

云智未来^{9th}

第九届中国系统架构师大会
SYSTEM ARCHITECT CONFERENCE CHINA 2017

Android多分支代码自动同步

魅族 段启智

2017.10.20

目录

1

维护多个分支的原因

2

分支合并的方法比较

3

分支自动合并的策略

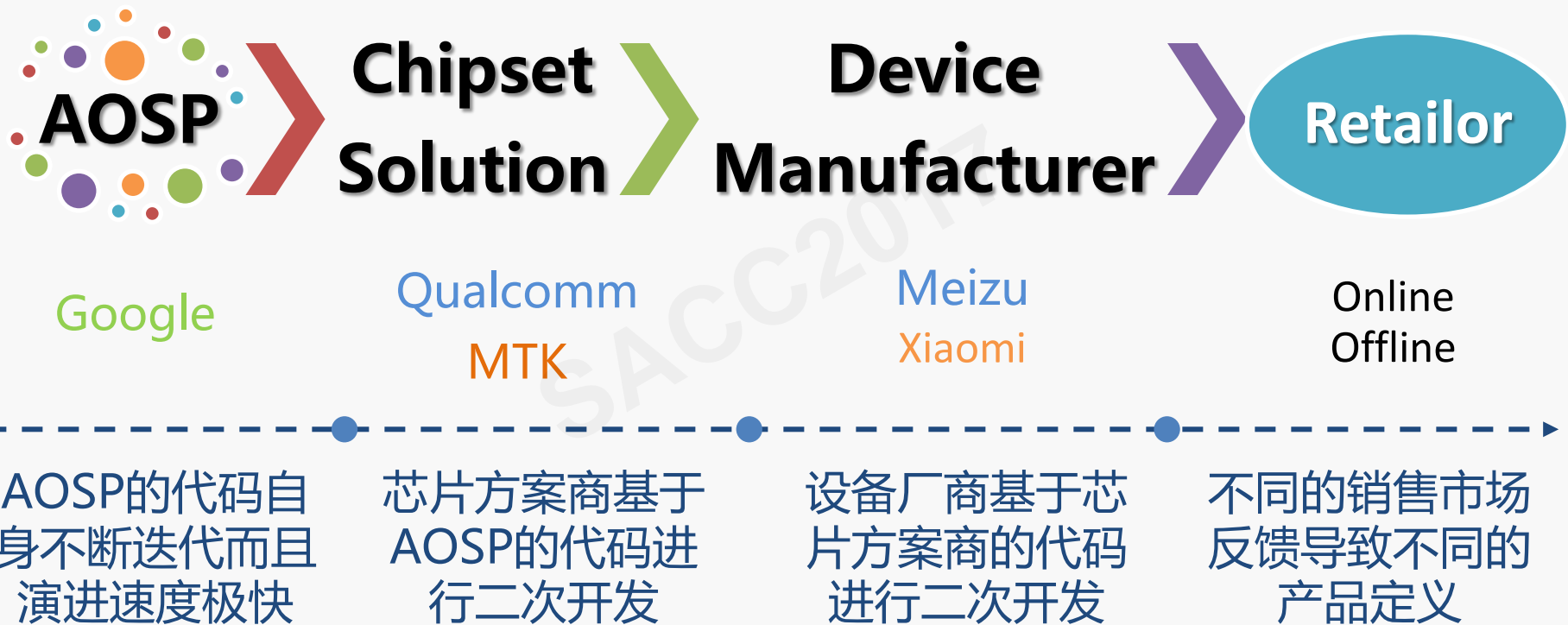
4

分支自动合并的实践

5

总结与展望

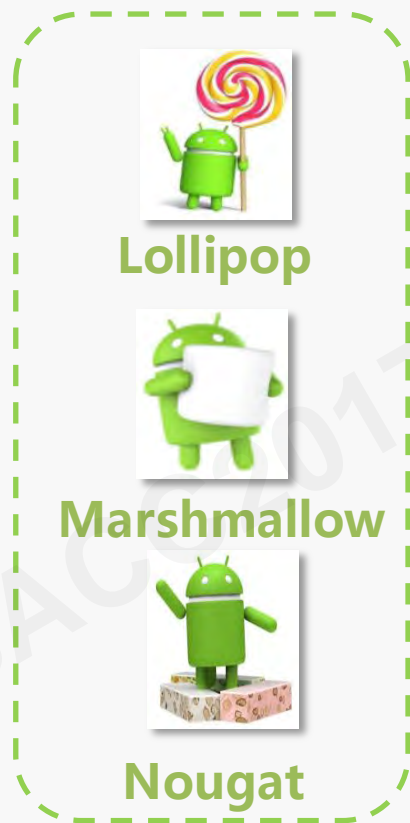
1. 维护多个分支的原因



1. 维护多个分支的原因



不同芯片平台



不同安卓版本



不同市场定制

1. 维护多个分支的原因

- ✓ 作为一个Android设备厂商，大都基于上游的芯片方案进行二次开发，而下游又有不同的市场销售诉求，导致不得不维护多个分支
- ✓ 诸如小米、华为、魅族等国内的手机厂商，同时在维护的机型多达数十款，意味着不得不维护大量的分支
- ✓ 多分支维护是设备厂商的噩梦，需重复投入大量的开发、测试、发布、运营等项目资源

代码宏开关

运行时属性开关

运行时反射

中间件技术

编译开关

多分支之间进行代码合并

设备厂商想尽办法减少分支

一个芯片平台尽量适配更多的机型

优秀的代码设计

Gradle Flavor

Gradle Build Types

机型退出机制，避免维护多版本

目录

1

维护多个分支的原因

2

分支合并的方法比较

3

分支自动合并的策略

4

分支自动合并的实践

5

总结与展望

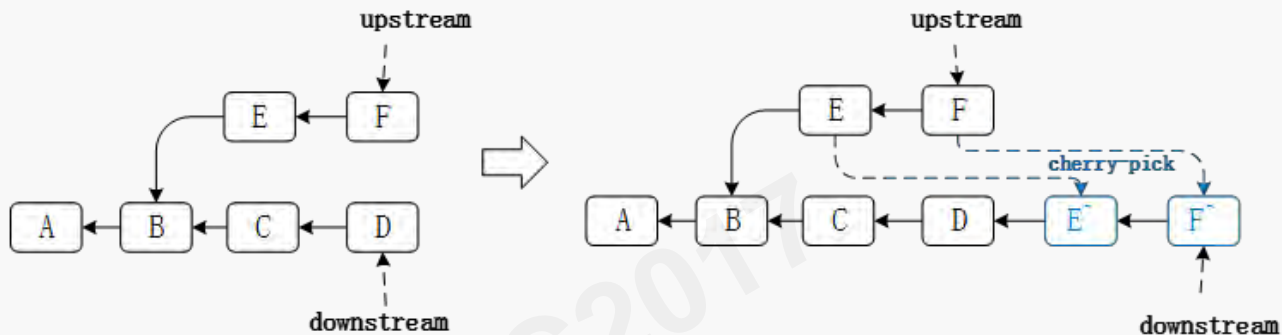
2. 分支合并的方法比较

- ✓ 手工通过对比工具进行代码合并
- ✓ git cherry-pick
- ✓ git rebase
- ✓ git merge

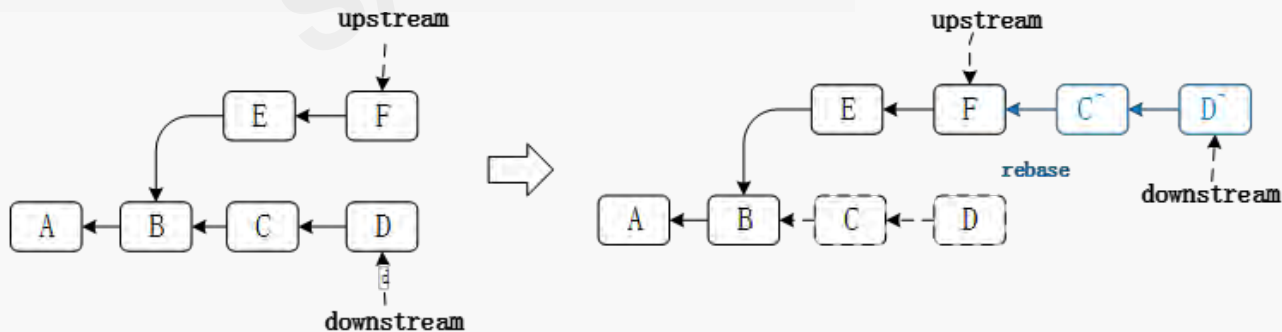
SACC2017

2. 分支合并的方法比较

```
$ git cherry-pick E,F
```

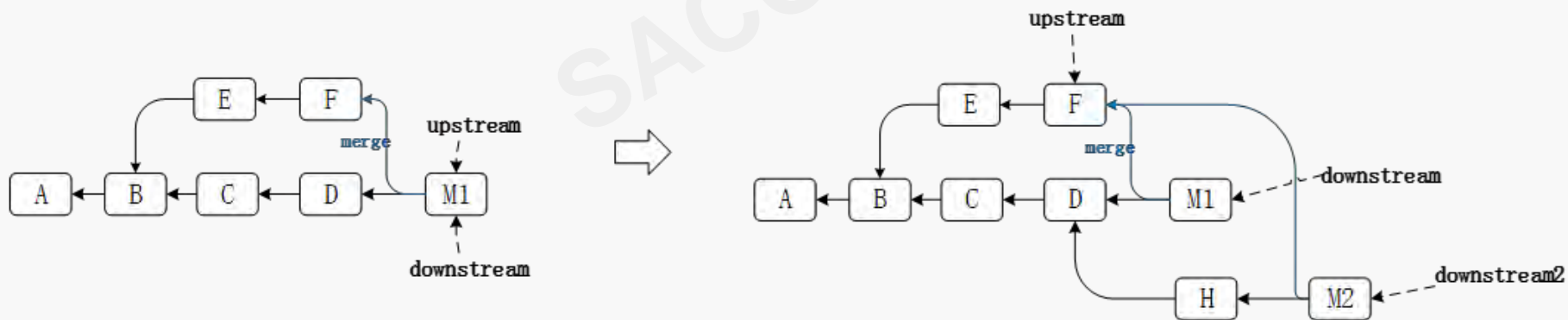
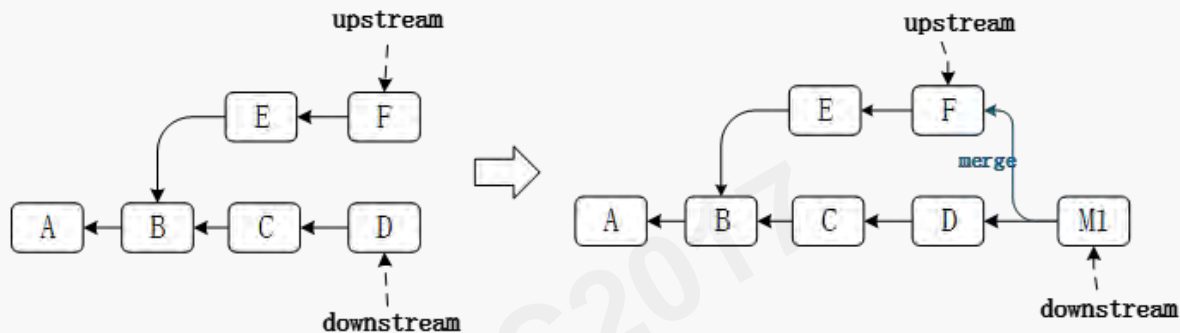


```
$ git rebase upstream
```



2. 分支合并的方法比较

`$ git merge upstream`



利用git-rerere机制自动解决冲突

2. 分支合并的方法比较

分支合并方式	不产生新提交	没有遗漏提交	容易解决冲突
cherry-pick	×	×	×
rebase	×	√	×
merge	√	√	√

与cherry-pick不同，merge一次性合并所有的提交，无法对提交进行精准选择。

2. 分支合并的方法比较

17cc44f	Import translations. DO NOT MERGE by Baligh Uddin - 10 months ago
32a22c4	Recover apps with malformed certificates. by Jeff Sharkey - 10 months ago android-5.0.0_r7
c482a9d	Increasing the min swipe thresholds for the nav bar gestures. (Bug 17109581) by Winson Chung - 10 months ago android-5.0.0_r5 android-5.0.0_r5.1 android-5.0.0_r6
5a150e4	[DO NOT MERGE] Increase min free levels for cached processes on 64bit by Andres Morales - 10 months ago
a14d782	DO NOT MERGE: Don't log passwords returned from vdc by Paul Lawrence - 10 months ago android-5.0.0_r4
b2ef7bf	merge in Imp-release history after reset to Imp-dev by The Android Automerger - 10 months ago
38c2eae	DO NOT MERGE: Don't log passwords returned from vdc by Paul Lawrence - 10 months ago android-5.0.0_r3
ba35a77	Add missing passive discovery flag by Jason Monk - 10 months ago android-5.0.0_r1 android-5.0.0_r2

AOSP官方负责人Jean-Baptiste Queru:

We (Google) routinely develop on multiple branches at the same time. In order to make sure that the later branch (e.g. ics-mr1) contains all the new features and bugfixes developed in an older branch (e.g. ics-mr0), we have a server that automatically takes every commit made in ics-mr0 and merges it into ics-mr1. However, sometimes the engineer making a change in ics-mr0 knows that this change doesn't apply to ics-mr1, e.g. because a similar issue was fixed differently in ics-mr1 and the fix from ics-mr0 wasn't necessary. In that case, the engineer includes the words "do not merge" in their change description, and the auto-merger performs a "git merge -s ours" instead of "git merge" when handling that change. There's a bit more complexity involved, but that's the high-level view.

实践中，采用merge合并分支能够快速进行代码移植

目录

1

维护多个分支的原因

2

分支合并的方法比较

3

分支自动合并的策略

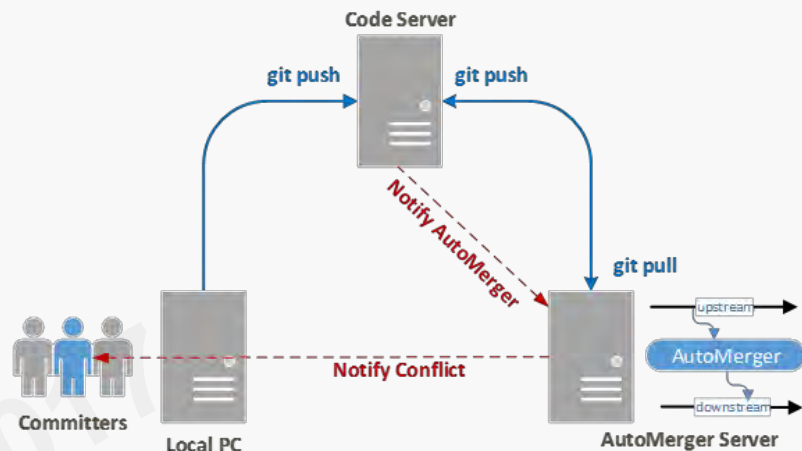
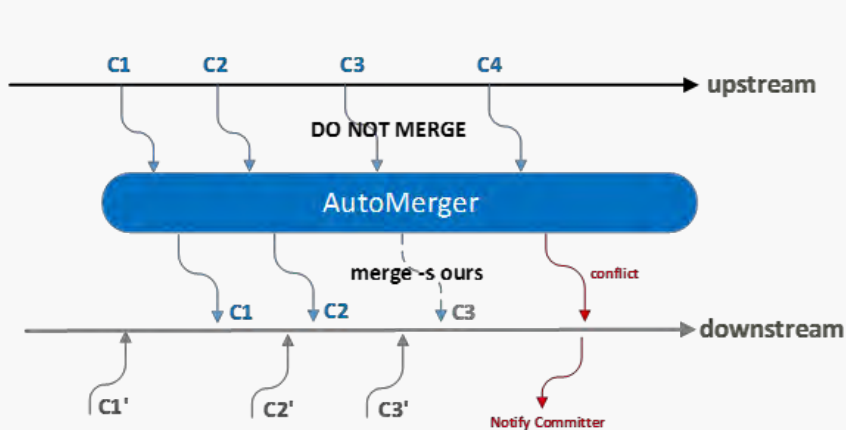
4

分支自动合并的实践

5

总结与展望

3. 分支自动合并的策略

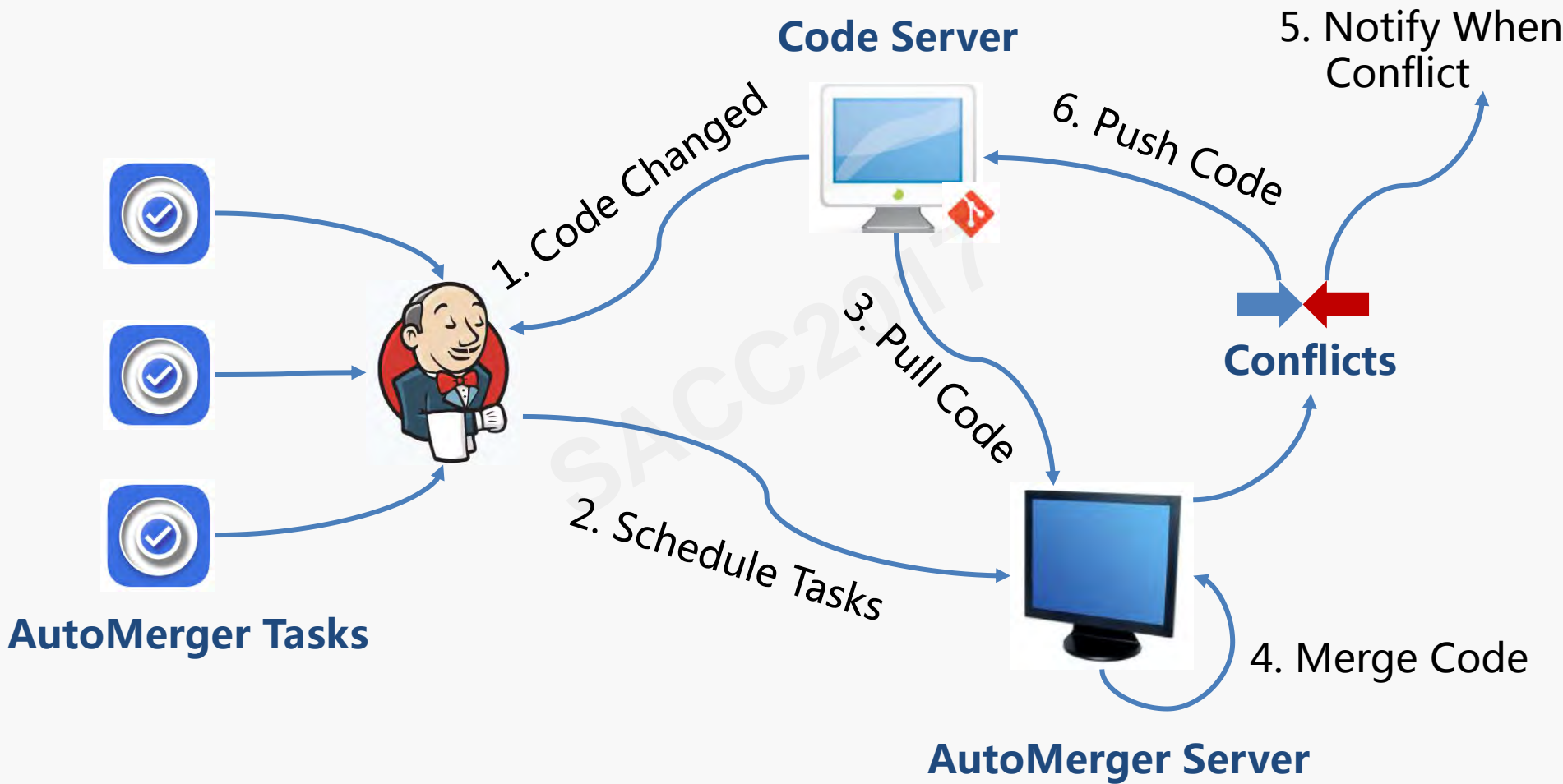


1. 对于上游分支的每一次提交，都会自动触发 AutoMerger 的执行，正常提交会采用 merge 的方式合并到下游分支；
2. 如果某个提交注明不需要合并，则通过关键字判断(DO NOT MERGER)，采用 merge -s ours 的方式合并到下游分支。
3. 如果在合并过程中发生了冲突，则需要通知提交人解决冲突。

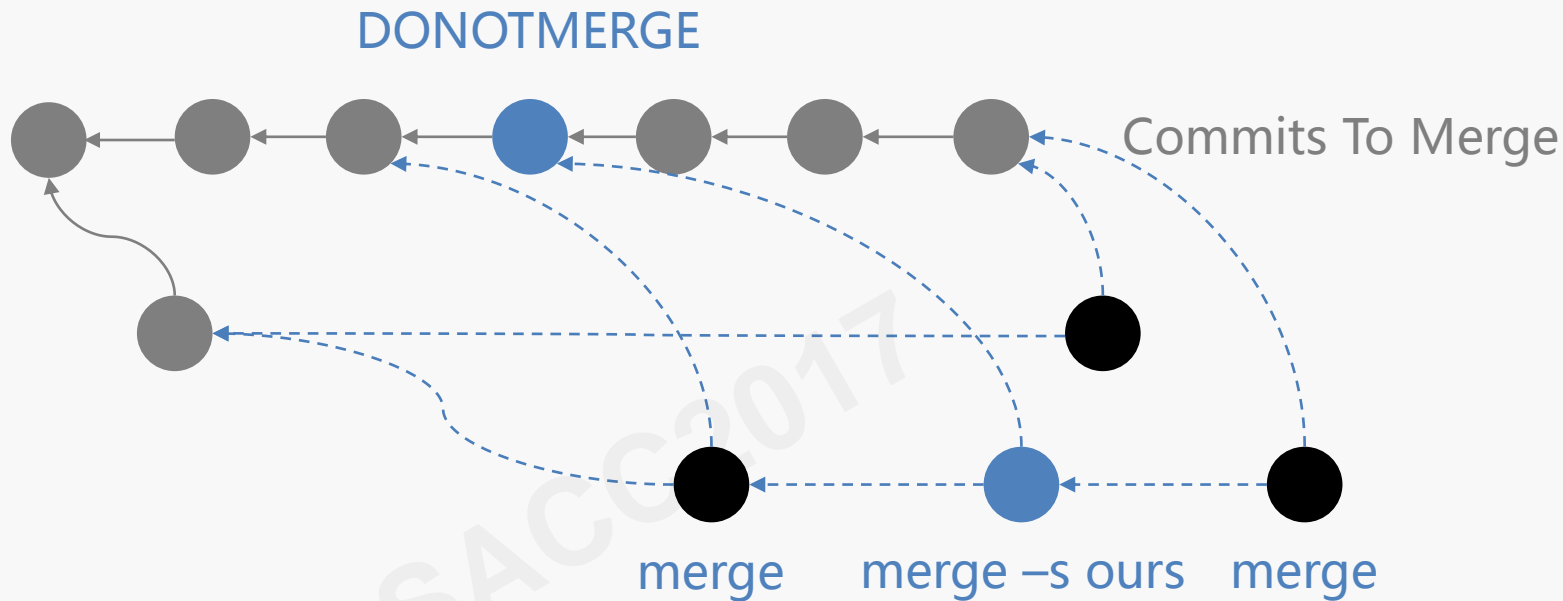
在一个服务器上部署 AutoMerger，该服务器具备以下能力：

1. 接收从代码服务器发出的代码合并通知；
2. 与代码服务器进行代码同步；
3. 通知冲突责任人解决冲突。

3. 分支自动合并的策略



3. 分支自动合并的策略



遇到不需要Merge到下游的提交，则对待Merge的提交进行分割，多次Merge

目录

1

维护多个分支的原因

2

分支合并的方法比较

3

分支自动合并的策略

4

分支自动合并的实践

5

总结与展望

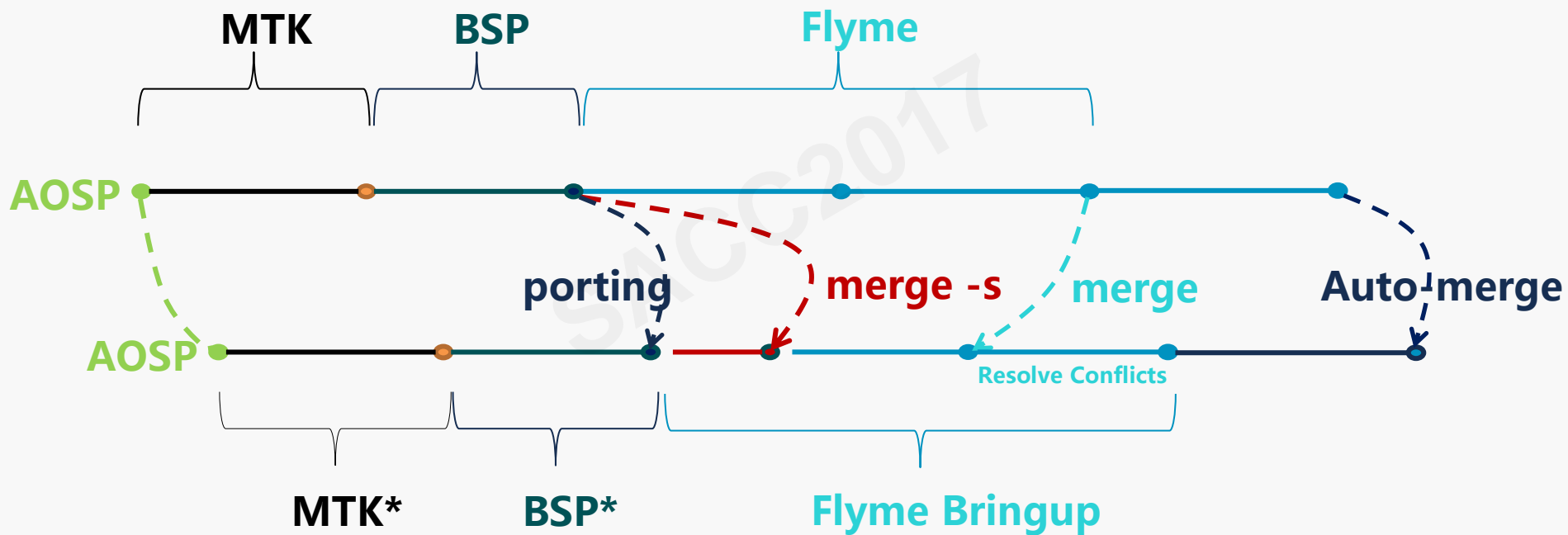
4. 分支自动合并的实践

新机型移植 ~~120人, 60天~~
1人, 7天

旧机型维护 ~~同时维护2款~~
同时维护 24款

4. 分支自动合并的实践

新机型移植



两次Merge + 工具化
极大的减少了新机型移植的冲突

4. 分支自动合并的实践

旧机型维护：分支同步方向可配置

□62	platform/frameworks/base	自删除
M86_flyme5	» F516-I1_base	自删除
M85-I1_base	» L91-I1_base	自删除
M85-I1_base	» M71-I1_base	自删除
M85-I1_base	» M75-I1_base	自删除
M86_flyme5	» M76-I1_base	自删除
M85-I1_base	» M81-I1_base	自删除
M85-I1_base	» M86_flyme5	自删除
M85-I1_base	» M88-I1_base	自删除
M85-I1_base	» M88-I1_intl	自删除
M85-I1_base	» M91-I1_base	自删除
SS8890-m_base	» odc/JP01-m_base	自删除
M85-I1_base	» odc/M98-I1_base	自删除

+添加分支

```
$ git automerger
M85-I1_base
├── M71-I1_base
├── M86_flyme5
├── M76-I1_base
│   └── M81-I1_base
│       └── M88-I1_base
├── M91-I1_base
├── M98-I1_base
├── L91-I1_base
└── M75-I1_base
```

通过界面配置分支同步的上下游关系，可树状展示

4. 分支自动合并的实践

旧机型维护：同步冲突邮件可配置

发邮件配置

汇总报告接收列表

tangliuxiang@meizu.com	删除
------------------------	----

+添加邮箱

提交冲突接收列表

zhoushengmeng@meizu.com	删除
sunqingqing@meizu.com	删除
zhongjian@meizu.com	删除

+添加邮箱

[AutoMerger] 请解决代码冲突 app/Shell/AlphaMe: flyme6_base >> flyme6_base_intl

bringup@meizu.com
发给 yuxuehai@meizu.com, chengbiao@meizu.com, kangweidan@meizu.com, zhaowenjie@meizu.com, shaowei@meizu.com, zuojie@meizu.com, shilian@meizu.com, duanqizhi@... 2017-06-11 01:30 隐藏信息

发件人: bringup@meizu.com <bringup@meizu.com>
收件人: yuxuehai@meizu.com <yuxuehai@meizu.com>, chengbiao@meizu.com <chengbiao@meizu.com>, kangweidan@meizu.com <kangweidan@meizu.com>, zhaowenjie@meizu.com <zhaowenjie@meizu.com>, shaowei@meizu.com <shaowei@meizu.com>, zuojie@meizu.com <zuojie@meizu.com>, shilian@meizu.com <shilian@meizu.com>, duanqizhi@meizu.com <duanqizhi@meizu.com>
时间: 2017年6月11日 (周日) 01:30
大小: 7 KB

lxh@meizu.com, 阁下的提交: 6b7b375 [系统研发部][OneMind] #545925将应用名称 AlphaMe 改成 One Mind, 版本号升级来1.2.1 导致 app/Shell/AlphaMe: flyme6_base >> flyme6_base_intl 代码流冲突。
请严格按照如下方式解决冲突, 否则会阻塞其他收件人的代码流!

```
[app/Shell/AlphaMe]
$ git fetch meizuapp
$ git checkout -b flyme6_base_intl meizuapp/flyme6_base_intl
$ git merge 6b7b375
$ [ 本地解决冲突 ]
$ git push meizuapp HEAD:refs/for/flyme6_base_intl
```

被阻塞的提交:

lxh@meizu.com 6b7b375 [系统研发部][OneMind] #545925将应用名称 AlphaMe 改成 One Mind, 版本号升级来1.2.1

冲突内容:

```
diff --cc app/gradle.properties
index 701261a..c3a4c42..0000000
--- a/app/gradle.properties
```

产生代码合并冲突时，提交人会实时收到邮件提醒

4. 分支自动合并的实践

旧机型维护：冲突解决在线化

OTHER — Exchange (26336 封邮件, 13786 封未读)

收件人: 陈建波 抄送: 周圣孟
[AutoMerger] 请解决代码流冲突 platform/packages/apps/Settings: M1792_NF6_base >> M1712_NF6_base

chenjianbo@meizu.com, 阁下的提交:b4f864f [CoreSystem][HMI]: 将M80系统应用卸载恢复功能移植到1792 Redmine : #null Details : 由于目前PackageManager目前暂时未合入卸载应用功能到1792上, 所以暂时先注释掉相关导致 platform/packages/apps/Settings: M1792_NF6_base >> M1712_NF6_base 代码流冲突。
请严格按照如下方式解决冲突, 否则会阻塞其他收件人的代码流!

方式一:

```
[platform/packages/apps/Settings]
$ git remote add meizu ssh://chenjianbo@review.rnd.meizu.com:29999/platform/packages/apps/Settings
$ git fetch meizu
$ git remote add odc ssh://chenjianbo@odc.review.meizu.com:29418/platform/packages/apps/Settings
$ git fetch odc
$ git checkout -b M1712_NF6_base odc/M1712_NF6_base
$ git merge b4f864f
$ [本地解决冲突]
$ git push odc HEAD:refs/for/M1712_NF6_base
```

说明: 以上命令序列中, 需要添加remote的url, 因为这是跨服务器的代码流, 所以需要先将不同的服务器地址添加到本地。
如果remote已经存在, 则忽略git remote add的操作即可。

方式二:

```
[platform/packages/apps/Settings]
您还可以尝试我们的在线冲突工具, 方便快捷:
点这里解决冲突吧
注意事项:
1. 本工具只能支持最新版本的chrome和firefox浏览器, 不支持IE, sarify等
2. 解决完冲突后请关闭网页, 否则会维护一个长连接, 占用服务器资源!!!
3. 该工具目前处于内测阶段, 请大家有问题及时反馈给zhouhengmeng@meizu.com
4. 这里有个在线解决冲突教程
5. 目前只有魅族内部人员可以使用该工具
```

被阻塞的提交:

```
chenjianbo@meizu.com b4f864f [CoreSystem][HMI]: 将M80系统应用卸载恢复功能移植到1792 Redmine : #null Details : 由于目前PackageManager目前暂时未合入卸载应用功能到1792上, 所以暂时先注释掉N
```

冲突内容:

```
diff --cc res_mz/values-zh-rTW/mz_strings_flyme6.xml
index 08e4237,8b12945..0000000
--- a/res_mz/values-zh-rTW/mz_strings_flyme6.xml
+++ b/res_mz/values-zh-rTW/mz_strings_flyme6.terms
```


同步策略可配，在线解决冲突
多个分支维护成本大为降低

目录

1

维护多个分支的原因

2

分支合并的方法比较

3

分支自动合并的策略

4

分支自动合并的实践

5

总结与展望

5. 总结与展望

- ✓ 多分支的代码自动同步方案包含：
 - ✓ 多分支合并的方法
 - ✓ 新机型移植的方法
 - ✓ 旧机型维护的方法
- ✓ 整套方案已经健康运转2年，是魅族内部的分支管理的技术支撑，多达30个分支在同时维护
- ✓ 本方案涉及到虽然技术细节较多，从前端到后端，从分支策略到代码管理，但可以迅速落地到其他设备厂商
- ✓ 计划开源

THANKS

The background features a dark, almost black, space filled with numerous small, bright blue particles. These particles are arranged in several distinct, curved paths that sweep across the frame from the bottom left towards the top right. A bright, white-to-blue gradient light source is positioned behind the word 'THANKS', creating a lens flare effect that illuminates the surrounding particles and the text itself.