



(12) 发明专利

(10) 授权公告号 CN 110119283 B

(45) 授权公告日 2023.04.21

(21) 申请号 201910404507.6

(22) 申请日 2019.05.15

(65) 同一申请的已公布的文献号
申请公布号 CN 110119283 A

(43) 申请公布日 2019.08.13

(73) 专利权人 浪潮通用软件有限公司
地址 250100 山东省济南市高新区浪潮路
1036号

(72) 发明人 纪鹏

(74) 专利代理机构 济南信达专利事务所有限公
司 37100
专利代理师 李世喆

(51) Int. Cl.
G06F 8/65 (2018.01)
G06F 8/61 (2018.01)

(56) 对比文件

CN 108241493 A, 2018.07.03

CN 104077140 A, 2014.10.01

US 2011/0239195 A1, 2011.09.29

审查员 刘升

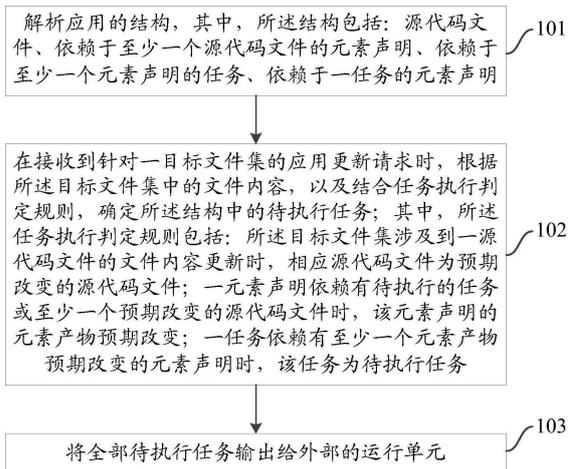
权利要求书3页 说明书12页 附图4页

(54) 发明名称

一种应用更新处理方法、装置及系统、应用更新系统

(57) 摘要

本发明提供了一种应用更新处理方法、装置及系统、应用更新系统,该方法包括:解析应用的结构,该结构包括源代码文件、依赖于若干源代码文件的元素声明、依赖于若干元素声明的任务、依赖于一个任务的元素声明;接收针对一文件集的应用更新请求,根据其中的文件内容并结合任务执行判定规则,确定结构中的待执行任务;将全部待执行任务输出给外部运行单元;规则包括:文件集涉及到一源代码文件的文件内容更新时,相应源代码文件预期改变;一元素声明依赖有待执行的任务或预期改变的源代码文件时,该元素声明的元素产物预期改变;一任务依赖有元素产物预期改变的元素声明时,该任务为待执行任务。本方案可自动识别待更新对象,故可提高应用更新效率。



1. 一种应用更新处理方法,其特征在于,包括:

解析应用的结构,其中,所述结构包括:源代码文件、依赖于至少一个源代码文件的元素声明、依赖于至少一个元素声明的任务、依赖于一个任务的元素声明;

在接收到针对一目标文件集的应用更新请求时,根据所述目标文件集中的文件内容,以及结合任务执行判定规则,确定所述结构中的待执行任务;

将全部待执行任务输出给外部的运行单元;

其中,所述任务执行判定规则包括:

所述目标文件集涉及到一源代码文件的文件内容更新时,相应源代码文件为预期改变的源代码文件;

一元素声明依赖有待执行的任务或至少一个预期改变的源代码文件时,该元素声明的元素产物预期改变;

一任务依赖有至少一个元素产物预期改变的元素声明时,该任务为待执行任务;

所述任务执行判定规则还包括:

一任务每一次成功执行所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容均不完全相同时,该任务为待执行任务;

所述任务执行判定规则还包括:

所述目标文件集涉及到一元素声明的元素声明文件的文件内容更新时,该元素声明的元素产物预期改变,该元素声明所依赖的任务为待执行任务;

所述任务执行判定规则还包括:

一任务每一次成功执行所间接依赖的全部文件的文件内容,与所述目标文件集中相应文件的文件内容均不完全相同时,该任务为待执行任务;

所述全部文件包括全部源代码文件和全部元素声明文件;

所述结构包括的元素声明有:依赖于服务编排任务的应用声明、依赖于镜像构建任务的镜像声明、依赖于制品构建任务的制品声明、依赖于至少一个源代码文件的源代码工程声明;

所述结构包括的任务有:依赖于至少一个镜像声明的服务编排任务、依赖于至少一个制品声明的镜像构建任务、依赖于至少一个源代码工程声明的制品构建任务;

所述根据所述目标文件集中的文件内容,以及结合任务执行判定规则,确定所述结构中的待执行任务,包括:

将所述结构中的每一个制品声明依次作为当前制品声明,并执行:判断制品仓库中是否存储有对应于所述目标文件集的当前制品声明的制品产物,若否,进一步判断所述制品仓库中是否存储有一当前制品声明的制品产物,该制品产物对应的文件集中包括的、该当前制品声明所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容完全相同,若否,将当前制品声明所依赖的制品构建任务确定为待执行任务;

将所述结构中的每一个镜像声明依次作为当前镜像声明,并执行:判断镜像仓库中是否存储有对应于所述目标文件集的当前镜像声明的镜像产物,若否,进一步判断所述镜像仓库中是否存储有一当前镜像声明的镜像产物,该镜像产物对应的文件集中包括的、该当前镜像声明所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容完全相同,若否,将当前镜像声明所依赖的制品构建任务确定为待执行任务。

2. 一种执行权利要求1所述应用更新处理方法的调度单元,其特征在于,包括:

解析子单元,用于解析应用的结构,其中,所述结构包括:源代码文件、依赖于至少一个源代码文件的元素声明、依赖于至少一个元素声明的任务、依赖于一任务的元素声明;

处理子单元,用于在接收到针对一目标文件集的应用更新请求时,根据所述目标文件集中的文件内容,以及结合任务执行判定规则,确定所述结构中的待执行任务;

输出子单元,用于将全部待执行任务输出给外部的运行单元;

其中,所述任务执行判定规则包括:

所述目标文件集涉及到一源代码文件的文件内容更新时,相应源代码文件为预期改变的源代码文件;

一元素声明依赖有待执行的任务或至少一个预期改变的源代码文件时,该元素声明的元素产物预期改变;

一任务依赖有至少一个元素产物预期改变的元素声明时,该任务为待执行任务。

3. 一种应用更新处理系统,其特征在于,包括:

配置单元、如权利要求2所述的调度单元、运行单元;

其中,所述配置单元,用于在接收到外部输入的应用的配置信息时,解析所述配置信息,以确定所述应用的结构并发送给所述调度单元;

所述运行单元,用于在接收到所述调度单元发来的全部待执行任务时,执行所述全部待执行任务中的制品构建任务和镜像构建任务,以生成至少一个镜像产物;将生成的全部镜像产物存储至外部的镜像仓库;将所述全部待执行任务中的服务编排任务发送给外部的服务编排系统,以触发所述服务编排系统根据所述全部镜像产物,通过执行接收到的服务编排任务以更新应用产物。

4. 根据权利要求3所述的应用更新处理系统,其特征在于,

所述运行单元,用于对于所述全部待执行任务中的每一个制品构建任务,根据所述目标文件集中包括的、该制品构建任务所间接依赖的全部源代码文件的文件内容,执行该制品构建任务,以生成对应于所述目标文件集的制品产物,并将生成的制品产物存储至制品仓库中;以及基于生成的制品产物,执行所述全部待执行任务中的每一个镜像构建任务,以生成对应于所述目标文件集的镜像产物,并将生成的镜像产物存储至镜像仓库中。

5. 一种应用更新系统,其特征在于,包括:

如权利要求4所述的应用更新处理系统、镜像仓库、服务编排系统、客户端开发工具、版本控制系统;

其中,所述镜像仓库,用于存储所述应用更新处理系统发来的全部镜像产物;

所述服务编排系统,用于经所述应用更新处理系统触发,根据所述全部镜像产物,执行所述应用更新处理系统发来的服务编排任务,以更新应用产物;

所述客户端开发工具,用于缓存外部输入的至少一个目标源代码文件的文件内容;在接收到外部输入的应用更新请求时,将所述至少一个目标源代码文件的文件内容发送给所述版本控制系统;

所述版本控制系统,用于在接收到所述至少一个目标源代码文件的文件内容时,生成当前文件集,并向所述应用更新处理系统发送应用更新请求;

其中,所述当前文件集中,任一所述目标源代码文件的文件内容为接收到的文件内容,

任一其他文件的文件内容为生成的上一个文件集中该其他文件的文件内容,且不同文件集的版本号不同。

一种应用更新处理方法、装置及系统、应用更新系统

技术领域

[0001] 本发明涉及计算机技术领域,特别涉及一种应用更新处理方法、装置及系统、应用更新系统。

背景技术

[0002] 软件生命周期管理系统在经过配置之后,可以自动化地完成从源代码编译到应用更新的整个过程。这里的应用,还可称作应用类软件。

[0003] 通常情况下,开发者可基于这类系统实践迭代开发和增量开发。比如会在一天之内提交多次代码更改,每一次提交的代码更改可能仅仅涉及到应用的一小部分元素。针对每次代码更改,均需对应实现应用的增量集成和部署,即基于更改的代码来更新应用。

[0004] 目前,已有的软件生命周期管理系统忽视了应用自身的内部结构,因此需要人工来识别应用的元素集合中需要更新的子集,以实现增量集成和部署。

[0005] 但是,待更新子集的人工识别会增大耗时,降低应用更新效率。

发明内容

[0006] 本发明提供了一种应用更新处理方法、装置及系统、应用更新系统,能够提高应用更新效率。

[0007] 为了达到上述目的,本发明是通过如下技术方案实现的:

[0008] 第一方面,本发明提供了一种应用更新处理方法,包括:

[0009] 解析应用的结构,其中,所述结构包括:源代码文件、依赖于至少一个源代码文件的元素声明、依赖于至少一个元素声明的任务、依赖于一任务的元素声明;

[0010] 在接收到针对一目标文件集的应用更新请求时,根据所述目标文件集中的文件内容,以及结合任务执行判定规则,确定所述结构中的待执行任务;

[0011] 将全部待执行任务输出给外部的运行单元;

[0012] 其中,所述任务执行判定规则包括:

[0013] 所述目标文件集涉及到一源代码文件的文件内容更新时,相应源代码文件为预期改变的源代码文件;

[0014] 一元素声明依赖有待执行的任务或至少一个预期改变的源代码文件时,该元素声明的元素产物预期改变;

[0015] 一任务依赖有至少一个元素产物预期改变的元素声明时,该任务为待执行任务。

[0016] 进一步地,所述任务执行判定规则还包括:一任务每一次成功执行所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容均不完全相同时,该任务为待执行任务。

[0017] 进一步地,所述任务执行判定规则还包括:所述目标文件集涉及到一元素声明的元素声明文件的文件内容更新时,该元素声明的元素产物预期改变,该元素声明所依赖的任务为待执行任务。

[0018] 进一步地,所述任务执行判定规则还包括:一任务每一次成功执行所间接依赖的全部文件的文件内容,与所述目标文件集中相应文件的文件内容均不完全相同时,该任务为待执行任务;

[0019] 所述全部文件包括全部源代码文件和全部元素声明文件。

[0020] 进一步地,所述结构包括的元素声明有:依赖于服务编排任务的应用声明、依赖于镜像构建任务的镜像声明、依赖于制品构建任务的制品声明、依赖于至少一个源代码文件的源代码工程声明;

[0021] 所述结构包括的任务有:依赖于至少一个镜像声明的服务编排任务、依赖于至少一个制品声明的镜像构建任务、依赖于至少一个源代码工程声明的制品构建任务。

[0022] 进一步地,所述根据所述目标文件集中的文件内容,以及结合任务执行判定规则,确定所述结构中的待执行任务,包括:

[0023] 将所述结构中的每一个制品声明依次作为当前制品声明,并执行:判断制品仓库中是否存储有对应于所述目标文件集的当前制品声明的制品产物,若否,进一步判断所述制品仓库中是否存储有一当前制品声明的制品产物,该制品产物对应的文件集中包括的、该当前制品声明所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容完全相同,若否,将当前制品声明所依赖的制品构建任务确定为待执行任务;

[0024] 将所述结构中的每一个镜像声明依次作为当前镜像声明,并执行:判断镜像仓库中是否存储有对应于所述目标文件集的当前镜像声明的镜像产物,若否,进一步判断所述镜像仓库中是否存储有一当前镜像声明的镜像产物,该镜像产物对应的文件集中包括的、该当前镜像声明所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容完全相同,若否,将当前镜像声明所依赖的制品构建任务确定为待执行任务。

[0025] 第二方面,本发明提供了一种执行上述任一所述应用更新处理方法的调度单元,包括:

[0026] 解析子单元,用于解析应用的结构,其中,所述结构包括:源代码文件、依赖于至少一个源代码文件的元素声明、依赖于至少一个元素声明的任务、依赖于任务的元素声明;

[0027] 处理子单元,用于在接收到针对一目标文件集的应用更新请求时,根据所述目标文件集中的文件内容,以及结合任务执行判定规则,确定所述结构中的待执行任务;

[0028] 输出子单元,用于将全部待执行任务输出给外部的运行单元;

[0029] 其中,所述任务执行判定规则包括:

[0030] 所述目标文件集涉及到一源代码文件的文件内容更新时,相应源代码文件为预期改变的源代码文件;

[0031] 一元素声明依赖有待执行的任务或至少一个预期改变的源代码文件时,该元素声明的元素产物预期改变;

[0032] 一任务依赖有至少一个元素产物预期改变的元素声明时,该任务为待执行任务。

[0033] 第三方面,本发明提供了一种应用更新处理系统,包括:配置单元、如上所述的调度单元、运行单元;

[0034] 其中,所述配置单元,用于在接收到外部输入的应用的配置信息时,解析所述配置

信息,以确定所述应用的结构并发送给所述调度单元;

[0035] 所述运行单元,用于在接收到所述调度单元发来的全部待执行任务时,执行所述全部待执行任务中的制品构建任务和镜像构建任务,以生成至少一个镜像产物;将生成的全部镜像产物存储至外部的镜像仓库;将所述全部待执行任务中的服务编排任务发送给外部的服务编排系统,以触发所述服务编排系统根据所述全部镜像产物,通过执行接收到的服务编排任务以更新应用产物。

[0036] 进一步地,所述运行单元,用于对于所述全部待执行任务中的每一个制品构建任务,根据所述目标文件集中包括的、该制品构建任务所间接依赖的全部源代码文件的文件内容,执行该制品构建任务,以生成对应于所述目标文件集的制品产物,并将生成的制品产物存储至制品仓库中;以及基于生成的制品产物,执行所述全部待执行任务中的每一个镜像构建任务,以生成对应于所述目标文件集的镜像产物,并将生成的镜像产物存储至镜像仓库中。

[0037] 第四方面,本发明提供了一种应用更新系统,包括:上述任一所述的应用更新处理系统、镜像仓库、服务编排系统、客户端开发工具、版本控制系统;

[0038] 其中,所述镜像仓库,用于存储所述应用更新处理系统发来的全部镜像产物;

[0039] 所述服务编排系统,用于经所述应用更新处理系统触发,根据所述全部镜像产物,执行所述应用更新处理系统发来的服务编排任务,以更新应用产物;

[0040] 所述客户端开发工具,用于缓存外部输入的至少一个目标源代码文件的文件内容;在接收到外部输入的应用更新请求时,将所述至少一个目标源代码文件的文件内容发送给所述版本控制系统;

[0041] 所述版本控制系统,用于在接收到所述至少一个目标源代码文件的文件内容时,生成当前文件集,并向所述应用更新处理系统发送应用更新请求;

[0042] 其中,所述当前文件集中,任一所述目标源代码文件的文件内容为接收到的文件内容,任一其他文件的文件内容为生成的上一个文件集中该其他文件的文件内容,且不同文件集的版本号不同。

[0043] 本发明提供了一种应用更新处理方法、装置及系统、应用更新系统,该方法包括:解析应用的结构,该结构包括源代码文件、依赖于若干源代码文件的元素声明、依赖于若干元素声明的任务、依赖于一个任务的元素声明;接收针对一文件集的应用更新请求,根据其中的文件内容并结合任务执行判定规则,确定结构中的待执行任务;将全部待执行任务输出给外部运行单元;规则包括:文件集涉及到一源代码文件的文件内容更新时,相应源代码文件预期改变;一元素声明依赖有待执行的任务或预期改变的源代码文件时,该元素声明的元素产物预期改变;一任务依赖有元素产物预期改变的元素声明时,该任务为待执行任务。本发明可自动识别待更新对象,故可提高应用更新效率。

附图说明

[0044] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

- [0045] 图1是本发明一实施例提供的一种应用更新处理方法的流程图；
- [0046] 图2是本发明一实施例提供的一种应用的结构示意图；
- [0047] 图3是本发明一实施例提供的另一种应用的结构示意图；
- [0048] 图4是本发明一实施例提供的一种调度单元的示意图；
- [0049] 图5是本发明一实施例提供的一种应用更新处理系统的示意图；
- [0050] 图6是本发明一实施例提供的一种应用更新系统的示意图。

具体实施方式

[0051] 为使本发明实施例的目的、技术方案和优点更加清楚，下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例是本发明一部分实施例，而不是全部的实施例，基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动的前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0052] 如图1所示，本发明实施例提供了一种应用更新处理方法，可以包括以下步骤：

[0053] 步骤101：解析应用的结构，其中，所述结构包括：源代码文件、依赖于至少一个源代码文件的元素声明、依赖于至少一个元素声明的任务、依赖于一个任务的元素声明。

[0054] 步骤102：在接收到针对一目标文件集的应用更新请求时，根据所述目标文件集中的文件内容，以及结合任务执行判定规则，确定所述结构中的待执行任务；

[0055] 其中，所述任务执行判定规则包括：

[0056] 所述目标文件集涉及到一源代码文件的文件内容更新时，相应源代码文件为预期改变的源代码文件；

[0057] 一元素声明依赖有待执行的任务或至少一个预期改变的源代码文件时，该元素声明的元素产物预期改变；

[0058] 一任务依赖有至少一个元素产物预期改变的元素声明时，该任务为待执行任务。

[0059] 步骤103：将全部待执行任务输出给外部的运行单元。

[0060] 本发明实施例提供了一种应用更新处理方法，该方法包括：解析应用的结构，该结构包括源代码文件、依赖于若干源代码文件的元素声明、依赖于若干元素声明的任务、依赖于一个任务的元素声明；接收针对一文件集的应用更新请求，根据其中的文件内容并结合任务执行判定规则，确定结构中的待执行任务；将全部待执行任务输出给外部运行单元；规则包括：文件集涉及到一源代码文件的文件内容更新时，相应源代码文件预期改变；一元素声明依赖有待执行的任务或预期改变的源代码文件时，该元素声明的元素产物预期改变；一任务依赖有元素产物预期改变的元素声明时，该任务为待执行任务。本发明实施例可自动识别待更新对象，故可提高应用更新效率。

[0061] 详细地，可以根据开发者编写好的应用的配置信息，来确定应用的结构，然后解析应用的结构以用于确定结构中的待执行任务。

[0062] 首次生成应用时，结构中的全部任务均为待执行任务，首次生成应用后，开发可按需更改一些源代码文件的文件内容，以请求基于这一更改操作再次执行应用生成处理，从而实现应用更新。当然，基于更新后的应用还可再次更新，如此循环，直至生成满足用户需求的应用。

[0063] 由于每次更新时，更改的文件内容通常并不会涉及到结构中的每一个源代码文

件,使得结构中部分任务需要执行而部分任务无需执行,如此,步骤102中,即可基于更改的文件内容,结合任务执行判定规则以将结构中的待执行任务确定出来。

[0064] 在本发明一个实施例中,更新应用时,还可能会涉及到结构的更改,即涉及到应用的配置信息的更改,如此可基于更改后的配置文件再次确定新的应用结构,并继续执行上述步骤101。

[0065] 需要说明的是,针对任一元素来说,结构中的元素是对元素的声明,而生成的元素是实在的产物。如此,结构中的元素可称之为元素声明,生成的元素可称之为元素产物,以作区分。此外,不同次生成同一元素所基于的源代码文件的文件内容不同时,生成的产物均为该元素的元素产物,但分别对应于不同的版本。

[0066] 请参考上述步骤101和步骤102可知,一元素声明依赖于源代码文件时,若文件内容更新,则该元素声明预期改变,即该元素声明为预期改变的元素声明;若一元素声明预期改变,且存在依赖于该元素声明的任务,则该任务即为待执行任务,进而依赖于该待执行任务的元素声明预期改变。基于结构中各对象间的依赖关系或称生成关系,以此类推,即可自动确定出结构中的每一个待执行的任务。

[0067] 在本发明一个实施例中,除了如上所述的因源代码文件更改而引发的应用更新外,还可以存在因其他原因而引发的应用更新处理。比如,至少可以有如下几种原因:

[0068] 原因1:考虑到任务执行时有可能未执行成功,故可执行自检以引发应用更新处理;

[0069] 原因2:考虑到元素声明的元素声明文件的文件内容也有更改的可能,故这类文件内容更改时可引发应用更新处理;

[0070] 原因3:原因1和原因2均需考虑。

[0071] 详细地,对应于上述原因1:

[0072] 在本发明一个实施例中,所述任务执行判定规则还包括:一任务每一次成功执行所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容均不完全相同时,该任务为待执行任务。

[0073] 举例来说,假设一任务仅成功执行了一次,且所依赖的文件集X包括源代码文件1-1、源代码文件2-1、源代码文件3-1、源代码文件4-1。目标文件集包括源代码文件1-2、源代码文件2-1、源代码文件3-1、源代码文件4-1。“-”前的数字用于区分不同的源代码文件,且对于同一源代码文件,文件内容不同时,“-”后的数字不同。

[0074] 若该任务的执行需间接依赖于源代码文件1,则说明基于文件集X来执行该任务时生成的元素产物,与基于目标文件集来执行该任务时生成的元素产物不一致,当前不存在基于目标文件集来执行该任务时应生成的元素产物,但当前又需要使用这一元素产物来执行基于目标文件集的应用更新,如此,该任务即为待执行任务。反之,若该任务的执行并不间接依赖于源代码文件1,该任务即为无需执行的任务,如此,可使用上次执行所生成的元素产物,来执行基于目标文件集的应用更新。

[0075] 详细地,对应于上述原因2:

[0076] 在本发明一个实施例中,所述任务执行判定规则还包括:所述目标文件集涉及到一元素声明的元素声明文件的文件内容更新时,该元素声明的元素产物预期改变,该元素声明所依赖的任务为待执行任务。

[0077] 与源代码文件的文件内容更新相类似,元素声明文件的更新同样会引发应用更新处理。由于元素声明文件已更新,说明相应元素声明的元素产物的生成依据已改变,故需基于更新后的元素声明文件重新生成元素产物,即该元素声明的元素产物预期改变,进而,该元素声明所依赖的任务为待执行任务。

[0078] 详细地,对应于上述原因3:

[0079] 在本发明一个实施例中,所述任务执行判定规则还包括:一任务每一次成功执行所间接依赖的全部文件的文件内容,与所述目标文件集中相应文件的文件内容均不完全相同时,该任务为待执行任务;

[0080] 所述全部文件包括全部源代码文件和全部元素声明文件。

[0081] 原因3与上述原因1相类似但又有所不同,不同之处在于,原因1的应用场景并未考虑元素声明文件,而原因3需考虑元素声明文件。如此,在原因1的判断结果的基础之上,即便一任务某次成功执行所间接依赖的全部源代码文件的文件内容,与目标文件集中相应源代码文件的文件内容完全相同,仍不能保证该任务即无需执行,因为该任务这次成功执行所间接依赖的全部元素声明文件的文件内容,或许与目标文件集中相应元素声明文件的文件内容并不完全相同。如此,若在原因1下并未将该任务确定为待执行任务,则经针对元素声明文件的文件内容是否完全相同的判断,若判断为是,则该任务才可被确定为无需执行的任务,否则,该任务为待执行任务。

[0082] 在本发明一个实施例中,所述结构包括的元素声明有:依赖于服务编排任务的应用声明、依赖于镜像构建任务的镜像声明、依赖于制品构建任务的制品声明、依赖于至少一个源代码文件的源代码工程声明;

[0083] 所述结构包括的任务有:依赖于至少一个镜像声明的服务编排任务、依赖于至少一个制品声明的镜像构建任务、依赖于至少一个源代码工程声明的制品构建任务。

[0084] 举例来说,图2所示内容即可以为确定出的一应用的结构。这一结构中,最底层元素为源代码工程,其上一层元素为制品,制品的上一层元素为镜像,最顶层元素为应用。如此,可先根据源代码文件以生成各个源代码工程产物,再基于生成的源代码工程产物以生成各个制品产物,再基于生成的制品产物以生成各个镜像产物,最后基于生成的镜像产物以生成应用产物,从而完成应用生成。

[0085] 请参考图2,结构中的元素声明可以有应用声明、镜像声明、制品声明、源代码工程声明,结构中的任务可以有服务编排任务、镜像构建任务、制品构建任务。执行制品构建任务可生成一制品产物,执行镜像构建任务可生成一镜像产物,执行服务编排任务可生成一应用产物。

[0086] 基于上述内容,在本发明一个实施例中,所述根据所述目标文件集中的文件内容,以及结合任务执行判定规则,确定所述结构中的待执行任务,包括:

[0087] 将所述结构中的每一个制品声明依次作为当前制品声明,并执行:判断制品仓库中是否存储有对应于所述目标文件集的当前制品声明的制品产物,若否,进一步判断所述制品仓库中是否存储有一当前制品声明的制品产物,该制品产物对应的文件集中包括的、该当前制品声明所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容完全相同,若否,将当前制品声明所依赖的制品构建任务确定为待执行任务;

[0088] 将所述结构中的每一个镜像声明依次作为当前镜像声明,并执行:判断镜像仓库中是否存储有对应于所述目标文件集的当前镜像声明的镜像产物,若否,进一步判断所述镜像仓库中是否存储有一当前镜像声明的镜像产物,该镜像产物对应的文件集中包括的、该当前镜像声明所间接依赖的全部源代码文件的文件内容,与所述目标文件集中相应源代码文件的文件内容完全相同,若否,将当前镜像声明所依赖的制品构建任务确定为待执行任务。

[0089] 详细地,如果一个任务执行的生成产物,与该任务某一次执行的生成产物是无差异的,则该任务对于此次请求是不需要执行的。即若上述两个进一步判断操作的判断结果为是,则说明可以得出这一无差异结论。

[0090] 基于上述原因3和图3所示的应用的结构,举例来说,为生产应用,开发者会首次输入各个文件的文件内容。版本控制系统会生成包括这些文件内容的文件集,版本号为V1,由于为首次生成文件集,故可发出应用生成请求。之后,开发者输入源代码文件1的更新后的文件内容,版本控制系统会生成包括该文件内容的文件集,版本号为V2,并发出应用更新请求。再然后,开发者输入源代码文件2的更新后的文件内容,版本控制系统会生成包括该文件内容的文件集,版本号为V3,并发出应用更新请求。

[0091] 基于此,生成的各个文件集中包括的文件内容可以如下述表1所示。

[0092] 表1

文件集的版本号	文件集内的文件内容				
V1	Y1	Y2	S0	S1	S2
V2	Y1'	Y2	S0	S1	S2
V3	Y1'	Y2'	S0	S1	S2

[0093] 表1中,V1、V2、V3分别表示先后生成的3个文件集的版本号;Y1表示源代码文件1的文件内容;Y2表示源代码文件2的文件内容;S0、S1、S2分别表示镜像声明、制品声明1、制品声明2的元素声明文件的文件内容;文件内容的代表符号的上标有无以及上标类别,用于表示同一元素声明的不同文件内容。

[0094] 基于表1所述内容,在生成版本V1时,图3中的镜像构建任务、制品构建任务1、制品构建任务2均为待执行任务,执行制品构建任务会分别生成制品产物1V1、制品产物2V1,再基于这两个制品产物,执行镜像构建任务,会生成镜像产物V1。之后,即可基于镜像产物V1生成应用产物V1。

[0095] 在生成版本V2时,由于仅源代码文件1更新,故图3中的制品构建任务1为待执行任务,执行制品构建任务1会生成制品产物1V2,再基于制品产物1V2和制品产物2V1,执行镜像构建任务,会生成镜像产物V2。之后,即可基于镜像产物V2生成应用产物V2。

[0096] 在生成版本V3时,由于仅源代码文件2更新,故图3中的制品构建任务2为待执行任务,执行制品构建任务2会生成制品产物2V3,再基于制品产物1V2和制品产物2V3,执行镜像构建任务,会生成镜像产物V3。之后,即可基于镜像产物V3生成应用产物V3。

[0097] 此时,制品仓库和镜像仓库中的产物可以如下述表2所示。

[0099] 表2

镜像仓库	制品仓库	
镜像产物 V1	制品产物 1V1	制品产物 2V1
镜像产物 V2	制品产物 1V2	制品产物 2V3
镜像产物 V3		

[0101] 在表1所示内容的基础之上,至少可以存在下述情况中的任意一种。

[0102] 情况1:源代码文件更新引发的应用更新处理。

[0103] 开发者期望改变源代码文件1的文件内容,并输入最新的文件内容。如此,版本控制系统会生成新的文件版本V4-1,具体内容如下述表3所示。

[0104] 表3

文件集的版本号	文件集内的文件内容				
V1	Y1	Y2	S0	S1	S2
V2	Y1'	Y2	S0	S1	S2
V3	Y1'	Y2'	S0	S1	S2
V4-1	Y1''	Y2'	S0	S1	S2

[0106] 表3中,Y1''表示开发者输入的这一文件内容。

[0107] 在生成版本V4-1时,由于仅源代码文件1更新,故图3中的制品构建任务1为待执行任务,执行制品构建任务1会生成制品产物1V4-1,再基于制品产物1V4-1和制品产物2V3,执行镜像构建任务,会生成镜像产物V4-1。之后,即可基于镜像产物V4-1生成应用产物V4-1。

[0108] 确定待执行任务的一种具体实现过程可以为:首先判断制品声明1,由于制品仓库中未存储有对应于V4-1的制品产物1,且存储的各个制品产物1所依赖的源代码文件1的文件内容,均与V4-1中源代码文件1的文件内容不同,故制品声明1所依赖的制品构建任务1为待执行任务。基于相同或相类似的实现原理,可判断出制品构建任务2不为待执行任务,镜像构建任务为待执行任务。可见,这一判断结果与上述分析描述相一致。

[0109] 情况2:元素声明文件更新引发的应用更新处理。

[0110] 开发者期望改变镜像声明文件的文件内容,并输入最新的文件内容。如此,版本控制系统会生成新的文件版本V4-2,具体内容如下述表4所示。

[0111] 表4

[0112]	文件集的版本号	文件集内的文件内容				
	V1	Y1	Y2	S0	S1	S2
V2	Y1'	Y2	S0	S1	S2	
V3	Y1'	Y2'	S0	S1	S2	
V4-2	Y1'	Y2'	S0'	S1	S2	

[0113] 表4中,S0'表示开发者输入的这一文件内容。

[0114] 在生成版本V4-2时,由于仅镜像声明文件更新,故图3中的镜像构建任务为待执行任务,执行镜像构建任务会生成镜像产物V4-2。之后,即可基于镜像产物V4-2生成应用产物V4-2。

[0115] 情况3:日常自检引发的应用更新处理。

[0116] 开发者期望系统自检各个任务是否成功执行,故可按需设置自检执行时间。此时,版本控制系统中的当前版本号为V3,即针对V3实现应用更新处理。

[0117] 由于需执行自检,故可依次判断各个制品声明和各个镜像声明,以确定结构中的各个任务是否为待执行任务。

[0118] 举例来说,若基于V3执行应用更新处理时,两制品构建任务均执行成功,而镜像构建任务未执行成功,故请参考表2,镜像仓库中将仅有镜像产物V1和镜像产物V2,而不存在镜像产物V3。

[0119] 如此,判断各个制品声明后,可确定两制品构建任务均无需执行。

[0120] 判断镜像声明时,首先判断镜像仓库中是否存储有对应于V3的镜像产物,可知判断结果为否。然后,进一步判断镜像仓库中的各个镜像产物,其所间接依赖的源代码文件1和源代码文件2的文件内容,是否与V3中相应源代码文件的文件内容完全相同,可知判断结果仍为否。如此,可以确定镜像构建任务为待执行任务。

[0121] 情况4:版本回滚引发的应用更新处理。

[0122] 开发者发现应用产物的一已有版本比现有版本更为优选,故可请求版本回滚。此时,版本控制系统中的当前版本号为V3,而开发者任务V2更好,故可将当前版本号改为V2。如此,可针对V2实现应用更新处理。

[0123] 由于版本回滚,故可依次判断各个制品声明和各个镜像声明,以确定结构中的各个任务是否为待执行任务。

[0124] 确定待执行任务的一种具体实现过程可以为:首先判断制品声明1,由于制品仓库中存储有对应于V2的制品产物1,故制品构建任务1不为待执行任务。然后判断制品声明2,由于制品仓库中虽然未存储有对应于V2的制品产物2,但经判断,制品仓库中存储的对应于V1的制品产物2,与对应于V2的制品产物2为同一制品产物,故制品构建任务2不为待执行任务。最后,判断镜像声明,由于镜像仓库中存储有对应于V2的镜像产物,故镜像构建任务不为待执行任务。如此,即可基于已有的这一对应于V2的镜像产物,经执行服务编排任务,以再次生成针对V2的应用产物。对应地,可以确定服务编排任务为待执行任务。

[0125] 基于上述内容,优选地,在本发明一个实施例中,接收到的应用更新请求可以标识

有请求类型。

[0126] 详细地,接收到的应用更新请求可以标识有请求类型,比如是涉及到源代码文件的文件内容更新,还是涉及到元素声明文件的文件内容更新,或是日常自检,或是版本回滚,甚至可以是这些类型的部分结合,以使应用更新处理系统可以基于不同的请求类型,来确定待执行任务。

[0127] 比如,涉及到文件更新时,可基于所更新的文件,找出预期改变的元素声明、待执行的任務,而无需逐一分析各个制品声明和各个镜像声明。

[0128] 再比如,涉及到日常自检、版本回滚时,可逐一分析各个制品声明和各个镜像声明,以确定出每一个待执行任务。

[0129] 当然,接收到的应用更新请求未标识有请求类型时,无需区分,可直接逐一分析各个制品声明和各个镜像声明,以确定出每一个待执行任务。比如,版本控制系统在确定出当前版本号变化时,可将当前版本号发送给应用更新处理系统,以提出应用更新请求,但未说明请求类型。此外,自检操作通常也是针对当前版本号的自检。

[0130] 详细地,新生成一版本号时,该版本号为当前版本号,直至生成下一个版本号。版本回滚时,回滚后的版本号为当前版本号。

[0131] 比如,在上述情况1下,发送的版本号为V4-1;在上述情况2下,发送的版本号为V4-2;在上述情况3下,发送的版本号为V3;在上述情况4下,发送的版本号为V2。

[0132] 综上所述,执行应用更新处理的这一集成和部署服务,可以根据应用的结构,对应用生命周期中所涉及到的各个组成要素所发生的改变进行判定,以决定结构中的各个任务是否需要执行。这种增量式的方法使得集成和部署的耗时减少,开发者的工作效率得到提高。

[0133] 如图4所示,本发明一个实施例提供了一种执行上述任一所述应用更新处理方法的调度单元,可以包括:

[0134] 解析子单元401,用于解析应用的结构,其中,所述结构包括:源代码文件、依赖于至少一个源代码文件的元素声明、依赖于至少一个元素声明的任务、依赖于任务的元素声明;

[0135] 处理子单元402,用于在接收到针对一目标文件集的应用更新请求时,根据所述目标文件集中的文件内容,以及结合任务执行判定规则,确定所述结构中的待执行任务;

[0136] 输出子单元403,用于将全部待执行任务输出给外部的运行单元;

[0137] 其中,所述任务执行判定规则包括:

[0138] 所述目标文件集涉及到一源代码文件的文件内容更新时,相应源代码文件为预期改变的源代码文件;

[0139] 一元素声明依赖有待执行的任务或至少一个预期改变的源代码文件时,该元素声明的元素产物预期改变;

[0140] 一任务依赖有至少一个元素产物预期改变的元素声明时,该任务为待执行任务。

[0141] 上述单元内的各子单元之间的信息交互、执行过程等内容,由于与本发明方法实施例基于同一构思,具体内容可参见本发明方法实施例中的叙述,此处不再赘述。

[0142] 如图5所示,本发明一个实施例提供了一种应用更新处理系统,可以包括:配置单元501、上述调度单元502、运行单元503;

[0143] 其中,所述配置单元501,用于在接收到外部输入的应用的配置信息时,解析所述配置信息,以确定所述应用的结构并发送给所述调度单元502;

[0144] 所述运行单元503,用于在接收到所述调度单元502发来的全部待执行任务时,执行所述全部待执行任务中的制品构建任务和镜像构建任务,以生成至少一个镜像产物;将生成的全部镜像产物存储至外部的镜像仓库;将所述全部待执行任务中的服务编排任务发送给外部的服务编排系统,以触发所述服务编排系统根据所述全部镜像产物,通过执行接收到的服务编排任务以更新应用产物。

[0145] 详细地,开发者可以将编写好的配置信息录入配置单元,并由配置单元解析这一配置信息,以确定应用的结构。

[0146] 本发明实施例中,应用生成处理系统中的运行单元仅执行结构中的制品构建任务和镜像构建任务,并将服务编排任务交由现有的服务编排系统来执行,共同完成应用生成的过程。

[0147] 在本发明一个实施例中,所述运行单元503,用于对于所述全部待执行任务中的每一个制品构建任务,根据所述目标文件集中包括的、该制品构建任务所间接依赖的全部源代码文件的文件内容,执行该制品构建任务,以生成对应于所述目标文件集的制品产物,并将生成的制品产物存储至制品仓库中;以及基于生成的制品产物,执行所述全部待执行任务中的每一个镜像构建任务,以生成对应于所述目标文件集的镜像产物,并将生成的镜像产物存储至镜像仓库中。

[0148] 上述系统内的各单元之间的信息交互、执行过程等内容,由于与本发明方法实施例基于同一构思,具体内容可参见本发明方法实施例中的叙述,此处不再赘述。

[0149] 如图6所示,本发明一个实施例提供了一种应用更新系统,可以包括:

[0150] 上述任一所述的应用更新处理系统601、镜像仓库602、服务编排系统603、客户端开发工具604、版本控制系统605;

[0151] 其中,所述镜像仓库602,用于存储所述应用更新处理系统601发来的全部镜像产物;

[0152] 所述服务编排系统603,用于经所述应用更新处理系统601触发,根据所述全部镜像产物,执行所述应用更新处理系统601发来的服务编排任务,以更新应用产物;

[0153] 所述客户端开发工具604,用于缓存外部输入的至少一个目标源代码文件的文件内容;在接收到外部输入的应用更新请求时,将所述至少一个目标源代码文件的文件内容发送给所述版本控制系统605;

[0154] 所述版本控制系统605,用于在接收到所述至少一个目标源代码文件的文件内容时,生成当前文件集,并向所述应用更新处理系统发送应用更新请求;

[0155] 其中,所述当前文件集中,任一所述目标源代码文件的文件内容为接收到的文件内容,任一其他文件的文件内容为生成的上一个文件集中该其他文件的文件内容,且不同文件集的版本号不同。

[0156] 本发明实施例中,开发者可经客户端开发工具以录入文件内容,录入的文件内容由客户端开发工具发送至版本控制系统。版本控制系统在收到文件内容时,即可生成一包括这些文件内容的文件集。一旦新生成有文件集,即可通知应用更新处理系统做相应处理。

[0157] 综上所述,本发明的实施例具有至少如下有益效果:

[0158] 1、本发明实施例中,解析应用的结构,该结构包括源代码文件、依赖于若干源代码文件的元素声明、依赖于若干元素声明的任务、依赖于一任务的元素声明;接收针对一文件集的应用更新请求,根据其中的文件内容并结合任务执行判定规则,确定结构中的待执行任务;将全部待执行任务输出给外部运行单元;规则包括:文件集涉及到一源代码文件的文件内容更新时,相应源代码文件预期改变;一元素声明依赖有待执行的任务或预期改变的源代码文件时,该元素声明的元素产物预期改变;一任务依赖有元素产物预期改变的元素声明时,该任务为待执行任务。本发明实施例可自动识别待更新对象,故可提高应用更新效率。

[0159] 2、本发明实施例中,执行应用更新处理的这一集成和部署服务,可以根据应用的结构,对应用生命周期中所涉及到的各个组成要素所发生的改变进行判定,以决定结构中的各个任务是否需要执行。这种增量式的方法使得集成和部署的耗时减少,开发者的工作效率得到提高。

[0160] 需要说明的是,在本文中,诸如第一和第二之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来,而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个·····”限定的要素,并不排除在包括所述要素的过程、方法、物品或者设备中还存在另外的相同因素。

[0161] 本领域普通技术人员可以理解:实现上述方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成,前述的程序可以存储在计算机可读取的存储介质中,该程序在执行时,执行包括上述方法实施例的步骤;而前述的存储介质包括:ROM、RAM、磁碟或者光盘等各种可以存储程序代码的介质中。

[0162] 最后需要说明的是:以上所述仅为本发明的较佳实施例,仅用于说明本发明的技术方案,并非用于限定本发明的保护范围。凡在本发明的精神和原则之内所做的任何修改、等同替换、改进等,均包含在本发明的保护范围内。

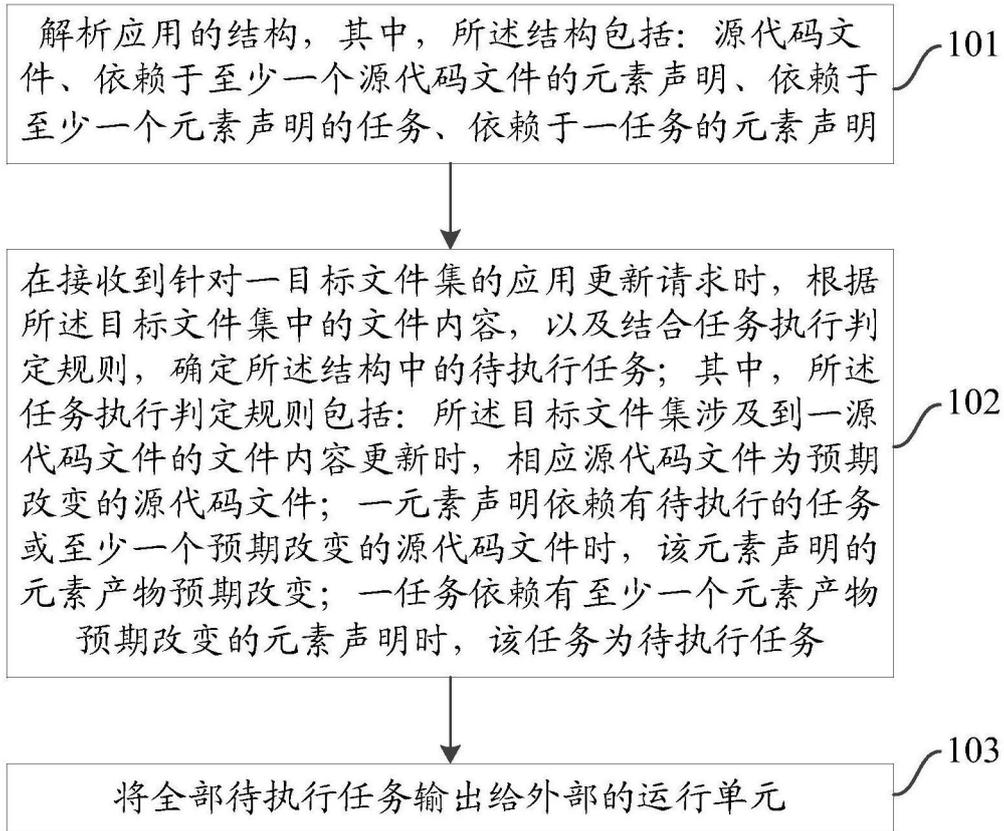


图1

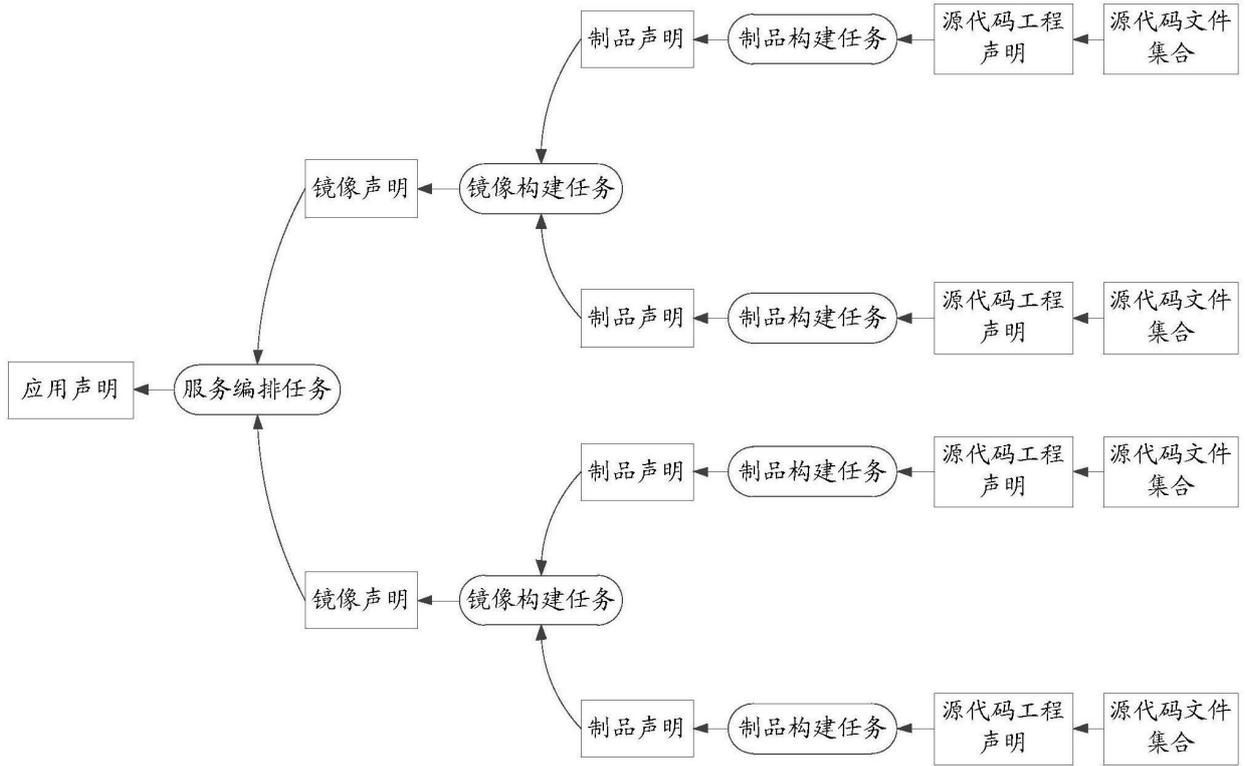


图2

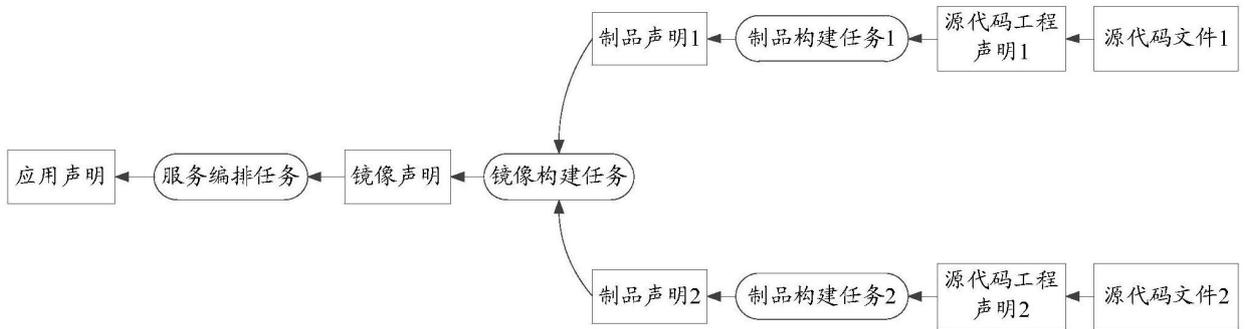


图3

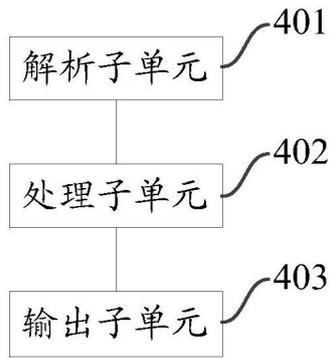


图4

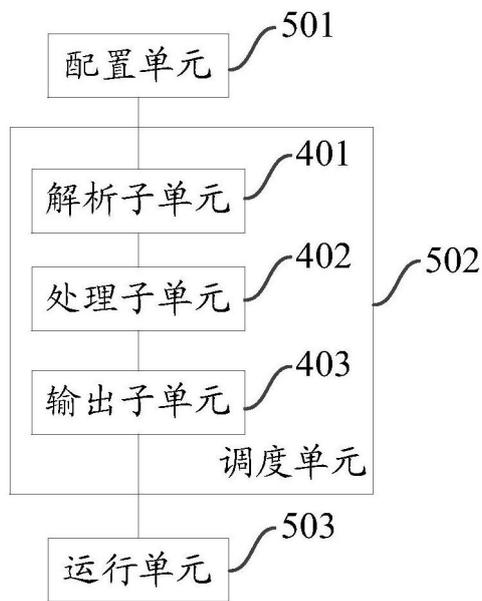


图5

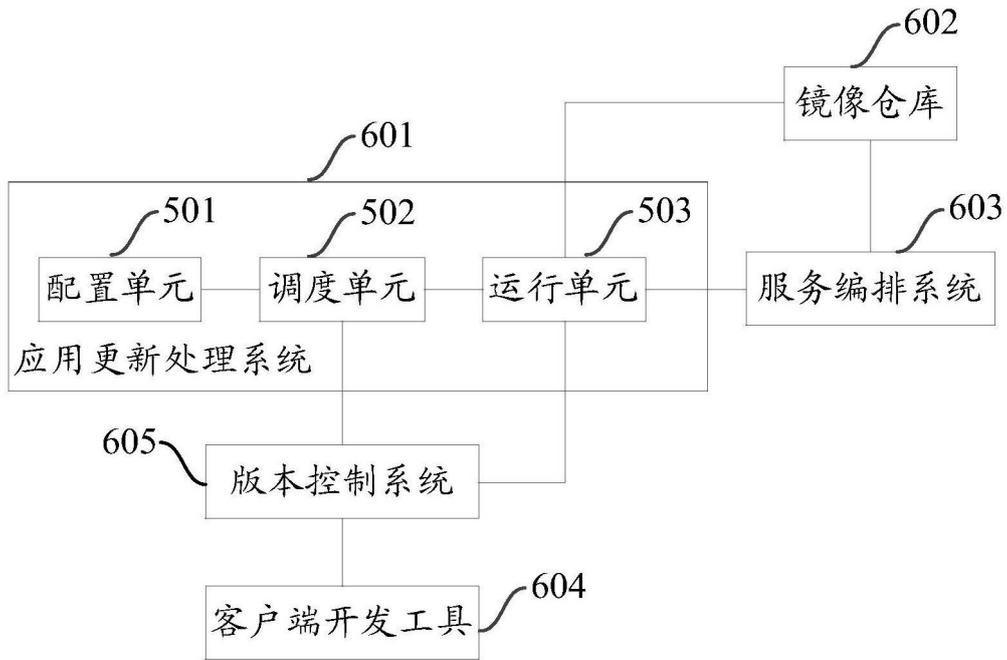


图6