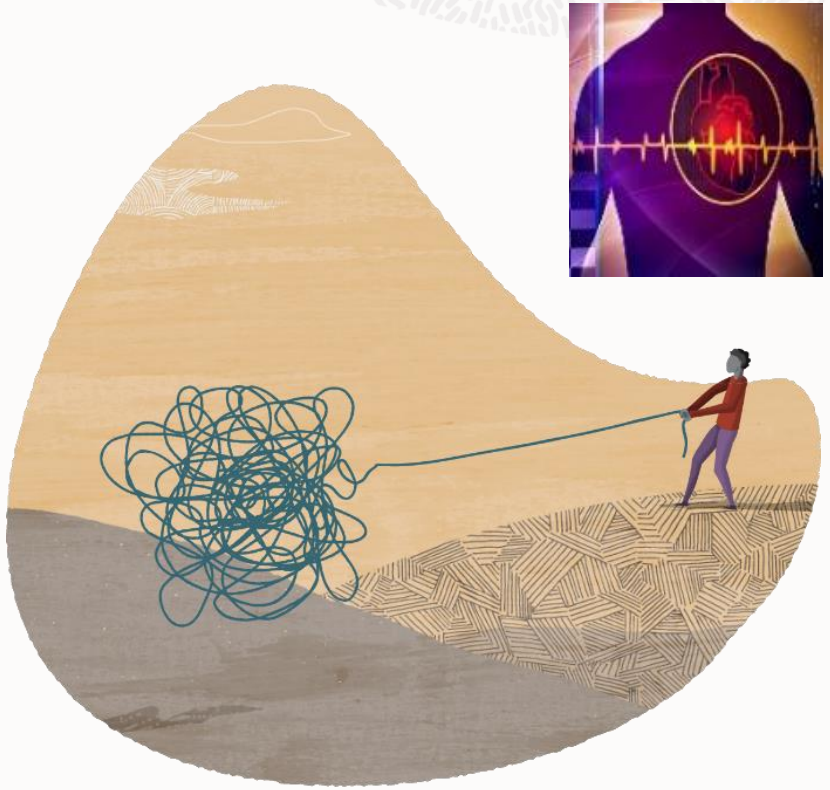
ORACLE

远距离容灾场景解决方案



赵靖宇（Alfred Zhao）

2022年7月22日

从一个金融客户A的迫切需求谈起

如何真正实现零数据丢失的目标?

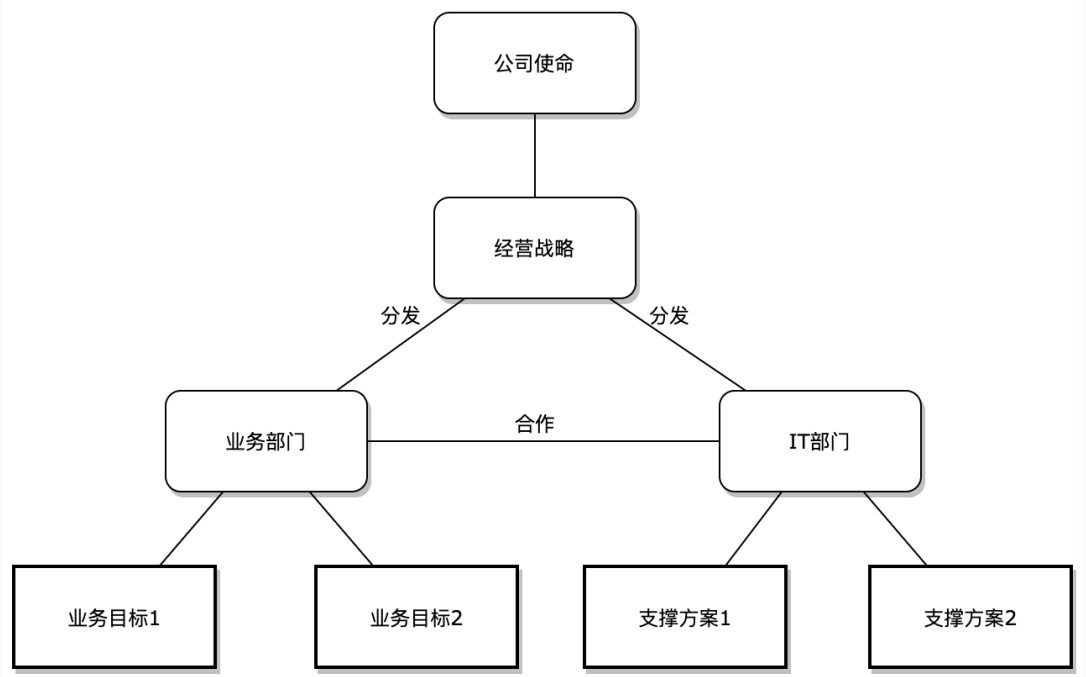
A客户所在公司的使命：增强国力、改善民生。

为了达成这个使命，公司高层管理开会反复讨论，终于制定出对应的经营战略：打造行业内领先的精品银行、特色银行balabala…

于是公司上下一心，团结一致，各部门同事也各司其职，开始为达成整体目标去分解成一个个的小目标：

1. 业务部门确认了要达成…的目标；
2. IT部门全力配合业务部门，分析并逐一确认落地方案…

在这个过程中，IT部门认真梳理需求后发现：业务部门明确提出了核心关键业务系统对应零数据丢失的要求，其中也包括异地灾难切换场景，这对IT来说无疑是一个巨大的挑战，那么要真正实现零数据丢失的目标，具体究竟该如何落地？是A客户当前迫切需要解决的问题。

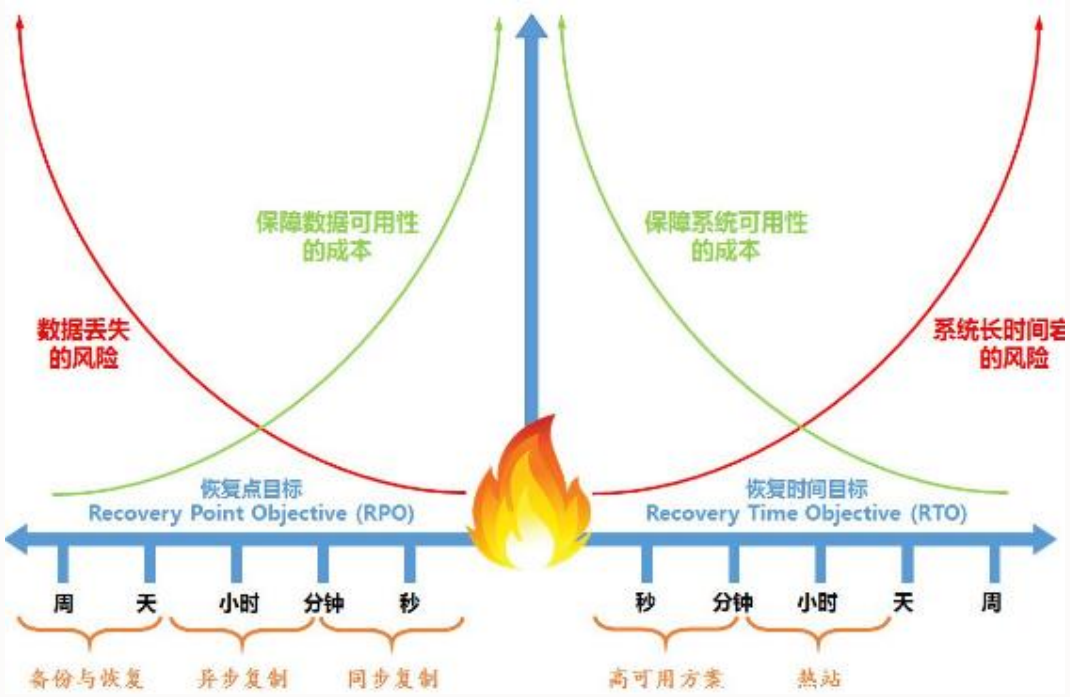


金融行业RTO/RPO等级标准定义

用户上报给监管机构的目标

表 C.1　RTO/RPO 与灾难恢复能力等级的关系

灾难恢复能力等级	RTO	RPO
1	2 天以上	1 天至 7 天
2	24 小时以上	1 天至 7 天
3	12 小时以上	数小时至 1 天
4	数小时至 2 天	数小时至 1 天
5	数分钟至 2 天	0 至 30 分钟
6	数分钟	0



Q：如何实现灾难恢复能力从5级到6级的提升?

可以看到RTO不是问题，只要具备整体向灾备切换的条件，数分钟完成切换并不难实现。

难点在于RPO=0（真正意义上的零数据丢失）这点该如何实现?

A客户调研同行业成熟方案

基于存储层面的复制 vs 基于数据库层面的复制

经调研，同业MAA部署实践中关于容灾部署部分，主要有以下两种方案：

➢ 1. 基于存储层面的复制

- 本质是存储级别的技术，通常需要相同存储和裸光纤网络，成本高昂；
- 同步模式对性能影响较大，并且坏块无感知，存在坏块从主库存储传播到备库存储的可能性；
- 备端始终处于闲置状态，投资回报率低。

➢ 2. 基于数据库层面的复制

- 本质是数据库级别的技术，通过redo实时传输应用，不需要相同存储，成本相对较低；
- 可自动修复坏块问题；
- 备库还可以用于分担查询负载，投资回报率高。

👍 目前基于数据库层面的复制已成为灾备建设的首选方案。

考量指标	基于存储层面的复制	基于数据库层面的复制
数据传输网络	裸光纤	广域网（千兆网络）
网络要求	高	低
数据传输量	传输IO操作所有原始块	仅传输数据更新操作日志
主备端距离	依赖于存储和光纤网络	依赖于广域网(可以达到比存储复制更远距离)
平台相关/依赖性	通常需要相同存储，相同平台	可以灵活使用不同存储
距离产生的影响	距离在数十公里接近100公里以后，只能使用IP协议，除距离增加的延迟外，使用IP使网络传输效率有所下降	可以消除距离影响（通过配置Far Sync实例）
数据一致性	受距离影响RPO窗口迅速增大（大量原始块等待传输）	可以保证数据一致性（通过配置Far Sync实例）
数据库可用性	灾难发生时数据库很可能无法启动，问题严重时很难恢复数据库，数据丢失 RPO≠0	同中心或同机房：备库随时可以验证查询，确保RPO=0； 同城或异地：远距离的同步可以引入Far Sync确保零数据丢失 RPO=0
设备投资利用率	备端设备相当于闲置	备端设备可以用来大量读，少量写，比如查询业务、投产前数据导出、实时主备数据一致性校验
坏库自动修复	不支持	支持

A客户当前核心系统的架构设计

标准的两地三中心，满足基本的监管要求

Q：当前架构下的灾难恢复能力?

可以满足灾难恢复能力5级的要求。

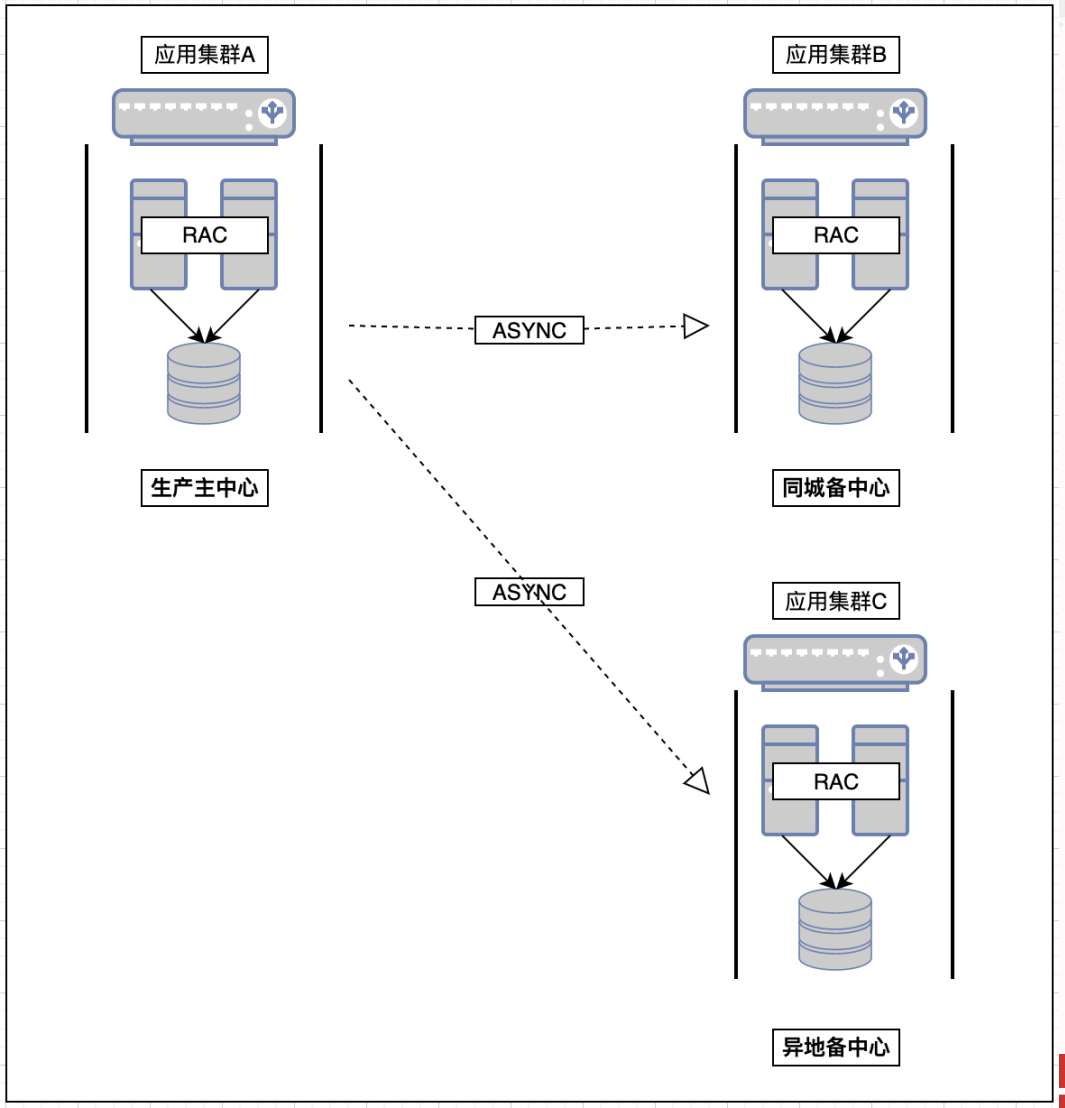
实际灾备切换演练评估结果，可以做到：

- 同城RPO趋近于0；
- 异地RPO小于30分钟。

如果想达到RPO=0的目标，就需要设置为SYNC模式才能够真正实现。

但是如果设置为SYNC模式，因为各中心之间有一定距离，受网络延迟影响，主库生产中心的正常数据库提交就会受到较大的性能影响。

由此就陷入了僵局，既然现在业务明确有了更高的要求，必须保证核心关键业务的RPO=0，那还有什么更好的设计方案吗?



利用Far Sync化解僵局

什么是Far Sync?

Far Sync可以看作是Oracle ADG 产品中的一个增强特性，在12.1.0.1版本引入，通过在距离主库较近的地方部署ADG Far Sync实例，能够实现包括本地和异地的任意距离的零数据丢失保护，同时对主库性能影响很小。即能够在保证RPO=0的同时，兼顾到主库性能影响最小化。

Oracle Data Guard Far Sync实例是一个远程 Oracle Data Guard 目标，它接受来自主数据库的redo日志，然后将该redo日志发送给 Oracle Data Guard 配置的其他成员。

需要特别注意的是：Far Sync实例并没有用户数据文件，因此无法打开访问，也无法应用redo日志，并且永远不能在主要角色中运行或转换为任何类型的备用数据库。

* Far Sync实例是 Oracle Active Data Guard Far Sync功能的一部分，需要 Oracle Active Data Guard 许可证。

Active Data Guard Far Sync

Availability → Data Guard

Far Sync is used to extend zero data loss protection to a remote standby database and avoid the impact to primary database performance of WAN network latency. A primary database ships synchronously to a light-weight instance referred to as a far sync instance (a control file and log files, no data files and no media recovery). The far sync instance then forwards the redo asynchronously to a remote standby database that is the failover target. Additional Far Sync features include the ability to directly service up to 29 remote destinations, and the ability to utilize Oracle Advanced Compression to compress redo for efficient transmission across a WAN. Far Sync is transparent to the administrator with regards to Data Guard role transitions. The same switchover or failover command used for any Data Guard configuration will transition any remote standby databases served by a far sync instance to the primary production role.

Business Benefit: Zero data loss protection can be achieved across long distances. The far sync instance is located within a distance of the primary database where synchronous transport does not impact application performance. Far Sync handles all communication with remote standby databases and is transparent when executing a zero data loss failover. Far Sync also offloads the production database of the overhead of servicing multiple remote destinations and redo transport compression.

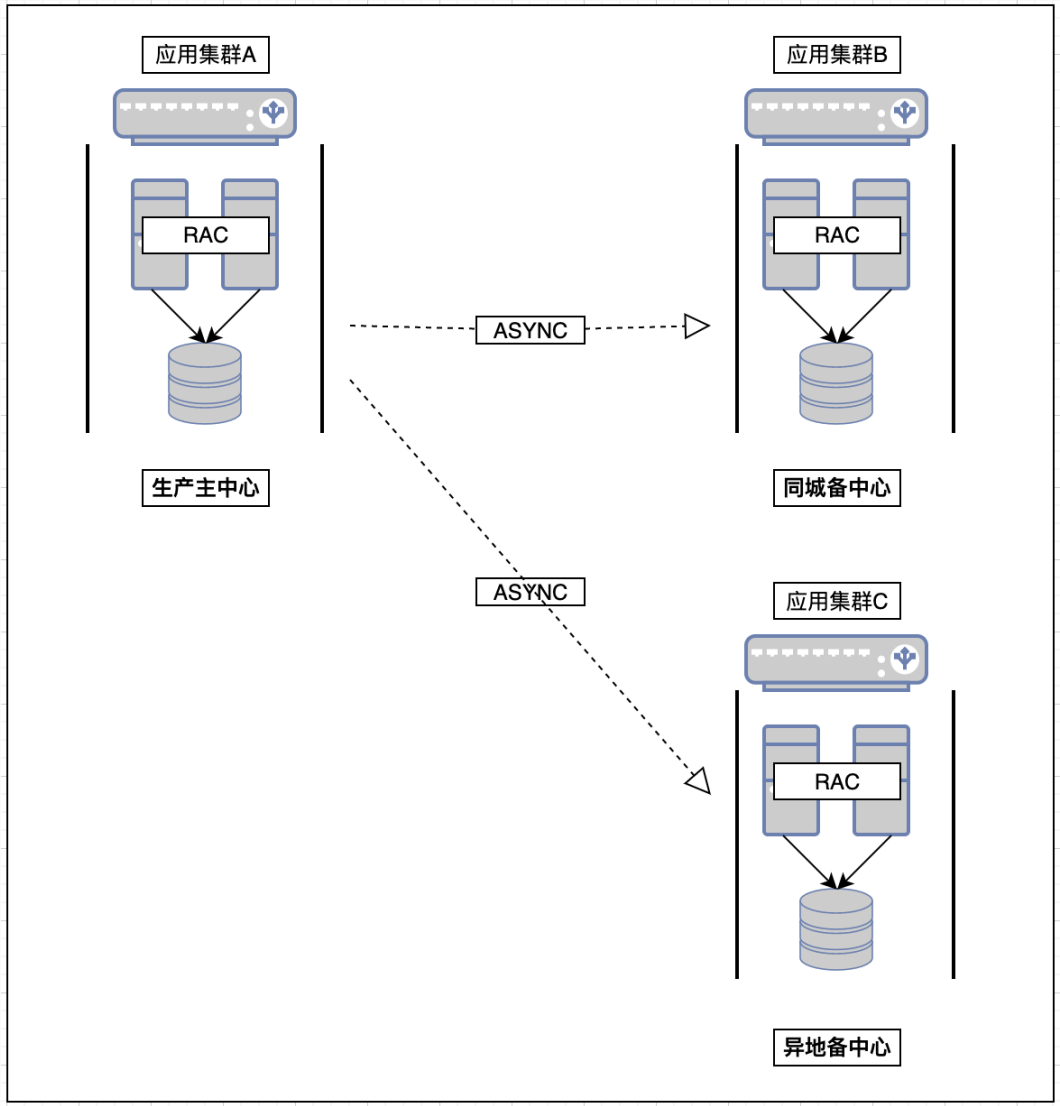
Release Availability	✗ 11.2 ✓ 12.1 ✓ 12.2 ✓ 18c ✓ 19c ✓ 21c
Parent Feature	Oracle Data Guard-Far Sync Standby
Available On	✓ Enterprise Edition ✓ Oracle Database Appliance ✓ Exadata ✓ Exadata Cloud Service / Cloud@Customer ✓ Database Cloud Service Enterprise Edition - Extreme Performance **Notes:** EE, ODA, and Exa: Requires the Oracle Active Data Guard option
Initial Release	12.1.0.1

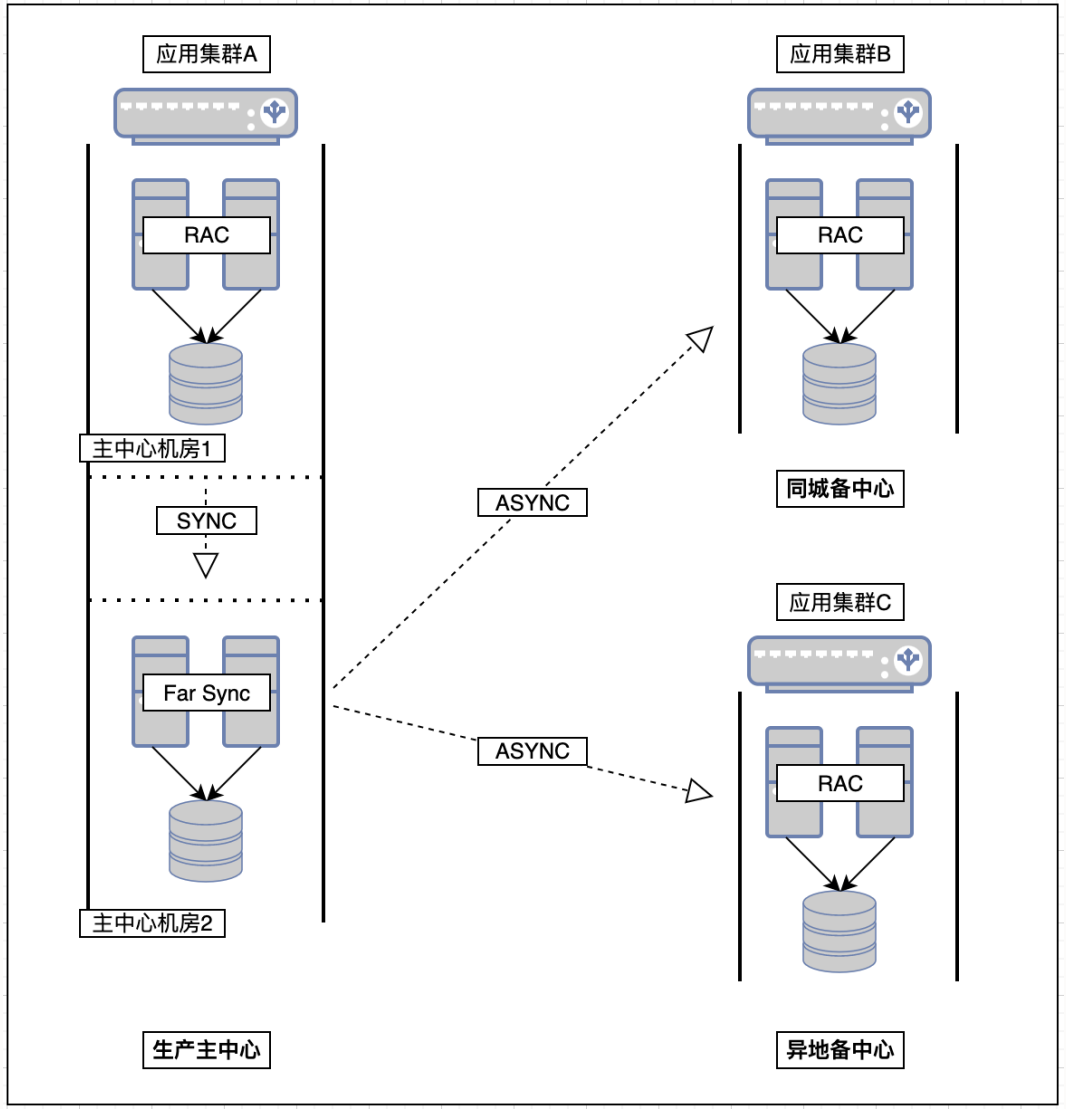
DISCLAIMER
The information in this application is subject to change at any time. Oracle Corporation and its affiliates disclaim any liability for incomplete or inaccurate information contained herein. If you have a question about your licensing needs, contact your Oracle sales representative, or refer to *Licensing Information User Manual* and *Licensing Guidelines*.

https://apex.oracle.com/database-features

A客户引入Far Sync之后的架构演进

在生产主中心的其他机房部署Far Sync实例



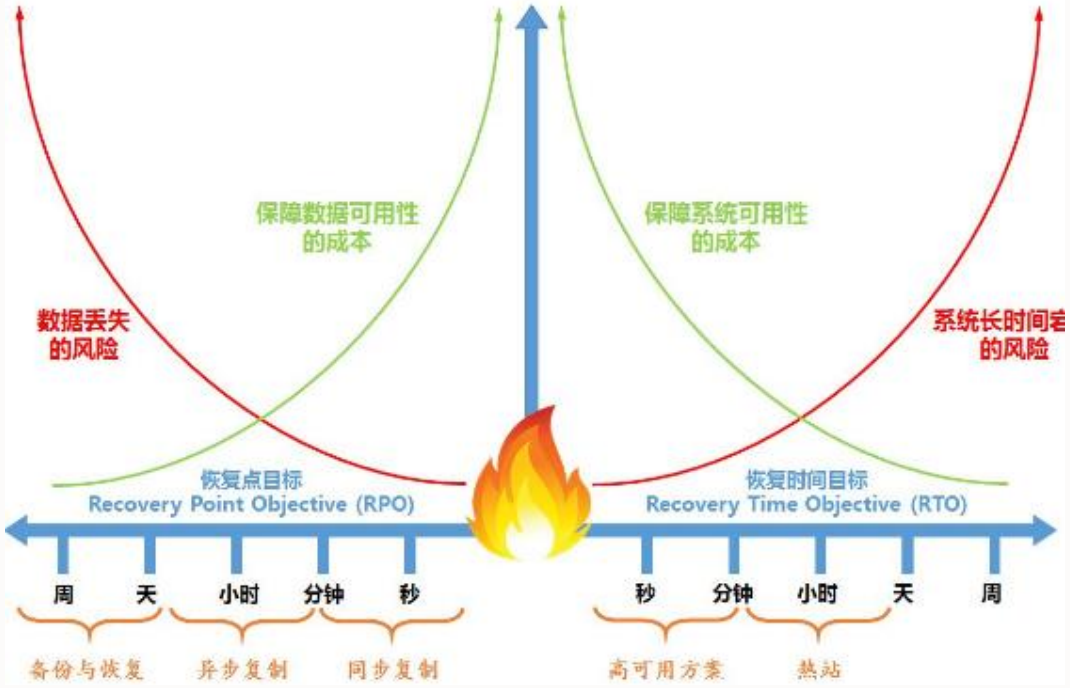


A客户引入Far Sync之后的架构演进

在生产主中心的其他机房部署Far Sync实例

新的架构演进，主要改动就是在生产主中心的其他机房新部署了一套Far Sync实例，而且这套实例对资源要求（空间、性能）并不高。

总的来说，ROI （投资回报率）达到一个很理想的状态！

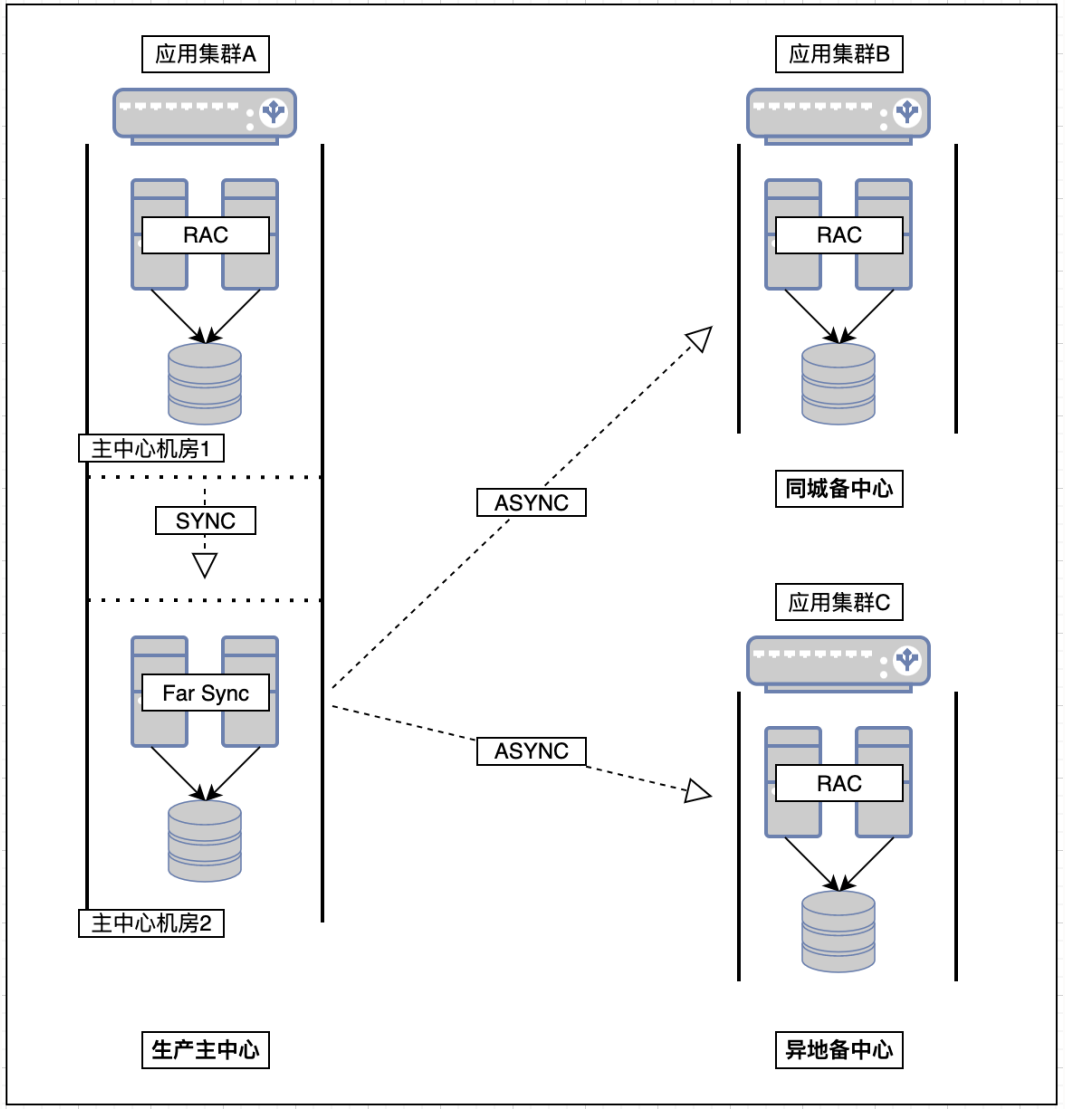


A客户引入Far Sync之后的架构演进

在生产主中心的其他机房部署Far Sync实例

嗯，新的架构在主库数据中心的不同机房内部署了Far Sync实例，基本保障了网络的稳定和低延迟，可是如果Far Sync实例本身IO问题影响到主库，那同样也会是个问题。

那能不能进一步优化呢?



Far Sync 和 Fast Sync

Far Sync 和 Fast Sync 傻傻分不清？甚至会把二者混为一谈

Far Sync实例支持的保护模式：

数据保护模式	事务提交前提	redo传输配置	RPO
最大保护	Redo在主备两端都写入成功	SYNC AFFIRM	RPO=0
最大可用	在不影响主库可用性前提下，当redo在主备两端都写入成功或者当redo在主库写入成功，在备库已经收到redo并启动写入动作（FASTSYNC模式）	正常配置：SYNC AFFIRM FASTSYNC配置：SYNC NOAFFIRM	RPO=0 （前提是最大可用模式没有发生降级）
最大性能	在不影响主库性能的前提下，提供最高的数据保护，主库事物提交不必等待redo传输和写入到备库	ASYNC NOAFFIRM	RPO趋近于零

- 最大可用性模式配置中的Far Sync实例
- 最大性能模式配置中的Far Sync实例

Fast Sync是最大可用模式中的增强：

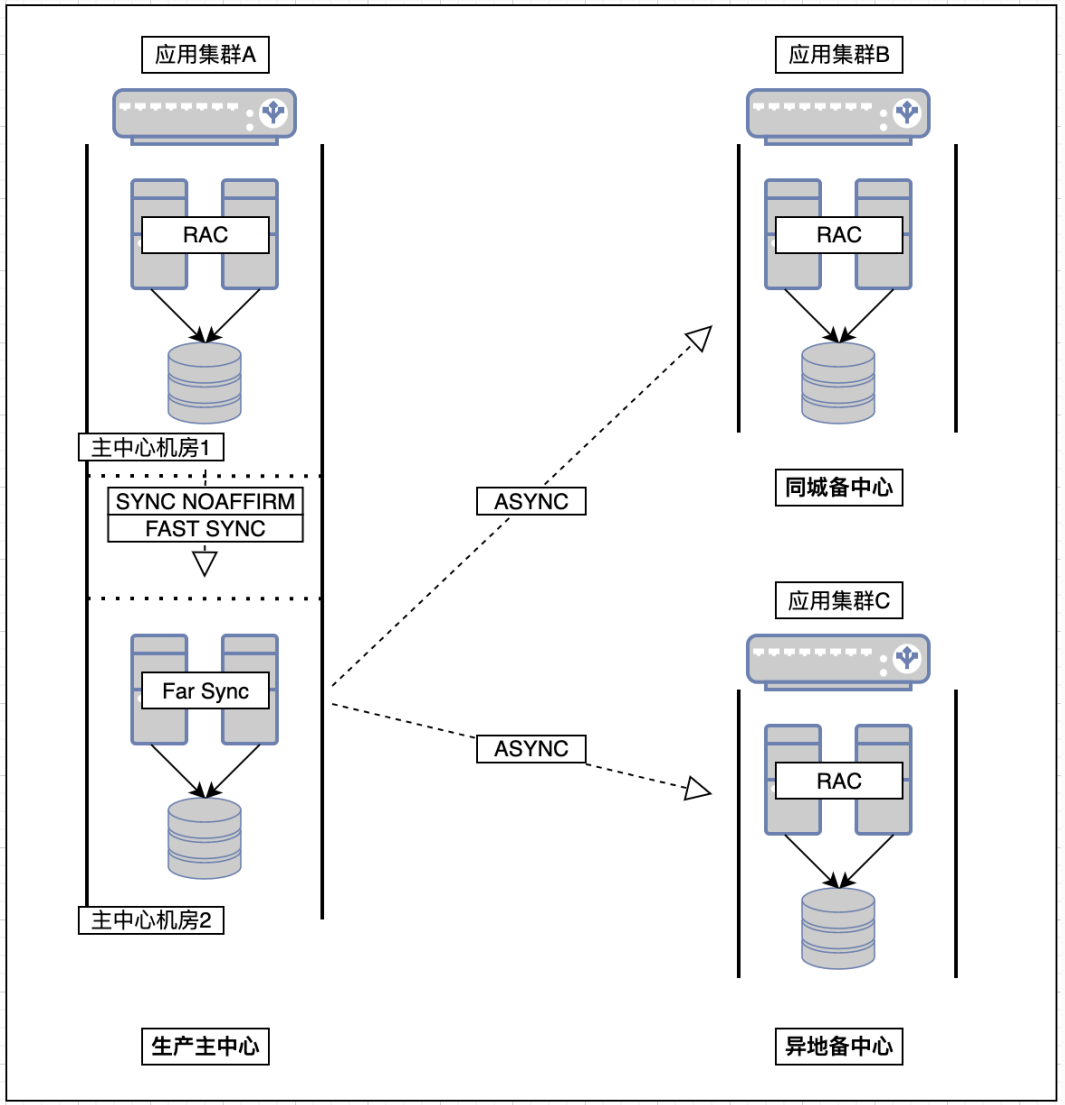
Maximum Availability	Maximum Performance	Maximum Protection
AFFIRM or NOAFFIRM	NOAFFIRM	AFFIRM
SYNC	ASYNC	SYNC
DB_UNIQUE_NAME	DB_UNIQUE_NAME	DB_UNIQUE_NAME

嘿，想想将这两个容易搞混的家伙组合起来使用还是不错的呢～

A客户引入Far Sync之后的架构演进

在生产主中心的其他机房部署Far Sync实例

所以可以将主中心主库到Far Sync实例的SYNC同步进一步优化为SYNC NOAFFIRM，消除Far Sync实例自身的IO影响，在保证RPO=0的同时进一步保障主库性能。



引入Far Sync实例的步骤参考

创建Far Sync实例与创建物理Standby数据库完全类似，只是Far Sync实例中不存在数据文件

1. 为Far Sync实例创建控制文件
2. 从主数据库使用的服务器参数文件 (SPFILE) 创建参数文件 (PFILE)
3. 从已编辑的参数文件 (PFILE) 创建服务器参数文件 (SPFILE)
4. 使用操作系统复制实用程序将步骤 1 中创建的远程同步实例控制文件和步骤 3 中创建的服务器参数文件 (SPFILE) 从主系统复制到远程同步实例系统上的适当位置
5. 创建备用重做日志（SRLs），方法和在普通standby创建SRLs相同
6. 如果要在 Windows 系统上托管远程同步实例，请使用 ORADIM 实用程序创建 Windows 服务
7. 如果操作系统身份验证用于管理用户并且 SSL 用于重做传输身份验证，则此步骤是可选的。如果不是，则将主数据库的远程登录密码文件复制到远程同步实例上的相应目录。无论何时授予或撤销管理权限（ SYSDG、SYSOPER、 SYSDBA等），以及更改任何具有管理权限的用户的密码后，都必须重新复制密码文件（在12.2.0.1之后，密码文件会自动同步更新）
8. 在Far Sync实例上，使用 Oracle Net Manager 为Far Sync实例配置监听
9. 在主数据库上，使用 Oracle Net Manager 为用于redo传输服务的Far Sync实例 (chicagoFS) 创建网络服务名称；在Far Sync实例上，使用Oracle Net Manager为重做传输服务使用的主(chicago)和最终备库(boston)创建网络服务名称
10. 以mount模式启动 Far Sync实例
11. 验证Far Sync实例是否正常运行
12. 将ADG配置的保护模式修改为最大可用。在主数据库上执行命令：

SQL> ALTER DATABASE SET STANDBY TO MAXIMIZE AVAILABILITY;

主库重新配置日志归档目标并分组

支持灵活配置LOG_ARCHIVE_DEST_N、 LOG_ARCHIVE_DEST_STATE_N参数

在主中心主库上配置两个日志归档目标，同属于一个组，状态分别为 ENABLE 和 ALTERNATE 。

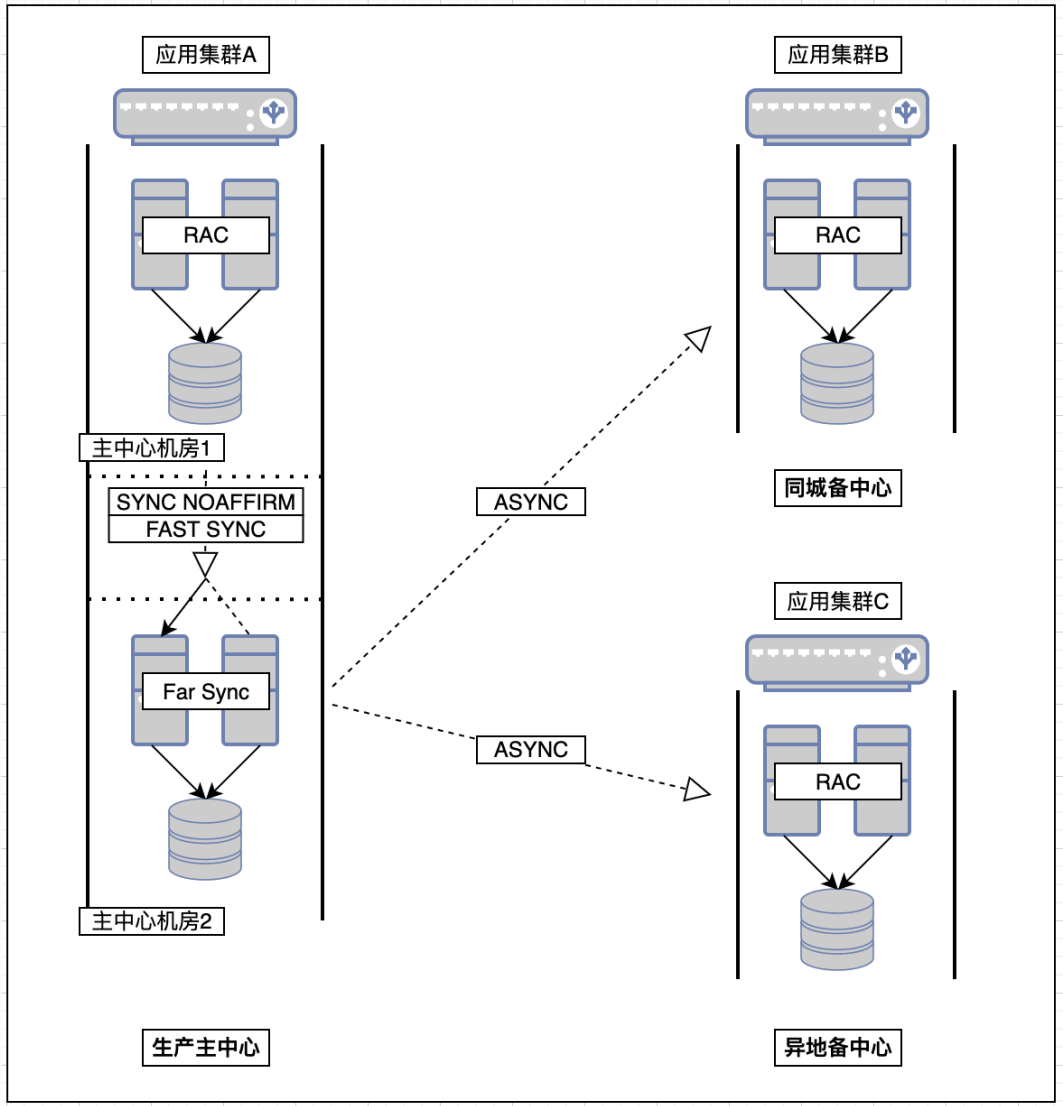
A客户这里Far Sync也利旧使用了RAC架构，可不做设置，如果是Far Sync是HA的部署方式，可以这样设置：

LOG_ARCHIVE_DEST_2='SERVICE=chicagoFS SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1'

LOG_ARCHIVE_DEST_STATE_2=ENABLE

LOG_ARCHIVE_DEST_3='SERVICE=chicagoFS1 SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1'

LOG_ARCHIVE_DEST_STATE_3=ALTERNATE



多个日志归档目标的优先级配置

为同一组中的日志归档目标分配优先级

使用初始化参数LOG_ARCHIVE_DEST_*n*的PRIORITY属性在日志归档目标组中配置，可让您控制fail back机制，尤其是在组内有多个成员的情况下。

优先级值是 1 到 8 范围内的整数。数字越小表示优先级越高。

A客户目前还没有区分优先级，如果日后需要，可以参考这样的配置：

LOG_ARCHIVE_DEST_2='SERVICE=chicagoFS SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=1'

LOG_ARCHIVE_DEST_STATE_2=ENABLE

LOG_ARCHIVE_DEST_3='SERVICE=chicagoFS1 SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=1'

LOG_ARCHIVE_DEST_STATE_3=ALTERNATE

LOG_ARCHIVE_DEST_4='SERVICE=chicagoFS2 ASYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=2'

LOG_ARCHIVE_DEST_STATE_4=ALTERNATE

17.13 PRIORITY

The `PRIORITY` attribute is used to specify preference within a collection of log archive destinations.

Priorities are numbered 1 through 8. A lower value represents a higher priority. The lowest priority (`PRIORITY=8`) is special in the sense that if that priority is active then all destinations at that priority are made active. If any higher priority destination returns to service, then that destination is made active and all low priority destinations are made inactive.

Category	**Priority=*integer***
Data Type	Integer
Valid Value	1 through 8
Default Value	1
Requires attributes	`SERVICE`
Conflicts with attributes	`ALTERNATE`
Corresponds to	Not applicable

配置到多个日志归档目标的同步

传输到同一个组中的多个日志归档目标

Far Sync实例将redo转发到两个最终standby数据库，一旦Far Sync实例故障，还可以同时启用priority=8的两个alternate状态的日志归档目标，A客户的同城备库和异地备库就参考这样的配置：

LOG_ARCHIVE_DEST_2='SERVICE=chicagoFS SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=1'

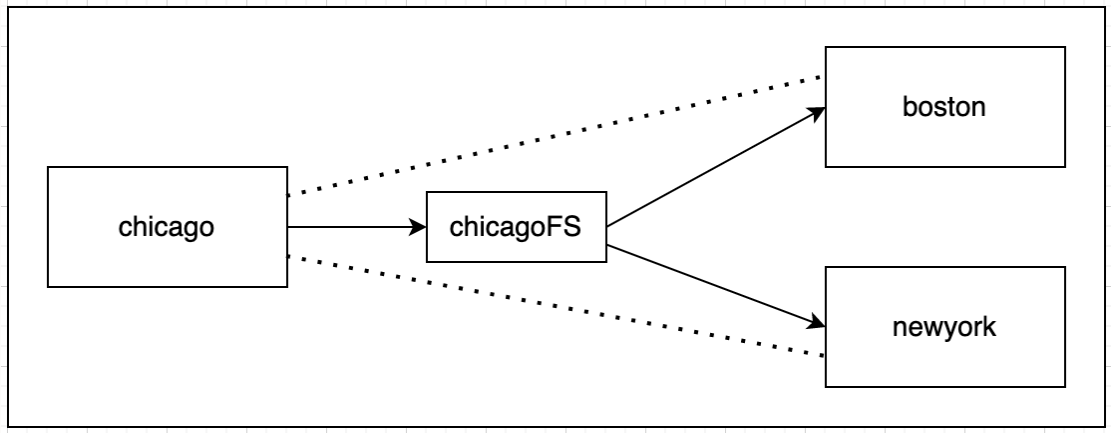
LOG_ARCHIVE_DEST_STATE_2=ENABLE

LOG_ARCHIVE_DEST_3='SERVICE=boston ASYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=8'

LOG_ARCHIVE_DEST_STATE_3=ALTERNATE

LOG_ARCHIVE_DEST_4='SERVICE=newyork ASYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=8'

LOG_ARCHIVE_DEST_STATE_4=ALTERNATE



A客户日志归档目标的高可用配置示例

只有当A客户所有Far Sync实例全部故障后保护级别才会降级

LOG_ARCHIVE_DEST_2='SERVICE=chicagoFS SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=1'

LOG_ARCHIVE_DEST_STATE_2=ENABLE

LOG_ARCHIVE_DEST_3='SERVICE=chicagoFS1 SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=1'

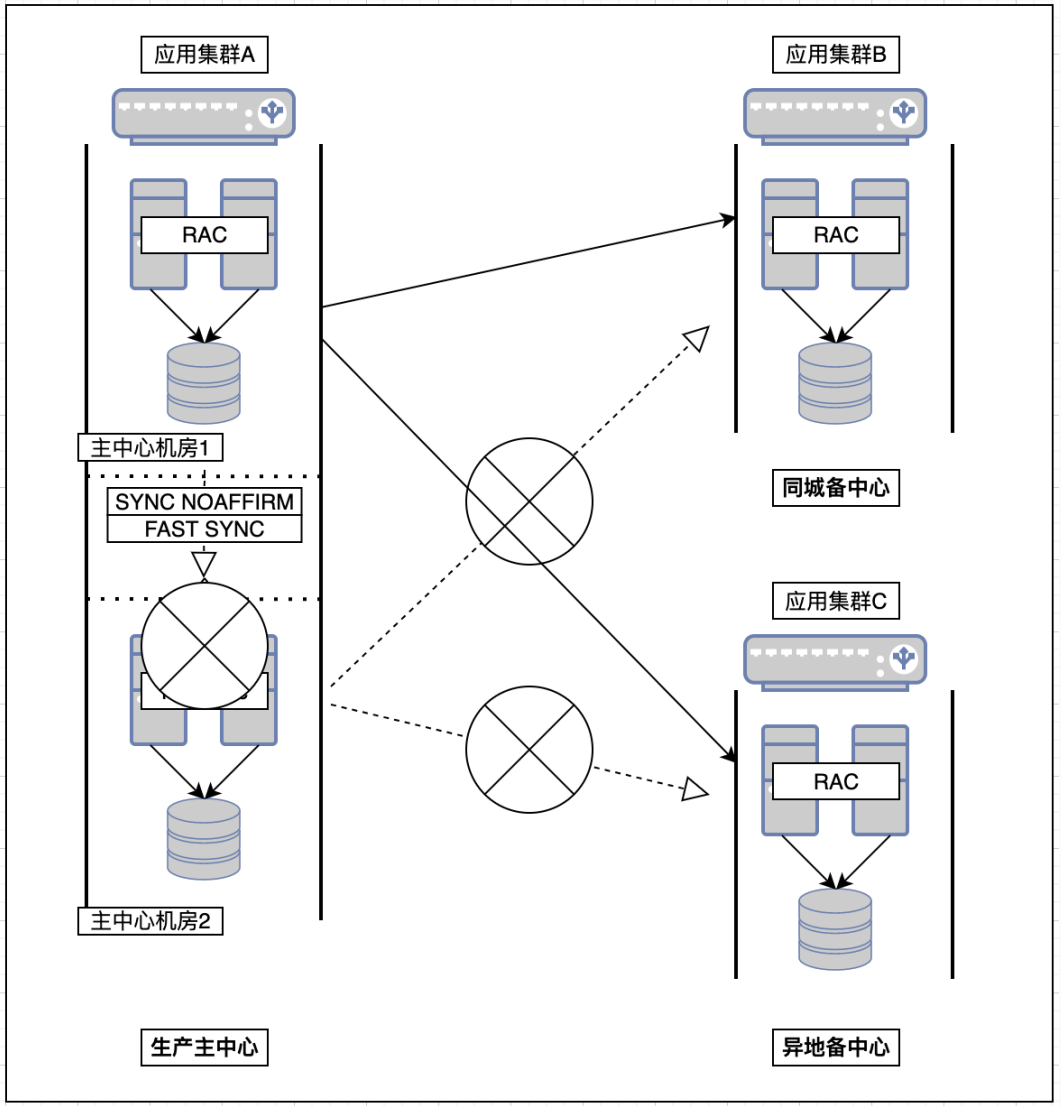
LOG_ARCHIVE_DEST_STATE_3=ALTERNATE

LOG_ARCHIVE_DEST_4='SERVICE=boston ASYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=8'

LOG_ARCHIVE_DEST_STATE_4=ALTERNATE

LOG_ARCHIVE_DEST_5='SERVICE=newyork ASYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=8'

LOG_ARCHIVE_DEST_STATE_5=ALTERNATE



典型客户场景1: Far Sync最简配置（生产环境不推荐）

实现RPO=0的目标的最低配

最简配置，只加入一个Far Sync实例，架构如右图所示：

优点：

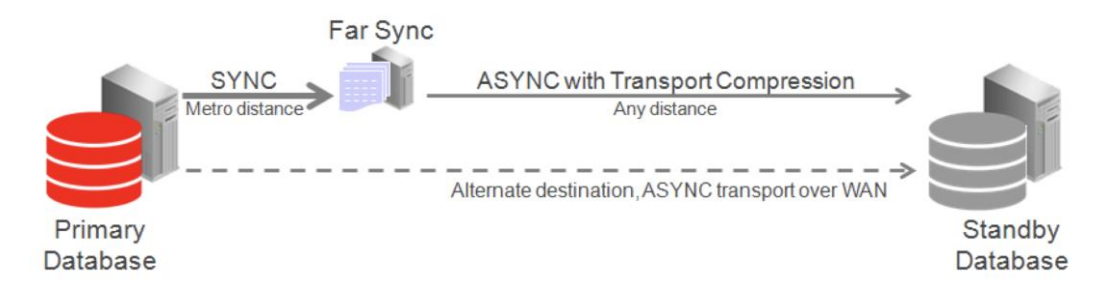
满足零数据丢失需求的最简配置。

缺点：

Far Sync缺少高可用性，可以看到一旦Far Sync实例故障，就只能降级为ASYNC的最大性能，此时就不能保证RPO=0的目标。

小结：

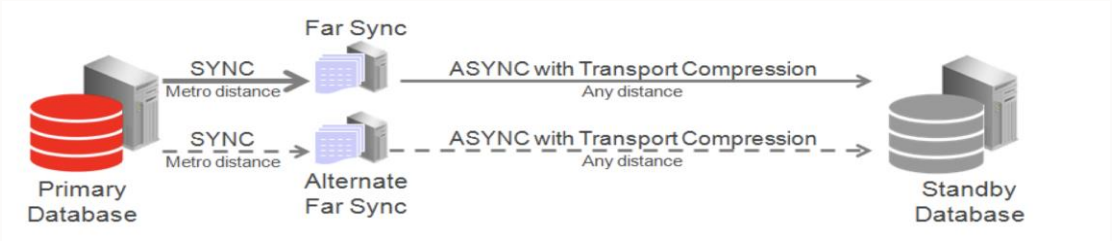
这种架构一般只用于测试验证Far Sync的场景，在重要的生产环境中通常不会推荐这种单点部署方式。



典型客户场景2: Far Sync高可用配置（A客户本次有采用）

实现RPO=0的目标，解决了Far Sync单点故障问题

Far Sync高可用配置，加入两个Far Sync实例，架构如右图所示：



优点：

能保证Far Sync实例具备高可用性，当其中任意一个Far Sync实例故障，依然能保证redo同步传输，RPO=0.

缺点：

只有单边Far Sync实例，一旦主备角色发生切换，远端的原备库成为新主库，因为没有距离近的Far Sync实例可用，所以只能选择ASYNC传输。

小结：

实际生产环境中，这种算是最基础的配置：至少保证Far Sync具备高可用配置。在角色切换后没有强制要求RPO=0的场景适用。

典型客户场景3: Far Sync对等配置（A客户本次未采用）

实现RPO=0的目标，主备角色切换后依然保证最大可用性保护级别

Far Sync对等配置，主备两边都有临近的Far Sync实例，架构如右图所示：

优点：

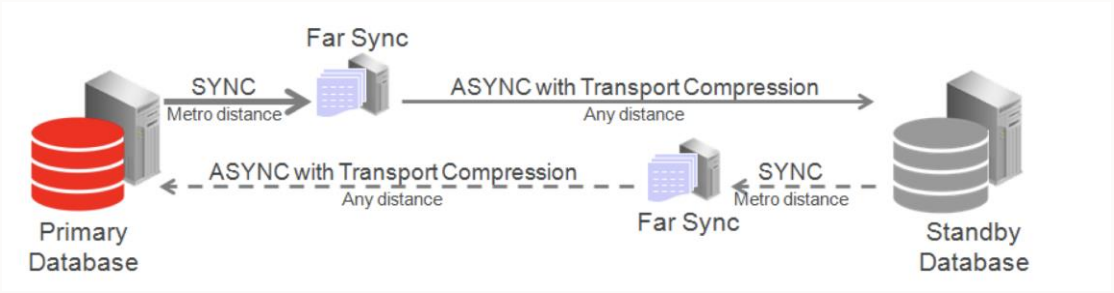
主备角色发生切换，依然可以保证有临近Far Sync实例，即依然能保证redo同步传输，RPO=0.

缺点：

两端都需要规划临近的Far Sync实例，对很多客户环境来说相对困难。

小结：

实际生产环境中，如果要求对等配置，通常都是要求较高的场景，此时建议同时结合场景2和场景3，同时满足Far Sync高可用配置和两边对等配置。



典型客户场景3: Far Sync对等配置（A客户本次未采用）

实现RPO=0的目标，主备角色切换后依然保证最大可用性保护级别

如果主备发生切换，原备库boston和原Far Sync实例chicagoFS两个站点之间的网络延迟足够大，会影响提交响应时间。

为了切换后也能保持零数据丢失的最大可用性保护级别，可以在原备库boston附近建立第二个far sync实例boston FS并配置。

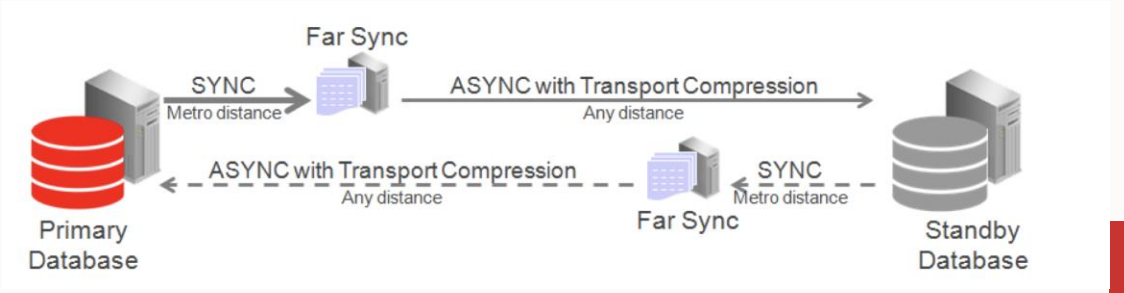
LOG_ARCHIVE_CONFIG='DG_CONFIG=(chicago, chicagoFS, boston, bostonFS)'

LOG_ARCHIVE_DEST_2='SERVICE=bostonFS SYNC AFFIRM
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=bostonFS GROUP=1 PRIORITY=1'

LOG_ARCHIVE_DEST_STATE_2=ENABLE

LOG_ARCHIVE_DEST_3='SERVICE=chicago ASYNC NOAFFIRM
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=chicago GROUP=1 PRIORITY=2'

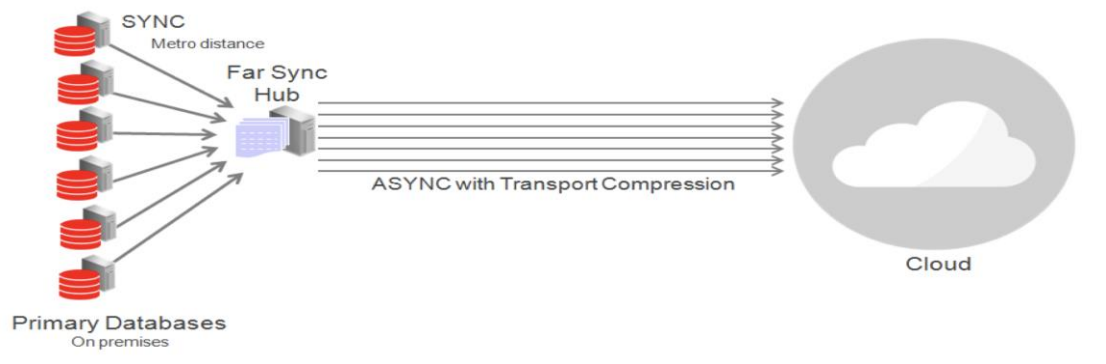
LOG_ARCHIVE_DEST_STATE_3=ALTERNATE



典型客户场景4: Far Sync Hub，同步到云端（A客户本次不适用）

实现RPO=0的目标，迁移上云

Far Sync Hub配置，同步到云端，架构如右图所示：



优点：

可以一定程度规避上云的网络不稳定影响，非常适用于从OP到Cloud的同步场景，Far Sync Hub保证OP端的所有数据库可以SYNC，然后再异步传输到Cloud端，保证RPO=0.

缺点：

虽然Far Sync所需资源相对很少，但承载大量实例的Far Sync Hub很可能成为传输瓶颈，需要合理规划。

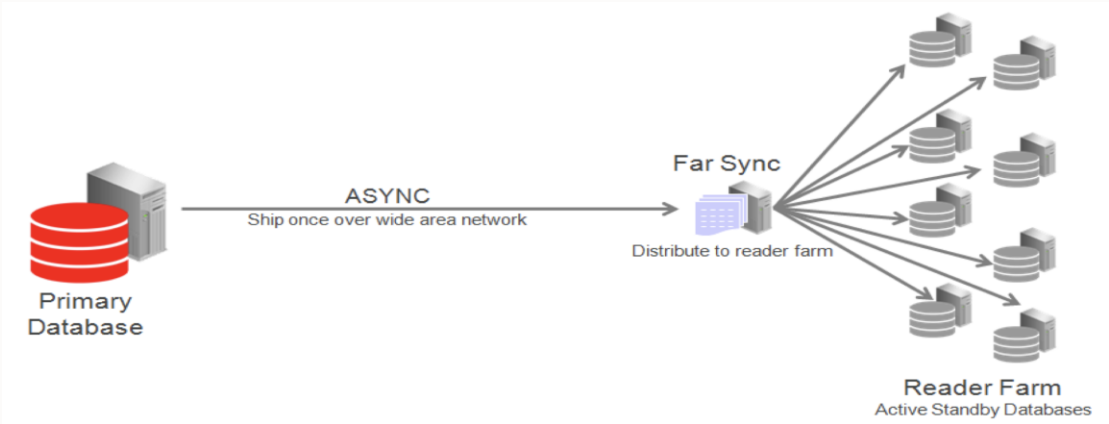
小结：

备库规划部署在云端的客户可以选择这种方式。

典型客户场景5: Far Sync实例集中分发（A客户本次有采用）

实现多备库集中分发，减少主库性能影响

Far Sync 集中分发配置，同步到多个备库，架构如右图所示：



优点：

主库只需要配置一个目标ASYNC或SYNC方式同步到Far Sync实例，由Far Sync实例分发到多个备库，可以减少主库的性能压力。

缺点：

只适用于特定场景。

小结：

有多个备库需求的客户可以考虑这种部署方式。

典型客户场景5: Far Sync实例集中分发（A客户本次有采用）

可配置多个Far Sync实例分流压力

A客户的Far Sync实例分发给同城备库以及异地备库。

LOG_ARCHIVE_DEST_2='SERVICE=chicagoFS SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=1'

LOG_ARCHIVE_DEST_STATE_2=ENABLE

LOG_ARCHIVE_DEST_3='SERVICE=chicagoFS1 SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=1 PRIORITY=1'

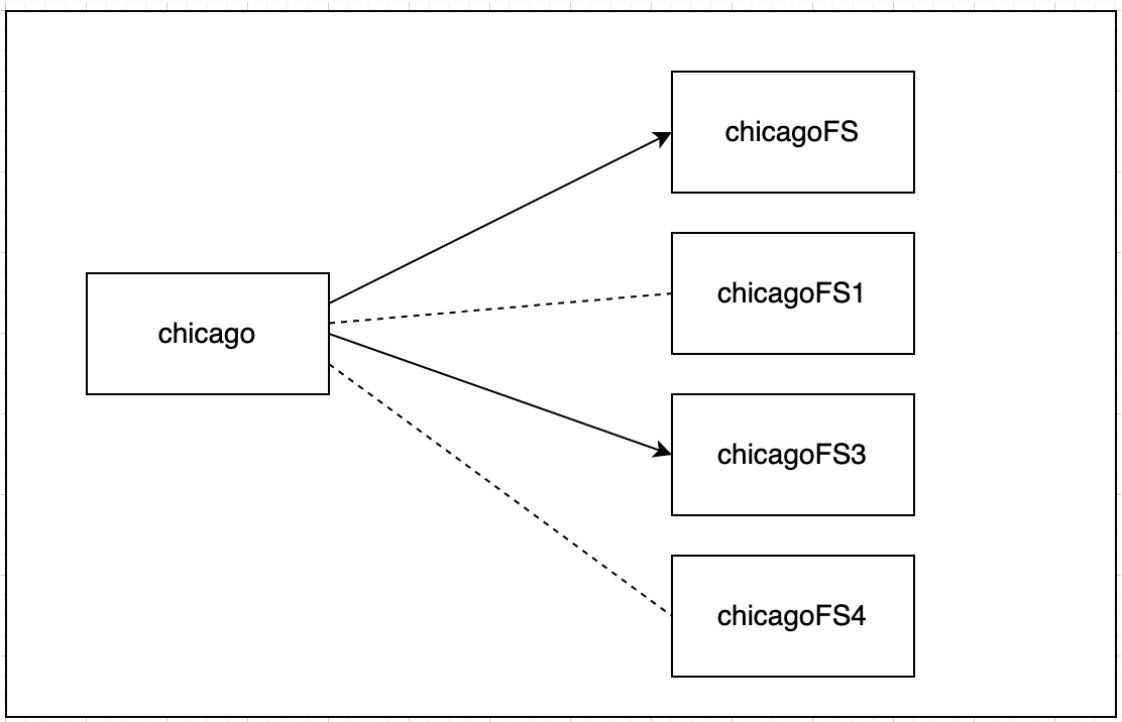
LOG_ARCHIVE_DEST_STATE_3=ALTERNATE

LOG_ARCHIVE_DEST_4='SERVICE=chicagoFS3 SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=2 PRIORITY=1'

LOG_ARCHIVE_DEST_STATE_4=ENABLE

LOG_ARCHIVE_DEST_5='SERVICE=chicagoFS4 SYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) GROUP=2 PRIORITY=1'

LOG_ARCHIVE_DEST_STATE_5=ALTERNATE



A客户使用ADG Far Sync 的总结

ADG Far Sync给A客户带来了哪些优势?

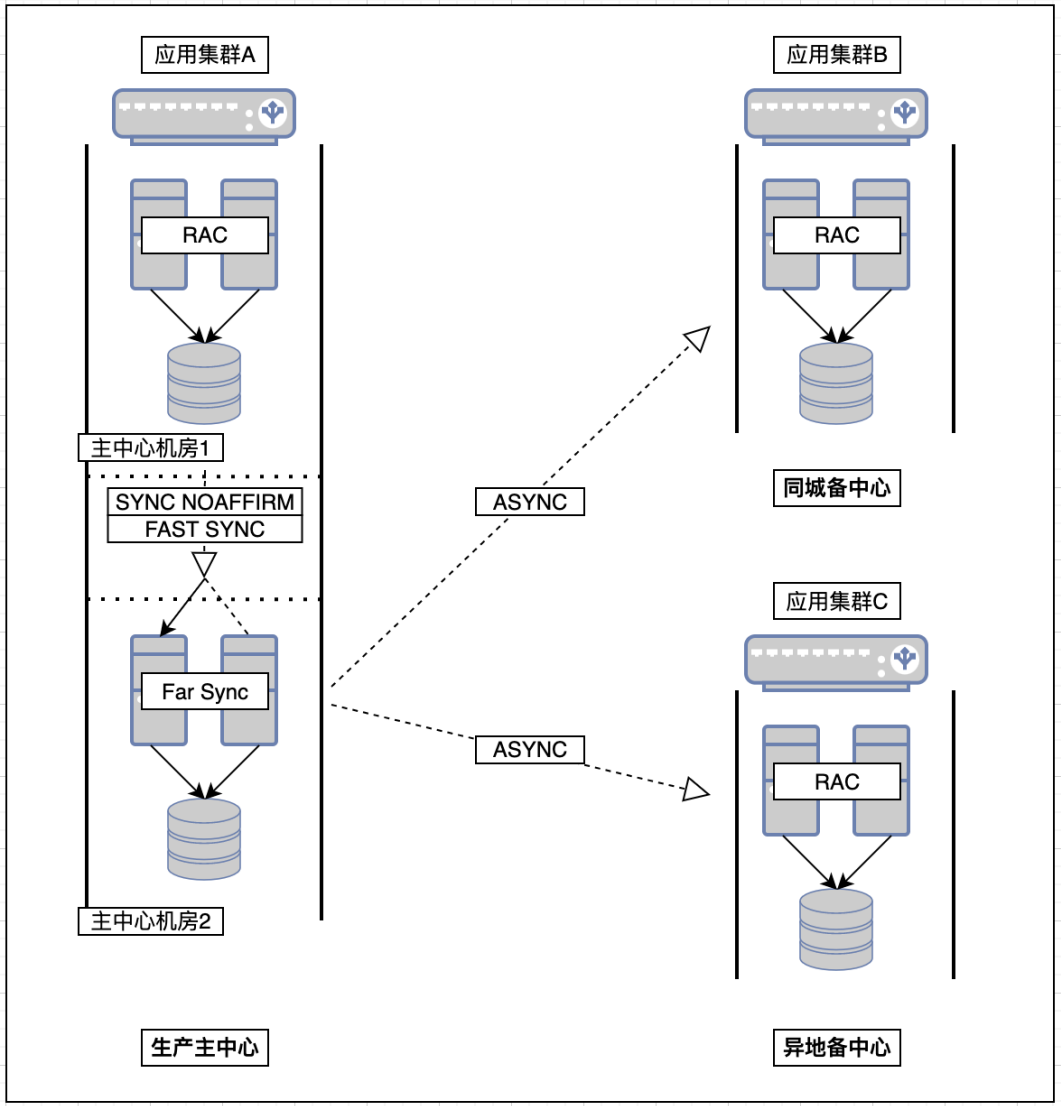
◆ A客户为什么选择ADG Far Sync?

- **业务连续性需求**，生产系统关乎民生，要求7*24稳定运行，可同步传输的灾备环境很容易受到网络影响。临近主库的Far Sync实例可以做到最大化满足主库性能影响最小的需求，保证业务连续性。
- **零数据丢失需求**，民生数据至关重要，要求备库零数据丢失（RPO=0），主备之前距离很远，日志直接同步传输带来的性能影响不可接受。因此需要临近主库的Far Sync实例中转，间接实现零数据丢失需求。
- **成本可控，ROI高**，Far Sync实例无需存放数据文件，消耗的磁盘和处理资源非常少，普通server就可以满足Far Sync实例部署需求，满足业务部门要求。

◆ 引入ADG Far Sync给A客户带来了哪些实际收益?

- **容灾保护等级提升至最高，**从现有的容灾保护5级提升到最高的6级，满足业务需求的同时也做到了行业内领先。

😄 今后不必再唯唯诺诺的说什么RPO趋近于零，可以自豪的讲我们核心系统的RPO就是0，真正实现了零数据丢失的目标。



真实世界的金融行业客户案例

核心银行系统灾备建设转型历程

客户的尝试论证新的技术ADG Far Sync，彻底改善了之前的难题：

➢ 业务连续性

曾经使用 HDC的 TrueCopy存储复制软件，开启同步的模式，在遇到网络不稳定的情况下，引发了长时间的性能问题（业务运行缓慢，awr显示的log file sync平均等待时间长达几十ms）。

使用Far Sync的成效：

引入ADG Far Sync之后，因为Far Sync实例部署在同数据中心不同机房内，从根源上解决了网络不稳定的问题，ADG最终备库部署在同城的另一个数据中心，保证RPO=0的前提下对主库性能影响最小。

客户对此的评价：

终于找到了实现RPO=0目标的最佳方案。

➢ 零数据丢失

曾经使用ADG但没引入Far Sync，最大性能异步传输，在一次主库存储发生故障，想启用备库接管业务时却发现延迟在半小时以上，不满足切换要求。

使用Far Sync的成效：

引入ADG Far Sync之后，因为Far Sync实例有全量的redo日志，可以追平后打开备库，实现RPO=0。

客户对此的评价：

单纯实现不丢数据不难，难的是可以同时兼顾最小化性能影响，ADG Far Sync做到了。

在正常业务运行期间，客户配置同步模式的Far Sync实例，性能表现优秀，业务层完全无感知。（redo生成量3M/s 左右，每秒300 Transactions左右）：

（1）客户日常使用期间， log file sync 是 1.08ms, log file parallel write 是 295.50us；

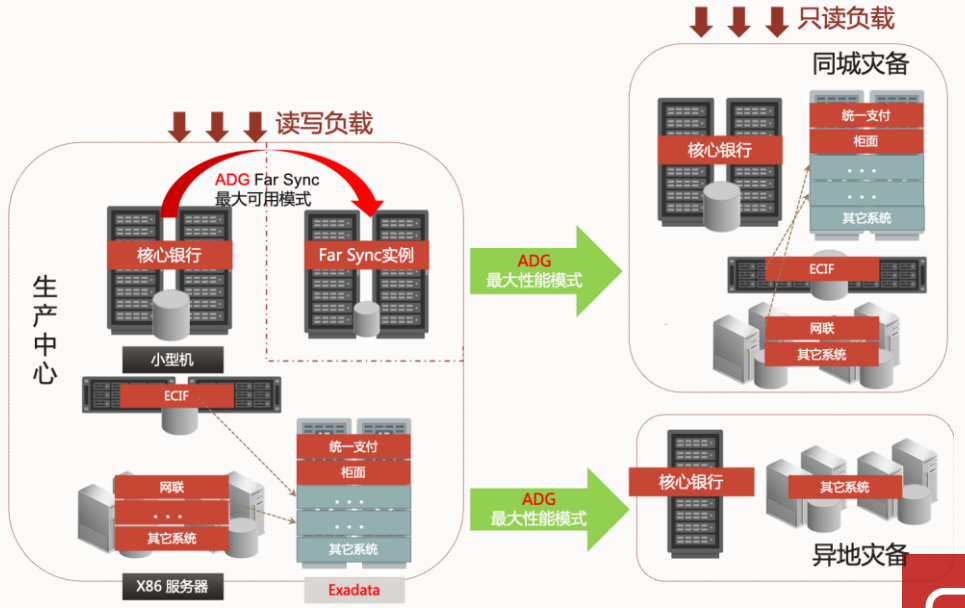
Event	Waits	Total Wait Time (sec)	Avg Wait	% DB time	Wait Class
log file sync	15,665,188	16.9K	1.08ms	1.7	Commit

Event	Waits	%Time -outs	Total Wait Time (s)	Avg wait	Waits /txn	% bg time
log file parallel write	23,779,404	0	7,027	295.50us	1.04	7.38

（2）客户日终批期间，log file sync 是 1.53ms, log file parallel write 是 397.42us；

Event	Waits	Total Wait Time (sec)	Avg Wait	% DB time	Wait Class
log file sync	836,745	1282.3	1.53ms	.9	Commit

Event	Waits	%Time -outs	Total Wait Time (s)	Avg wait	Waits /txn	% bg time
log file parallel write	793,600	0	315	397.42us	0.69	6.23



即刻扫码报名免费评估

您将体验到

- 八大维度高可用性架构评估
- 专业的数据库团队服务
- 高质量业务连续性改进建议