不是纯粹基于软件的方案。然而,在替换实施例中,可以根据此处描述的处理实现基于软件的解决方案。

[0103] 在生产流 202 中,BDE 用户工作于 BDE206,以开发和管理可被部署到同步 SDE208

的内容。BDE 被用于指示内容管理系统和网口服务器的组合。BDE 可以包括 TeamSite 和 PortalServer 两者,并且允许非技术用户创建网口页面和内容片段,并且将这些页面与功能 IT 制品组装在一起。同步 SDE208 被用于在部署到生产环境 212 之前,测试内容和代码两者。生产环境是资源的最后版本所在的环境,并且是顾客和前端用户将访问的环境。另外,处理可以包括部署到预生产系统 210,以便在部署给用户之前,更完整地测试整个流,包括内容、代码和 html。预生产环境是配置为模拟生产环境的配置的同步 SDE。这可以包括主机连接、防火墙或安全配置或附加服务器设置。这是不是所有项目都需要的选择性的环境,但是可被用于增加测试时的置信级别。该同步 SDE 直接连接到 BDE,允许能够将用户已经组装的代码和内容公开到可被用于各种类型的测试,即,UAT、OAT 等的环境中。多个 SDE 可被“连接到”单个 BDE,从而可以并行执行各种类型的测试。

[0104] 并行流 224 还具有使用独立的 BDE220 的用户 218,正在使用 BDE220 开发并行流。为了测试这个流上的代码和内容,使用独立的同步 SDE222。并行流还将具有自己的同步 SDE 集合,在其中可以发生各种测试。然而,相同的非同步 SDE 可被用作测试 IT 制品的系统,虽然如果这种测试必须被并行执行,则可能使用多个非同步 SDE 环境。由于并行流不能被部署到生产环境,并行流不需要预生产或实际生产环境。

[0105] MKS 是用于 IT 制品的源控制系统,包括主题、皮肤和 portlet 应用。MKS 具有内建并行开发支持和工具,以便允许开发者并行编写和管理 IT 制品。该系统内的内容和代码可被部署到多个并行流以用于进一步开发。MKS214 连接到非同步 SDE,以便测试代码和内容,而不与任何特定 BDE 和开发流同步。非同步环境是用于部署 IT 制品和执行系统测试的独立网口服务器实例(不连接到 TeamSite 的 BDE)。

[0106] 图 3 是示出了并行开发处理中所涉及的各种系统和实体以及它们的唯一网口 ID 的系统图。可以通过提供或映射完成在其上开发并行流的并行环境。在映射时,硬盘的镜像被拷贝到另一个系统,以便快速地产生相同的环境。在提供时,使用脚本或自动工具,以脚本定义的方式产生和配置并行环境。

[0107] 存在于并行流内的给定资源必须具有相同的对象 ID,确保相同的对象被在每个并行流中修改。图 3 示出了由于提供,并行流诸如实体 A 332 和实体 B324 内的对象 ID 环境如何可以不同。实体 A 的同步 SDE304、预生产环境 306 和生产环境 308 的对象 ID 与实体 B 的同步 SDE312、预生产 314 和生产 316 的对象 ID 不同。类似地,具有并行 BDE 318 和同步 SDE320 的基于实体 B310 的并行流也具有不同的对象 ID。可以使用脚本在生产环境诸如 302 和 310 之间同步对象 ID。

[0108] 图 4 是示出了可以如何使用环境配置器 406 提供可以在其上发生并行开发的新的并行 BDE 和 SDE 环境的系统图。环境配置器使用基础镜像 402,以及配置模板 404 和任意补丁 408,以便创建网口环境 410。然后 BDI 安装处理 412 增加附加组件,以便完成提供。

[0109] 环境配置器 406 是运行在工作站或其它计算机上的软件,并且它可被以任意数目的编程或脚本语言编写。环境配置器可以使用设置模板 404 或可被送入处理的参数,以便从单个基础镜像 402 创建不同结果配置。这允许对默认环境配置进行快速改变,而不必编写复杂的脚本或使用附加处理。该处理的输出是可被用作 BDE 或 SDE 的空白网口环境 410。然后,通过安装其它二代 (2G) 组件 412,可以使得该环境适合于更特定的功能。

[0110] 图 5 是示出了并行开发处理中所涉及的系统之间的关系的系统图。并行开发处理

包括应用开发系统 504、源代码管理系统 506 和非同步 SDE502。在测试代码和内容之后,在被最终部署到生产同步 SDE514 以便由终端用户使用之前,应用可被部署到具有网口平台 510 和内容管理系统 512 的并行 BDE508。

[0111] 开发系统 504 是任意源代码或内容开发系统,诸如 Websphere Studio Application Developer。工程师和设计师使用该系统,以创建最终被部署到生产系统的代码或内容。MKS 是存储正被开发的代码的源代码管理系统。当开发代码时,其可被在不连接到任意 BDE 的非同步 SDE502 上测试。这提供了一种测试代码的方便的方法。

[0112] BDE508 由网口平台 510 构成,其被用于创建和修改 web 应用的页面。该内容被存储在内容管理系统 512 例如 Interwoven TeamSite 内。一旦为终端用户准备就绪,生产系统 514 执行该代码和内容。网口平台 510 和内容管理系统 512 被通过网络连接。

[0113] 在操作中,使用开发系统 504 为作为并行开发环境项目的一部分的特定应用开发源代码,并且在源代码管理系统 506 中对其进行检查。当开发的源代码准备好发行时,在源代码管理系统 506 内创建检查点。然后,提取源代码,并且建立 web 应用档案 (WAR)、EAR 等,并且将其部署到非同步 SDE502 以便进行测试。

[0114] 接着,从源控制管理系统 506 和内容管理系统提取源代码、主题、皮肤等,并且建立 WAR、EAR 等。得到的“建立包”被部署到部件网口平台 510。页面、内容片段等被在 eBDE508 中创建 / 修改,并且被存储在内容管理系统 512 内。当内容制品已经准备用于发行时,创建编辑,以类似于在源代码控制系统中对以 IT 为中心的制品所做的方式,有效地确定非技术事务用户内容发行的版本。

[0115] 为了最终发行作为 web 应用的代码和内容,从部件网口平台 510 导出“发行包”(例如,具有 Websphere 网口对象 ID),并且在源代码管理系统 506 中进行检查。然后,从源代码管理系统中提取“发行包”,并且将其导入 SDE514。不论页面定义、内容片段和其它内容如何,它们被从内容管理系统 512 中“提取出来”,并且“被导入”SDE514。以这种方式,SDE 具有现在可被终端用户使用的应用的代码和内容两者的带有版本号版本。

[0116] 图 6 是示出了并行开发处理中所涉及的系统之间的关系的更详细的系统图。该图示出了开发队伍 602、开发建立 604、开发集成和测试系统 606 和发行环境 608 之间的关系。

[0117] 开发系统包括用于作为开发和测试工具和环境的主机的开发者工作站 610-616。开发环境给系统提供用于创建代码 604 的源代码。这些系统包括仓库管理工具 618,仓库管理工具 618 自身包括 SCM620。SCM620 从仓库向项目管理附件元件 622 提供源代码。这些附加元件包括用于创建源代码的创建服务器 624 和用于对创建处理进行报告的报告工具 626。

[0118] 在已经创建了源代码之后,它被提供给集成测试系统 606。系统 628 被用于创建的代码的测试,系统 628 包含测试系统 630-634 中的一个或多个和 BDE636, BDE636 包括 BDE638-640 中的一个或多个。

[0119] 在已经完成了测试之后,创建的代码和内容被提供到发行包版本管理服务器 642。该服务器提供存储和管理应用和内容的测试的位置,它还提供用于将应用部署到预生产系统 646 以便进行操作接受性测试 (OAT) 648 和用户接受测试 (UAT) 650 的平台。一旦完成了发行测试,它可被部署到生产服务器 644。

[0120] 图 7 是示出了第一发行的并行开发处理内的环境之间的信息流动的系统图。代码

和内容的流 1 发行 702 被示出为在被部署到生产系统 712 之前从开发环境通过 BDE 环境 706 和测试环境 708 和 710 移动。

[0121] 流 1702 表示正被开发的代码和内容的集合。在移动到 BDE706 以便与内容一起测试之前,代码在开发系统 704 上开发。在已经在 BDE 系统上成功测试代码之后,其可被部署到用于用户接受测试 (UAT) 708 的另一个环境。在已经执行了用户接受测试之后,该发行可被部署到紧密相应于生产环境 712 的预生产环境 710,以便进行终端用户使用之前的最终测试。

[0122] 在开发和测试处理过程中,可以编辑和修复代码和内容。这些反向流被以箭头 714 示出,箭头 714 示出了 BDE706 上的代码和内容的测试和编辑循环。针对没有通过用户接受测试的代码,这还从 UAT 回到 BDE706 的箭头 716 和从 UAT 回到开发系统 704 的箭头 718 示出。还以从预生产到 BDE 的箭头 720 和从预生产到开发 704 的箭头 722 示出了修订代码和内容的另一路径。

[0123] 图 8 是示出了当代码在不同开发和测试环境中流动时,所述内容管理系统内的相应工作区域的系统图。内容管理系统 814 具有内部分支存储结构 802,以便存储内容的多个版本。根据 MPP,相应于具有附有代码的内容的发行形成这些内部版本。内部区域 802 在这个例子中被示出为具有单个代码分支的 3 个版本,804、806 和 808。这些将相应于图 1 所示的 V1.0、V2.0、V3.0。版本 808 被示出为被部署到生产,并且相应于代码的完全测试并行版本。其它版本 804 和 806 可以相应于被开发并且被合并到最终版本 808 内的代码的并行版本,或未通过测试并且需要修订的内容的版本。

[0124] 图 9 是示出了基于第一发行的第二发行的并行开发处理内的环境之间的信息流动的系统图。这个图示出了流 1902 和流 2914,以及当部署到生产 912 时它们之间的代码和内容的流动。

[0125] 流 1 类似于图 8 所示的流,并且沿着从开发环境 904 到 BDE960 到 UAT908 到预生产 910 并且最终到达生产 912 的路径。类似于图 8,代码可被修改并且通过部署流向后移动以便重新测试。

[0126] 流 2 表示相对于生产流的应用的并行开发流。该流也具有开发环境 920、BDE 环境 922 和 UAT 测试环境 924。这些开发和测试环境被与主路径同步,允许合并任何改变。

[0127] 除了如以箭头 916 所示,流 2 914 从主路径的发行开始之外,流 2 914 沿着与为流 1 902 所示类似的路径。流 2914 然后移动到 eBDE922 和 UAT 924 以便进行测试。它从测试移动到生产环境,并且在被合并到生产 912 内之前,被 eBDE906、UAT 908 和预生产 910 进一步测试。在流 2 被部署到生产 912 时之前,它首先被合并到流 1 内。这样,仅存在一个分支,即,生产分支。箭头 938 示出了流 2 的 BDE 922 和 BDE906 之间的这种提升或合并。

[0128] 图 10 是示出了相应于代码和内容的发行版本的内容管理系统内的相应工作区域 1002 和 1034 的系统图;类似于图 8,该图示出了内容管理系统的内部分支结构。

[0129] 内容管理系统 1022 被示出为具有工作区域 1002,工作区域 1002 具有三个项目版本(版本)1004、1006 和 1008。该代码被在 BDE1020 上测试,其被示出为安装有 TeamSite 分支 1 1022 和 Websphere 平台 1024。然后,代码被部署到生产 1026。

[0130] 内容管理系统还包含用于并行流 2 内的内容的版本的另一个工作区域 1010。如图所示,这个工作区域 1010 包含项目的多个版本。这个例子示出了四个版本 1012、1014、

1016 和 1018。这个工作区域中的代码的版本被部署到 BDE 环境 1022, 其被示出为安装在 TeamSite 分支 21030 和 Websphere 网口平台 1032 内。

[0131] 箭头 1034 示出一个并行流中的项目可被如何分支到另一个并行流以便进一步开发。在该情况下, 流 1 的工作区域 1002 的 ed_003 1008 被作为 ed2_0001 分支到流 2 的工作区域 1010。从此, 示出了已经创建的其它版本 (例如, 1014、1016 和 1018)。类似地, 在使用 BDE1022 进行了并行版本的开发和测试之后, 代码可被合并回流 1。例如, 代码可被合并到第一个流的 ed_0003 1008 内。然后, 代码可被部署到生产 1026。

[0132] 图 11 是示出了第二发行的并行开发处理内的环境之间的信息流动的系统图。这更清楚地示出了在被部署到生产之前, 第一个流的第二发行 (例如, 非并行版本 1.1) 经过与流 1 的第一个版本相同的环境。这包括开发环境 1101、BDE1104、UAT1106、预生产 1108 和生产 1110。

[0133] 图 12 是示出了所述环境内的相应工作区域的系统图。图 12 更直接地示出代码的版本 1218 可被从并行分支 1220 移回生产分支 1222 (例如, 使用路径 1212 和 1214 从 ed2_0004 到 ed_0004 1216)。然后, 这个新版本 1216 可被部署到生产 1206。

[0134] 图 13 是示出了 MPP 的部署组件的设计的系统图。部署组件具有 SCM 服务器 1302、包括 SCM 客户端脚本 1306 和部署脚本 1308 的脚本集合 1304。使用这些脚本, 代码和内容被部署到一个或多个数据中心, 例如, 数据中心 1 1310 和数据中心 2 1316。

[0135] SCM 服务器 1302 包含将被部署到测试和生产系统的代码和内容。Getzip 函数是将 SCM 服务器连接到开放部署组件的 MPP 系统的函数。Getzip 函数检索保存在内容管理系统服务器 1304 上的建立权证 (ticket), 并且使用建立权证中的信息创建部署包。ConfigLabel 函数将内容管理系统连接到 SCM 服务器, 并且在内容管理服务器上保存检索到的被标记的文件。ConfigLabel 函数还合并检索到的建立权证 zip 文件。部署工作流程中的最后一个函数是 DeployZip 函数, 其使用开放部署脚本将保存的 zip 文件传送到特定数据中心内的目的地运行时部署帮助器 (RDH)。

[0136] 图 14 是示出了 MPP 的安装组件的设计的系统图。安装工作流程安装部署的建立权证。开放部署组件 1404 控制从 RDH 服务器 1414 和 1424 到数据中心 1416 和 1428 的建立权证的部署处理。开放部署处理包括 DeActWeb 1406、安装 1408、ActWeb 1410 和回滚 1412 任务。

[0137] DeActWeb 处理使用开放部署服务 1418 触发目的地 RDH 服务器 1414 上的脚本, 以便在准备新建立的安装时去活 webserver 实例。如果没有问题, 工作流向用户报告处理的成功, 并且自动地开始作为开放部署处理的一部分的下一个处理。

[0138] 安装处理 1408 使用开放部署服务 1418 触发目的地 RDH 服务器 1414 上的脚本, 以便安装在部署 zip 任务过程中创建的建立权证 zip 文件。

[0139] ActWeb 处理 1410 使用开放部署服务 1418 触发目的地 RDH 服务器 1414 上的脚本, 以便激活 webserver 实例。如果没有问题, 工作流向用户报告当前处理的成功, 并且自动开始开放部署处理的下一个部分。

[0140] 回滚处理 1412 使用开放部署服务 1418 触发目的地 RDH 服务器 1414 上的脚本, 以便恢复备份文件系统和网口数据库。如果开放部署服务停机或如果在开放部署处理过程中安装脚本发生失效, 回滚可能是必须的。

[0141] 图 15 是示出了 BDE 环境的实现的系统图。BDE 环境包括通过客户端 1504 (例如, web 接口) 连接到 BDE 环境 1508 的用户 1502。BDE 环境 1508 包括 web 服务器 1514 和支持系统 1510 以及数据库系统 1512。

[0142] 根据本发明的至少一个实施例,网口服务器内的资源具有使用网口 GUPID 产生的唯一标识符。如果网口 GUPID 在两个环境之间相同,存在流之间的两个或多个对象 ID 可能相同的风险。这将意味着管理具有相同唯一标识符的两个不同资源。并行流 BDE 的行为方式与同步 SDE 类似。然而不同于 SDE,用户可以在并行 BDE 上创建新资源,从而确保 GUPID 在环境之间是唯一的。以这种方式,确保所有新创建的资源真正具有唯一标识符。在替换实施例中,可以使用其它标识机制标识资源。

[0143] 详细地,用户 1502 是任意用户,诸如访问 MPP 系统的软件开发者。该用户通过计算机 1504 使用由 BDE 环境 1508 通过网络链路 1506 提供的 web 界面。由例如 IBM http 服务器 1514 结合应用服务器 1516、网口服务器 1518 和 LDAP 服务器 1520 的 web 服务器 1510 提供该 web 界面。应用服务器 1516,诸如 websphere,提供运行 web 应用的能力。以这种方式,可以使用 java 应用建立 MPP 系统。Websphere 网口服务器 1518 提供应用层的表示层。

[0144] BDE 环境还包括数据库层 1512。该数据库层包括用于存储在网口内使用的对象和页面的网口数据库 1522。该数据库层还包括用于存储正工作的项目的 BDE 数据库 1526,以及出于标识和验证目的,用于存储用户简档信息的 ID&V 数据库 1528。

[0145] 图 16 是示出了 BDE 环境的进一步细节的系统图。BDE 环境包括由网口服务器、标识和验证组件、数据库组件 1616、应用服务器 1614 和网口服务器 1612 构成的基础基础构造和配置层。BDE 环境还包括由基础 BDE1604、BDE 运行时 1606、ID&V 组件 1608 和用于网口和应用的代码和内容 1610 构成的基础层。BDE 环境还包括具有相应于并行项目的实体特定代码和内容的实体层 1602。

[0146] 图 17 是示出了 MPP 系统内的组件之间的信息流动的系统图。该系统包括生产部件网口实例 1702 和并行网口部件实例 1714。该系统还包括 teamsite 仓库 1704 和 MKS 仓库 1716。MPP 系统的其它组件包括设计时建立帮助器 1706、开放部署服务器 1708 和运行时部署帮助器 1710。

[0147] MPP 系统由生产部件网口 1702 构成,生产部件网口 1702 是已被测试和部署的生产应用。在应用被部署到生产系统之前,它被在并行部件网口实例 1714 上并行开发。这种并行开发环境可以访问源代码控制系统 1716 内的代码的并行分支。通过源代码控制系统的客户端界面进行访问。源代码控制系统的一个例子是 MKS。并行开发环境还被连接到内容管理系统 1704,诸如 TeamSite,其具有相应于正被开发的代码的并行分支的内容的子分支 1712。可以通过类似 CIFS 协议的协议在应用机器上访问存储在内容管理系统内的内容。

[0148] 内容管理系统 1704 和源代码控制 1716 系统被连接到设计时建立帮助器 1706,设计时建立帮助器 1706 和开放部署服务器 1708 包含用于管理如上所述的并行流的部署的脚本和代码。首先,向运行时 DH

[0149] 1710 进行部署,并且然后通过 SSH 向生产部件网口实例 1702 进行部署。

[0150] 图 18 是示出了系统之间的目录信息的传递的系统图。图 18 示出了安装在刀片机架 1812 内的生产 BDE 环境 1802。部件网口刀片 1806 包括用于存储与部件网口界面 1810 有关的内容的数据库 1806。BDE 环境还包括用于验证目的的 LDPA 目录服务器 1808。

[0151] 并行开发环境 1810 被连接到生产 BDE 环境 1802, 并行开发环境 1810 也被安装在刀片机架 1812 内。并行开发环境还包括内容数据库 1814 和部件网口实例 1816。为了简化构造, 并行开发环境可以使用来自生产环境的目录信息。可替换地, 生产和并行环境两者可以具有其自己的可被使用复制同步的目录信息。这也使得故障恢复成为可能, 增加了容错性。

[0152] 图 19 是示出了生产部件网口实例 1902 和并行开发部件网口实例 1910 之间共享的目录信息的更高级版本的系统图。该系统类似于图 18 所示的系统, 然而, 它使用集中的并且冗余的目录服务器。生产部件网口实例 1902 被通过内容交换机 1918 连接到第一目录服务器 1920。第一目录服务器 1920 被连接到第二目录服务器 1922。这两个目录服务器可以在它们自己之间复制信息, 并且彼此提供故障恢复支持。以这种方式, 并行部件网口实例可以与生产部件网口实例一样连接到相同的集中式目录服务器 1920 和 1922。

[0153] 图 20 是示出了能够用于 MPP 的目录系统的细节的系统图。目录系统 2001 是用于管理用户信息的分层目录系统。目录系统 2001 被划分为不同的区域 (例如, 美国区域 2002), 每个区域还可被划分为多个环境 2006 和 2007。在这种系统内, 能够在生产和并行环境之间移动、拷贝和重新产生用户账户。还能够从单个位置和界面同步所有并行流并且管理整个目录。

[0154] 图 21 是示出了在并行 BDE 系统之间共享目录信息的系统图。该图示出了生产 BDE2102 和并行 BDE2104。生产 BDE 包括应用环境 2112 和支持基础构造 2108。它还包括目录服务器 2106, 目录服务器 2106 可被生产 BDE2102 和并行 BDE2104 两者使用。并行 BDE2104 仅包括应用环境 2114 和支持基础构造 2110。

[0155] 图 22 是示出了 BDE 环境之间的目录信息的传递的系统图。该图示出了生产环境 2202 和多个并行流 (例如, 2204、2206、2208 和 2210)。第二个流 2204 和第三个流 2206 具有与生产流 2204 不同的 ID&V 组件版本 (例如, 2218 和 2220)。然而, 由于它们是兼容的, 它们可以共享生产流的目录服务器 2216。第四个并行流 2208 被示出为具有 ID&V 组件的不兼容版本, 并且因此指向本地 LDAP 服务器 2222。第五个并行流 2210 具有相同版本的 ID&V 组件, 但是例如由于它需要单独数据使用, 仍然能够使用本地目录服务器。

[0156] 图 23 是 MPP 的用户界面的一部分, 并且示出了完成的任务的记录以及其它系统消息。MPP 系统被设计为记录 workflow 消息。该记录可以具有事件日期 2302、创建事件的任务 2304、执行事件的用户 2306 和事件细节 2308。

[0157] 该记录可以包括与部署和合并处理有关的消息。例如, 其可被显示在比较结果旁边, 以便提供在比较处理过程中做出的决定的全面视图。可以通过直接观看记录文件获得关于 MPP 的执行的更详细的信息, 例如, 关于失败的执行的信息、关于过早退出或 workflow 初始化的信息。可替换地, 信息还可以包括在基于 portlet 的观看器中。

[0158] 图 24 是示出了在并行开发和部署多个 web 应用时所涉及的系统的细节的系统图。该图示出了构成 MPP 平台的系统。这包括 ID&VCAM 数据库 2402、部件网口实例 2404 和部件网口数据库 2406。该系统还包括 MKS 仓库 2408、应用平台 2410 (包括 J2EE 应用部件 2412 和编译器 2414)、以及 team site 仓库 2416。开放部署处理 2422 运行在这个平台上, 开放部署处理 2422 将项目部署到登台环境 2426 和生产环境 2428。部署处理中所涉及的其它个体包括调度发行的个体 2420、帮助质量控制的个体 2424 和内容编辑器 2418。

[0159] ID&V CAM用于确保顾客账户被允许接入网口应用。正在执行的部件网口实例 2404 访问这个账户信息以便检验用户。它还组合与应用的内容有关的部件网口数据库 2406 内的数据。该应用运行在 J2EE 平台 2140 上, J2EE 平台 2140 包括用于组装应用的应用组装器 2412 和用于编译应用代码的编译器 2414。

[0160] 项目存储在 MKS 仓库 2408 内。应用的其它部分, 诸如内容和网口制品存储在 TeamSite 仓库 2416 内。以 TeamSite 内容编辑器 2418 编辑内容。

[0161] 编译后的应用被开放部署处理 2422 部署, 开放部署处理 2422 将应用部署到登台环境 2426 以便进行质量控制 2424。当被根据调度系统或项目管理者 2420 触发时进行部署。可以在登台环境中使用测试工具 2432 或其它工具 2430 测试应用。在测试并且部署到登台环境之后, 应用可被部署到生产环境 2428。

[0162] 图 25 是示出了并行流部署中所涉及的系统之间的链接的系统图。部件网口实例 2502 内的信息被使用 CIFS 传递到 TeamSite 仓库 2504。Team site 仓库内容 2504 和 MKS 仓库内容被分别使用 CIFS 和 MKS 客户端传递到设计时建立帮助器 2508。设计时建立帮助器 2508 然后将其输出发送到开放部署服务器 2510。从开放部署服务器, 应用被使用开放部署和 SSH 发送到 SDE 运行时部署帮助器 2512。应用还被使用开放部署和 SSH 发送到生产运行时部署帮助器 2518。从 SDE2512 和生产运行时 DH2518, 应用被使用开放部署处理和网络文件系统 (NFS) 发送到 IHS2514 和 2520, 并且被发送到 websphere 平台 2516 和 2522。

[0163] 图 26 是示出了通过 BDE 环境和部署系统的内容流动的系统图。IT 开发者 2602 使用 Rational Application Developer (RAD) 2610 创建内容。事务用户 2604 也可以使用 TeamSite 内容编辑器 2614 和部件网口实例 2612 输入内容。然后 IT 制品被存储在 MKS2608 内, 并且事务管理内容和页面部件 / 组成物被存储在 TeamSite 文件服务器 2616 内。从此处, IT 制品被转移到设计时建立帮助器 2622。设计时建立帮助器包括用于检索关于 MKS 仓库 2608 的信息的 MKS 客户端 2624。它还包括用于组装关于 TeamSite 文件服务器 2616 的信息的应用部件工具 2632。

[0164] 从此处, 内容被提供给开放部署服务器 2618, 开放部署服务器 2618 具有开放部署 GUI 2620。然后, 内容被提供给运行时部署帮助器 2636 (其具有脚本组件 2638 和开放部署服务器 2640)。从运行时部署帮助器, 内容被提供给 web 部署管理器 2642 和 websphere 网口 2644。部署的内容可被存储在 Oracle 数据库 2646 内。

[0165] 图 27 是示出了 MPP 的实现的细节的系统图。可视化刀片 2714 内的部件网口实例 2712 可以通过 CIFS 获得工作区域 2702 和 team site 内容。该刀片被包括在刀片机架 2710 内, 刀片机架 2710 包括相应于其它 BDE 实例的若干其它刀片。应用被使用 IBM RAD2704、MKS 2706、设计时建立帮助器 2720 和前面所述的应用部件工具 2722、Net HSM 2724 和 SDE 2726 部署到刀片机架。

[0166] 图 28 是示出了 MPP 系统的灾难恢复功能的细节的系统图。该图示出了使用刀片机架 2804 和刀片机架 2820 上的 MPP 处理开发的两个部署的应用 2802 和 2818。刀片机架 2802 包括如前所述的部件网口刀片 2804 以及同样如前面所述的设计时建立帮助器 2806。刀片机架 2802 被通过 NAS 网关 2808、SAN 2810、SRDF 连接 2812 和刀片机架 2818 的相应 SAN 2814 和 NAS 网关 2816 连接到刀片机架 2818。

[0167] NAS 网关 2808 提供对存储在 SAN 2810 上的文件的访问。该 SAN2810 被使用 SRDF

协议与 SAN 2814 同步,虽然如本领域已知的还可以使用其它协议。同步的存储允许刀片机架 2802 和刀片机架 2818 之间的故障恢复。

[0168] 图 29 是示出了 MPP 的灾难恢复功能的进一步细节的系统图。通过被修改为指向 DR 站点的主机名称 IP 的 URL,用户被定向到灾难恢复 (DR) 站点 2920。通过使用将 BDE Staff 用户自动带到处于运行的站点的公共 URL,BDE Staff 用户将被带到适当的站点。部件网口刀片 2904 被以通常的方式 (即,通常的 BDE 环境安装)2906 安装在 NAS 网关 2916 和 SAN 2918 上,如前所述,其与 DR 站点 2920 同步。

[0169] 通常的 BDE 环境包括用户数据库 2908 和应用部件网口 2912。应用网口为目录信息与 LDAP 2914 通信。

[0170] 图 30 是示出了 MPP 的灾难恢复功能的进一步细节的系统图。用户 3002 连接到与数据库系统 3006 通信的主应用系统 3004。数据库系统 3006 包括主数据库 3008、主记录目录 3010 和复制的备用记录 3012。类似地,备用环境包括备用应用系统 3018、备份数据库系统 3020。备份数据库系统 3020 包括副数据库 3022、备用记录目录 3010 和复制的主记录目录 3014。复制的主记录目录 3014 和备用记录目录与主站点保持同步。

[0171] 图 31 是 MPP 的故障恢复功能所涉及的组件的系统图。用户 3102 连接到 DNS URL 重定向组件 3104,DNS URL 重定向组件 3104 可以使用户指向主站点或故障恢复站点。DNS URL 组件 3106 被连接到 Veritas VIP 3106。主设计时建立帮助器 3108 和备用设计时建立帮助器 3101 两者被连接到 Veritas VIP3106 以便为请求服务。可替换地,附加的系统可以是故障恢复集群的一部分。使用来自 SAN 3114 进行服务请求,并且通过 NAS 网关 3112 对服务请求进行检索。SAN3114 使用 SRDF 复制保持同步。使用 DNS 故障恢复将用户重定向到故障恢复站点。

[0172] 图 32 是示出了部署并行流时所涉及的系统的组件的系统图。该系统包括运行时部署帮助器 3202,运行时部署帮助器 3202 包括用于将 EAR 包部署到 Websphere 部署管理器 3210,以及用于将 WAR 部署到 Websphere 网口 3212 的 ANT 和脚本 3204。部署的环境 3203 还包括用于支持 Websphere 网口 3212 的 Oracle 数据库 3214。对象被由 ANT 和脚本组件 3204 使用诸如 XML Access 的 Websphere 网口脚本工具加载到数据库 3214。

[0173] 图 33 是示出了用于创建新的并行流的整体处理的流程图。该处理以用户决定正在开发的项目究竟是否需要并行流而在步骤 3302 处开始。在步骤 3304,用户完成并行 BDE 请求。这是用户提供的需要创建并行流的信息。它可以包含对生产流的引用 (服务器名称和关于将访问 MKS 和其它环境组件的用户的信息)。

[0174] 在步骤 3306,为新的并行流提供环境。在步骤 3308,为了支持这个新的流,建立一个环境,其包括安装软件组件,诸如 IBM、BDE、ID&V 核心组件。在步骤 3310,测试安装,并且根据并行流请求配置环境和其它组件。在步骤 3312,在已经建立了环境之后,执行初始化工作流。在步骤 3314,同步新测试并且配置的环境。

[0175] 图 34 是示出了用于开始新的并行流的处理的流程图。该处理由启动器 3410 开始创建并行流的处理而在步骤 3402 处开始。然后在步骤 3404,由批准器 3412 批准在并行环境中初始化新的并行流。参考上文图 33 描述,在为并行流创建的环境中创建并行流。在步骤 3406,检验并行流的目标环境,以便确保它是可接受的。在步骤 3408,进行项目的加载以便创建并行流。

[0176] 初始化新的并行流是这样处理的，借此，以生产流 BDE 的内容配置、设置和安装并行流的 BDE 环境。一旦已经初始化了并行流 BDE，它基本上是生产流的完整克隆。这允许用户有利地管理并行前进的代码、内容和配置。为了并行管理网口资源，它们必须是相同的，这意味着这些资源的内部唯一标识符必须相同。然而重要的是，当在并行或生产流中创建新的资源时，它们被给予不与已经在其它流上创建的资源冲突的新的唯一标识符。

[0177] 当重新初始化并行流时必须小心，这是由于该并行流的内容将被删除。为了减轻初始化工作流错误运行的风险，一旦工作流已经正确运行，该工作流可被禁止，并且通过发出工作命令可被重新启动。在开始初始化工作流之前，用户退出目标上的部件 WPS 工作台，保存并且关闭目标工作区域内的文件，没有其它批准工作流在目标环境上处于进行中，并且已经创建了用于初始化工作流的源版本。可以从目标流的工作区域的内容窗格开始初始化工作流。

[0178] 如果初始化工作流被拒绝，可以向用户发送电子邮件。然后可以解锁目标，这使得再次为进一步的开发读 / 写目标上的资源。还去除针对该处理的工作流锁。然后以发生的动作更新记录。

[0179] 检验目标任务确保正确地执行初始化工作流。使用工作流窗格的任务视图（下面描述），用户可以选择任务列表中的检验目标任务。然后，用户选择开始输入任务，并且选择检验目标链接。然后用户通过观看所选择的页面检验目标流。用户还可以观看公共可用的网站。为了在检验之后接受部署，用户可以选择接受部署按钮。

[0180] 作为响应，系统执行下面的任务，禁止工作流，发送完成的电子邮件，解锁工作台，解除工作流锁并且记录执行的动作。

[0181] 用户还可以恢复目标流，这将流放回到任何改变被作为该工作流的一部分之前的状态。当用户拒绝目标流的部署时，开始这个任务。恢复功能是工作流窗格内的任务。为了执行恢复，系统解锁备份工作区域，从获取的备份中恢复目标流，向用户发送电子邮件，解锁工作台以便用于进一步的开发，解锁工作流，并且记录执行的工作流任务的细节。

[0182] 图 35 是示出了用于获得最新的和可由用户执行的 MPP 的提升处理的流程图。获得最新工作流被设计为将资源从生产流移动到并行流，或从并行流移动到另一个并行流。作为获得最新工作流的一部分，对目标环境 / 并行流 TeamSite 工作区域（其包含所有内容，包括带有版本的网口资源）和基线或“MPP 版本”进行详细比较。这是一种由中间处理自动形成的特殊类型的 TeamSite 版本。基线还包括该环境中安装的应用以及它们相关联的版本的列表。

[0183] 一旦执行了比较，结果被显示给用户（仍然在该工作流的上下文中），然后用户可以决定他们希望被同步的资源。在某些实施例中，仅向用户显示不同的资源。对于识别出的流之间的每个差异，用户可以决定忽略、覆盖、添加或删除该资源。在做出决定之后，获得最新将执行每个决定。一旦已经执行了这个操作，流被同步。即使流的内容可能不是完全相同，用户也可进行同步处理，并且做出关于合并流的知情决定。

[0184] 在步骤 3502，启动该工作流。这将结束的改变从生产流带入并行流，或将经测试的改变从一个并行流带入另一个并行流。当首次提供并行流时，使用初始化工作流，并且初始化工作流负责将生产流内的基线中的内容移动到并行流，并且创建并行流内的资源的新的基线。基线事实上是包含给定时刻的内容和资源的 TeamSite 版本。当执行并行开发中所

涉及的各种比较操作时,基线(即,版本或检查点)可被用作比较操作中的参考点。

[0185] 在步骤 3504,批准器 4128 决定是否创建并行流,以及并行流中包括什么文件。可以进行代码比较,以便决定将包括哪些改变或文件。当启动新工作流时,可以向批准器自动发送电子邮件或其它通知。在步骤 3508,系统准备代码部署,例如,退出用户、创建锁文件。在步骤 3510,开始部署,并且将代码和内容拷贝到新环境内的并行流。

[0186] 在步骤 3512,检验目标。这个步骤包括检验部署是否被正确执行,并且它解锁系统以便进行访问。在步骤 3514,如果处理中存在任何错误,可以进行恢复,以便将系统带回其以前状态,从而可以重试部署。

[0187] 为了通过图形用户界面执行获得最新工作流,用户一般退出目标环境上的部件 WPS 工作台,目标工作区域内的文件一般被保存并且被关闭,目标工作区域内一般没有其它批准的工作流在进行中,目标工作区域上一般没有其它代码部署工作流在进行中,并且用于该处理的源版本可能已经被创建。然后,用户使用目标流的工作区域的内容窗格,并且选择“新任务”。然后,用户可以选择获得最新按钮。然后工作流显示用户的 ID 和源分支,该源分支是到目标工作区域的路径。然后,用户完成其余必选字段,并且开始任务。

[0188] 然后系统更新锁文件,锁定该工作区域和工作台,向用户发送电子邮件,进行目标流的备份,并且通过创建工作区域的拷贝和初始的比较文件来执行比较预处理。然后,随同用户,通过向用户显示不同文件,执行内容和代码比较。

[0189] 图 36 是示出了初始化处理的细节的流程图。初始化工作流的目的是使得实体能够将生产流中的代码和内容同步到新提供的并行流。在已经交付了新的并行流 BDE 环境之后,它仅仅包含最少的软件、环境组件和所需的配置。这包括诸如网口管理 portlet 和页面的组件。初始化工作流将创建包含源生产环境的内容的发行包。在新环境的提供处理过程中,该工作流仅需要被执行一次。

[0190] 一旦已经提供了并行 BDE,该环境也被初始化(使用初始化工作流),并且代码和内容已经被同步。然后还可以同步与 BDE 相关联的 SDE 环境。与并行流相关联的同步的 SDE 和与生产或任意其它流同步的 SDE 没有两样。当首次交付并行流 BDE 时,与其相关联的 SDE 不包含应用的任意代码或内容。

[0191] 已经作为生产流的同步环境,或作为用于系统测试或 IT 目的的非同步环境使用的已有 SDE 可被移入该并行流,并且被使用。这将潜在地减少整体环境成本,然而,为了与并行流相关联,可以重新提供该环境。不存在可以简单地通过改变环境配置,简单地移动 SDE 或将其与另一个源 BDE 重新关联的通用方法。

[0192] 通过重新提供环境,该环境上的所有数据将丢失,并且实际上,交付的环境将与请求新环境相同。

[0193] 处理在步骤 3604 开始,其中输入用于初始化处理的信息。在步骤 3606,更新系统上的锁文件,从而不在相同时刻进行其它更新或改变。在步骤 3608,出于相同原因锁定工作区域和工作台。在步骤 3610,验证初始化处理,以便确保它可被正确执行。

[0194] 在步骤 3612,向新的并行流初始化的批准器发送电子邮件或其它通知,如果批准了初始化(步骤 3614),那么在步骤 3620,通知启动器可以继续初始化。

[0195] 如果未批准初始化,则在步骤 3616,可向启动器发送电子邮件或其它通知,以便让他们知道其请求未被批准。在步骤 3618,撤消目标上的锁。然后处理进入步骤 3648,其中

还撤消工作流锁,并在步骤 3650,更新工作流记录。

[0196] 如果批准了初始化,则在步骤 3622,进行备份以防在初始化过程中发生错误。在步骤 3624,解锁工作区域从而可以进行改变。在步骤 3626,初始化 AWPS,并且在步骤 3628,初始化工作区域。然后在步骤 3630 通知启动器工作区域也被初始化。

[0197] 在步骤 3632,检验目标,以便确保初始化被正确执行,如果不是这样,处理进入步骤 3634,以便执行恢复回以前的系统配置。在步骤 3636,解锁备份工作区域,并且在步骤 3638 恢复系统。在步骤 3640,向启动器发送电子邮件通知其恢复已被完成。然后,处理进入步骤 3646,并且执行上面描述的当未批准工作流的初始化时步骤 3618 之后的步骤。

[0198] 在步骤 3642,如果检验了目标,则部署被接受,并且工作流锁被禁止。在步骤 3644,向启动器发送通知,通知它们完成了初始化。然后,处理进入步骤 3646,执行上面所述当工作流初始化未被批准时步骤 3618 之后的步骤。

[0199] 图 37 是示出了整体部署处理的细节的流程图。在步骤 3702,开发第一个发行。在步骤 3704,该发行被部署到非同步 SDE。在步骤 3706,测试代码缺陷。如果没有缺陷,则在步骤 3708,该发行被部署到 eBDE。否则,处理回到步骤 3702,并且开发新的发行。在步骤 3710,执行站点组装,并且在步骤 3712 检查站点缺陷。如果未发现缺陷,则在步骤 3714,该发行被部署到一个环境以便进一步测试。同样,如果没有发现缺陷,该发行被在步骤 3718 移动到预生产环境。最终,如果在步骤 3720 未发现缺陷,该发行被在步骤 3722 部署到生产以便由用户使用。缺陷测试在步骤 3724 继续。

[0200] 在执行缺陷检测的任意步骤(步骤 3712、3716、3720 和 3724),在检测到一个或多个缺陷之后,处理回到步骤 3728,其中可以修复代码并且开发新发行以便部署。

[0201] 图 38 是示出了提升处理的细节的流程图。提升处理实际上与获得最新工作流相同,然而,它被特别设计为将资源从并行流移动到生产流。由于生产流是用于将资源部署到预生产和生产环境的唯一环境,存在将资源合并到生产流中的潜在的较高风险。例如,如果需要回滚,则在回滚期间生产流不可用,并且用户将不能执行到生产环境中的任何发行。

[0202] 用于提升处理的回滚处理类似于用于获得最新工作流的回滚处理。然而,由于生产流的架构的重要性,提升工作流中的备份和恢复处理略微不同,因为生产流是可被提升到预生产或生产环境的环境,重点必须放在确保该生产流的内容的完整性。生产流还可以具有灾难恢复环境。这是驻留在不同数据中心的同步的环境,并且如果主生产环境存在严重错误,其允许无缝的故障恢复。这需要生产流和灾难恢复环境之间的实时同步。

[0203] 当执行生产环境的备份时,该处理可以与执行获得最新操作时的相同,然而,当执行以前备份的恢复时,灾难恢复环境必须被同步。

[0204] 该处理在步骤 3802 处开始,其中正在开发第二个流。在步骤 3804,第二发行被部署到非同步 SDE。如果在步骤 3806 检测到缺陷,则处理回到步骤 3802 以便开发修改后的发行。如果未发现缺陷,则在步骤 3808,初始化第二个流。在步骤 3810,基于第一个流初始化第二个流。在步骤 3812,部署第二个流。在步骤 3814,执行站点组装。在步骤 3816,对第二个流进行缺陷测试。如果未发现缺陷,则在步骤 3818,对第一个流进行更新测试,并且如果发现了更新,它们被在步骤 3820 结合到第二个流,此后再次执行站点组装。

[0205] 如果在步骤 3822 发现了更新,则第二个流被发行到 UAT 以便进行测试。在步骤 3824,测试第二个流,并且然后如果未发现缺陷,在步骤 3826 将其部署到 eBDE。在步骤

3828, 该发行被部署到 UAT, 并且在步骤 3830 对其再次进行缺陷测试。如果未发现缺陷, 则发行第二个流以便部署到预生产环境。在步骤 3838, 再次对第二个流进行缺陷测试, 并且如果未发现缺陷, 该发行被在步骤 3840 部署到生产环境。在步骤 3842, 可以对发行的版本进行附加测试。

[0206] 如果在步骤 3830、3838 或 3842 中的任意一个检测到缺陷, 处理进入步骤 3832, 其中存在到第一个流的回滚。在步骤 3830, 可以修复代码。在步骤 3824, 当检测到缺陷时, 处理直接进入步骤 3834, 而不执行回滚。当执行操作, 例如, 获得最新时存在这样的固有风险, 即, 在并行流的合并过程中做出的决定将引起一个或所有并行流失效。例如, 它将引发与其它代码或内容的冲突。如果在该环境中不存在相关性或先决条件, 这更可能发生。如果发生了这种情况, 那么可以将该环境回滚到执行该操作之前的状态。由于每次执行一个操作, 例如, 获得最新工作流, 可以进行回滚, 所以可以进行并行流的完整备份。操作的最后阶段允许用户检验该环境如预期那样运转。如果不是, 用户可以执行回滚。

[0207] 图 39 是示出了部署处理和随时间的信息流动的系统图。示出了开发第一个流 3902 的步骤。这些步骤包括从开发环境 3914 移动到 eBDE 3906 环境, 移动到测试环境 3916, 并且然后移动到预生产环境 3918。最后, 第一个流 3902 从预生产环境移动到生产环境 3910。

[0208] 当开发并行流时, 在开发第一个流并且将其移动到 eBDE 环境 3906 之后, 其可被用于形成第二个流 4064。如这个图中所示, 在开发环境 3920 中开发第二个流 3904, 此后它可在 eBDE 环境 3922 中与第一个流 3902 组合。新组合的第二个流然后可被移动到测试环境 3924, 此后其可被移动到第二个 eBDE 环境 3926, 第二个 eBDE 环境 3926 处于到生产环境 3910 的路径上。从 eBDE 环境 3926, 在移动到生产环境 3910 之前, 可以在测试环境 3928 和预生产环境 3912 中进一步测试第二个流。

[0209] 图 40 是示出了获得最新处理和随时间在三个并行流之间的流动的系统图。流 1 4002 被在 T1 4208 处发行。在时间 T2 4010, 可以执行获得最新以便创建第二个流 4004。从这个第二个流, 可以在 T3 4012 以相同方式创建第三个流 4006。

[0210] 然后在被发行到生产之前, 这些并行流可被合并。图 40 示出在 T4 4014, 第二个流 4004 被合并到第一个流 4002。在 T5 4014, 并行流 34006 被合并到合并后的流 1 和流 2, 创建单个合并流。

[0211] 图 41 是示出了可以如何配置不同安全组的 MPP 的用户界面。该 MPP 界面包括包含各种管理功能的管理组件 4102。该图示出了可以如何创建批准组。该界面包括活动的文本字段 4110 和测试字段 4108, 其中可以输入信息。该界面包括用于添加批准组的“添加”按钮 4104, 以及用于重新命名批准组的“重新命名”按钮 4106。

[0212] 存在三种不同类型的用户访问。对于 TeamSite 用户, 他们被以 UNIX 组和 TeamSite 内的用户管理控制台控制, 这些用户仅能访问 TeamSite 6.5 Content Center。TeamSite 内用户账户的创建不提供对整个 BDE 的访问。对于网口服务器用户, 这些用户被在 LDAP 目录内创建, 并且被 BDE 用户用于执行管理任务, 诸如管理站点, 即, 使用 BDE 功能。对于 ID&V 网口用户, 这些用户是基本网口注册用户账户, 他们具有将他们标识为“顾客”的 LDAP 条目内配置的附加属性。这些用户一般被本地实体用于测试目的, 以便查看对于模拟的前端“顾客”, 环境如何运转。还可以提供可替换的用户访问类型。

[0213] 图 42 是示出了获得最新处理的配置的 MPP 的用户界面。该管理界面示出了各种用户 4202-4212, 以及他们是否是贡献者的状态, 并且每个用户具有其所属的相应批准组。

[0214] 图 43 是示出了系统的总体设计的 MPP 的用户界面。该系统被示出为具有 web 界面, 虽然可以使用其它界面。该界面包括 workflow 任务窗格 4302 和管理任务窗格 4306。workflow 任务包括新任务按钮 4312, 以及示出进行中任务的窗口 4310。该显示包括任务名称 4322、描述 4312、请求者 4314、所有者 4316、到期日期 4318 和需要进行的动作 4320。从这个界面, 用户可以控制创建和部署并行流的整个 workflow。

[0215] 图 44 是示出了代码部署 workflow 的开始的 MPP 的用户界面。通过这个界面, 创建新任务的用户可以选择实用工具任务 4402、简单实用工具任务 4404 或代码部署 workflow 4406。在选择了任务类型之后, 用户可以选择下一个 4408, 或使用按钮 4410 关闭处理。

[0216] 图 45 是示出了用于部署处理的配置设置的 MPP 的用户界面。用户可以输入改变请求权证号 4506 (当部署到生产环境时)、网站实体的类型 4508、目标环境的类型 4510 (例如, 生产、预生产、测试)、以及他们是否希望配置标记处理 4512。该界面还允许用户检查工作流配置 4514, 以便修改 workflow 配置 4516、查看部署状态页面 4518、以及查看 TeamSite workflow 流程图 4520。按钮 4502 和 4522 可用于开始任务, 而按钮 4504 和 4524 可用于取消创建处理。通过选择开始任务, 将以输入的配置创建代码部署 workflow 任务。

[0217] 图 46 是示出了获得最新处理的一部分的 MPP 的用户界面。该界面示出了两个任务 4602 和 4604, 其中如以动作栏 4606 所示, 已经完成了获得最新处理。窗口 4608 示出了例如 workflow 记录中的获得最新处理的结果的细节。该界面还示出使用开始输入任务按钮 4610, 用户可以开始处理的下一个部分, 该部分是代码部署处理。

[0218] 图 47 是示出了部署处理的一部分的 MPP 的用户界面。该界面用于向目标环境部署代码。该界面包括关于当前并行流 4702 的信息。它还包括描述将被执行的下一个处理的描述 4704。该界面允许用户选择 MKS 项目 4706、建立权证号 4708 以及配置标记 4710。它还显示目标环境 4712 (例如, 生产) 和目标部署服务器 4714。该界面包括用于提交部署任务的提交按钮 4716。

[0219] 图 48 是示出了部署处理的结果的 MPP 的用户界面。该界面示出了关于当前并行流 4802 的信息, 以及关于代码部署处理的结果的信息 4804。例如, 这可以包括来自记录的信息, 诸如该处理是否成功。该界面还示出关于任务 4806 的概要信息, 包括, 任务名称、处理名称、完成状态和下一个动作 4808。

[0220] 图 49 是示出了将代码部署到生产的 MPP 的用户界面。该界面示出了说明已经发送了关于代码部署的通知的输出 4902。它还提供按钮 4904, 从而在测试之后用户可以将代码部署到生产环境。

[0221] 图 50 是示出了将代码部署到生产的 MPP 的用户界面。该界面允许用户以类似于其它界面的方式将代码部署到生产环境。

[0222] 图 51 是示出了部署处理的结果的 MPP 的用户界面。该界面类似于图 47, 并且允许用户输入必需的配置信息, 以便将并行流部署到生产环境。它还示出了关于当前部署处理和下一个任务的信息。其包括在部署过程中启动和停止 web 服务器的选项 5102。

[0223] 图 52 是示出了结果记录的 MPP 的用户界面。该界面类似于图 48, 并且示出了部署到生产处理 5201 的结果。它还示出了关于到生产的部署和下一个步骤 5202 的概要信息。

[0224] 还可以进行对 MPP 系统的操作的报告。例如,产生用户可以观看的审计历史。以这种方式,用户可以检查在过去执行的任何获得最新或提升操作期间做出的决定。用户可以确定环境的当前基线,其包括部署到环境的所有当前应用、相关项目修订和当前的内容基线的列表。

[0225] TeamSite 可被设置为自动记录向其提交资源的工作流。记录的重要信息包括在工作流期间执行的任务、谁执行该任务、谁批准该任务和该任务何时被执行。不论操作是否成功,可以使用通常的工作流记录组件记录 MPP 任务和操作中的信息。记录还可以包含诸如历史文件位置、任务名称、任务 ID 和日期 / 时间的信息。

[0226] 图 53 是示出了可以如何比较项目的生产和并行版本的 MPP 的用户界面。该界面示出了已经开发的生产版本 5302 和并行版本 5304。示出了生产版本 5302 的页面 5306、内容 5322 和应用 5310。还示出了并行流 5304 的相应新版本。

[0227] 用于管理流之间的发行的工作流本质上是复杂的处理,用户通过分析流之间的不同以及做出继续工作流的决定提供帮助。在比较处理期间的任意时刻,用户可以保存其决定和 / 或关闭比较差异的页面。当工作于大量差异时,这是有用的。用户可以选择他们希望检查的制品的类型的标签。这显示自从发生最后比较以来的制品差异的列表。该表标识出制品类型和当前做出的决定的数目(列出的差异的总数中的)。单选按钮(增加、删除、忽略、覆盖)仅对该值是有效选择的制品有效。

[0228] 差异被以颜色编码,标识出自从最后比较发生以来已经针对制品发生的不同动作。用户可以将其光标放置在颜色上,以便获得关于每种颜色表示什么的提示。用户选择制品,以便为该制品开启比较细节页面。用户还可以选择观看文件源差异,以便并排观看两个制品之间的代码差异。

[0229] 然后用户对每个项目做出其选择。这可以使用设置动作值完成,该值适用于一组制品或单个选择。重要的是与对其相应组件的决定一前一后地做出 URL 映像决定。对于 CAM 级别和代码标签,用户可以点击是或否。如果环境之间的差异仅是 CAM 级别差异,则不推荐代码部署。在这种情况下,用户可以手动更新 CAM 级别。在结束一个标签上的项目的检查之后,用户可以简单地点击另一个标签以便选择它。最终,一旦用户完成了其检查,他们可以点击提交选择按钮。

[0230] 用户可以为删除的成分进行这种相同的处理。用户可以检查删除的文件,并且决定对其做什么。用户可以针对相同动作将文件包在一起。当用户完成其检查之后,他们可以点击提交选择按钮。

[0231] 使用包括添加 5312、删除 5314、忽略 5316、合并 5318 和覆盖按钮 5320 的按钮列,用户可以将生产和并行流合并到新版本内。在某些情况下,用户可能希望增加文件,而在其它情况下,他们可能希望忽略并行流中的改变,并且保持生产版本。

[0232] 图 54 是示出了如何比较 web 页面布局的生产和并行版本的 MPP 的用户界面。该界面帮助用户在生产版本 5402 和并行版本 5404 之间比较内容的页面布局。该界面示出了页面标题 5406、概要文本 5408、关键字 5410 和事务线索 5412 之间的差异。

[0233] 图 55 是示出了如何比较 web 页面的生产和并行版本的 MPP 的用户界面。该界面示出了页面的生产版本(例如, home.html)和并行流 5504 中的页面的版本(例如, home.html 的更新版本)。该预览还可示出包括在页面内的其它内容。该预览允许用户在合并并

行流时更容易地比较内容文件。用户可以使用标签 5506 和 5508 容易地在预览视图和合并视图之间切换。

[0234] 图 56 是示出了比较和合并处理的概要的 MPP 的用户界面。该界面示出了流 5602 的生产版本、该流的并行版本、以及如何合并每个文件以及合并到哪个版本的概述 5606。以这种方式,用户可以容易地检查合并处理,并且如果希望,进行改变。概要包括页面 4980、内容 5610 和应用 5612。

[0235] 图 57 是示出了如何比较代码的生产和并行版本的 MPP 的用户界面。该界面示出源代码文件 5702 的生产版本、并行流内的源代码文件 5704 的版本。这允许用户在决定如何合并文件时容易地比较源代码的两个版本。可以使用突出显示和其它视觉指示使得差异更明显。

[0236] 可以使用标准 web 技术诸如 HTML、JAVA 和 JavaScript,以及用于部署和管理文件的脚本语言和用于在开发过程中使用的提供系统实现本发明的实施例。例如,可选择地可以使用来自 RedHat 的 Kickstart 程序。另外,可以使用已知的标准数据库系统,例如, DB2 存储内容和文件。

[0237] 在审阅本公开之后,本领域的技术人员将领会和想到上面给出的思想的其它实施例、扩展和修改。因此,本发明的范围在各个方面不受上面给出的例子和实施例的限制。本发明的各个方面和其整体应当被认为允许在本公开的范围内的这些设计修改和将来的发展。仅以下面的权利要求限定本发明。

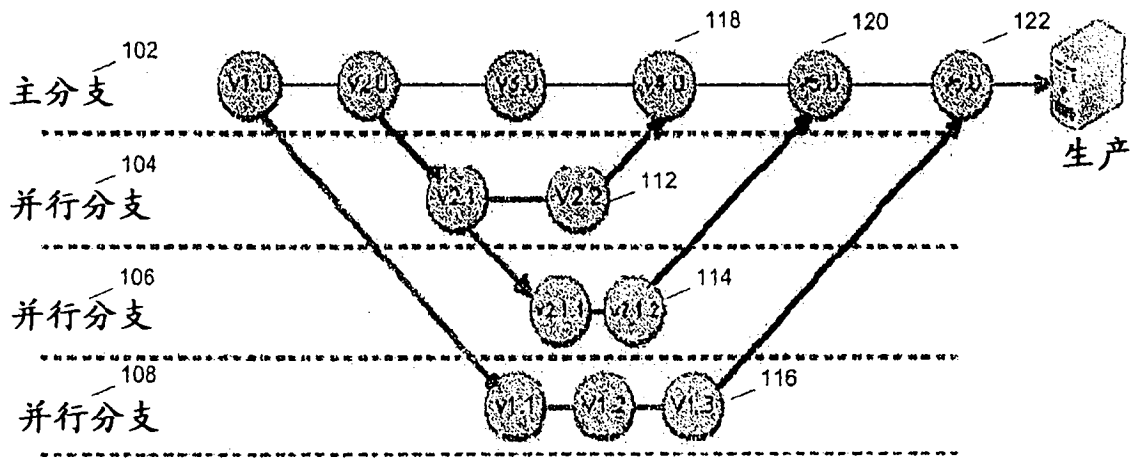


图 1

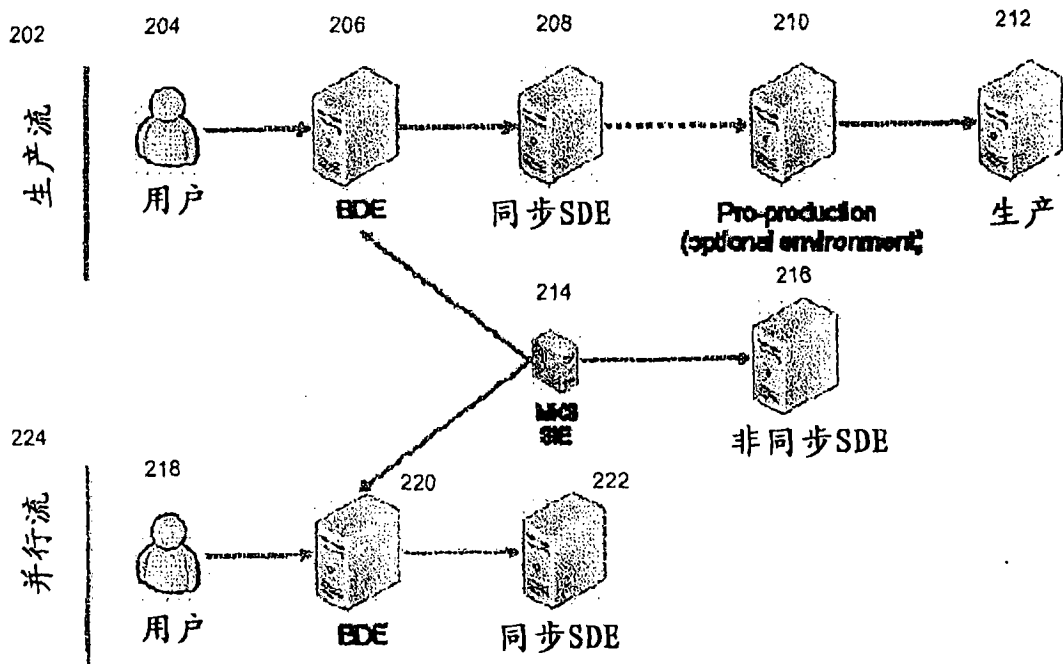


图4: 多个流开发中使用的环境

图 2

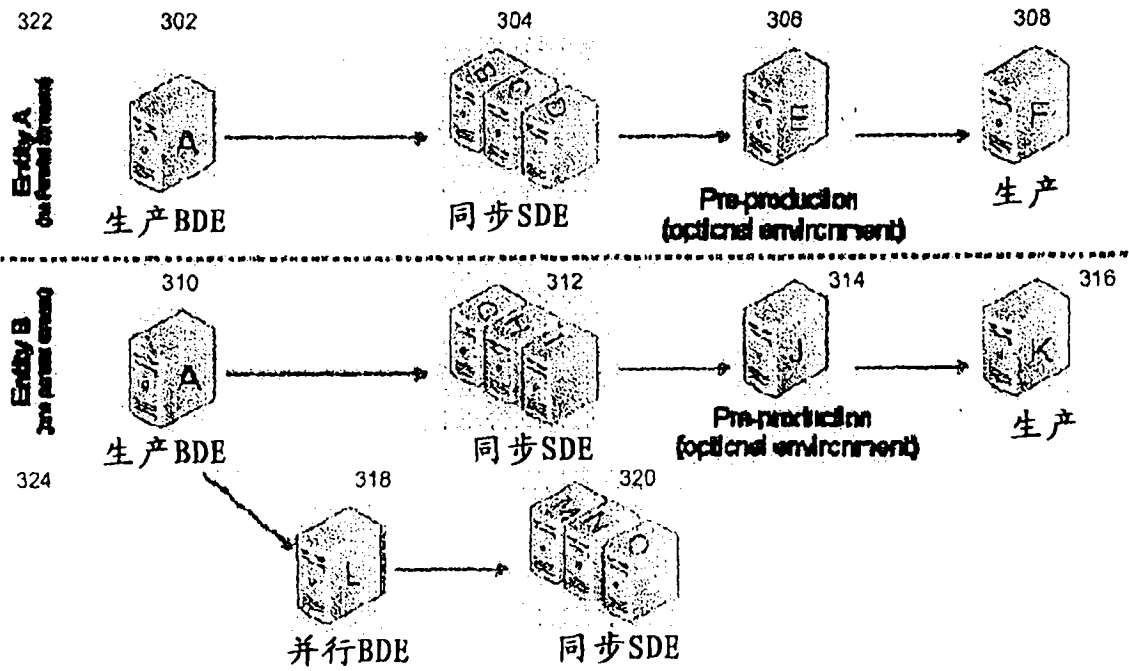


图11: 流之间网口GUPID的唯一性

图 3

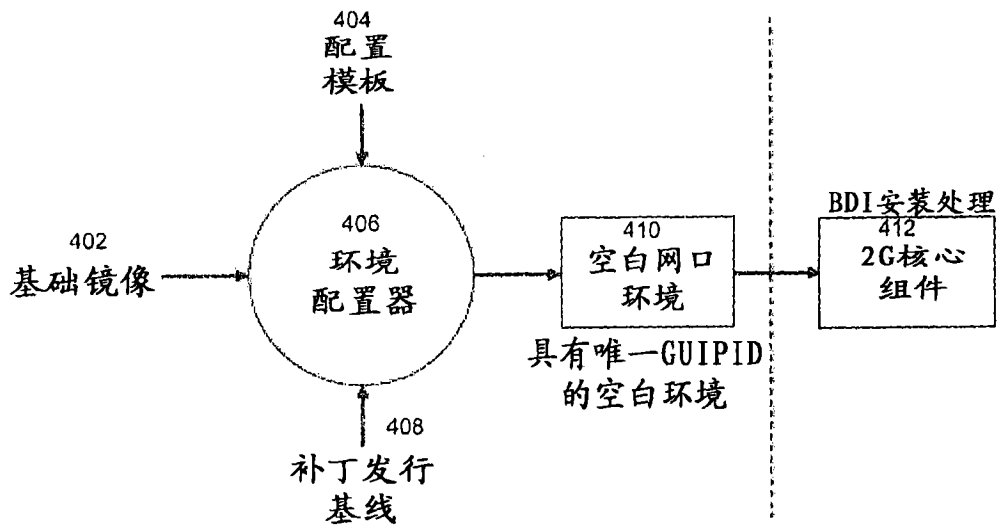


图 4

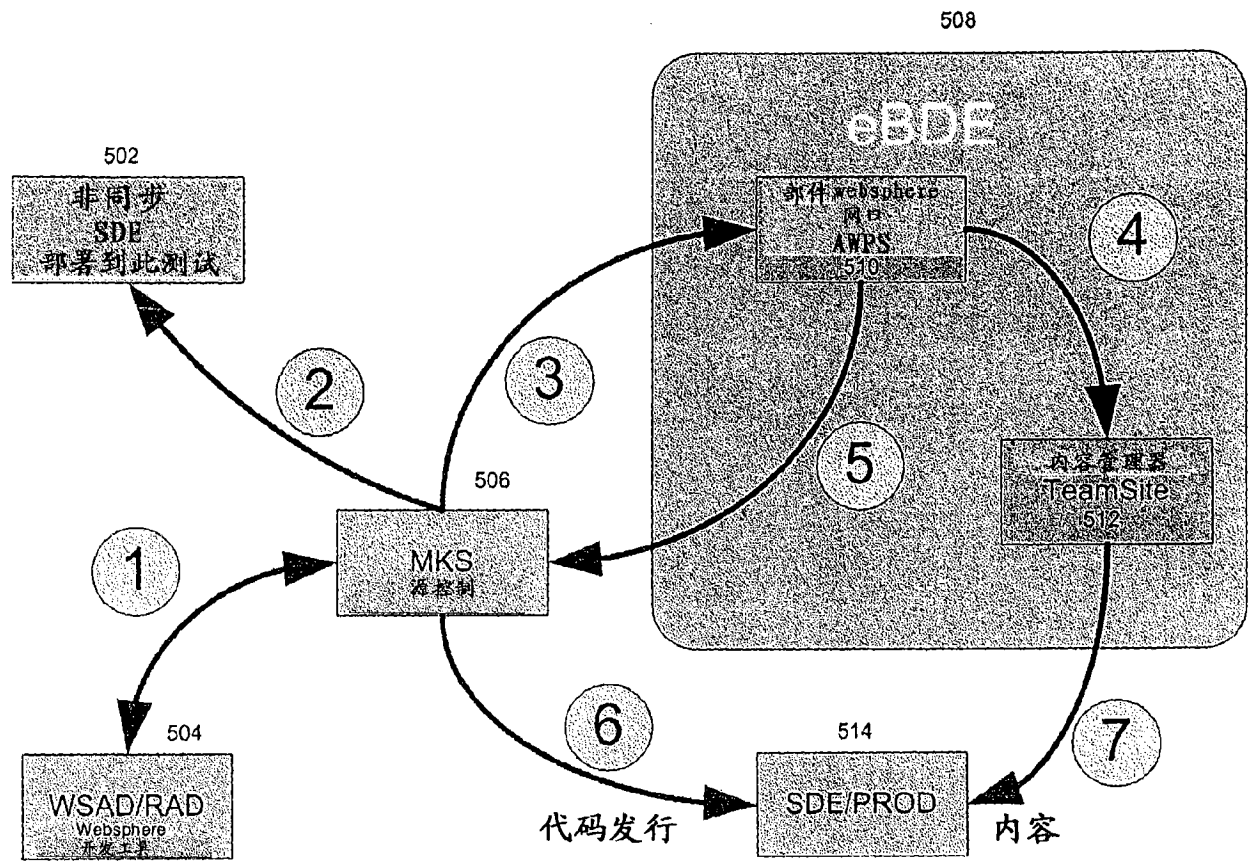


图 5

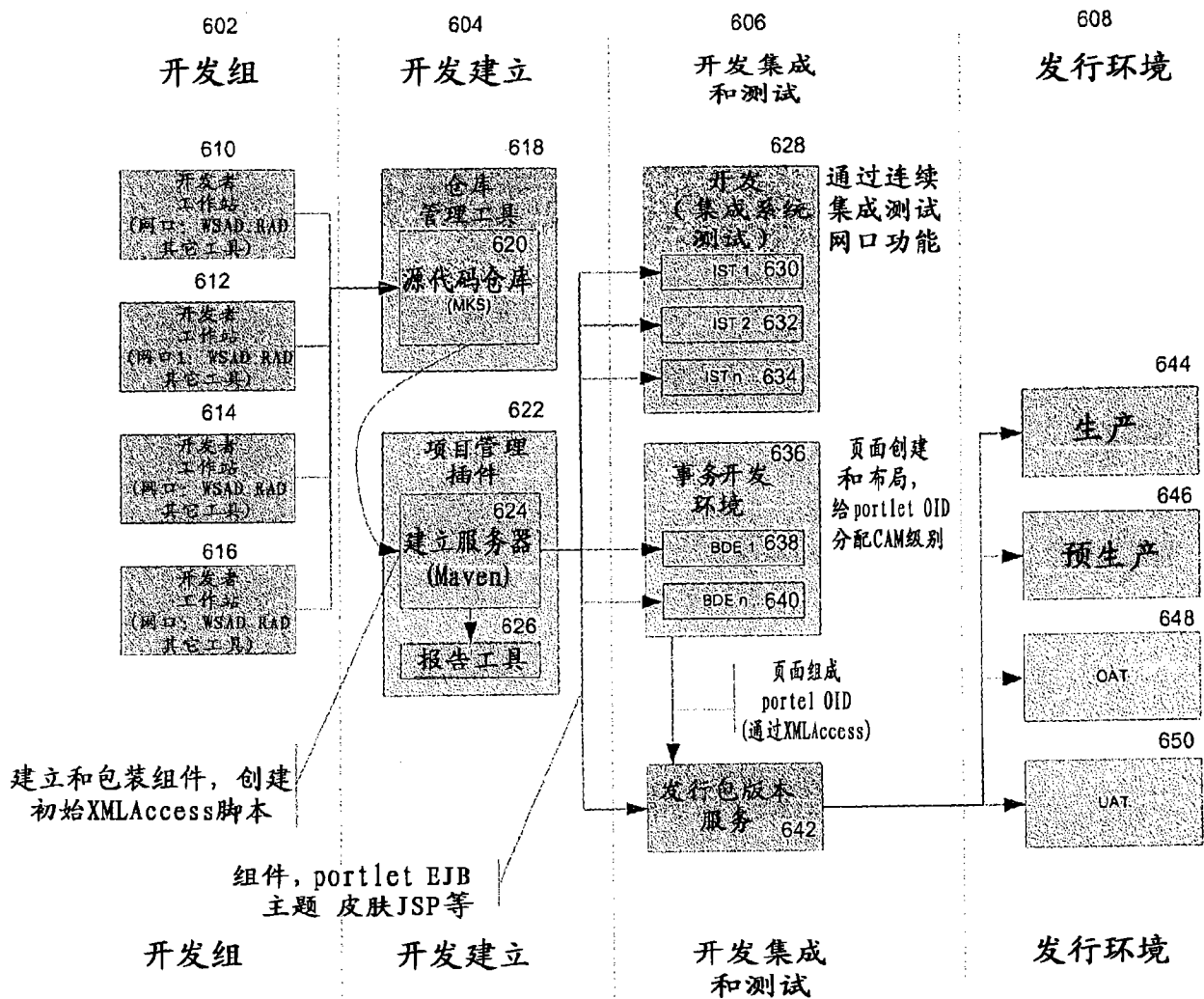


图 6

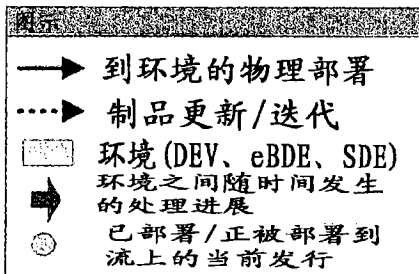
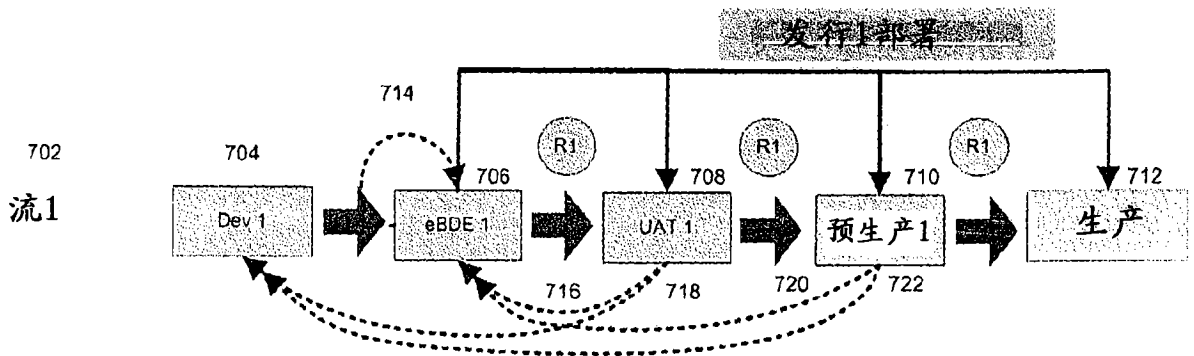
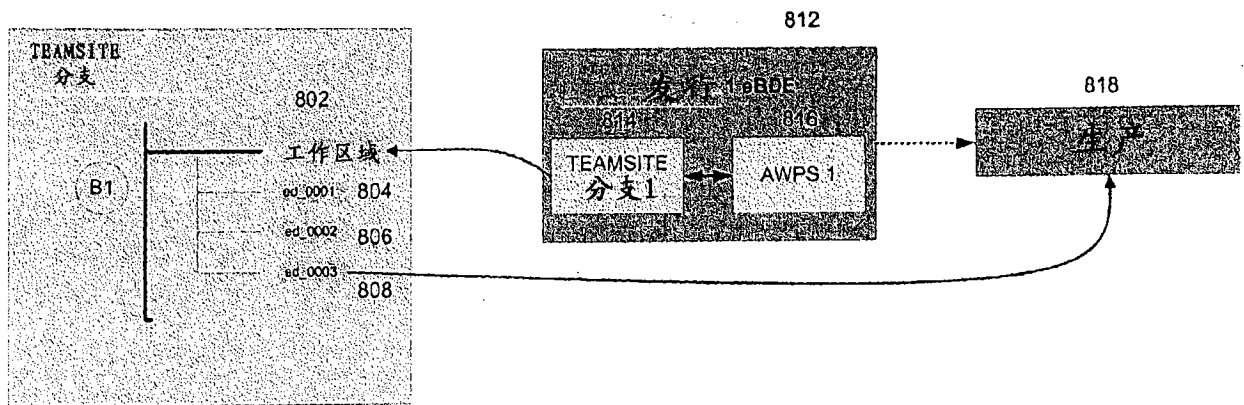


图 7



注意：
 默认，根据下列约定标记teamsite “版本”：
 ed-**<Edition#>**，其中**<Edition#>**是从0001开始的4位整数

图 8

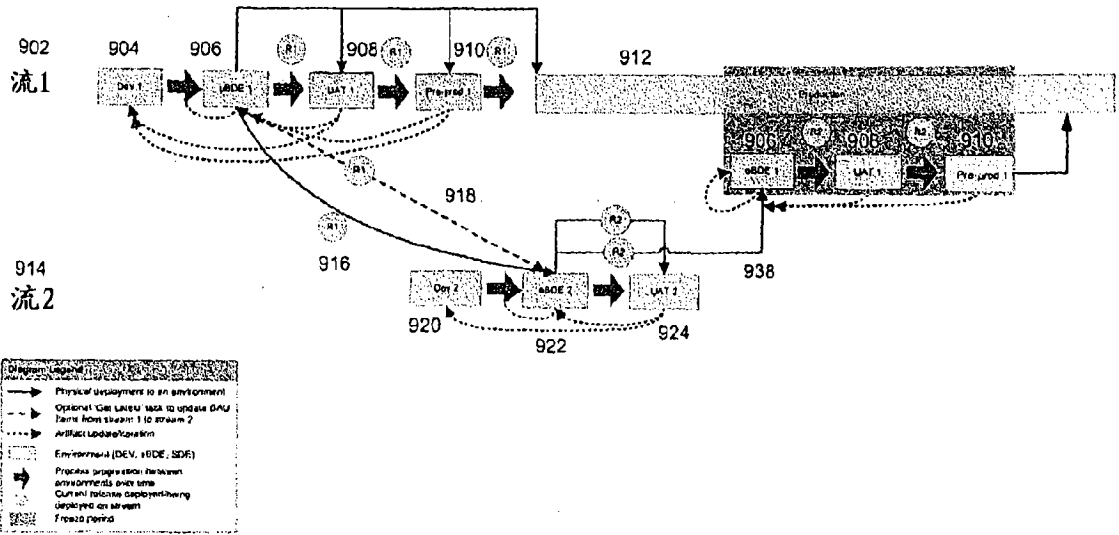
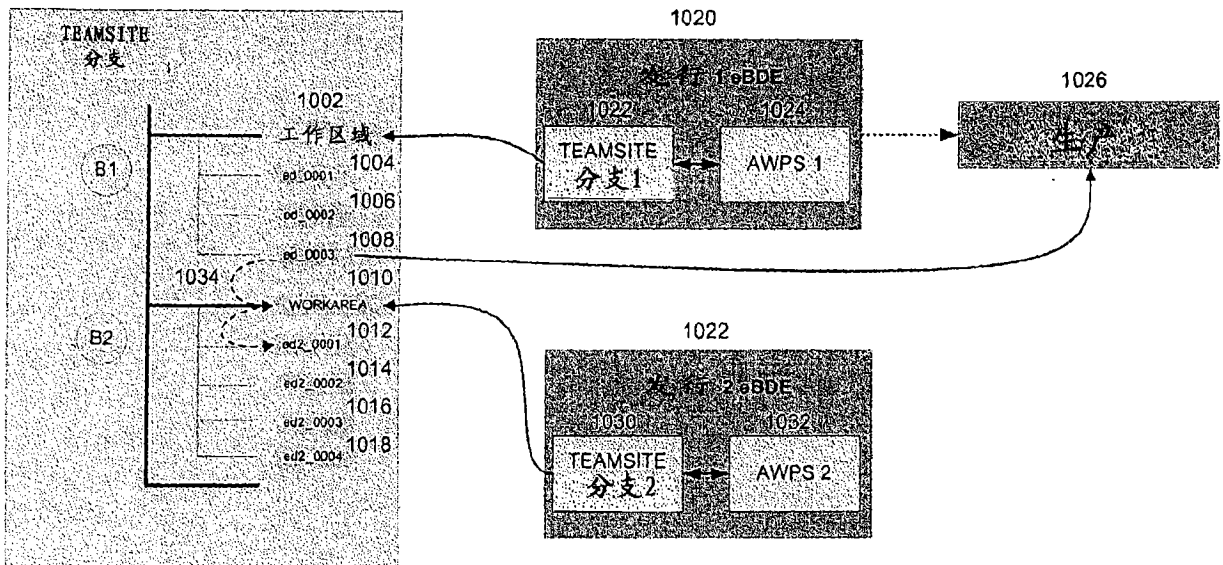


图 9



注意:

- 1 默认, 根据下列约定标记teamsite “版本” ed_<Edition>, 其中<Edition>是从0001开始的4位整数。
- 2 为了清楚起见, 根据下列约定标记分支B2内的Teamsite “版本”: ed2_<Edition>, 其中<Edition>是从001开始的4位整数

图 10

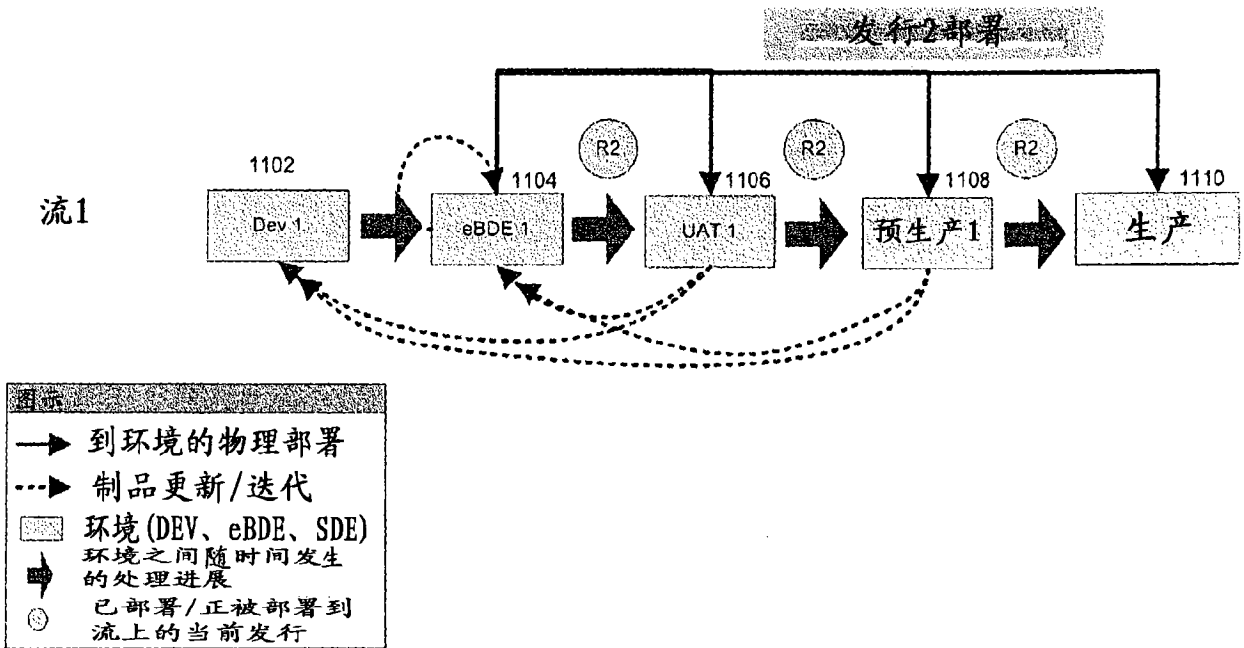
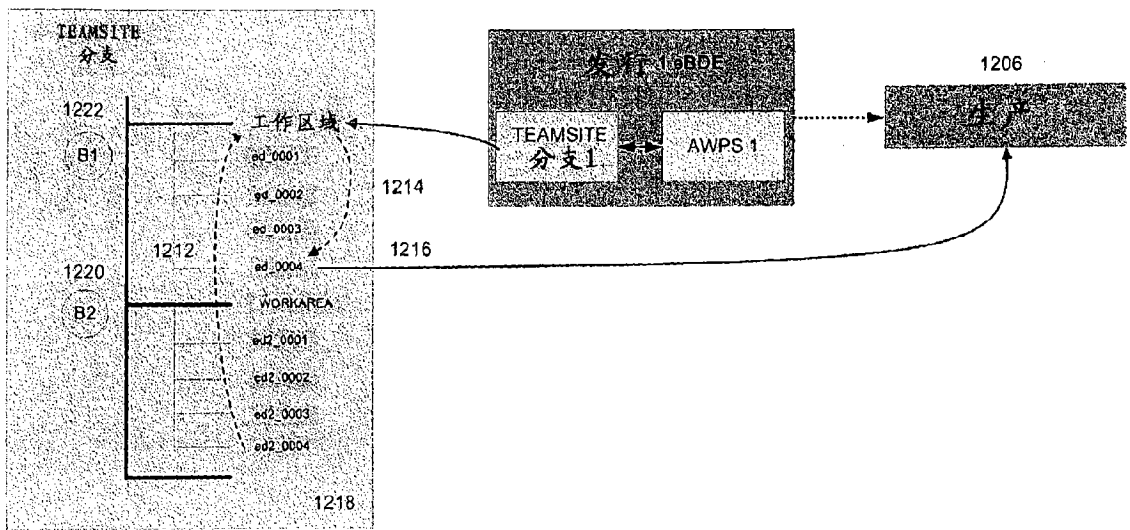


图 11



注意:

- 1 默认, 根据下列约定标记teamsite “版本” :
ed_<Edition>, 其中<Edition>是从0001开始的4位整数.
- 2 为了清楚起见, 根据下列约定标记分支B2内的Teamsite “版本” :
ed2_<Edition>, 其中<Edition>是从001开始的4位整数

图 12

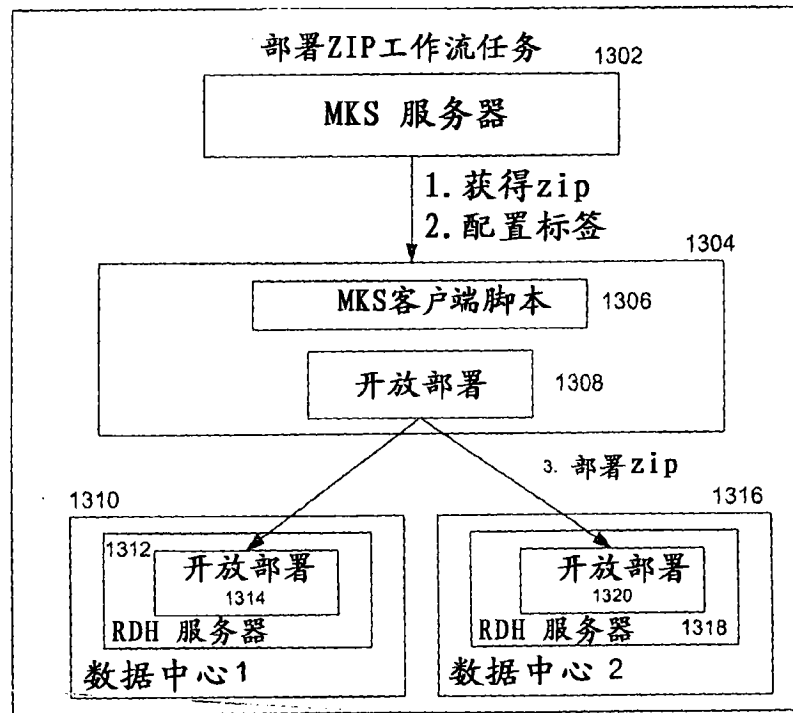


图 13

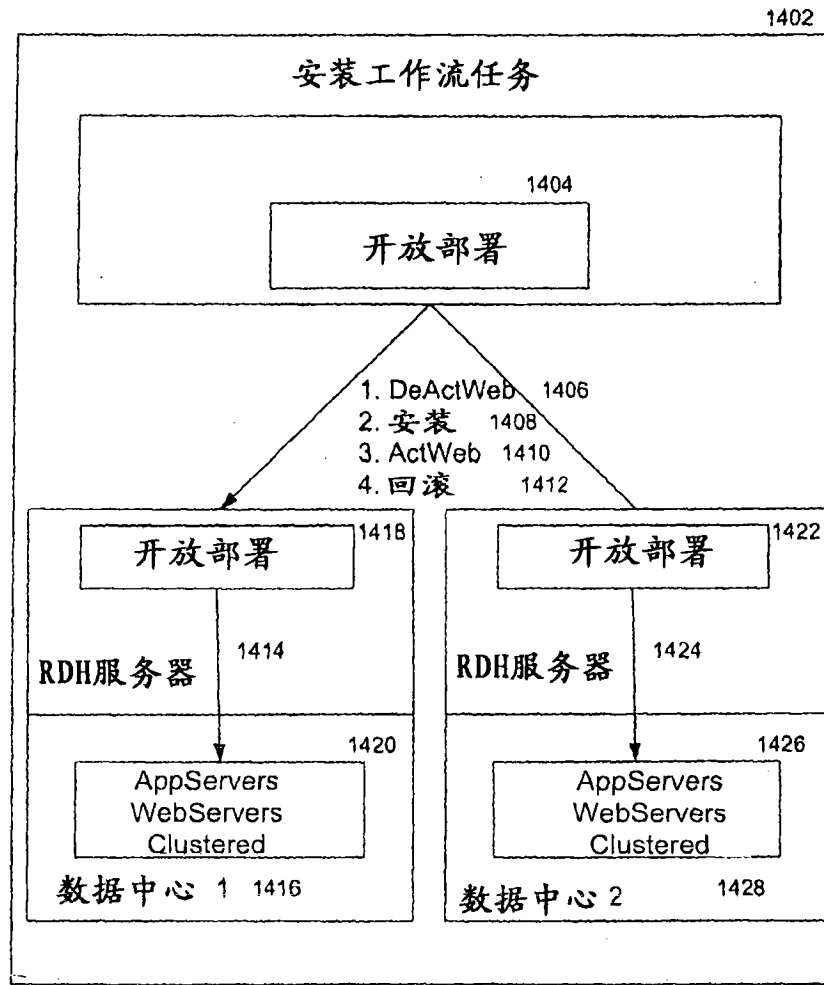


图 14

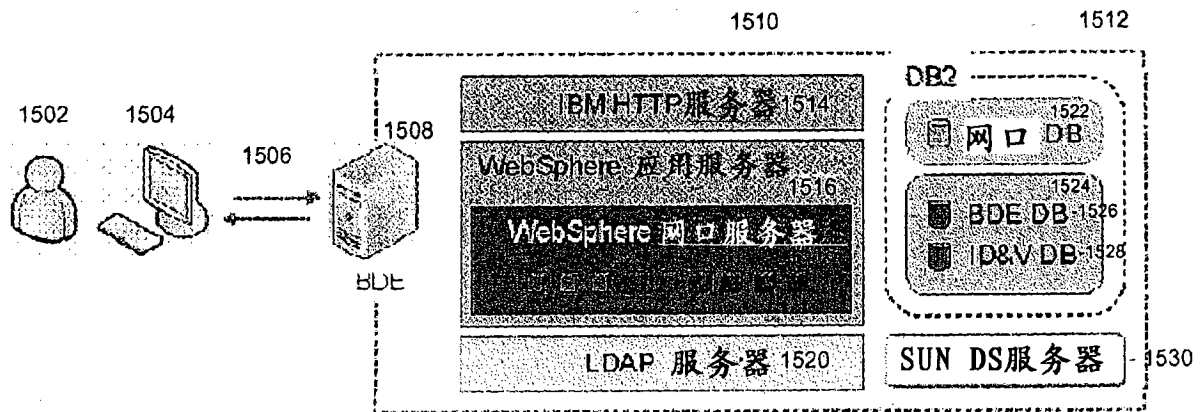
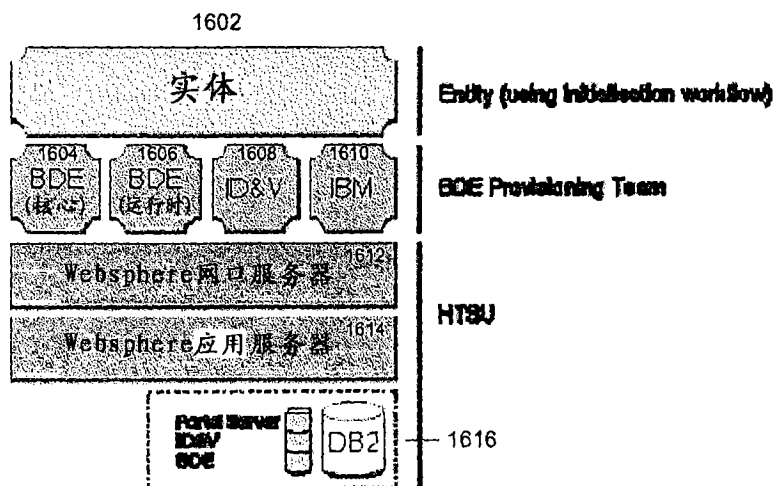


图5: 生产流和已有 (非MPP) BDE服务器环境构架

图 15



初始化并行环境所需的高层组件和所有者

图 16

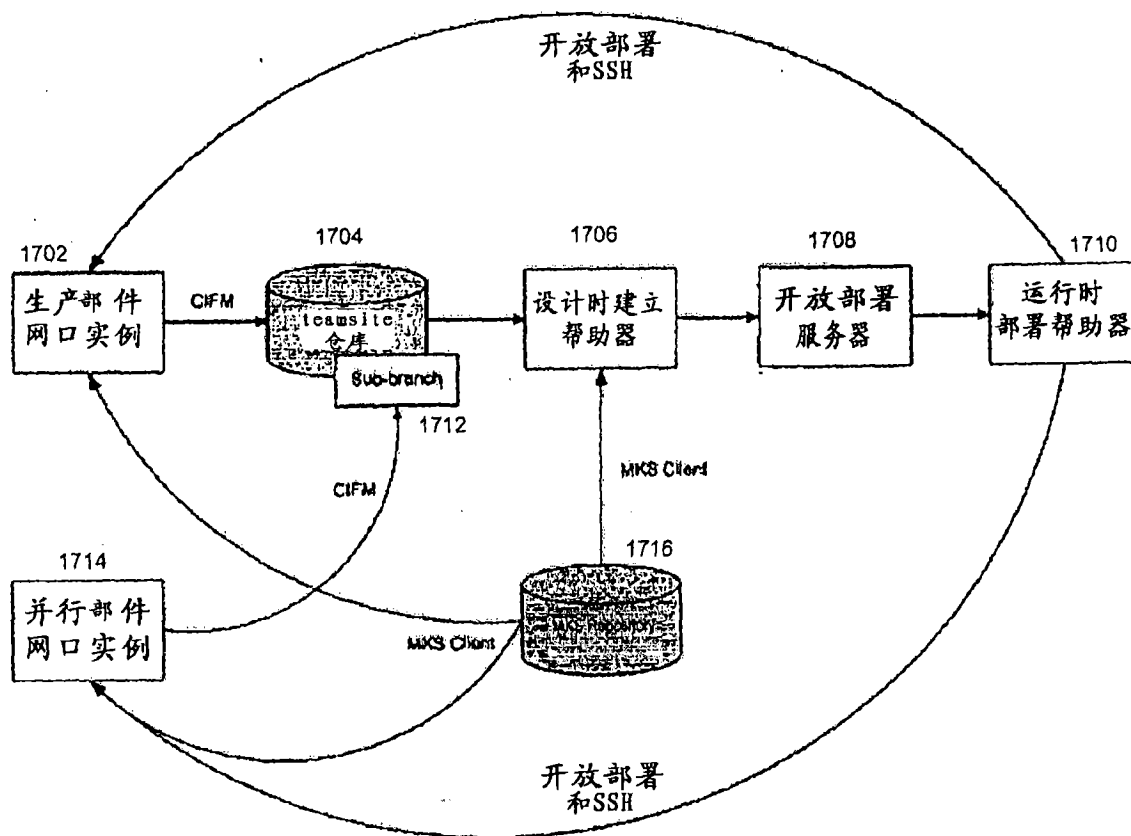


图 17

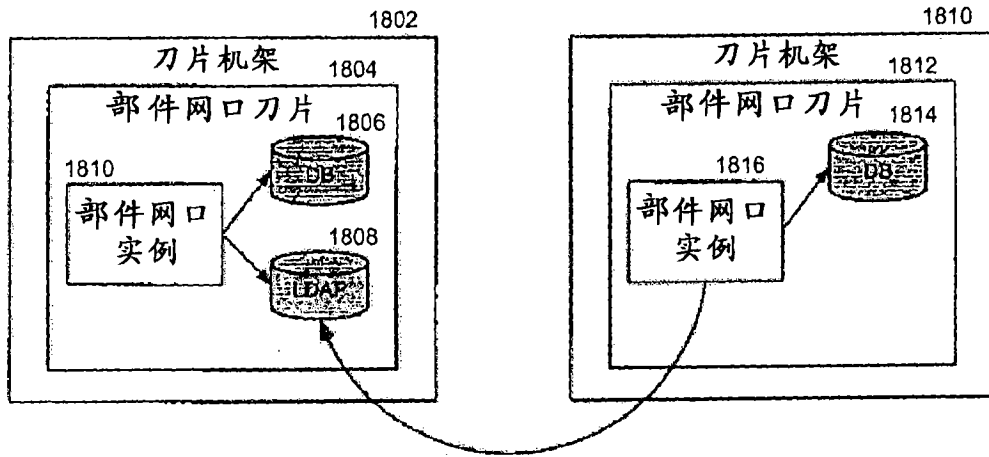


图 18

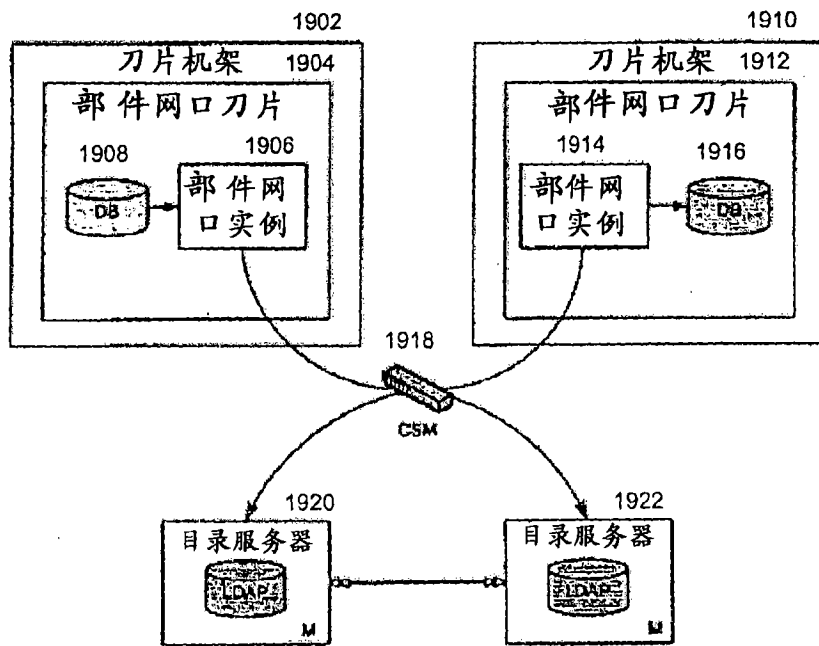


图 19

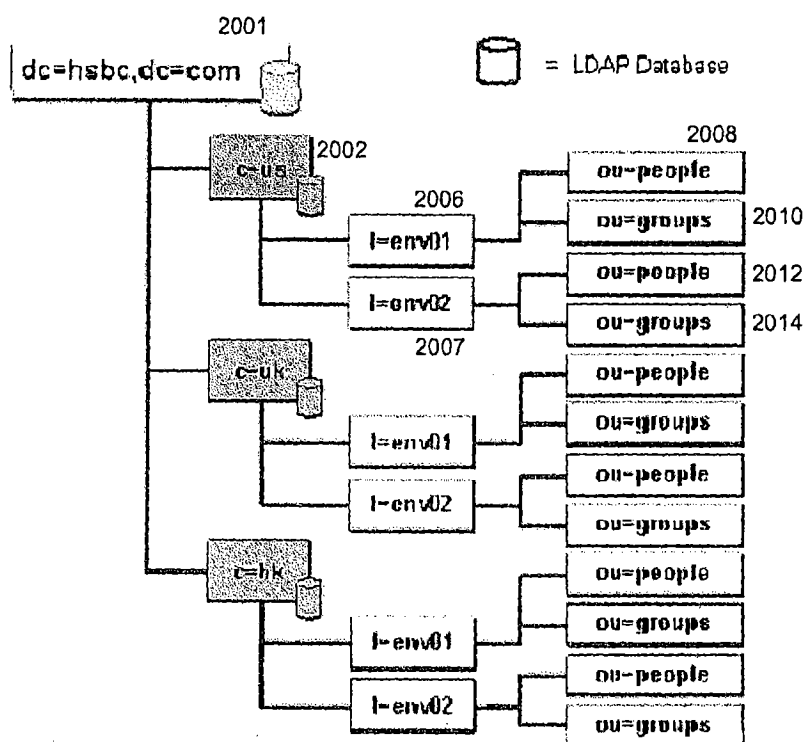


图18: SDE LDAP目录信息树 (DIT)

图 20

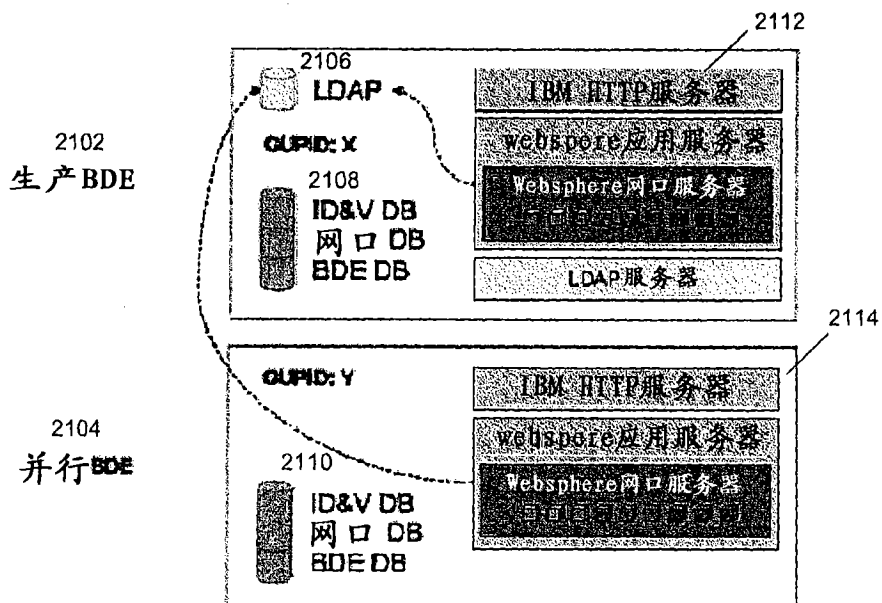


图19: 用于并行流的MPP共享的LDAP基础构造

图 21

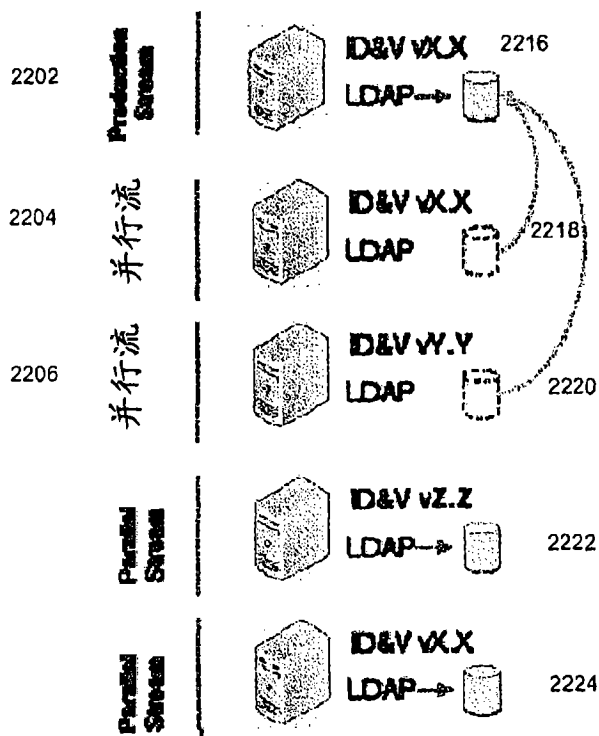


图20: 支持流之间的ID&V的多个版本

图 22

Date	Task	User	Details
Wed Dec 20 20:51:24 2006	90879	ivui	Update Lock File completed successfully.
Wed Dec 20 20:51:25 2006	90882	cmk08	Lock Workarea and Workbench completed successfully.
Wed Dec 20 20:51:27 2006	90883	ivui	Completed email notification.
Wed Dec 20 20:51:29 2006	90884	wasadm	Error Occured while trying to execute action. Failure occured, transitioning to exception handling task.
Wed Dec 20 20:51:31 2006	90910	ivui	Exception Handler task set transition to retry: Take Backup
Wed Dec 20 20:51:33 2006	90911	caced833	Completed email notification.
Wed Dec 20 20:52:34 2006	90912	ivui	End Job: 90878 transition: 0
Wed Dec 20 20:52:36 2006	90913	caced833	Completed email notification.
Wed Dec 20 20:52:38 2006	90914	cmk08	Unlock Target completed successfully.
Wed Dec 20 20:52:39 2006	90908	ivui	Unlock Workflow Lock completed successfully.
Wed Dec 20 20:52:41 2006	90909	ivui	Completed email notification.

图22: 比较历史portlet的工作流历史视图

图 23

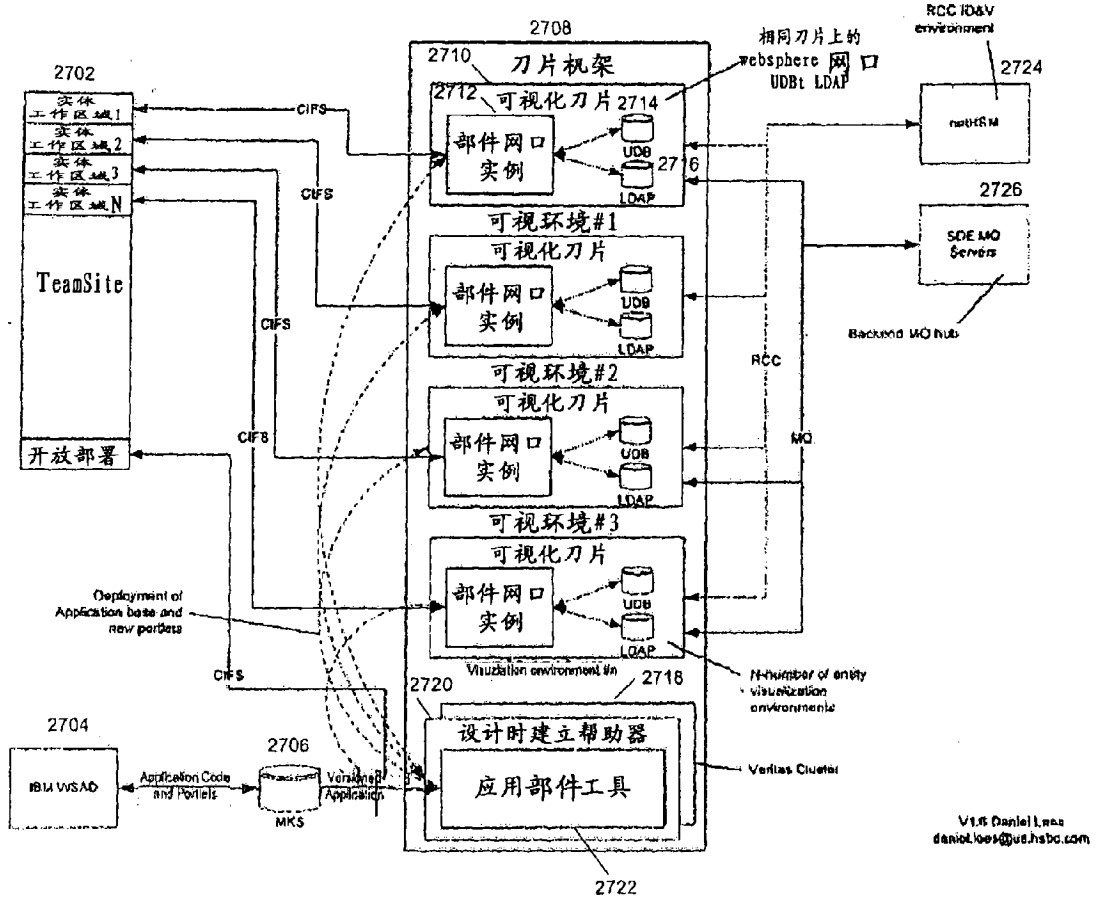


图 27

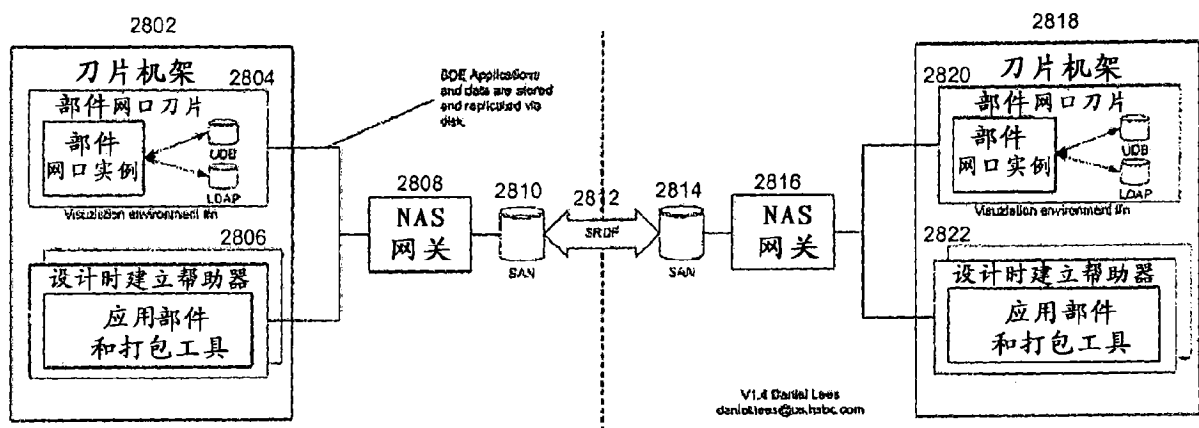


图 5: 虚拟刀片灾难恢复

图 28

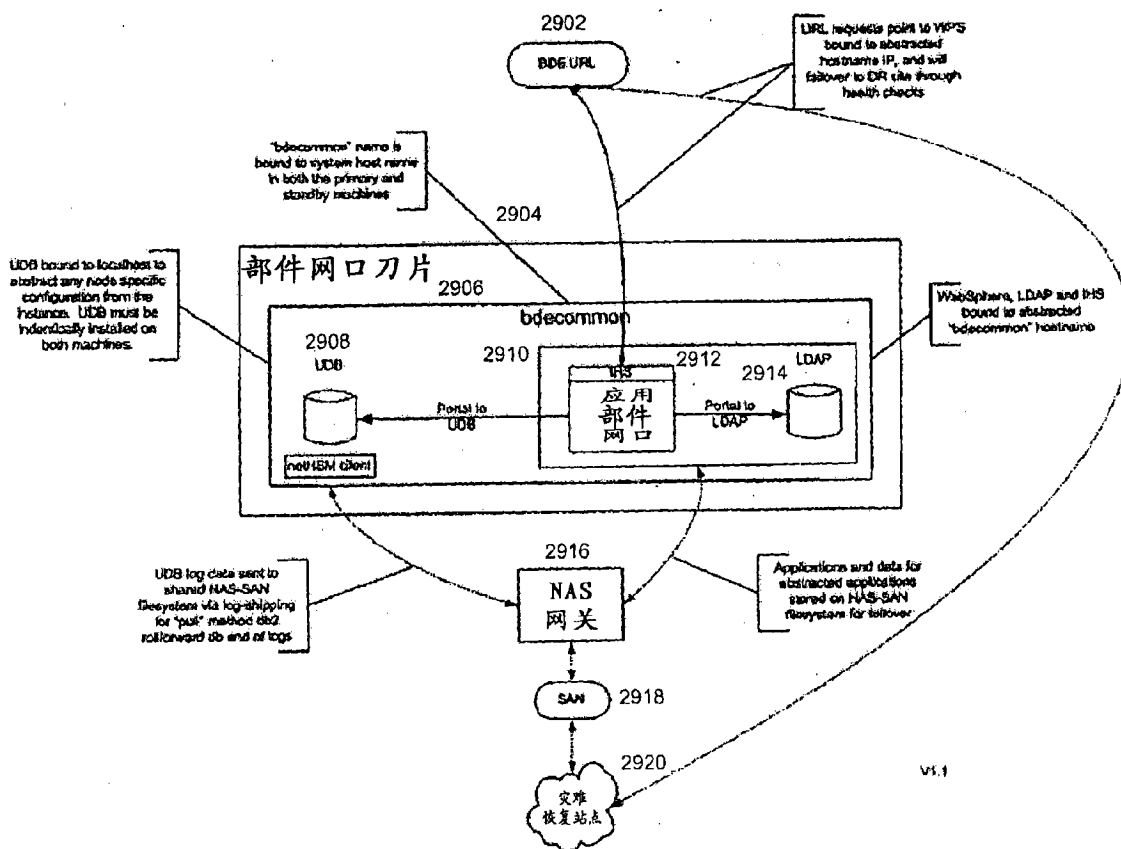


图6: 部件网口故障恢复设计

图 29

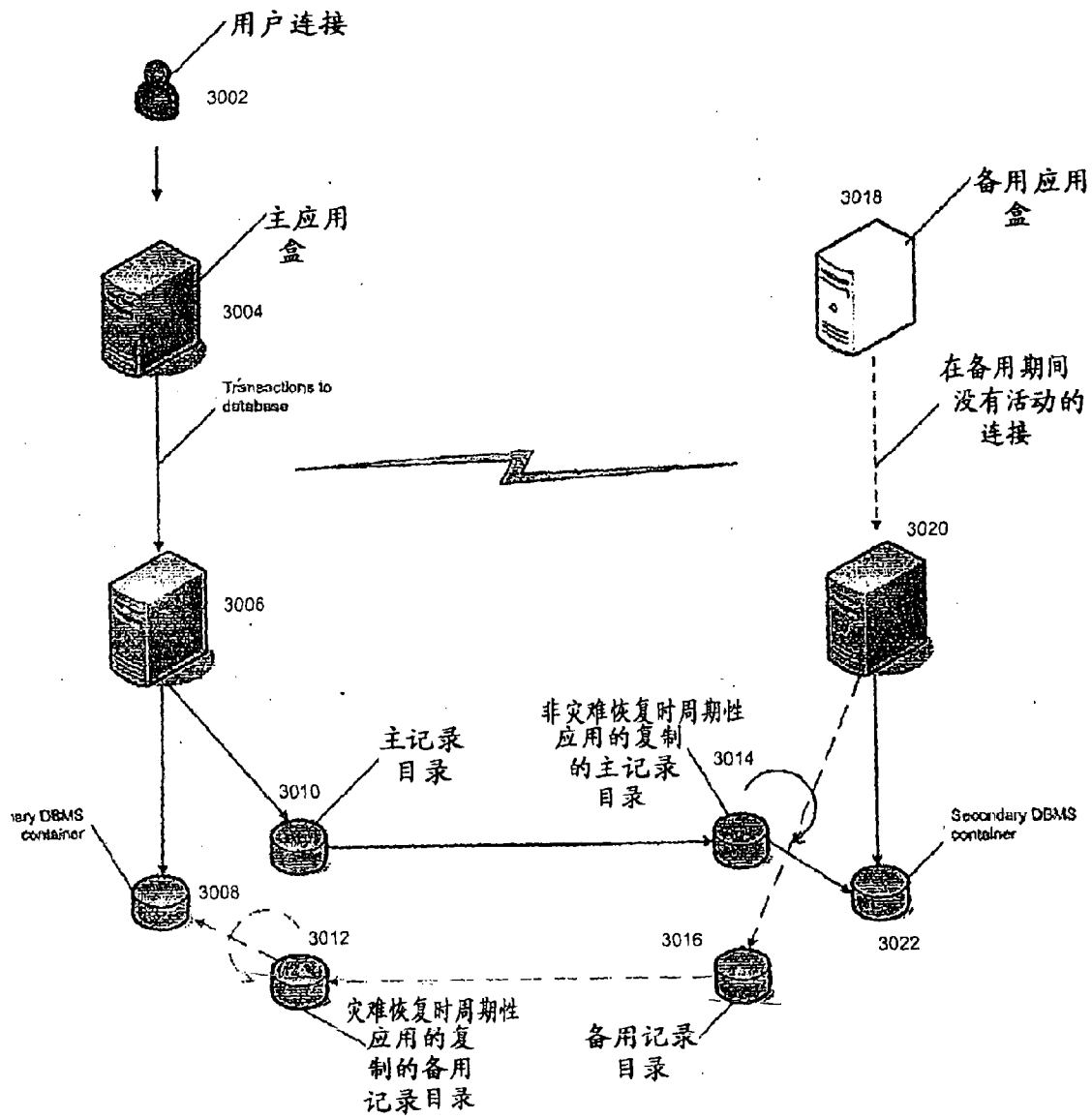


图 30

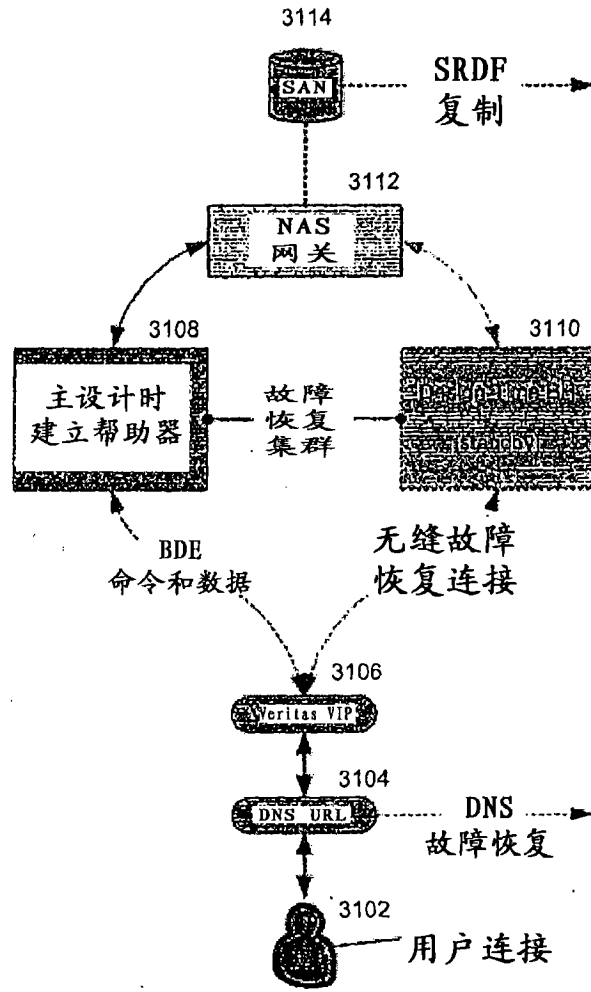


图 31

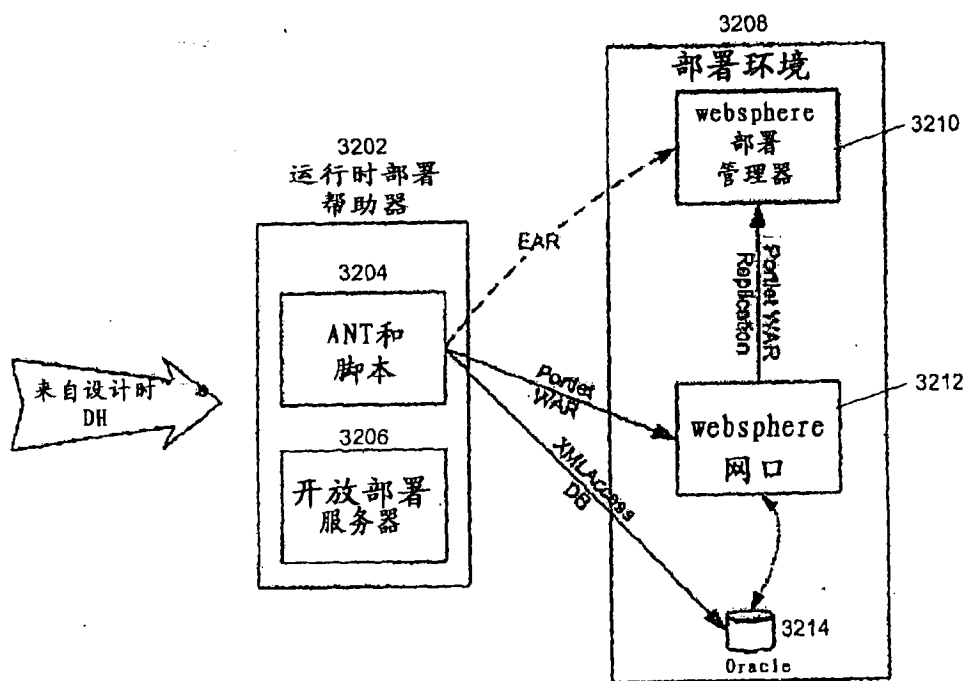


图18: 运行时部署帮助器应用

图 32

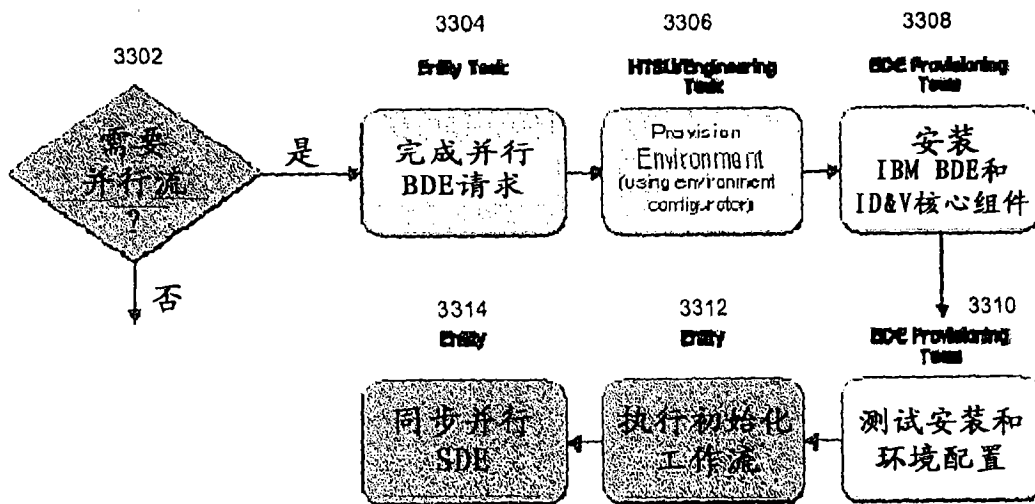


图8: 新并行流的创建的高层处理

图 33

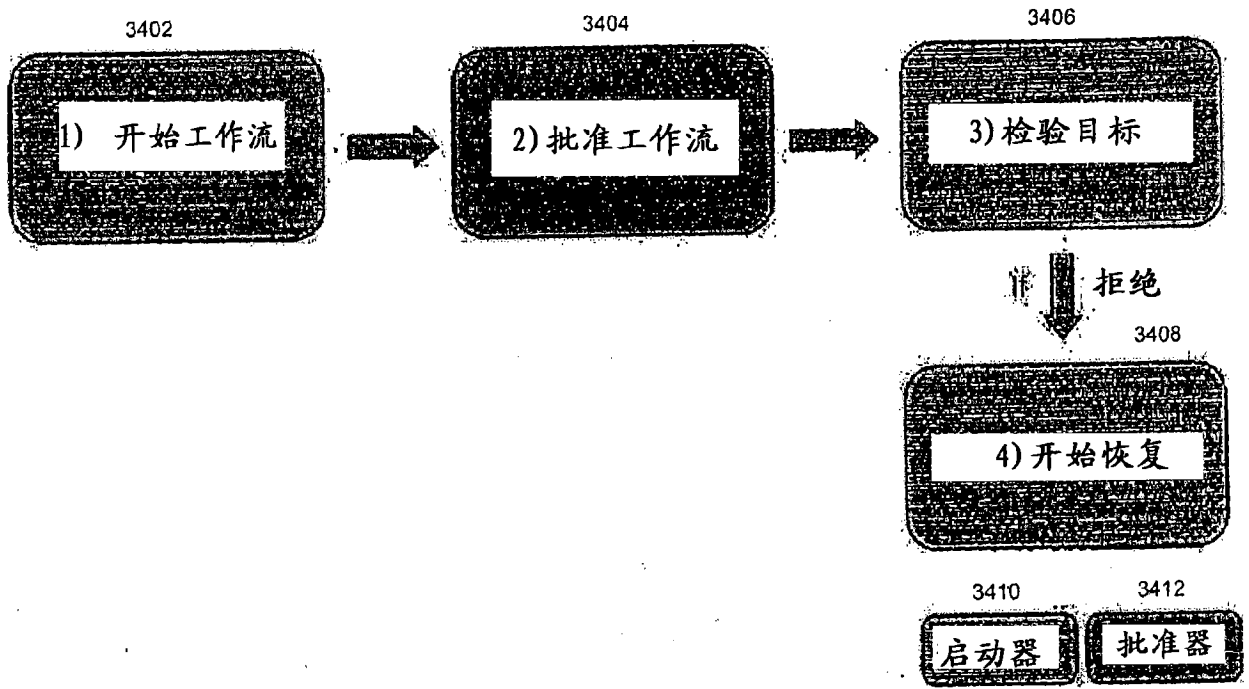


图 34

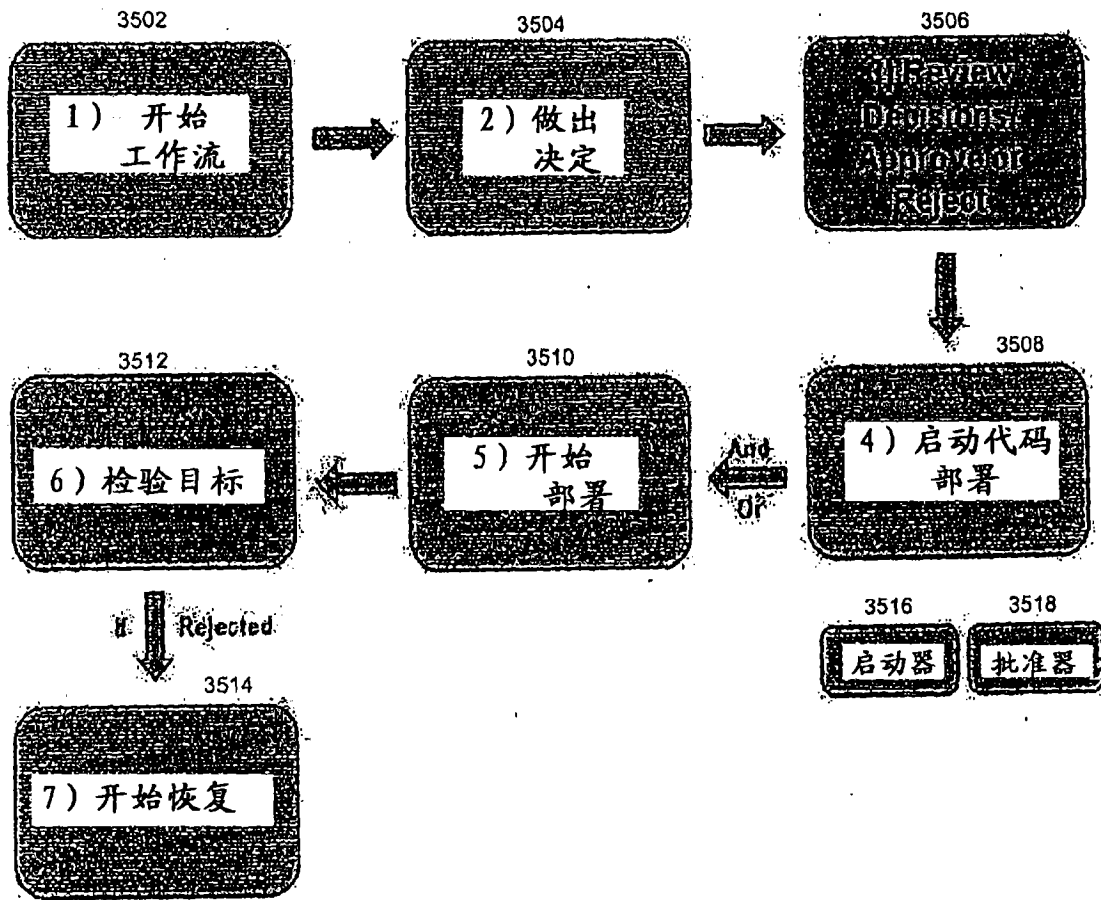


图 35

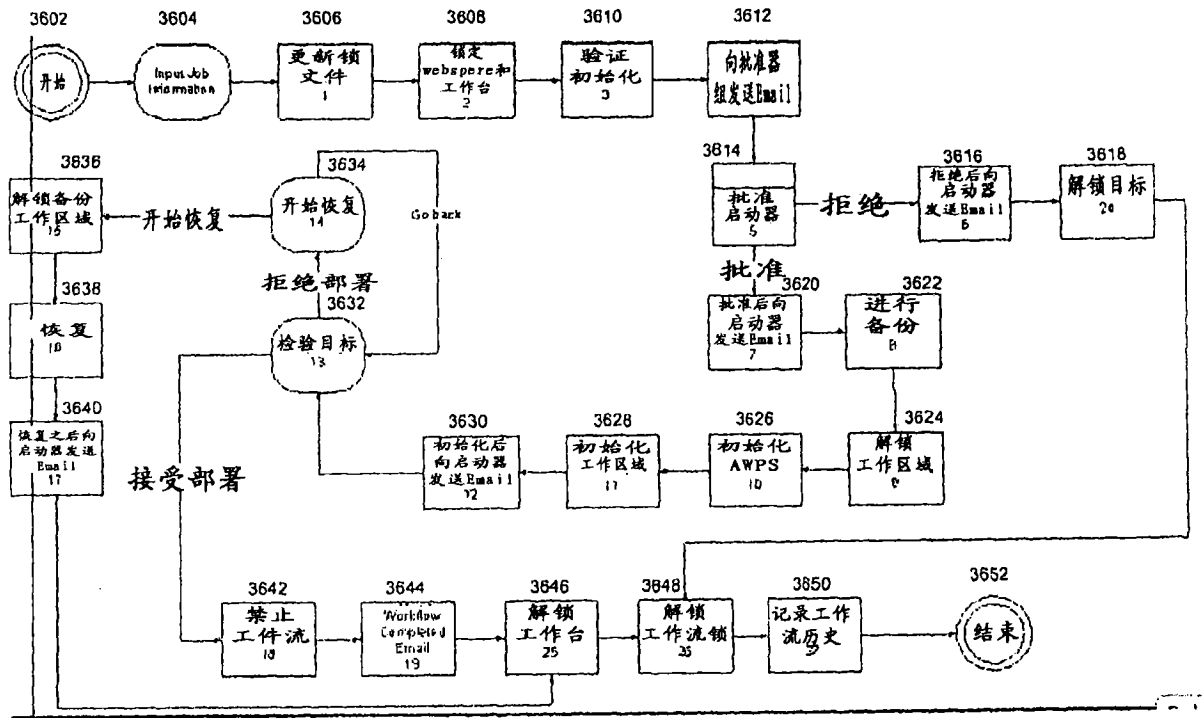


图 36

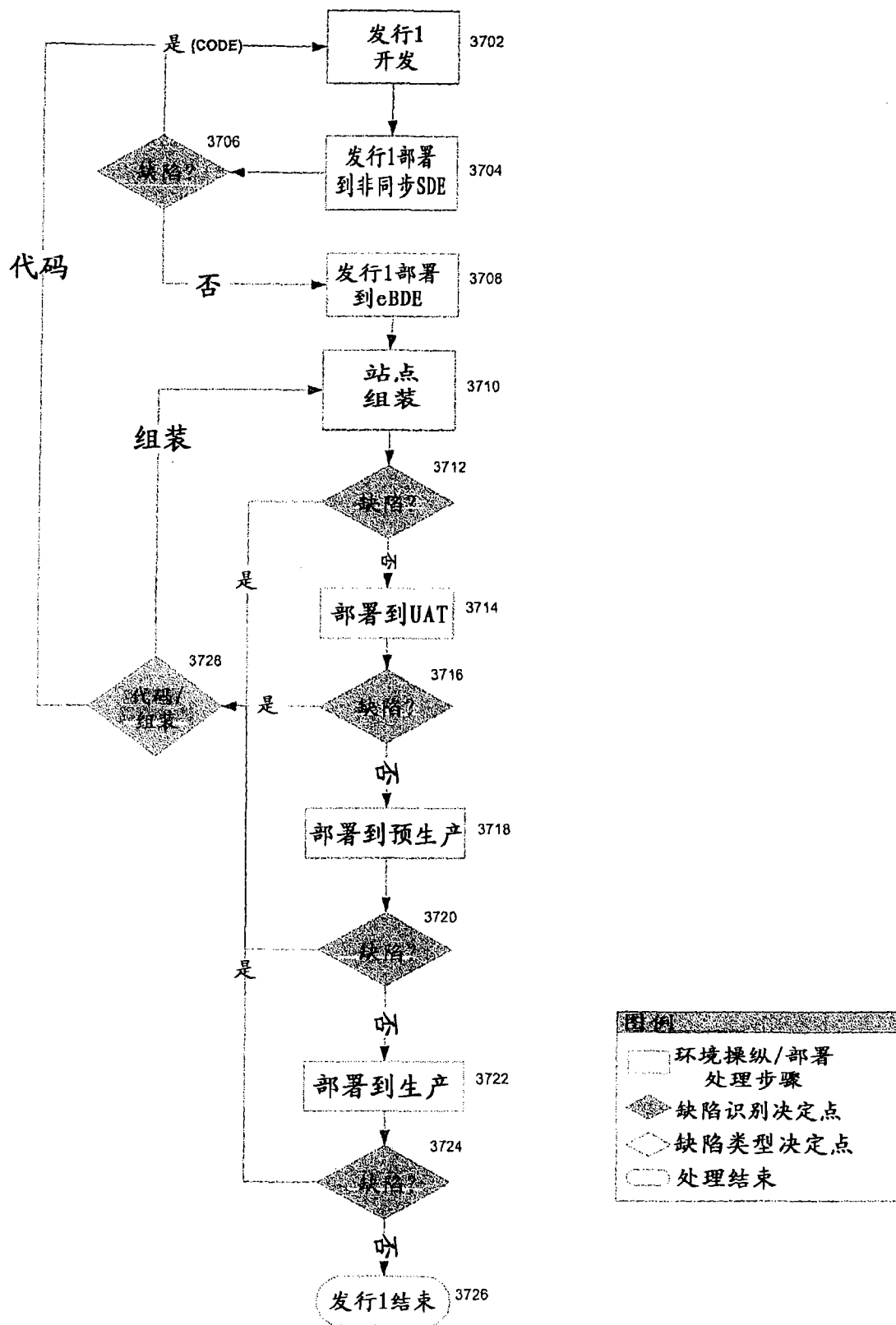


图 37

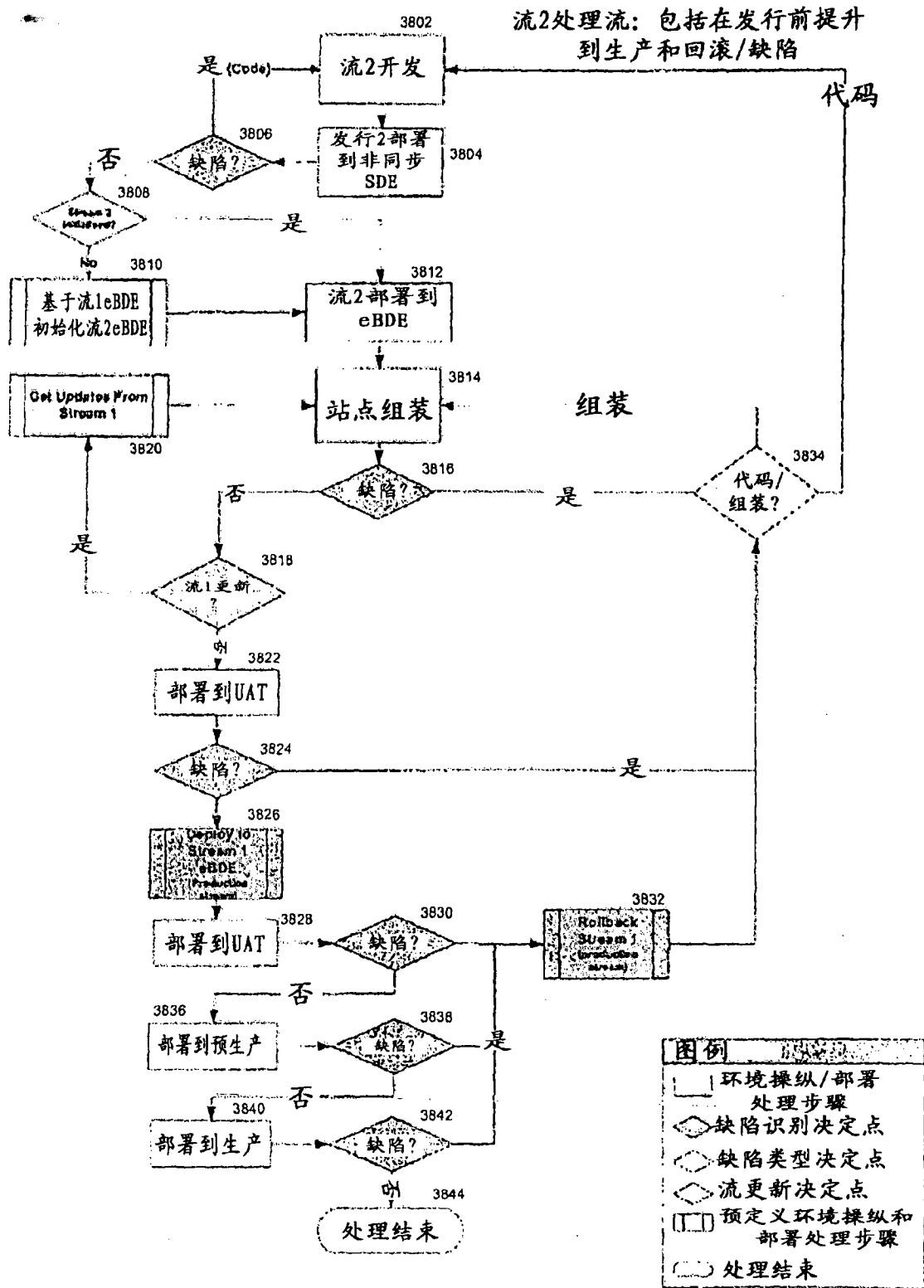


图 38

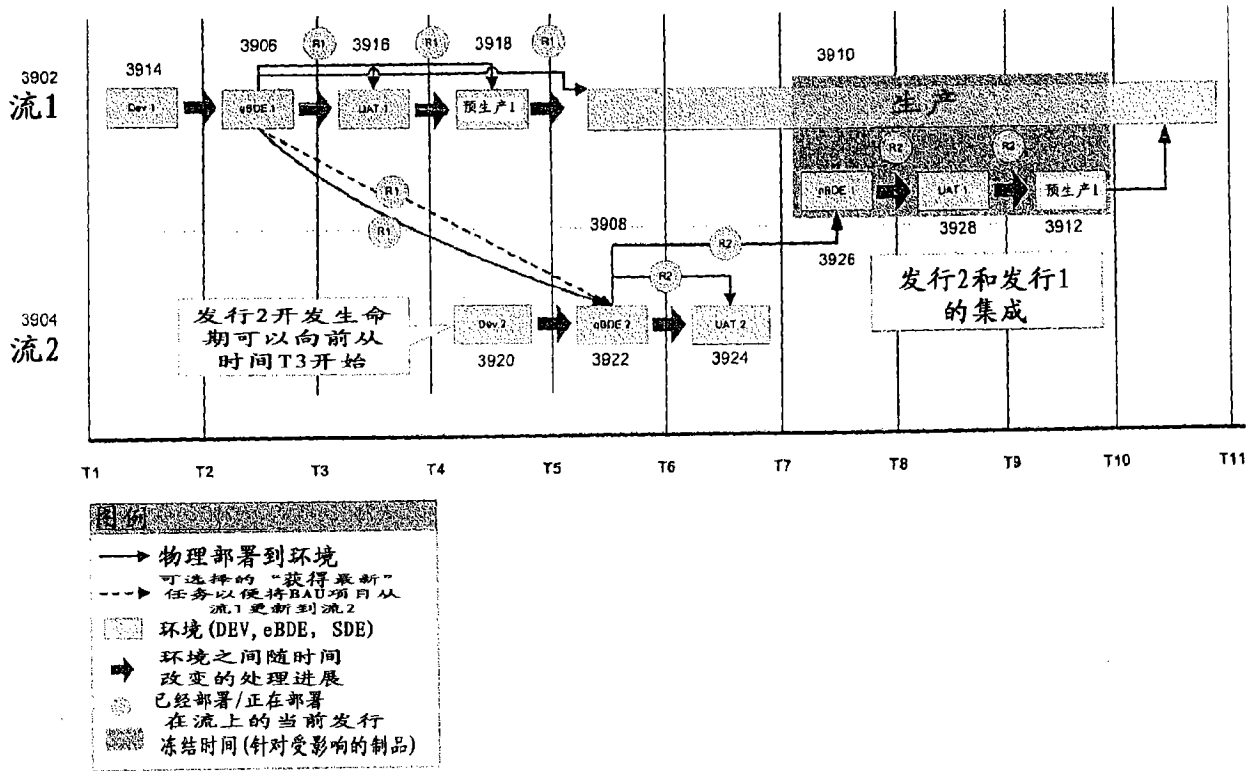


图 39

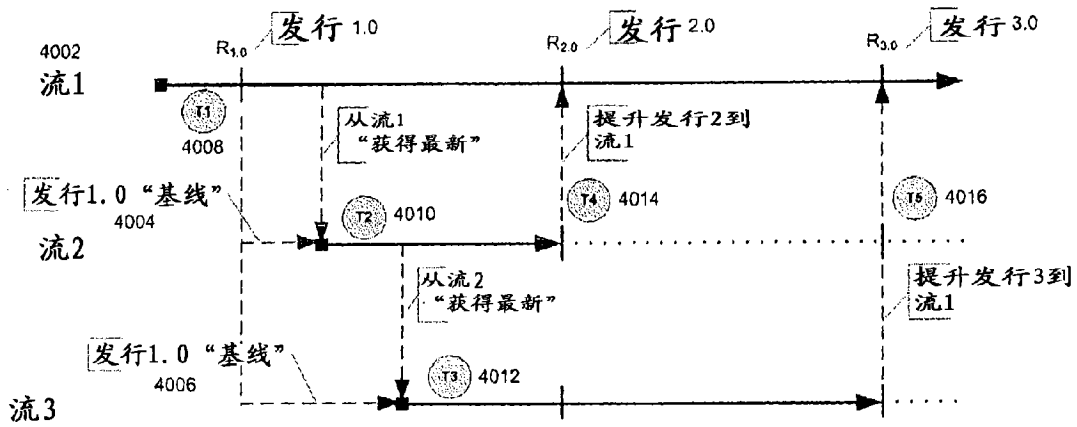


图 40

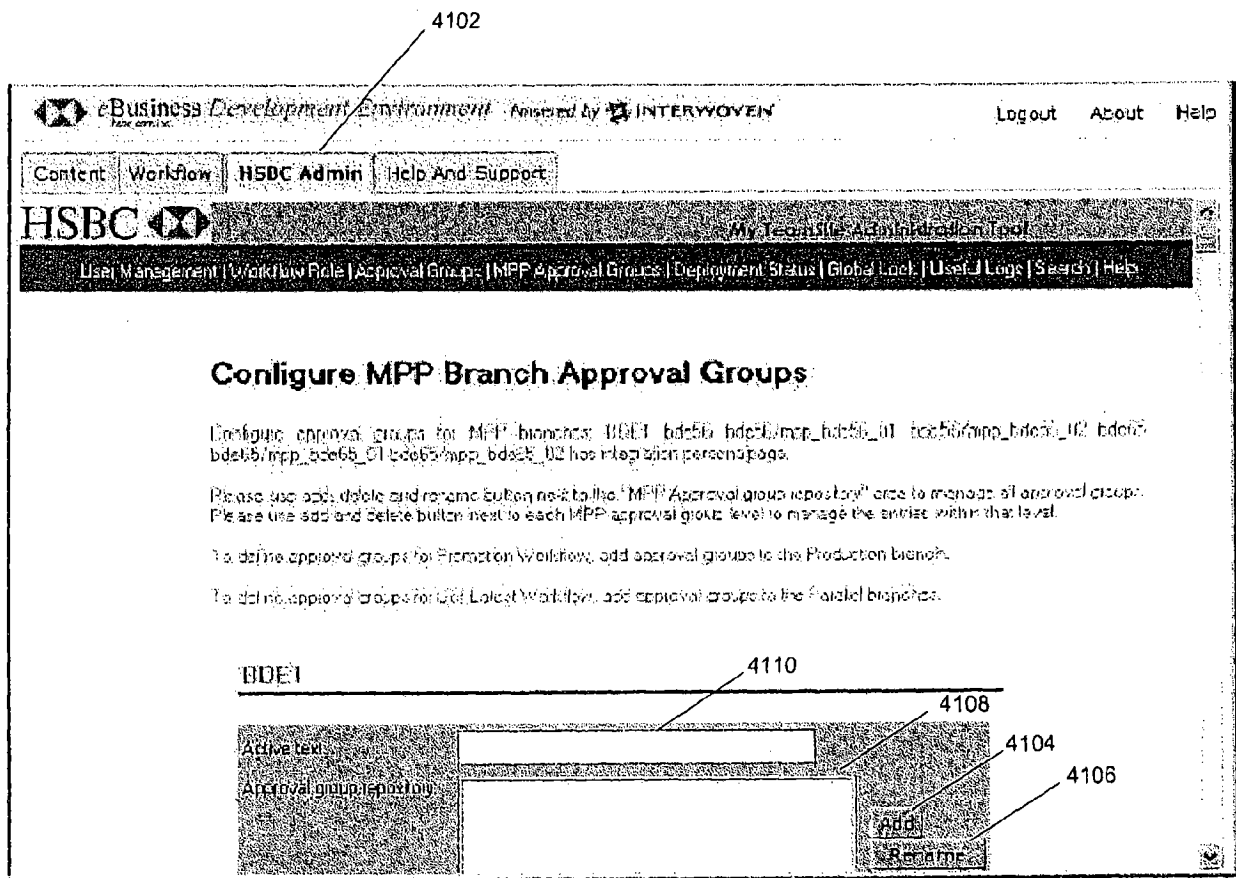


Figure 14: The TeamSite MPP Approval Groups administration page.

图 41

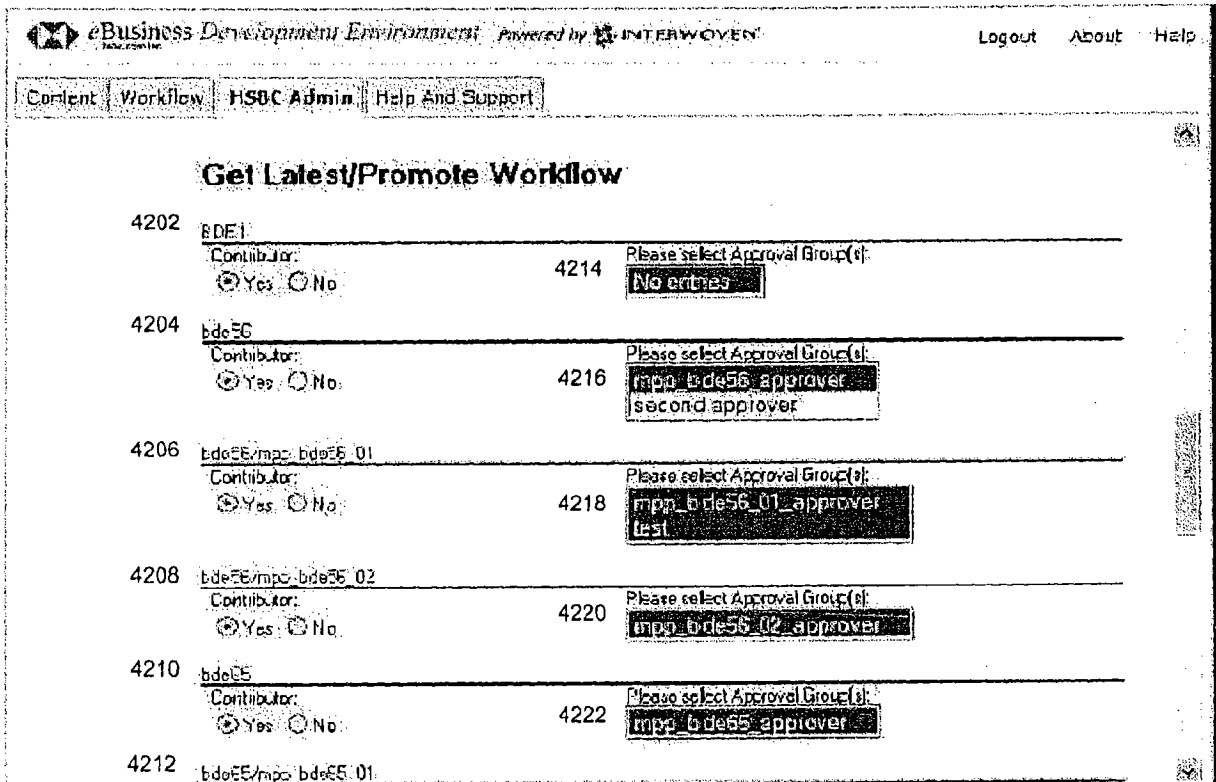


Figure 15: MPP workflow role configuration page

图 42

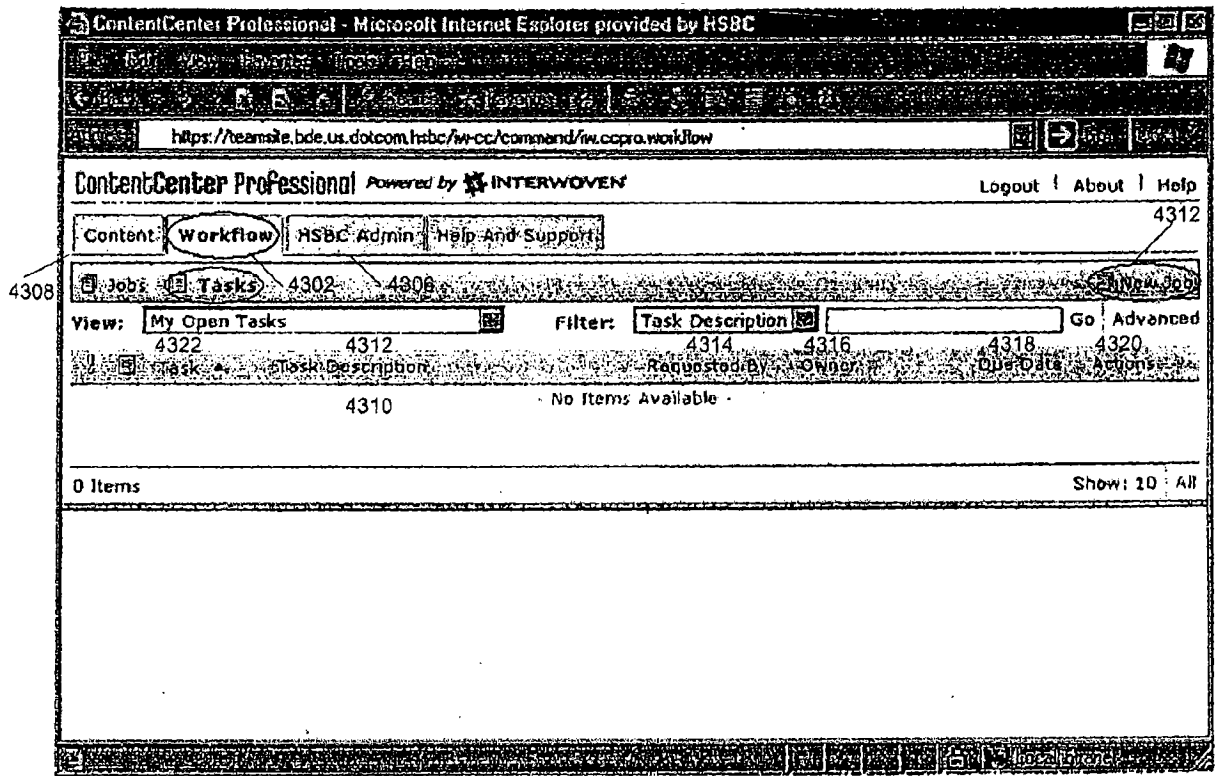


图 43

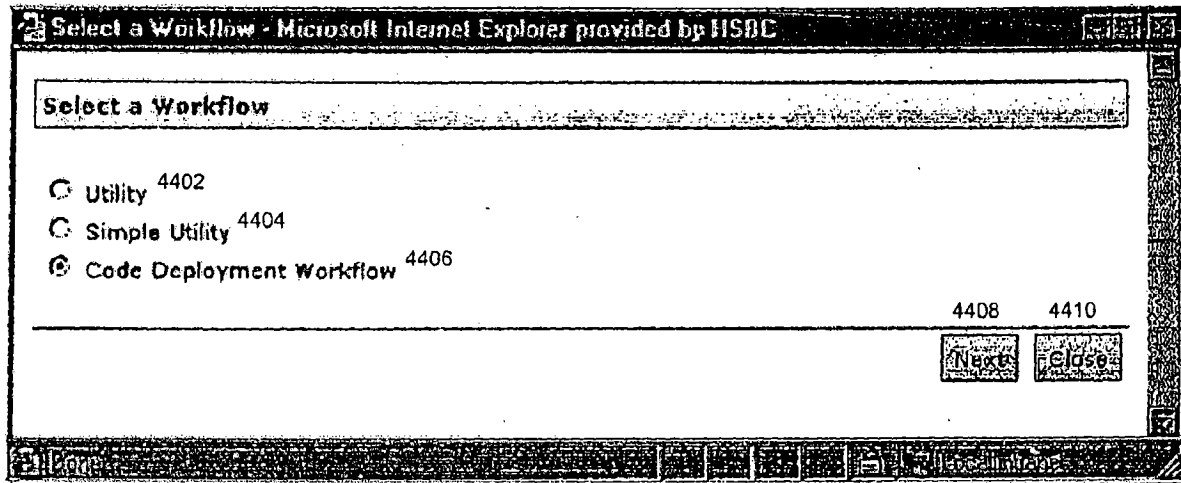


图 44

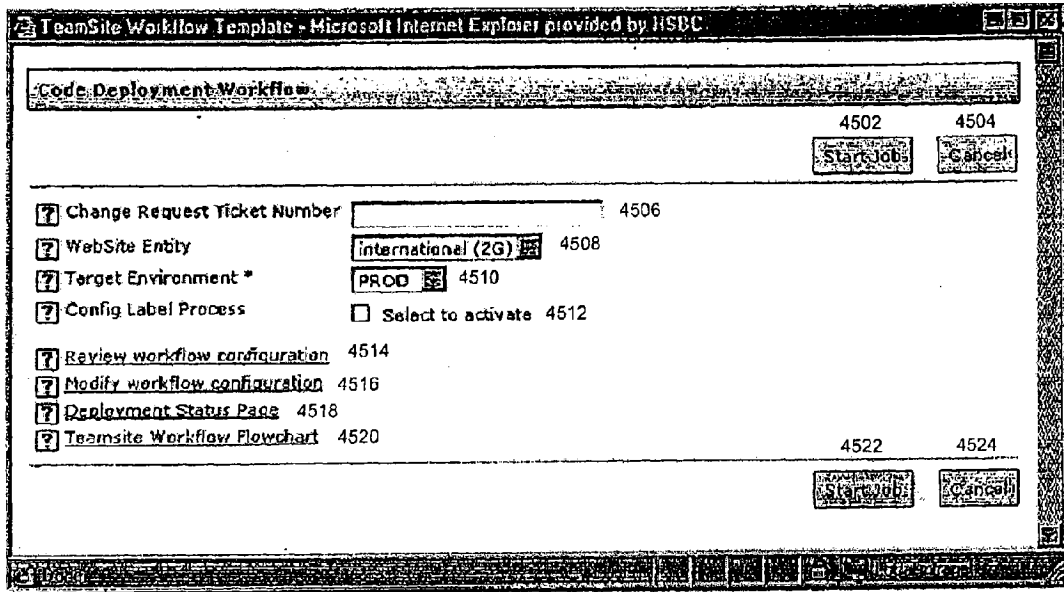


图 45

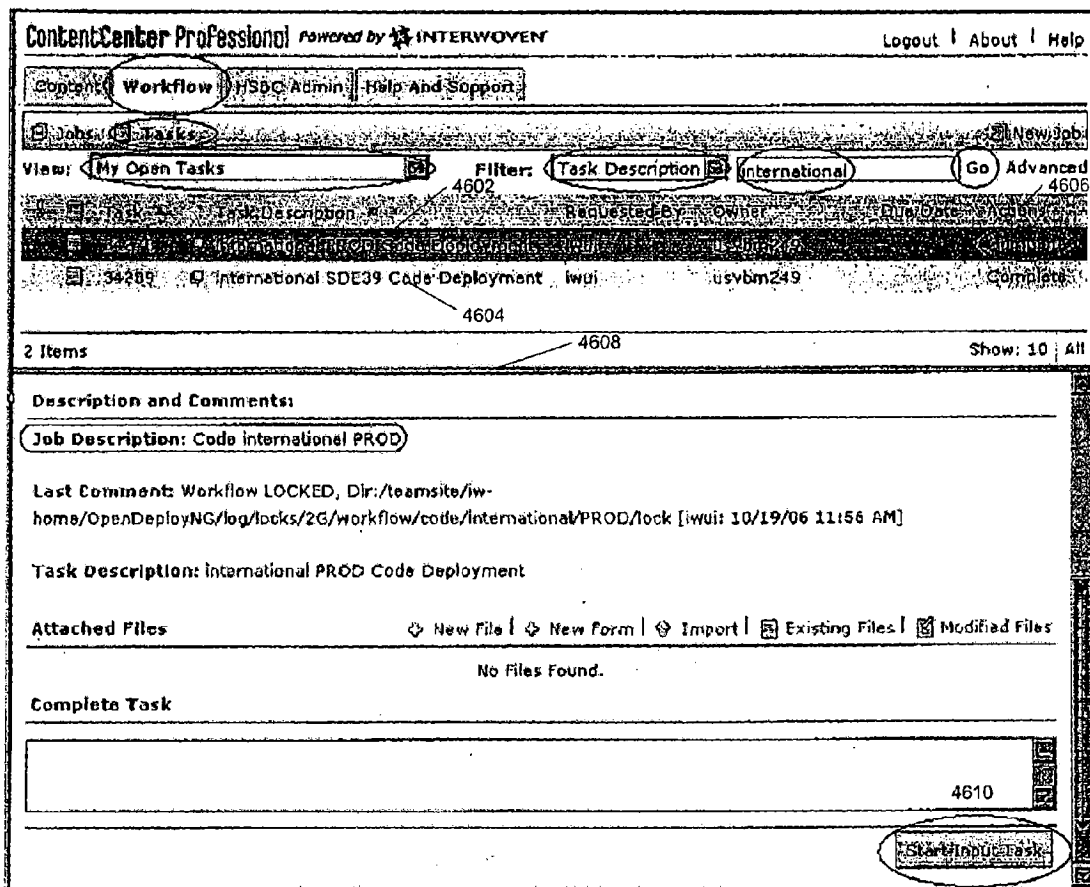


图 46

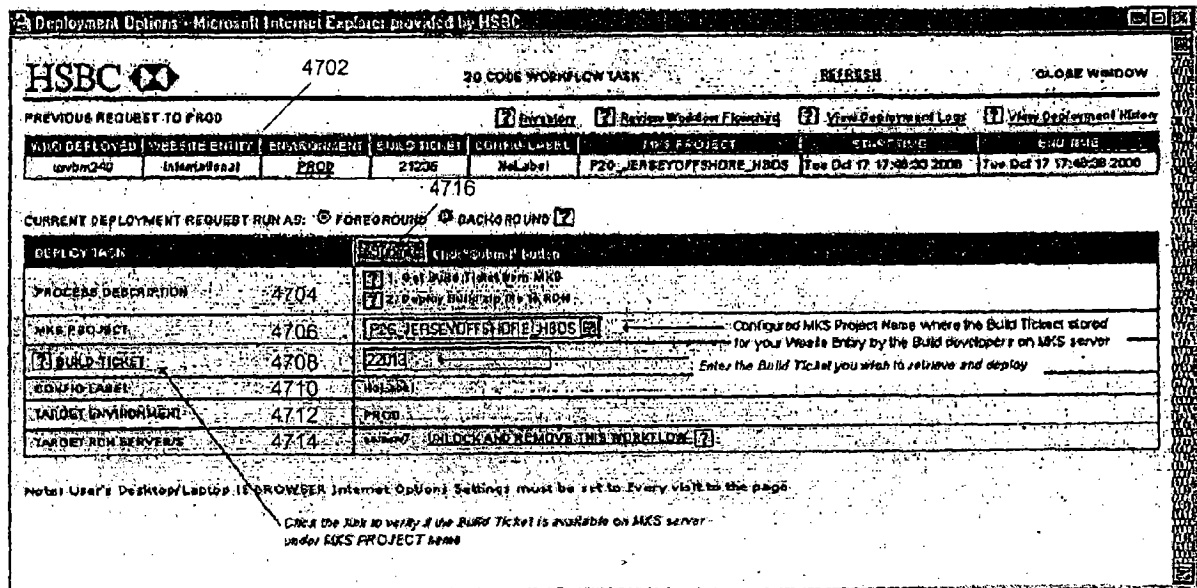


图 47

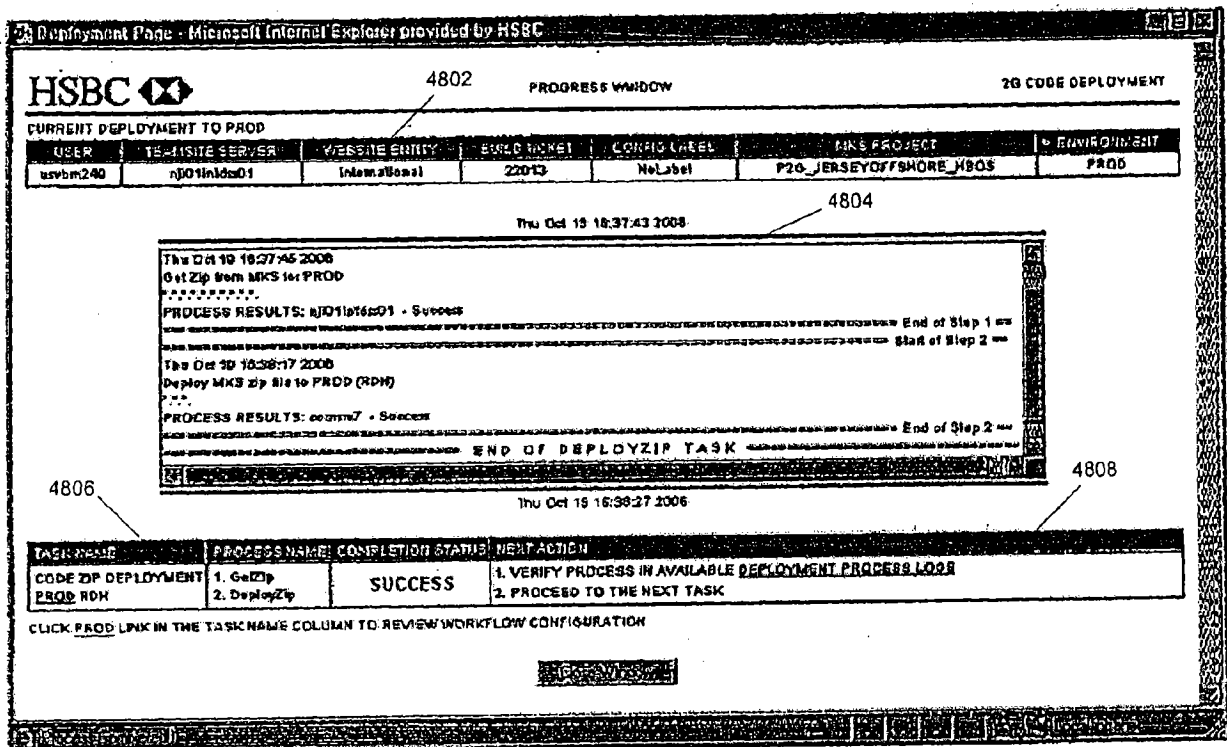


图 48

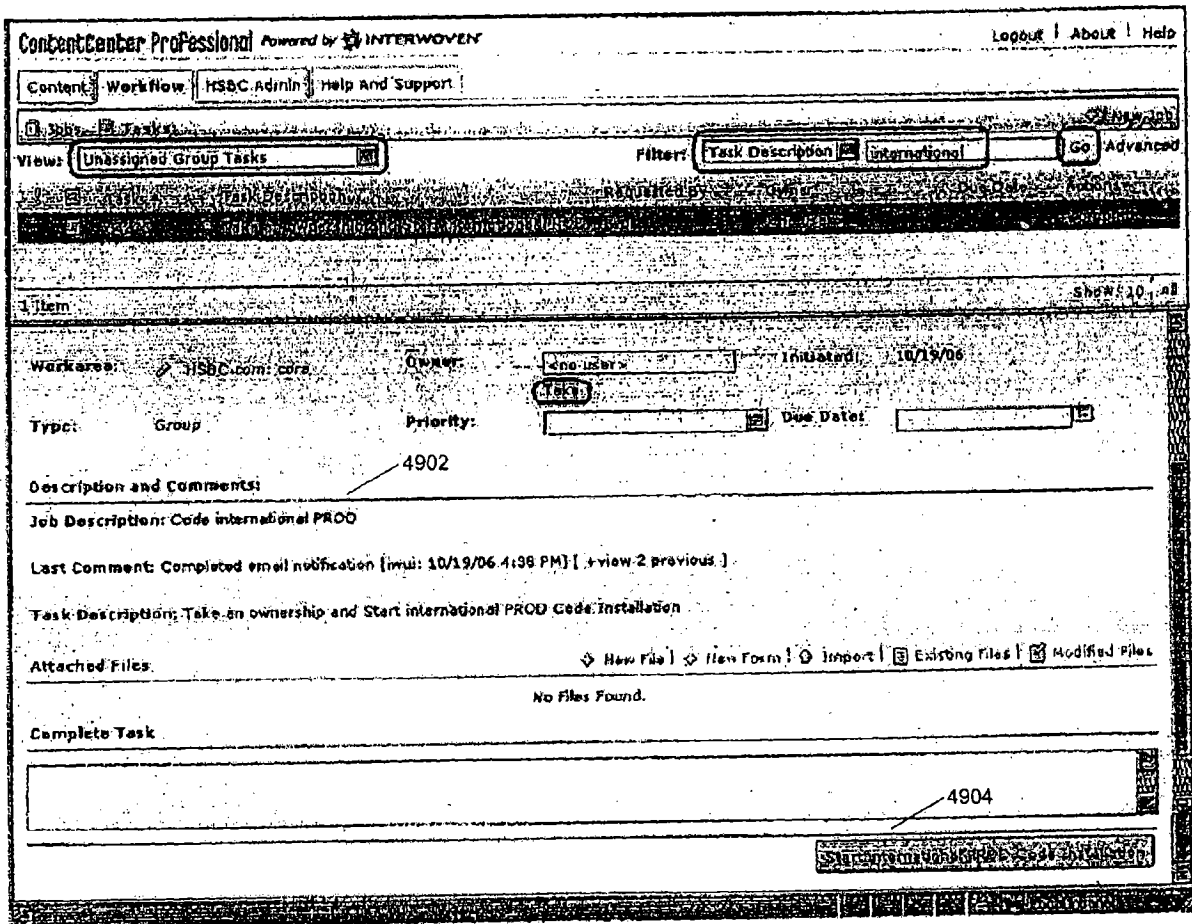


图 49

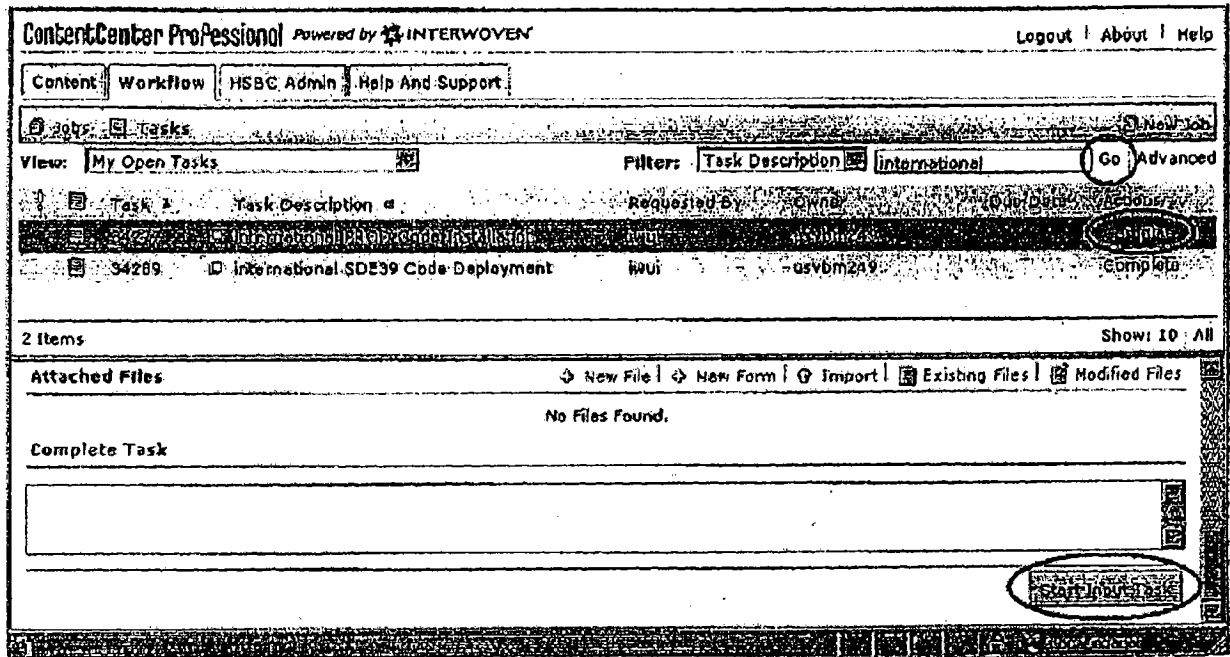


图 50

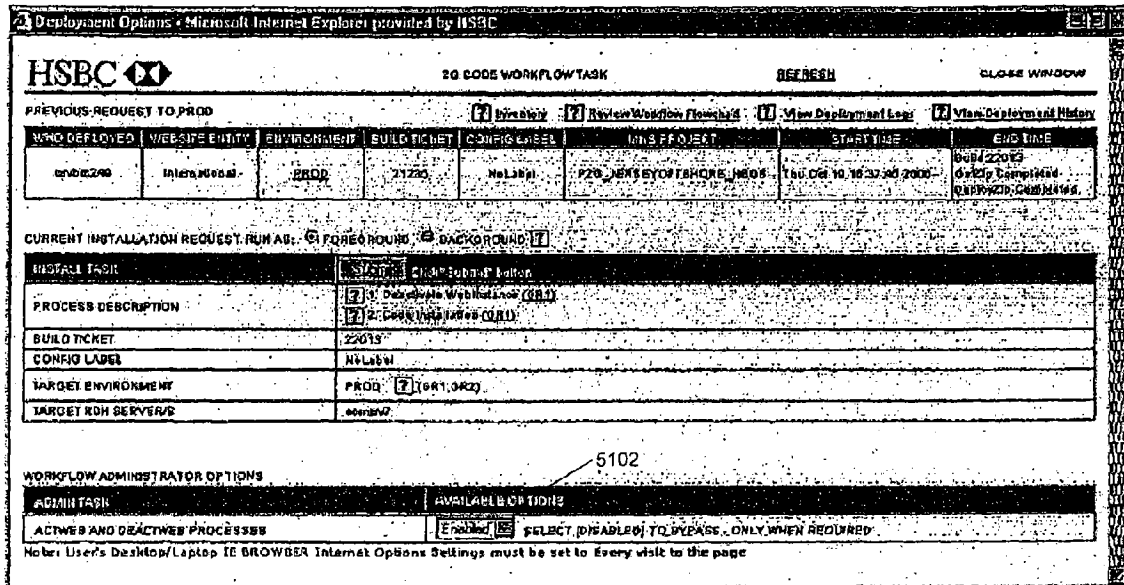



图 51

Deployment Page - Microsoft Internet Explorer provided by HSBC

HSBC  **PROGRESS WINDOW** **20 CODE INSTALLATION**

CURRENT INSTALLATION TO PROD

USER	INSTANCE SERVER	WEBSITE ENTITY	BUILD TARGET	CONFIG LABEL	TARGET GROUP	ENVIRONMENT
usdm249	sj011inbx01	International	22013	NoLabel	GR1,GR2	PROD

5201

START OF INSTALLATION TASK PROCESS

Start of Step 1

Fri Oct 20 14:16:50 2006

DeActWebInstance process on PROD (GR1)

0 R1 zym1cnbx01:80 zym2cnbx01:80

PROCESS RESULTS:NAS cmd.sh: PROD zym1cnbx01:80 = SUCCESS

PROCESS RESULTS:NAS cmd.sh: PROD zym2cnbx01:80 = SUCCESS

FINAL RDH com7 DeActWeb process = Success

End of Step 1

Start of Step 2

Fri Oct 20 14:18:04 2006

5202

TASK NAME	PROCESS NAME	COMPLETION STATUS	NEXT ACTION
CODE INSTALLATION PROD (GR1)	1. DeActWeb 2. Install	SUCCESS	1. VERIFY PROCESS IN AVAILABLE DEPLOYMENT PROCESS LOGS 2. PROCEED TO THE NEXT TASK

CLICK [PROD](#) LINK IN THE TASK NAME COLUMN TO REVIEW WORKFLOW CONFIGURATION

图 52

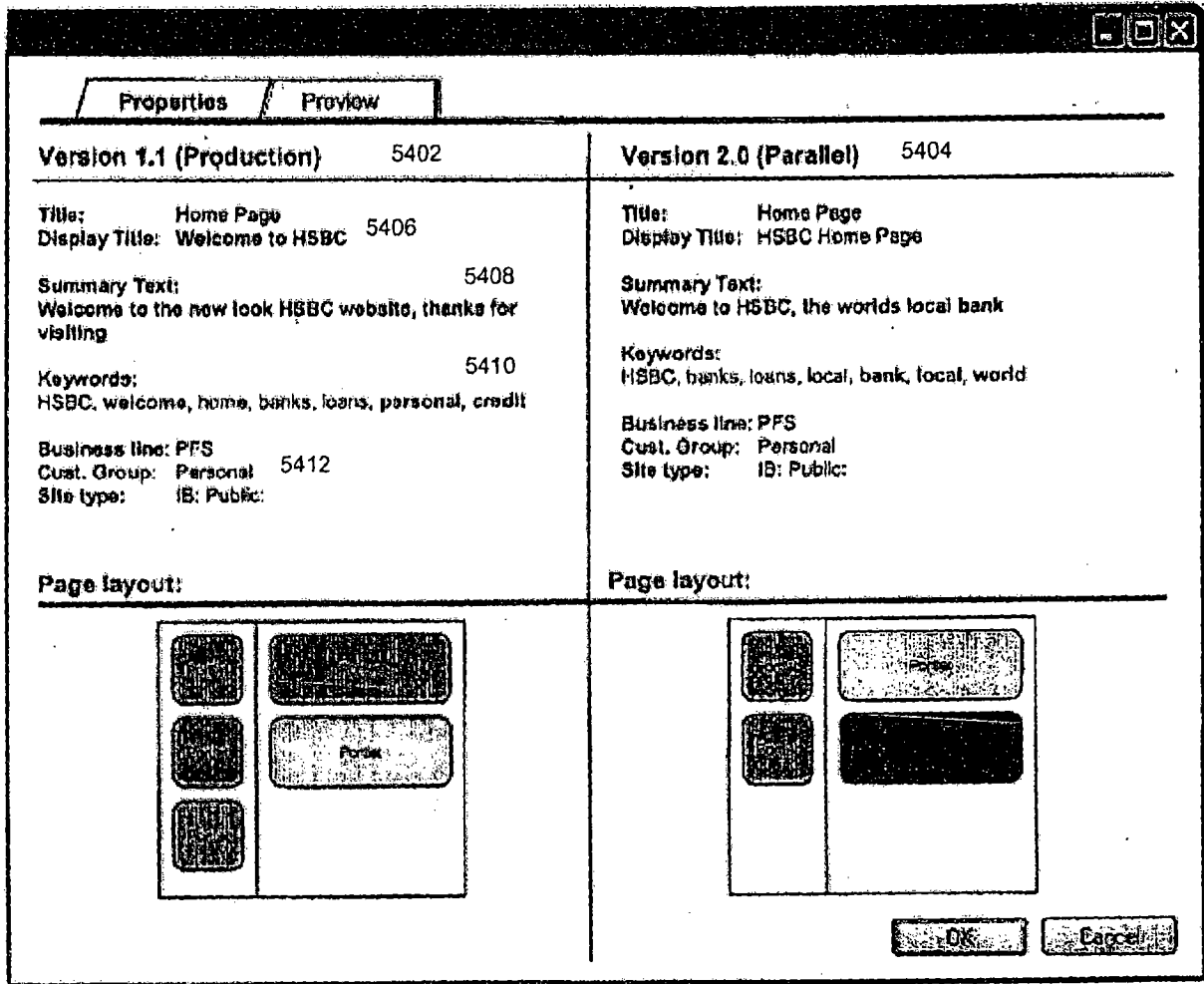


图 54

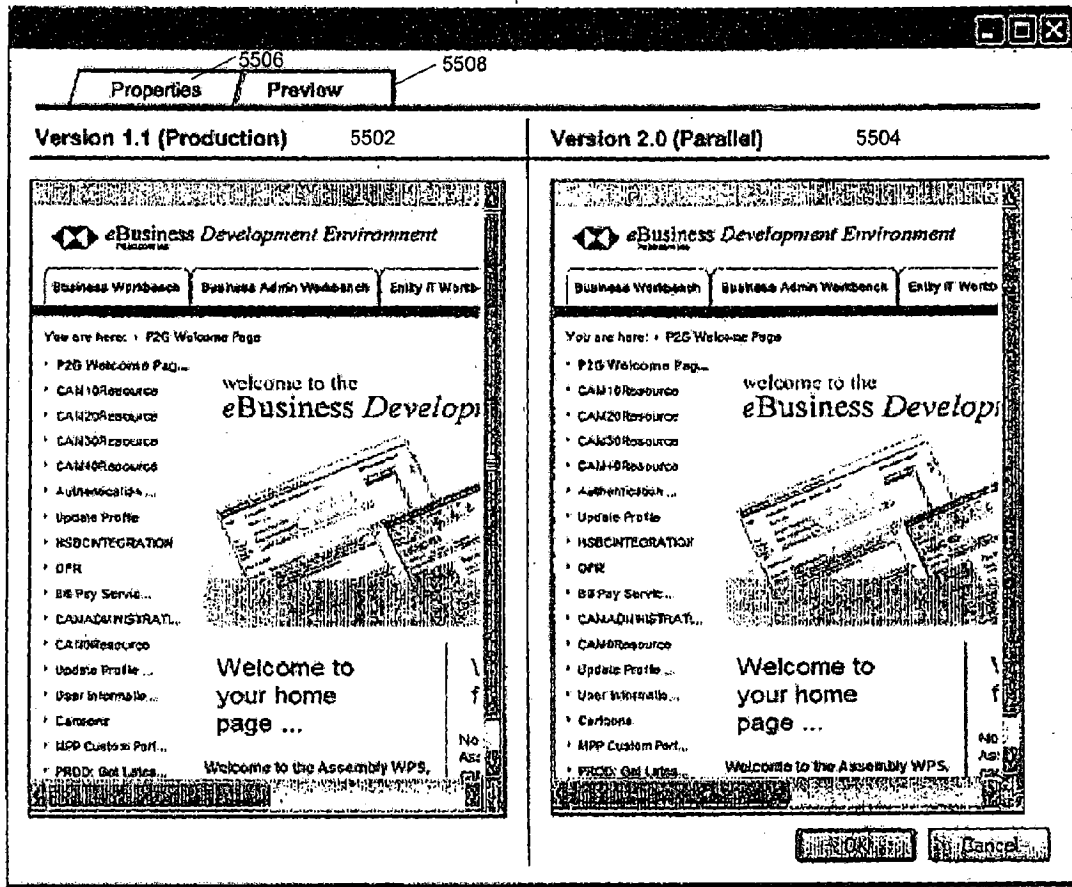


图 55

User Jon Smith synchronised Version 2.1 from production-branch to parallel-branch on 5 th May 2006 performing the following operations:		
Version 1.1 (Production) Pages: 5608	Version 2.0 (Parallel) 5604	5606
Home Page Credit card home page Credit card product info Business credit card Premier credit card Credit card contacts 5610 Content dependencies:	Home Page Credit card home page Credit card product info Business credit card Premier credit card Credit card contacts	Edition 2.0 changes kept Change merged Page added to Edition 2.0 Page added to Edition 2.0 Page removed from Edition 2.0 Change merged
HomePage1.jsp HomePage2.jsp CreditCard.jsp CreditInfo.jsp Premier.jsp BusinessCards.jsp ContactUs.jsp Applications: 5612	HomePage1.jsp HomePage2.jsp CreditCard.jsp CreditInfo.jsp Premier.jsp BusinessCards.jsp ContactUs.jsp	Changes merged Content added to Edition 2.0 Changes merged Content added to Edition 2.0 Content removed from Edition 2.0 Content added to Edition 2.0 Changes merged
ES_CC_Balance_web UK_Prem_Rewards_web	ES_CC_Balance_web UK_Prem_Rewards_web	v2.3 deployed UK_Prem_Rewards_web removed

图 56

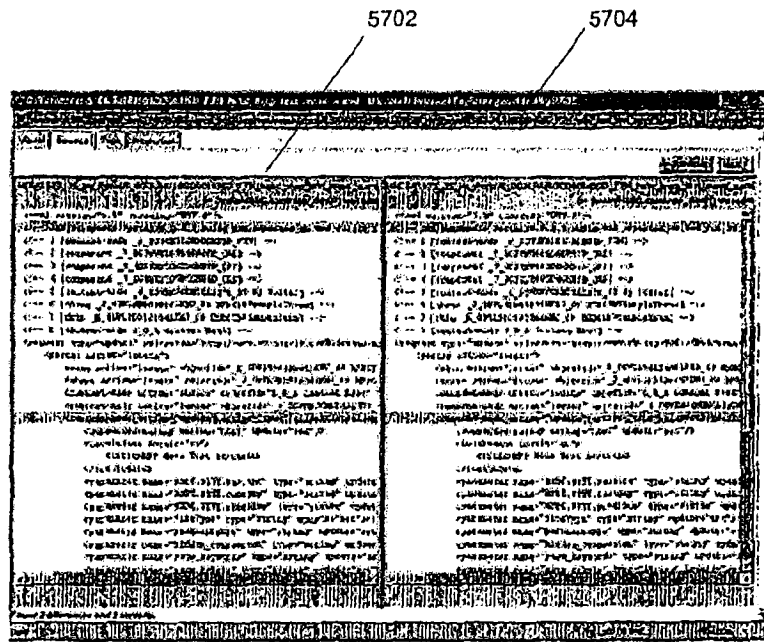


图 57