 PL/SQL是基于Oracle的一个主流应用程序编程语言，它的主要特点是将SQL语句与过程化程序开发语言相结合，以实现更为复杂的商业逻辑。本文主要就其中多进程通信进行讨论。 显然，多进程技术是用来提高应用的并发性，进而提高整个系统的执行效率，那么如何在PL/SQL中实现多进程的通信呢？其实，PL/SQL其设计的初衷主要是增强SQL语句的功能，而没有考虑到其他编程语言的高级功能，所以在PL/SQL中实现多进程通信只能借助于Oracle提供的两个开发包：DBMS_PIPE和DBMS_ALERT. 1.DBMS_PIPE 该包提供多进程之间管道通信的方法，比如连接到同一个数据库的两个独立会话之间可以通过管道进行通信，另外也可以在存储过程和Pro*C之间进行通信，这样就大大地增强了PL/SQL的处理能力。该包主要提供两对函数：pack_message（v_msg varchar2）将v_msg信息打包放入到缓冲器中，准备发送；send_message（v_pipename varchar2）发送名为v_pipename的管道的缓冲器；unpack_message（v_msg varchar2）将信息解析到v_msg中；receive_message（v_pipename varchar2）接受名为v_pipename的管道的缓冲器；其执行的原理是：首先建立有名管道（这点熟悉unix很清楚），管道的发送端和接受端都有相应的缓冲器进行接受和发送处理，要注意的是，文本信息必须打包来发送，通过解析来读取信息。 为了理解前面的描述，下面列举一个两个会话之间通信的实例。 发送进程：declarev_pipename

varchar2(30):=pipe1.v_status integer.begindbms_pipe.pack_message(hello,this is sending process!).v_status:=dbms_pipe.send_message(v_pipename).if v_status !=0 thendbms_output.put_line(error!).end if.end./ 接受进程：declarev_pipename varchar2(30):=pipe1.v_status integer.v_msg varchar2(20).beginv_status:=dbms_pipe.receive_message(v_pipename).if v_status !=0 thendbms_output.put_line(error).end if.dbms_pipe.unpack_message(v_msg).dbms_output.put_line(v_msg).end./ 2.DBMS_ALERT 与DBMS_PIPE类似，DBMS_ALERT可以实现多个进程（会话）之间的通信。其基本的执行过程为：首先建立一个报警通道，然后通过报警通道来发送报警信号；在接收端需要先注册该报警通道，对该通道进行监听，然后等待报警信号的到来。下面是该包中的主要函数说明：dbms_alert.signal（报警管道名，待发送消息）发送报警信息；dbms_alert.reGISter（报警管道名）注册报警管道；dbms_alert.waitone（报警管道名，接受消息值，返回状态值）对报警管道进行监听，等待消息的到来；同样，这里也给出使用DBMS_ALERT的一个实例。 发送进程：declarev_alertName varchar2(30):=alert1.begindbms_alert.signal(v_alertName, hello,this is sending process!).commit.end./ 接受进程：declarev_alertName varchar2(30):=alert1.v_status integer.v_msg varchar2(20).begindbms_alert.register(v_alertName).dbms_alert.waitone(v_alertName,v_msg,v_status).if v_status !=0 thendbms_output.put_line(error).end

if.dbms_output.put_line(v_msg).end./ 3.说明 DBMS_PIPE和DBMS_ALERT这两个包都可以进行多进程间的通信，但两者任然是有一些区别：（1） 报警信号是同步的。报警信号直到会话发出Commit时才被发出；如果它所处的事务回滚，则该信号就不发送了。但是管道信号是异步的，而且其通信过程不受事务提交和回滚的影响。（2） 沿着管道进行传递的消息只能被一个进程进行处理，其消息的接收方式是拷贝后删除，但是报警信号可以被多个进程所获得，即消息的接收方式仅仅是拷贝。（3） 管道消息不仅可以传递文本信息，还可以传递其他信息，比如对象等，这一好处得益于其对消息的打包预处理；而报警消息只能是文本，不能是其他类型。 4. 总结 近年来，虽然多进程应用有逐渐被多线程应用取代的趋势，但是多进程仍然有其适用的空间，希望读者通过本文初浅的介绍，对基于PL/SQL的多进程通信这一技术有一个初步的认识，这样为以后深入的开发打下一定的基础。