第 35 卷 第 6 期 计 算 机 工 程 2009 年 3 月
Vol.35 No.6 Computer Engineering March 2009
·软件技术与数据库· 文章编号：1000—3428(2009)06—0069—04 文献标识码：A 中图分类号：TP303

# 存储网络自适应容错协议和算法的研究

韩德志[1,2]，傅 丰[3]
(1. 广东外语外贸大学信息学院，广州 510420；2. 中山大学广东省信息安全重点实验室，广州 510275；
3. 黄淮学院计算机系，驻马店 463000)

**摘 要：**针对传统复制技术存在的问题，在研究复杂存储网络高可用系统结构的基础上，分别设计了面向恢复的自适应复制协议和算法，它们允许系统中同时有多个复制组，且每个复制组成员可动态变化。实验结果表明，新协议和算法能适用于各种复杂的存储网络环境。
**关键词：**存储网络；高可用；复制协议；容错

# Research on Self-adaptable Fault Tolerance Protocol and Algorithm in Storage Network

HAN De-zhi[1,2], FU Feng[3]
(1. School of Information, Guangdong University of Foreign Studies, Guangzhou 510420; 2. Guangdong Key Lab of Information Security, Sun Yat-sen University, Guangzhou 510275; 3. Dept. of Computer, Huanghuai University, Zhumadian 463000)

**【Abst*RA*ct】**Aiming at the problems existed in traditional replication technology, and on basis of analyzing the High Availability(HA) system structure for complex storage network, a Recovery-Oriented self-Adaptable Replication Protocol(ROARP) and a Recovery-Oriented Request Recording and Checkpoint(RORRC) are designed, which allows multiple replication groups exist in one system, and each member of the group is active. Experimental results show these novel protocol and algorithm are fit for varied complex storage networks.
**【Key words】**storage network; High Availability(HA); replication protocol; fault tolerance

## 1 概述

在分布式环境中，面向恢复的容错技术可以分为 3 类：检查点技术，系统日志技术和模块冗余技术。检查点算法在当前国际学术界研究较多，目前检查点算法已从研究阶段走向实用化阶段，如 condor, Fail-safe PVM, Dome, Cocheck 等[1]。检查点算法可分成 2 类：独立检查点和一致性检查点。系统日志文件是预测系统故障的主要工具，通过日志文件，系统管理员可以再现导致故障的事件，在大多数情况下可以发现故障的原因。模块复制技术主要有 2 种：主动复制和被动复制。在主动复制中，复制实体的所有副本同时响应客户请求并发送应答，副本的失效对客户是透明的，但消耗系统资源；在被动复制中，只有主实体 primary 执行客户请求，并将必要的信息通过某种方式传给复制副本实体 backup，其失效恢复时间长，但节约系统资源。典型的复制协议包括 Active Replication, Primary Backup 和 ROWA 等[2]。但目前多数复制技术都存在以下缺陷：(1)在设计复制协议和复制算法时施加了许多限制，从而导致它们大多只能适应某一类型的分布式应用；(2)复制协议与通信协议是一种紧耦合关系，不能将复制协议同通信协议有效地分离，从而使复制协议或复制算法不能完全对用户透明；(3)往往只从局部考虑可用性，未能从整体上考虑可用性。

本文在研究融合了 NAS, SAN 和 iSCSI 的 USN 高可用技术[3]以及存储网络服务器端 I/O 请求响应进程迁移技术和存储子系统文件数据复制技术的基础上，针对传统的复制技术存在的问题，提出一种针对高可用存储网络的面向恢复的自适应的复制协议(Recovery-Oriented self-Adaptable Replication Protocol, ROARP)，在此基础上提出面向恢复的动态容错算法(Recovery-Oriented Request Recording and Checkpoint, RORRC)，这在满足复杂存储网络系统可用性需求的同时，也可支持不同存储网络系统对自适应性的要求。

## 2 USN 的高可用结构

图 1 中方框内的部分就是 USN 的系统结构。

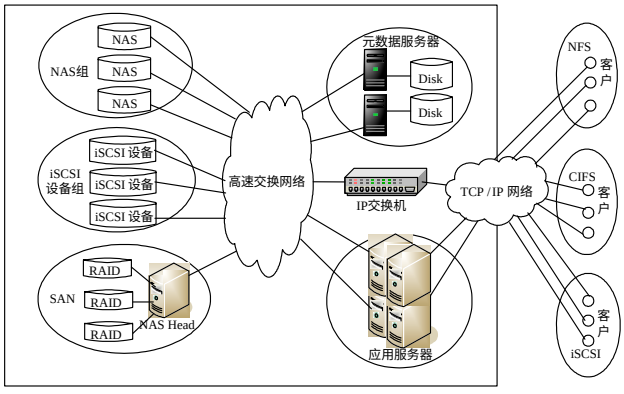


**图 1 USN 的系统结构**

**基金项目：**国家“863”计划基金资助项目(2007AA01Z449)；国家自然科学基金资助项目(60673191)；广东省信息安全重点实验室基金资助项目(200603)；河南省科技计划基金资助项目(072100451230)；广东外语外贸大学科研创新团队基金资助项目(GW2006-AT-005)
**作者简介：**韩德志(1966－)，男，教授、博士，主研方向：高可用、高扩展的海量存储系统；傅 丰，副教授
**收稿日期：**2008-10-10 **E-mail：**han_dezhi88@tom.com.cn

它由应用服务器组、元数据服务器组、NAS 组、iSCSI 设备组及 SAN 设备组组成。这几部分之间通过高速交换网络连接，在 USN 中将不同的存储设备分成不同的组主要便于整个 USN 高可用系统的研究与设计，应用服务器组是指加载了 USN 文件系统核心扩展与相关进程并用来接收客户 I/O 请求的服务器节点。这些应用服务器由 Unix 服务器、Windows 服务器以及加载了 iSCSI 启动器模块和目标器模块的服务器节点组成。元数据服务器组负责管理 USN 的文件数据分布、USN 目录文件及普通 USN 文件元数据的存储。NAS 组中包括多种 NAS 设备，且 NAS 设备又可分为多个 NAS 子组，NAS 组中的 NAS 主要存放客户端的文件数据信息，支持 NFS 客户和 CIFS 客户的文件 I/O 访问。iSCSI 设备组主要由支持 iSCSI 协议的存储设备组成。SAN 设备组主要由各种支持光纤通道协议的存储设备组成，如 FC RAID、FC 磁带库、FC 光盘库等。

图 2 是 USN 的高可用结构模型。其中，Client 是客户端；$RG_1$ 表示第 1 个复制组；$RG_k$ 表示第 $k$ 个复制组。$\text{Replica}_1$，$\text{Replica}_2$，…，$\text{Replica}_n$ 表示 $RG_1$ 中有 $n$ 个复制实例，即一个写请求可将数据写到 $n$ 个存储节点，一个读请求可以从 $n$ 个存储节点的任意一个中读取。HAM 协助主复制代理 *PRA* 完成复制组的创建、*PRA* 的失效监测及替换、复制组中存储节点的失效监测及处理。HAM 是存储网络系统高可用性系统模块，主要由高可用管理模块 HA_ADMIN、高可用核心模块 HA_KERNEL、高可用代理模块 HA_AGENT 组成，同时有相应的监测机制、日志记录机制、失效替换和状态恢复机制[4]。

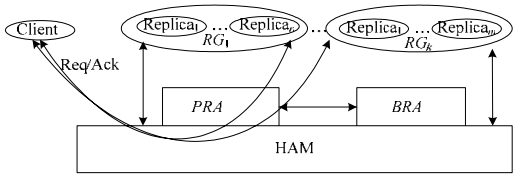


**图 2　USN 的高可用结构模型**

## 3　面向恢复的自适应复制协议

针对传统支持高可用的复制协议存在的问题，ROARP 采用：(1)在应用服务器端引入复制代理来适应遗留系统需求，保证使用不同协议的客户都能得到存储网络提供的高可用服务；(2)为保证存储代理的高可用性，引入主-从复制代理结构，主代理响应客户请求并在高可用模块 HAM 的控制下定期把接收请求状态信息复制给备份代理，并协助日志管理器 LM 写日志检查点文件；(3)为保证用户 I/O 数据信息的高可用性，将用户所写数据信息被同时写入 2 个或多个存储节点；(4)复制实例数(每个复制组成员数)可根据用户信息高可用性级别的需求以及存储网络中存储节点的性能，由系统自适应地配置。

### 3.1　ROARP 模型

ROARP 是基于分布式的存储网络，采用冗余分层设计，按不同请求的高可用性要求，复制代理可自适应地将 I/O 请求递交给不同的复制组。一个复制组可以包含多个复制实例，即对应多个存储节点。一个存储节点可以运行多个复制实例，即对应多个复制组。

用户在注册成为存储网络用户时，存储网络高可用模块 HAM 为不同的用户分配不同的高可用级别，并且控制 *PRA*(主复制代理)在接到写请求时根据不同的级别的用户请求形成不同数目的复制实例数目，对应不同的复制组。存储网络环境下的 ROARP 模型如图 3 所示。

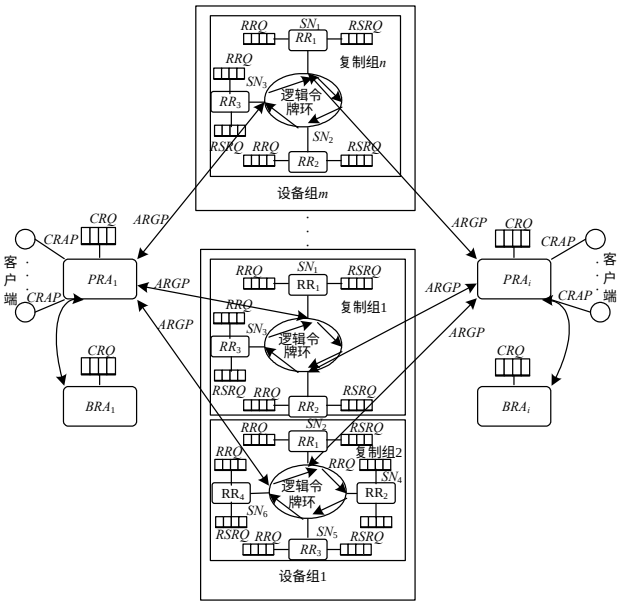


**图 3　存储网络环境下的 ROARP 协议模型**

整个存储网络被分成 $m$ 个设备组，每个设备又包括若干个存储节点 *SN*，在某一时刻 $t$ 所有设备组动态分成 $n$ 个复制组。设备组 1 有 2 个复制组，其中，复制组 1 包括 3 个复制实例，分别位于 3 个存储节点上；复制组 2 包括 4 个复制实例分别位于 4 个存储节点上，其中，复制组 1 的复制实例 $RR_1$ 与复制组 2 的复制实例 $RR_2$ 同位于一个存储节点 $SN_2$ 上，复制服务有自己的内部状态和服务方法。客户端服务 I/O 请求体现为相应的方法调用，其中，方法分为“读”方法和“写”方法。

在 ROARP 中引入复制代理 *RA* 的主要目的是对客户端屏蔽存储网络的异构性，使持各类型协议用户能通过复制服务透明地访问存储网络。*RA* 除对持不同协议的用户提供透明地访问存储网络外，还能对不同的用户提供不同级别的高可用性的复制服务。

在该模型中，当 *PRA* 每接到一个客户 I/O 请求时，一方面放入自己的客户请求队列 *CRQ* 中，并且在日志管理器 *LM* 的控制下按固定时间间隔写日志记录并对备份自制代理 *BRA* 的 *CRQ* 进行更新。*BRA* 不作递交操作，只有当 *PRA* 失效且接替其业务时，才进行递交操作。*BRA* 通过心跳机制监测 *PRA* 的工作状态，一旦发现 *PRA* 失效，*BRA* 立即通知 HAM，由其确认后 *BRA* 才接替 *PRA* 的业务。在图 3 中，主复制代理 $PRA_1$ 的备份复制代理是 $BRA_1$，主复制代理 $PRA_i$ 的备份复制代理为 $BRA_i$。

存储网络同时有多个复制组，每个复制组中的复制实例构成一个逻辑令牌环，所有的复制组构成多令牌环结构。在任意时刻，一个复制组中只有一个复制实例持有令牌。*PRA* 可通过代理复制组协议 *ARGP* 向复制组发送信息和接收复制组成员的消息。*PRA* 向复制组发送的消息是一种广播消息，即发送的消息广播到复制组的所有成员，但应答消息只能由在接收广播消息时持有令牌的成员发送，其他成员的应答消息只能发送到接收消息时持有令牌的成员，复制组成员之间使用的是一种令牌协议(Token Ring Protocol, TRP)。*PRA* 在接到客户请求消息 *rm* 时，在 HAM 和元数据服务器的控制下形成一个四元组复制代理 *RA* 请求消息(*ran*, *dgn*, *rgn*, *rm*)，然后

将该消息广播到对应的复制组的所有成员。其中，*ran* 是复制代理号；*dgn* 是设备组号；*rgn* 是复制组号；*rm* 是客户端请求信息。当复制组中持有令牌的成员接到四元组请求消息时，它把该消息加上自己的组成员号 *gmn* 随令牌信息一起多播给复制组中所有其他成员，然后将令牌按序传给下一个成员，如此同时将加 *gmn* 的四元组信息存入自己的 *RRQ*(Replica Request Queue)中。复制组中的其他成员接到持令牌成员的令牌消息后与 *RA* 广播消息进行对比若相匹配，则将四元组消息(*ran, dgn, rgn, rm*)和 *gmn* 一起放入自己的 *RRQ*，若不匹配则丢弃。每个成员持令牌的时间是受限制的。当超过限制的时间时，若持有令牌的成员仍未收到 *RA* 多播的请求消息，则它多播一个空令牌消息给复制组的所有成员告诉它们令牌已转移后继者。为保证整个存储系统的高可用性对每个复制实例都有一个复制请求队列 *RRQ* 和一个复制响应应答队列(Replica Acknowledging Queue, *RA*Q)。*RRQ* 和 *RA*Q 有 2 个主要作用：(1)对请求和应答起缓存作用；(2)保证用户请求的连续性。

### 3.2 数据一致性

由图 3 可知，当多个复制组共用某一个存储节点时，即多个复制实例都位于同一个 *SN* 上，并且各个复制实例对文件的操作有读、有写，这很容易引起数据的一致性问题，ROARP 协议在解决数据一致性问题具体方法：客户所有的 I/O 请求都先经过 HAM 处理，即为每个请求加上一个全局顺序号，HAM 然后往下传递，当 *PRA* 拦截到 I/O 请求时，首先分析所请求的文件信息，如果是已有的文件，则根据元数据信息将其发往对应的复制组；如果是新文件则根据系统负载平衡和可用性要求将其发已有的复制组，或 HAM 新创建的复制组。每个复制组的复制实例在执行请求队列中的请求时，主要依据全局顺序号小优先的原则。这样可充分保证整个系统的全局一致性。

### 3.3 ROARP 的性能分析

ROARP 是面向恢复的自适应复制协议，它提供两级复制，一级是应用服务器端的软件模块(*RA* 模块)复制，另一级用户所写数据复制(通过复制实例在多个存储节点中复制)。软件模块复制能在客户端保证用户 I/O 请求响应的连续性，即当主应用服务器或其上的主复制模块出故障时，将由冗余的应用服务器上的备份复制代理接替其的工作；用户数据冗余保证当存储节点或所存数据出故障时，由冗余的存储节点代替其响应用户对数据的 I/O 请求。ROARP 是种全新的适应复杂存储网络环境的支持多令牌协议结构的高可用复制协议，与 ROWC, Active Replication 和 Primary Backup 相比有如下优点：(1)ROWC 引入单令牌结构只能适应简单的网络环境，ROARP 引入了多令牌结构可以适应各种复杂的存储网络环境。(2)ROWC 没有充分考虑复制适配器 SA 失效和复制实例在响应客户请求失效的情况系统的可用性问题，ROARP 充分考虑到复制代理 *RA* 失效、复制组中复制实例失效或存储节点失效时客户 I/O 请求的连续性。(3)由文献[2]可知，当 ROWC 在服务器为 $n(n\geq 2)$时，复制系统提供的可用度为 $A_n=1-P_n=1-\left(\frac{\lambda}{\mu+\lambda}\right)^x$，其中，$1\leq x\leq n$，而 Active Replication 和 Primary Backup 的可用度都为 $A_2=1-P_2=1-\left(\frac{\lambda}{\mu+\lambda}\right)^2$，由文献[4]可知，ROARP 的可用度为 $A_n'=1-P_n=1-\left(\frac{\lambda}{\mu+\lambda}\right)^n$。所以，ROARP 的可用度优于 ROWC, Active Replication 和 Primary Backup。(4)ROARP 能在复杂的网络环境中为整个系统提供高的可用性，而 ROWC, Active Replication 和 Primary Backup 只能适用于同构的双机或多机构成的集群系统。

## 4 面向恢复的自适应容错算法

针对静态复制算法存在的问题，结合已有的日志检查点方法，在 ROARP 协议的基础上，提出一种面向恢复的动态容错算法 RORRC。与以往的算法相比，该算法有以下优点：

(1)基于 RORAP 协议，该项算法有效地综合了被动复制和日志检查点机制；

(2)实现了复制协议与可靠通信协议的分离，一方面复制协议无需涉及消息传递细节，另一方面可以无需改变复制协议就更好地利用通信协议或底层网络拓扑来提高性能；

(3)该算法所采用的容错机制对客户完全透明。

### 4.1 算法描述

为便于描述，这里将 HAM 模型中的 client 假设是个生成用户 I/O 请求的应用程序逻辑，它同 HAM 及 *PRA* 同时位于存储网络的服务器(可以是应用服务器，也可以是元数据服务器)。RORRC 算法可描述为：

(1)接收用户 I/O 请求的应用服务器 *AS* 或 *PRA* 可保证客户方请求的连续性，即重定向客户请求到一个 *BRA*，并保证 at-most-once 语义，即客户只会构造一次请求且最多只会被执行一次。因为一个客户请求被 *PRA* 拦截后在 LM 通过相应的接口将其记录在日志文件的同时也复制到 *BRA* 队列中。当 *PRA* 失效时，由 *BRA* 根据其 CRQ 保证客户请求的连续性，即保证 at-most-once 语义。

(2)当处理客户请求的组信息 group_inf 与当前的 group_inf 不一致时的情况。为了区别不同的客户和不同的请求，*PRA* 代理在接到一个客户请求 *req* 时，在其请求信息中加上每个客户的唯一标识 c_id 和每个请求的唯一标识 req_id 及对应的复制组信息 group_inf，即在其 *CRQ* 中存放的 cri=<C_id, req_id, req, group_inf>，并且将 *cri* 记录在日志文件和复制到对应的 *BRA* 的 *CRQ* 中。同时形成 rri=<*ran, dgn, rgn, cri*>并广播到复制组的所有成员。复制组每个成员能根据此结构识别重复的请求，保证 RORAP 中的 ROWARO(Read One Write All and Reply One)的语义及上述中提到的 at_most_once 语义。其中，*rri* 为代理请求信息；*rgn* 是个复制组的唯一标识，而 group_inf 记录一个复制组相关的信息。在 RORAP 协议中，每个复制实例在其复制请求队列 RRQ 和复制请求应答队列 RSRRQ 保存请求信息和应答信息。

当服务器或 *PRA* 失效时，由 HAM 模块启动 *BRA* 作为 *PRA*，从而保证客户 I/O 请求的连续性。当客户通过 *PRA* 发请求到复制组时，存在 group_inf 已过时的情况(例如，假设当前引用复制组为 group_inf={$SN_1$,$SN_2$,$SN_3$}，客户从名字服务或元数据服务器中获得 group_inf 后，提供高可用 HAM 中的失效监测代理会监测到失效并完成失效处理，将失效的存储节点 $SN_1$ 从复制组中删除，新的组信息变为 group_inf={$SN_2$,$SN_3$})。为了让服务方确定客户请求的是否是当前的 group_inf，引入一个复制组版本号 rgvn(replica group version number)，标识 *PRA* 产生的组引用 group_inf 的版本。每当组成员发生变化时，HAM 产生一个新的 group_inf，并修改其

对应的版本号 *rgvn*。

当 *PRA* 每拦截一个客户 I/O 请求时，先要查看所发请求所对应的元数据信息，然后比较元数据中所存文件对应的复制组 group_inf 和 *rgvn* 与服务方所持的 group_inf 和 *rgvn* 的匹配情况并作相应的处理，这里的服务方是指整个复制组。

### 4.2 算法证明

本节将从正确性和有效性 2 个方面对 RORRC 进行证明。正确性是指客户请求对应 *PRA* 的唯一性和复制组所有成员所存信息的一致性，有效性包括动态复制特性和 ROWARO 语义的满足。

**定理 1** RORRC 算法能满足客户请求对应 *PRA* 的唯一性：在任何时候，系统虽然有多个 *PRA*，但对任一客户端的 I/O 请求，系统都有一个唯一的 *PRA* 与之对应。

证明 假设某个时刻 $t$，系统中的客户请求对应一对 *RA*，如果 *PRA* 不失效，由 *PRA* 执行该客户请求，在这个过程系统只有一个 *PRA*；假设在时刻 $t_1(t_1>t)$时，*rap* 失效，存储网络的 HAM 中的失效监测器会在 $t_2(t_2=t_1+\delta$，$\delta$ 为 TIMEOUT 值)时刻 *PRA* 失效，将失效报告给 HAM 中的失效处理部件，由该失效处理部件在时刻 $t_3(t_3=t_1+\delta+\delta_1)$启动 *PRA* 的 *BRA*，即 *BRA* 作为 *PRA*。所以，在任何时刻一个客户请求只能有一个 *PRA*。

**定理 2** RORRC 算法能满足复制组成员信息的一致性：每个客户请求执行完后复制组所有成员执行该请求时的状态是一致的。

证明 设复制组 *RG* 有 $k$ 个成员 $SN_1,SN_2,\cdots,SN_k$，当客户请求是写请求时，由于该请求的 *PRA* 将该写请求广播给 *RG* 中的 $k$ 个成员，由 RORAP 中复制组通信协议可以保证 $SN_1,SN_2,\cdots,SN_k$ 每个成员中所写信息的一致性。当客户请求是读请求时，由于只有接请求时持令牌的 $SN_j$ 响应该读请求，RORAP 中复制组通信协议同样可使所有成员状态的一致性。

**定理 3** RORRC 能满足动态复制特性：复制成员的动态变化不会影响复制组一致性。

证明 复制组成员的动态变化主要包括成员的动态加入或删除，下面分 2 种情况证明：

(1)当删除复制组成员时，若成员在持令牌时未接收过客户请求，删除该成员对整个系统未有任何影响。

(2)当删除复制组成员时，若成员在持令牌时接收过客户请求，删除该成员对整个系统有影响。因为必须由该项成员向对应的 *PRA* 发应答消息。为保持一致性，必须由相应的成员向 *PRA* 发请求完成应答消息。这可通过存储网络高可用系统来保证。在 HAM 中，该复制组所对应的 HA_agent 监测节点失效并通告给 HAM，由 HAM 通过 *LM* 和 *PRA* 重发最近的检查点的 *cri* 给该复制组的所有成员。复制组的所有成员比较发现有重复请求则丢弃，非重复请求则按一致性服务要求执行，这样可保证所有节点的状态一致性。

(3)若增加 *SN* 时，只要该 *SN* 在加入该复制组时的时刻与其成员同步即可。因为 *SN* 加入后又成为一个新的复制组，与原复制组在逻辑上相互独立。

总之，复制组成员的动态变化会带来某些一致性问题，HAM 中的相关部件可保证每个复制组成员的状态一致性。

**定理 4** RORRC 能保证 ROWARO 语义，即每个客户读请求只能被执行一次，写请求必须在复制组中所有成员中执行，并且读/写请求只能由一个成员应答。

因为由 RORAP 协议可知，所以其证明略。

## 5 结束语

提出一种面向恢复的自适应复制协议 RORAP 和面向恢复的自适应复制算法 RORRC，与传统的复制技术相比，具有更好的可扩展性和更好的适应性。对于 RORAP 而言，由于其采用多令牌结构，因此比传统采用单令牌的复制协议具有更好的整体性能，因为它允许系统中同时有多个复制组，每个复制组成员可动态变化，并且由性能、容量相近的存储节点组成，这就使复制协议对系统的负载平衡和 I/O 调度有更好的适应性。RORRC 算法由于有效地综合了被动复制和日志检查点机制，因此能克服传统静态复制算法存在的不足。

### 参教文献


[1] Litzkow M J, Livny M, Mutka M W. A Hunter of Idle Workstations[C]//Proc. of the 8th Int'l Conf. on Distributed Computing Systems. [S. l.]: SPIE Press, 1998.

[2] 赵 东, 姚绍文, 周明天. 一种适应性协议的研究与设计[J]. 电子学报, 2002, 30(12): 1991-1994.

[3] 韩德志, 余顺争, 谢长生. 融合 NAS 和 SAN 的统一存储网络系统的设计与实现[J]. 电子学报, 2006, 34(11): 3012-3017.

[4] 韩德志. 统一存储网络高可用关键技术研究[D]. 武汉: 华中科技大学, 2005.


编辑 陈 文

(上接第 68 页)

## 4 结束语

本文将过程模型作为一种可视化的工作流建模语言，根据过程模型的语法和工作流系统的特点，从活动的就绪变迁、活跃变迁、完成变迁等状态变迁为过程模型定义了执行语义，为实现其模拟、验证和执行奠定了基础。目前已成功开发了基于 VPML 和过程模型执行语义的可视化建模环境 PMBE 和模拟环境 PMSE，并已进行推广应用。

### 参考文献


[1] Hollingsworth D. The Workflow Reference Model[Z]. Workflow Management Coalition. 1994.

[2] 赵志崑, 盛秋戬, 史忠植. UML 活动图描述工作流模型的执行语义[J]. 计算机研究与发展, 2005, 44(10): 1801-1807.

[3] Endl R, Knolmayer G, Pfahrer M. Modeling Processes and Workflows by Business Rules[C]//Proc. of the 1st European Workshop on Workflow and Process Management. Zurich, Switzerland, [s. n.], 1998: 47-56.

[4] 王 聪, 王智学. UML 活动图的操作语义[J]. 计算机研究与发展, 2007, 42(2): 300-307.

[5] 周伯生, 张社英. 可视化过程建模语言 VPML[J]. 软件学报, 1997, 8(增刊): 535-545.

[6] 谭文安. 企业过程动态优化技术及其支持环境的研究与开发[D]. 北京: 北京航空航天大学研究生院, 2001.


编辑 金胡考