

一种基于虚拟日志的数据复制解决方案^{*})

倪泳智 王 珊

(中国人民大学信息学院 北京 100872)

摘 要 本文从实际应用的角度出发,为数据库系统设计并实现了一种基于虚拟日志的高可用性的复制解决方案。采用一主多从模式(Single Master, Multi Slaves),使用异步分发技术,实行延迟远程存取和延迟传播技术对数据进行更新,提供跨平台支持。针对具有自主知识产权的国产大型数据库系统 KingbaseES 具体实现了该复制方案,从而提供了一个现实的、可扩展的复制系统框架。它能够提供稳定的数据复制,可用性高、准时复制、响应时间短,在及时同步业务数据、灾难备份恢复、改善决策支持系统等许多关键服务中有很高的应用前景。

关键词 数据复制,一主多从模式,异步分发,虚拟日志

A Virtual Log-based Data Replication Solution

NI Yong-Zhi WANG Shan

(Information School, Renmin University of China, Beijing 100872)

Abstract This paper designs and completes a virtual log-based, high availability data replication solution for real applications. The replication solution uses "Single Master, Multi Slaves" model, asynchronous technique, deferred remote procedure calls to apply update data, and supports multi platforms. We implement this replication solution on the national large database system—Kingbase ES, and provide a realism, high extendable replication system framework. This solution can provide steady, high performance data replication for many key services.

Keywords Data replication, Single master, Multi slaves, Asynchronous, Virtual log

1 引言

1.1 数据复制简介

数据库复制技术是在数据库之间对数据和数据库对象进行复制和分发并进行同步以确保其完整性和一致性的一组技术。它将数据库中的数据拷贝到不同物理地点的数据库中以支持分布式应用,是整个分布式计算解决方案的一个重要组成部分。

使用复制可以将数据通过局域网、拨号连接、Internet 等方式分发到不同位置,供远程或移动用户使用,还能够提高应用程序性能,根据数据的使用方式物理分隔数据(例如,将联机事务处理业务和决策支持系统分开),或者跨越多个服务器分布数据库处理。

1.2 数据复制优点

根据不同的复制模型和参数选择,以及实际应用需要,数据复制能够提供不同的优点。其共性在于数据需要时的高可用性。

主要的优点包括:

(1)将复制作为备用恢复策略的一部分,在主服务器出现故障的情况下提供数据副本;还可保证不间断地提供服务。

(2)在多个服务器保存相同数据的副本。当多个服务器需要为报表等应用程序读取相同数据或需要各个服务器能独立运行承担业务时,这将非常有用。

(3)从大量读取数据的应用程序,如联机分析处理 OLAP 数据库、数据集市或数据仓库中分离 OLTP 应用程序。另

外,也可利用复制分离出需要的部分数据。

(4)提供更好的独立性。用户可以在连接断开的情况下继续使用数据的副本,然后在连接恢复时将数据库所做的更改传播到其它数据库,保持一致状态。

1.3 问题描述与分析

目前大多数主流数据库系统,如 Oracle, Sybase ASE, IBM DB2, Microsoft SQL Server 等都已经实现了相应配套的数据复制功能。但对于国内的数据库来说,尚缺乏能够真正配套的复制方案。因此,设计一个可扩展的复制解决方案,并实现该复制系统框架,是提高我国数据库产业,增强数据库性能,提高数据可用性以及应用可靠性的一个关键。

2 复制类型概述与选择

2.1 复制基本术语

2.1.1 主数据库(Master Database)

主数据库是提供数据以便复制到其它从数据库的数据库服务器。主数据库可以发布一个或多个项目内容(如表),它们各自代表一组逻辑相关的数据。除了指定其中哪些数据需要复制之外,主数据库还检测事务复制期间发生更改的数据并维护记录相关的信息,生成快照。在下文中,简称主库。

在不同的数据库系统中,此术语有不同的表述,如 Oracle 称为主站点(Master Site), IBM DB2 称为源服务器或源数据库(Source Database), Sybase 称为主数据库(Primary Database), SQL Server 称为发布服务器(数据库)等等。

2.1.2 复制服务器(Replication Server)

^{*})国家 863 数据库重大专项课题(2003AA4Z3030)、北京市科技计划重大项目(H030130060011)。倪泳智 硕士研究生;王 珊 教授,博士生导师。

复制服务器是分发数据更新的服务器,它的角色根据执行的复制类型而不同。

在一般的情况下,复制服务器可以与主数据库、从数据库分离,位于不同的服务器上。它在主数据库上扫描快照,在合适的时候启动复制进程,将更新快照应用到从数据库,协调主、从数据库,并确保数据的一致性。

2.1.3 从数据库(Slave Database)

从数据库是接收复制数据的数据库服务器。它可以提供远程的数据备份,保证数据的可靠性,以便主库发生灾难故障时恢复,或者立即启用并提供服务。也可以为数据查询提供负载均衡,以提高数据库性能。在下文中,简称从库。

在不同的数据库系统中,此术语有不同的表述,如 Oracle 称为复制站点(Replication Site),IBM DB2 称为目标服务器或目标数据库(Target Database),Sybase 称为复制数据库(Replicate Database),SQL Server 称为订阅服务器(数据库)等等。

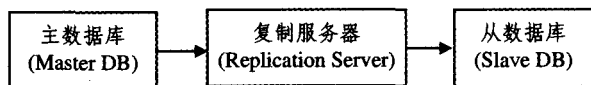


图 1 复制基本术语图示

2.2 基本复制类型

目前存在着两种类型的复制解决方法:同步数据复制和异步数据复制,也就是实时数据更新复制和延迟数据更新复制。

2.2.1 同步复制

同步复制是一种实时远程存取和实时更新数据的分发复制技术。这种技术可以保证应用的完整性,而且降低了应用的复杂性。当主库发生更改时,同步复制会立即把更新传播到从库,作为事务处理的一个部分。这种复制技术适用于那些对于实时性要求较高的应用。

如果主库和从库连接便利,网络畅通,那么同步复制可以立即保证主库和从库之间数据的一致性。但是这种复制方式也存在着很大的弊端:在从库网络无法连通,或者出现物理故障的情况下,同步复制将无法进行,因为这个进程必须等待主库和从库都提交(Commit)的情况下才能完成。假若从库端无法提交,那么主库端的事务也将无法进行下去而被强制回滚。也就是说,同步复制可能对主库的事务产生极大的阻碍和影响。

2.2.2 异步复制

异步复制是一种延迟远程存取和延迟传播对数据更新的复制分发技术。这种技术具有很高的可用性和很短的响应时间,但相比同步分发数据库技术显得复杂一些。

在异步复制的模式下,数据更新的快照会保存在主库中,然后根据实际的情况,在合适的时候传播到各个从库。当所有的从库都完成更新后,复制进程会给主库发一个确认信息,表示主、从库已经保持了一致性。也就是说,主库和从库的数据在一定时间内是不同步的,但这种不同步现象是临时的,复制的传播最终将保证主库和所有从库间的数据一致。

因此,异步复制中从库是在可行的情况下才进行更新。典型的情况是主库更新后的几分钟或者几个小时内。当主库更新后,如果不能和某些从库连接,那么可以延迟到以后某个时刻再次进行通信,但不会影响主库上事务的正常执行,较好地保证了健壮性的要求。

2.2.3 更新冲突

在同步复制中,由于主、从库必须同时更新成功事务才算完成,否则会回滚,因此数据总是保持一致的,并不存在更新冲突的问题。

而在异步复制环境中,对于所有应用最关键的就是要确保数据的一致性。如果允许多主复制的存在,也就是说,从库也可以独立进行数据更新,那么在同一时间对同一个表的同一行数据的同一列在两个不同的地点做更新,就会产生称之为更新冲突的错误。为保证数据的一致性,更新冲突必须被检测到并且处理,以确保在不同地点的数据元素保持同样的值。这样就产生了冲突避免或者冲突检测和消除的问题。一般来说,更新冲突可以通过定义解决规则,限制“所有权”到单一数据库或者将更新某个特定数据元素的权利限制到某一具体数据库的方法来避免。

2.3 各大主流数据库的复制比较

目前各大主流数据库都提供了自己的复制方案,譬如 Oracle 的 Advanced Replication, Sybase 的 Replication Server, IBM DB2 的 DataPropagator 和 DataJoiner, 以及 Microsoft SQL Server 的 Replication。它们大都基于日志(或触发器)捕捉更新,支持异步复制,支持异源数据(Oracle 还支持同步复制)。

2.4 复制类型的选择

为了提供高性能、高可用的复制方案,我们参考了各大主流数据库的复制功能,并且仔细权衡了各种技术的利弊,选择了异步复制方式。同时,考虑实际应用的需要,我们简化了复制模型,最终采用了一主多从(Single Master, Multi Slaves)的异步复制方式,这样既可以满足大部分日常应用的需求,又可以避免更新冲突。

考虑到复制方案的跨平台性,以及支持异源数据,支持各种主流数据库的复制和数据迁移的扩展性要求,我们设计了基于虚拟日志(Virtual Log)的更新捕捉机制。当数据更新(包括更改、插入、删除)操作在主库端发生时,触发器机制会捕捉更新,将其记录在一个预定义的数据表中(即虚拟日志)以供复制服务器使用。

因此,该复制方案设计了一个不与特定数据库相关的复制框架,既保证了复制的效率,又可以在此体系下方便地实现多种异源数据的复制,具备极大的可用性和可扩展性。

3 复制原理与步骤

3.1 复制系统原理概述

复制服务器方案的设计实施充分利用多数主流数据库的现有特征机制:触发器(trigger)和事务标识来进行工作。当更新、插入、删除等操作在主数据库监控的表上发生时,触发器机制会将这些事件记录在一个虚拟日志中(如上所述,在实现中使用的存储方式可以是在主库端专门为复制服务器所定义的一系列数据、表、文件等等),这些日志记录形成了数据更新快照。

复制服务器在用户设定的时间间隔内定期扫描快照,如果存在未应用于从数据库的更新,并且从数据库可通过网络进行连接通信,则将更新应用于从数据库,从数据库在完成更新后返回一个成功确认。如果从数据库暂时未能连通,则将更新延迟到下一个扫描周期再应用。在这期间,主数据库和从数据库间都能够独立地运行和正常地工作,互不影响。

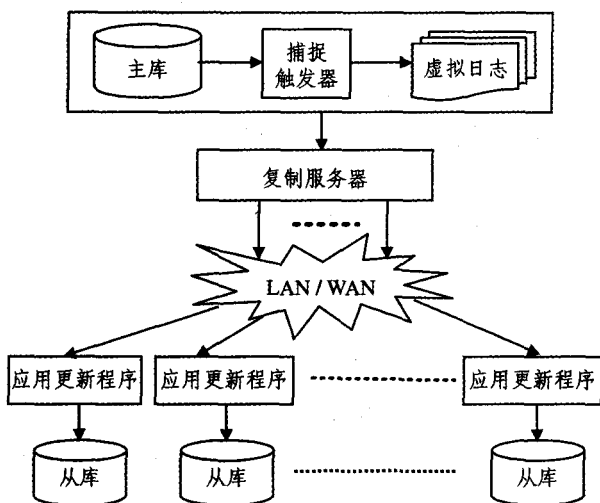


图2 复制系统

3.2 捕捉程序

捕捉程序运行在主数据库端，利用触发器机制来记录主库端的更新。

当主库中定义为需要进行复制的表执行某些特定的数据库操作(譬如 Update, Insert, Delete)时，会触发捕捉程序，将这些操作暂时记录在主库端的复制系统表中(即虚拟日志)，形成快照数据。

在初始化设置的时候，复制服务器会自动为需要复制的表建立触发器机制，以便进行更新捕捉(对于一些特定的数据库，可能无法使用此方式，比如 Microsoft Access, Microsoft Jet, Teradata 等)。

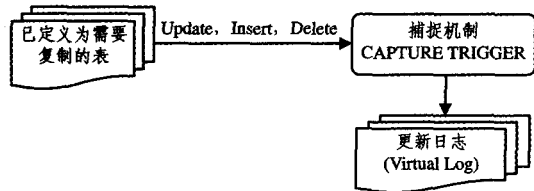


图3 更新捕捉

3.3 快照程序

快照程序按照一定的时间间隔(可以为几秒钟、几分钟、几小时等等，视具体的应用需要而定)定期扫描更新日志。如果存在未应用于从库的更新，则请求服务器控制程序，将更新传播至从库端的缓冲队列；如果不存在未应用的数据更新，则结束该次执行过程，等待下一次扫描。

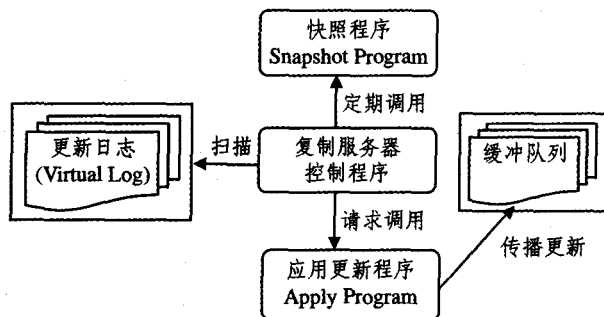


图4 快照程序

3.4 应用更新程序

应用更新程序从缓冲队列中读取主数据库的更新信息，

并且将更新应用于从数据库。

如果更新应用成功，则给服务器返回一个确认信息。如果因为某些原因导致更新失败，则返回一个错误信息。

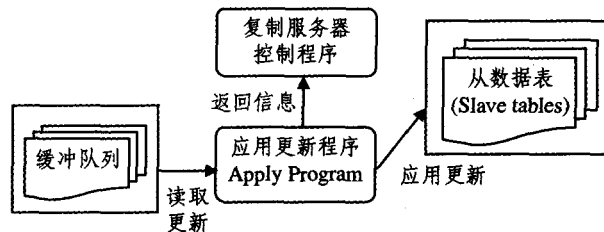


图5 应用更新程序

3.5 复制步骤

3.5.1 一主一从(Single Master, Single Slave)

下面以一对一的主、从数据库复制(Single Master, Single Slave)来简要说明复制的过程。一对多(Single Master, Multi Slaves)的情况与此类似。

首先是初始化，即全更新过程(full-refresh)。它将主库中需要复制的表完全拷贝到从库中(包括表定义和所有数据)，从而使主、从数据库保持了某个时刻的数据一致性。

然后就进入了通常的事务更新复制，也就是以下的异步更新流程。此时复制服务器的异步复制通过以下几个步骤来实施：

(1)产生标识和快照

主库中可能含有从库所不知道的数据。触发器机制在主库更新的时候被触发，捕捉程序根据更新的信息产生一个相关标识，将其内容记录在主库端的复制系统表(即虚拟日志)中，形成数据更新快照。



图6 复制步骤1

(2)将更新传送到从库

复制监控程序定期调用快照程序进行快照扫描，发现已经更新的快照记录，便在指定时间间隔内(如果无法连接成功，则推迟一个周期，至网络可达的时候)将更新传送到从库。

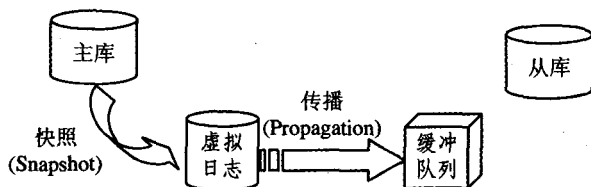


图7 复制步骤2

(3)对从库应用更新

在从库处，可用的更新信息被反馈到应用更新程序上。这时候，复制服务器会启动应用更新程序，在从库端进行数据的更新，以便与主库端的数据保持一致。

(4)同步主库

最后一步，从库的数据更新成功应用后，会给复制监控进程返回一个操作成功的信号，使主库确认更新成功，从当前的

复制任务中清除信息。

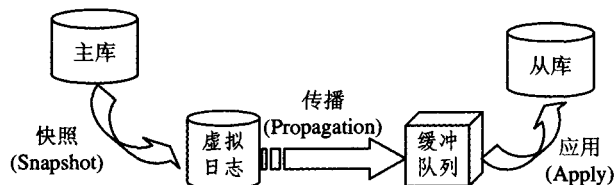


图 8 复制步骤 3

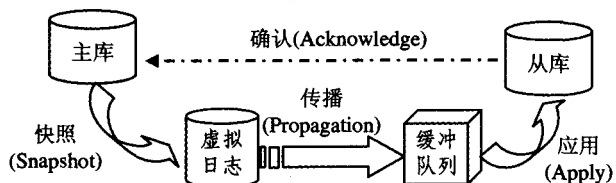


图 9 复制步骤 4

3.5.2 一主多从(Single Master, Multi Slaves)

一主多从(Single Master, Multi Slaves)的复制流程与一主一从类似,如图 10 所示。

需要说明的是,复制服务器为每一个从库开启一个进程,用于复制的监控,它们之间是独立工作,互不干扰的。也就是说,某一个从库的复制失败(可能是由于网络不可达、从库无法连接等等),并不会影响其他从库的复制进程,它们仍然可以正常进行工作。

另外,某个时刻无法连接并进行正常复制的从库监控进程,仍然能够定期唤醒,尝试进行连接。如果连接成功,那么监控进程会扫描该从库自最后一次更新后的所有快照记录,并将更新应用于该从库。

此种工作模式提高了程序的容错能力,能够保证主、从数据库的独立运行,并最终保持数据的一致性。

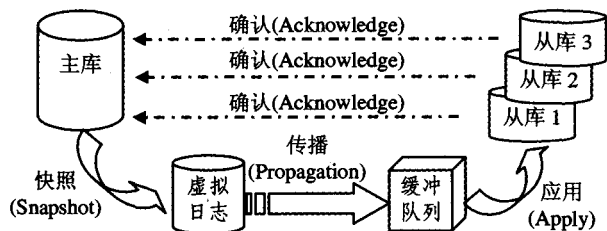


图 10 一主多从复制过程

4 实施

我们选取国产数据库的典型——人大金仓的 KingbaseES 作为复制方案实现的数据库系统平台,根据上述的原理和系统框架,使用 JAVA 语言具体实现了复制服务器,达到高可用性和高扩展性的目标,取得了良好的效果。

复制服务器的运行需要在主、从库端定义一些辅助表和存储过程,下面简要列出一些关键部分。

在主库端需要定义的复制辅助表:Replication_log(用于记录更新的日志)、Master_sync(主库同步记录)、Master_tables(需要复制的表)等。

在从库端需要定义的复制辅助表:Slave_sync(从库同步记录)、Slave_tables(需要复制的表)等。

另外,为了捕捉主库的更新,还需要为主库定义一些 PL/SQL 的存储过程,如捕捉更新的 Replication_catchlog、获取

快照的 Server_getSnapshot 等。

为了达到高效率的数据复制,我们还设计了图 11 所示的快照数据结构,以便批量应用更新。

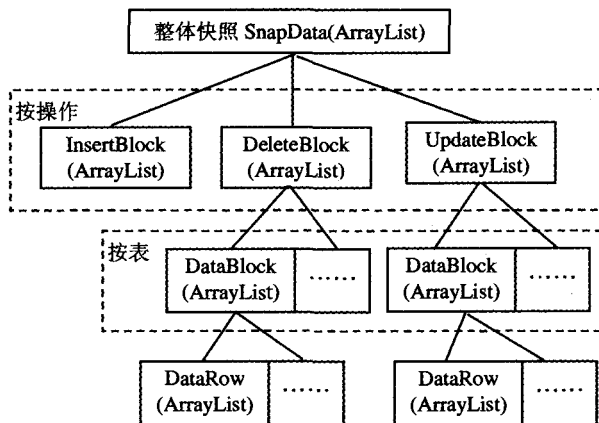


图 11 复制服务器快照数据结构

4.1 服务器运行关键流程

该复制服务器的运行过程大体分为 3 个部分:

- (1)更新触发部分,即主表的更新引起复制服务器系统数据的相应更新,这些数据主要存储在主库端的日志表中;
- (2)检测过程,即在指定的时间间隔里检测主库中是否有更新;
- (3)如果检测到更新,对从库应用更新的过程。

这 3 个主要过程的大体流程如图 12 所示。

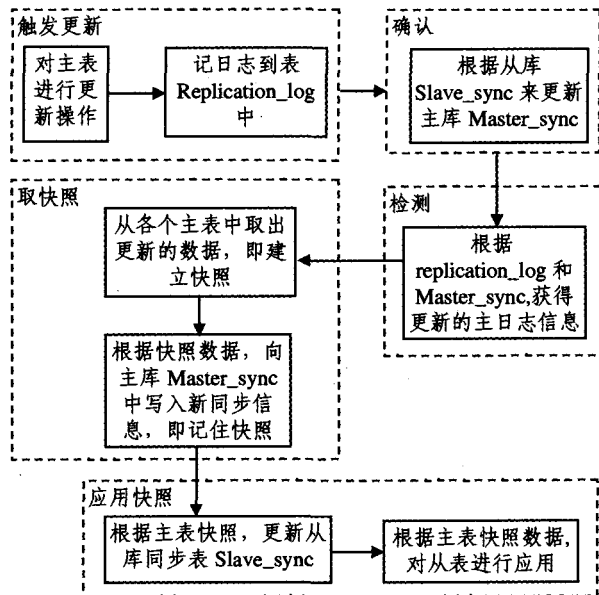


图 12 复制服务器运行关键流程图

4.2 实际测试结果

我们采用衡量数据库性能的 TPCC 基准测试,通过对比不加载复制服务器/加载复制服务器两种情况下的数据来比较复制服务器的实际性能。主库服务器配置为双 CPU(P4 2.0G, 1G RAM),从库为普通 PC 机(P4 2.0, 1G RAM),复制服务器和主库运行于同一服务器上。加载 10 个 Warehouse(每个 Warehouse 10 个连接),在不使用复制服务器、使用复制服务器(设置为每隔 30s 同步一次)的情况下各运行测试 3600s(1h),比较其性能差异,结果如图 13 所示。

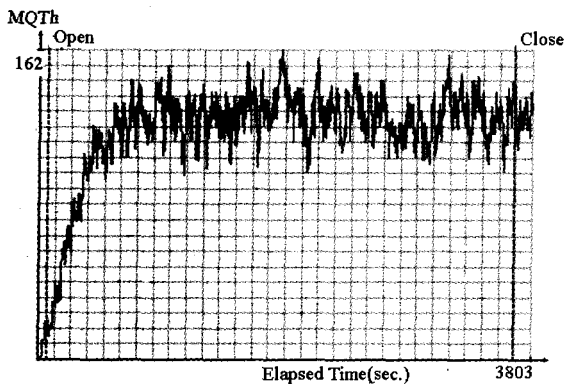


图 13 不加载复制服务器运行 3600s

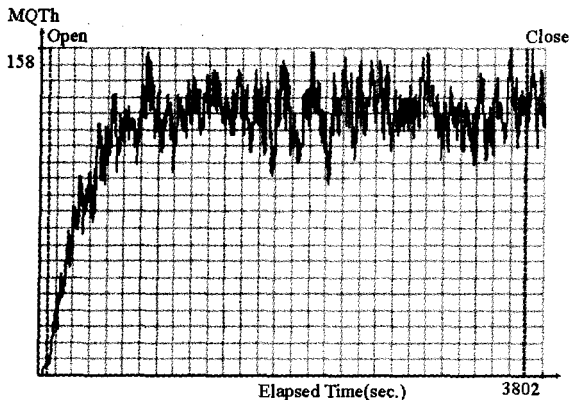


图 14 加载复制服务器运行 3600s

	不加载复制服务器	加载复制服务器
MQTH	122.1 tpmC	121.2 tpmC
Total Completed	7325	7277

从上面的测试数据可以看出,10个 Warehouse 远没有达到压力的上限,两个测试都顺利地通过(在第二个测试中,从库的数据也得到了正确准时的同步)。虽然使用复制服务器要在主库上定义、执行 Trigger、存储过程等一些操作,对主库的性能带来了一定的影响,使得测试中总的提交事务相对少了一些,反应时间也较不加载时稍长,但是总体来说,主库在使用复制服务器的情况下性能稳定,可用性强。

结论 本文从实际应用的角度出发,为大型数据库系统

设计并实现了一种基于虚拟日志的(由触发器触发生成)的高性能/高可用性的复制解决方案。该复制方案采用一主多从模式(Single Master, Multi Slaves),使用异步分发技术,实行延迟远程存取和延迟传播技术对数据进行更新,提供跨平台支持。

同时,我们针对国产大型数据库系统 KingbaseES 具体实现了该复制方案,从而提供了一个现实的、可扩展的复制系统框架。

由于采用触发器机制捕捉主库的数据更新,而主流的数据库系统均支持触发器,因此该方案并不依赖于特定的数据库,可以方便地实现对异源数据的复制支持,具备良好的扩展性。针对不同的数据库系统,如果作为主库,需要根据该数据库的特点进行触发器的定义并实现相应的快照扫描接口;如果作为从库,需要定制相应的更新应用程序接口。目前,我们也已经成功地实现了主库对 Oracle、SQL Server 等异源数据库的复制。

在实际的测试中,复制服务器也表现出了优异的稳定性和可靠性,能够很好地与数据库系统配合,提供稳定高效的数据复制解决方案,具有准时复制、响应时间短等优点,在及时同步业务数据、灾难备份恢复,改善决策支持系统等许多关键服务中有很高的应用前景。

参考文献

- Oracle. Replication with Oracle Streams[M/OL]. June 2002. 3~10
- Oracle. Oracle9i Database Online Documentation (Release 2 (9.2)) Advanced Replication[M/DK]. Part Number A96567-01, 2002
- Microsoft. SQL Server 2000 联机丛书[M/DK]. 2000 复制部分
- Rodolphe M, Silvia C, Joji J, et al. IBM Replication Solutions for Pervasive Computing with DB2 Everyplace and DB2 Satellite Edition[M/OL]. ibm.com/redbooks, April 2001
- Shirai T, Crompton G, Priest V. Migrating to the IBM Replication Solution[M/OL]. ibm.com/redbooks, February 2001
- Gu Li-jun, Budd L I, Cayci A, et al. A Practical Guide to DB2 UDB Data Replication V8[M/OL]. ibm.com/redbooks, December 2002
- Sybase. Replication Server 管理指南[M]. 2000. 1~44
- Sybase. Replication Server 参考手册[M]. 2000. 415~426
- BaseSoft. Kingbase ES V4.0 程序员参考手册[M]. 2004
- 石晓群. Oracle 8 高级数据复制技术[N/OL]. 2001-03-28. <http://www0.ccidnet.com/tech/guide/2001/03/28/58-1900.html>

(上接第 57 页)

参考文献

- Blakley G R, Meadows C, Purdy G B. Fingerprinting Long Forging Messages. Crypto'85, LNCS218, Springer-Verlag, Berlin, 1986. 180~189
- Boneh D, Shaw J. Collusion-Secure Fingerprinting for Digital Data. Crypto'95, LNCS963, Springer-Verlag, Berlin, 1995. 452~465
- Pfitzmann B, Schunter M. Asymmetric Fingerprinting. Eurocrypt'96, LNCS1070, Springer-Verlag, Berlin, 1996. 84~95
- Pfitzmann B, Waidner M. Anonymous Fingerprinting. Eurocrypt'97, LNCS1233, Springer-Verlag, Berlin, 1997. 88~102
- Ateniese G, Camenisch J, Joye M, Tsukik G. A Practical and Provably Secure Coalition-Resistant Group Signature Scheme. Crypto2000, LNCS 1880, Springer-Verlag, Berlin, 2000. 255~270
- Wang Yan, Lü Shuwang, Liu Zhenhua. A Simple Anonymous Fingerprinting Scheme Based on Blind Signature. ICICS2003, LNCS2836, Springer-Verlag, Berlin, 2003. 260~268
- Camenisch J. Efficient Anonymous Fingerprinting with Group Signatures. ASIACRYPT2000, LNCS1976, Springer-Verlag, Berlin, 2000. 425~428
- Domingo-Ferrer J, Herrera-Joancomarti J. Efficient Smart-card Based Anonymous Fingerprinting. Smart Card Research and Advanced Application-CARDIS'98. Springer-Verlag, Berlin, 1998. 221~228
- Domingo-Ferrer J. Anonymous Fingerprinting Based on Committed Oblivious Transfer. Public Key Cryptography'99, LNCS1560, Springer-Verlag, Berlin, 1999. 43~52
- Pfitzmann B, Sadeghi A R. Coin-Based Anonymous Fingerprinting. Eurocrypt'99, LNCS1592, Springer-Verlag, Berlin, 1999. 150~164
- 李勇, 杨波, 华翔. 一种高效匿名的数字指纹方案[J]. 西安电子科技大学学报, 2003, 30(3): 394~398