
第 1 章 Websphere Product Center 监视

可以通过使用 **rootadmin** 和 **rmi_status** 脚本或通过 GUI 来监视 WebSphere Product Center。WebSphere Product Center 未附带提供独立的监视工具。

本文档不阐述监视工具的创建；但是，可能会提到几个简单的想法：

- 在应用程序服务器上（也可能在独立的监视服务器上）创建虚拟主机。由于 **rmi_status** 和 **rootadmin.sh** 会造成系统负载，所以，使用单独的监视服务器对于许多环境来说可能更合适。
- 使用脚本语言（如 Perl）来为 **rootadmin.sh**（它控制和显示服务的状态）创建一个 CGI 包装器。
- 在 Web 服务器中创建一个指向 **\$TOP/logs** 的别名以便您可以更方便地浏览日志文件。
- 创建一个实用程序，它对 **\$TOP/logs** 中地日志进行语法分析以了解所发生的异常和其它错误并且可以选择检查服务状态。在发生错误时，此实用程序可以向管理员发送电子邮件，或者通过其它一些方法通知管理员。不间断地运行此实用程序

获取服务的短状态

要获取服务的短状态，请传递下列参数：

```
-cmd=check -svc=<service name>
```

短状态返回下列其中一种情况：

正在运行	该服务正在运行并且正在响应“脉动信号”功能。
找不到	找不到该服务。该服务可能尚未启动，或者可能已崩溃。
找到但未响应	找到该服务，并发现该服务已对 RMI 注册程序注册，但它不响应“脉动信号”功能。可能必须重新启动该服务。

获取服务的长状态

要获取服务的长状态，请向 **rootadmin.sh** 传递下列参数：

```
-cmd=status -svc=<service name>
```

它将生成一个 HTML 文件，可以使用任何浏览器来查看该文件。在终端上，使用 **Lynx**（或类似工具）来格式化输出。

此状态提供正在该服务中运行的不同线程的概述以及该服务当前所使用的数据库连接的状态。

示例:

要获取调度程序的状态:

```
rootadmin.sh -cmd=status -svc=scheduler > /tmp/sch_status.html;  
lynx /tmp/sch_status.html
```

或者

```
rootadmin.sh -cmd=status -svc=scheduler > /tmp/sch_status.html; lynx -  
dump /tmp/sch_status.html
```

注意: 以上示例中使用的 “>” 将状态详细信息定向至一个文件输出位置。

数据库监视和管理

由于关系数据库是大量的产品信息内容的主要存储场所，所以，提供能够防止性能下降和损失的管理操作十分重要。

通过在 WebSphere Product Center 中设置警报，可以针对可能发生的问题提供通知，以便在这些问题失去控制前解决它们。还应该实现监视系统以不间断地监视 WebSphere Product Center 数据库。

应该定期执行下列任务。

- 在有必要的时候分配更多的空间
- 根据需要应用修订包 / 补丁集
- 在有必要的时候启动和关闭数据库管理器、数据库以及其它服务
- 在有必要的时候分析数据库模式并收集统计信息以提高性能
- 定期重组表和索引，以提高性能
- 检查已调度的备份作业的状态
- 在有必要的时候复原和恢复数据库
- 调整数据库性能

1. 在有必要的时候分配更多的空间

对于大多数用户来说，空间管理是一项持续进行的任务。除非使用的是完全静态的数据库，否则，表和索引的大小将有规律地增大或缩小。您需要确保有足够的可用空间以使这种情况发生，而不会中断正在进行的处理。您还需要帮助确保有效地使用该空间。可以使用“DB2 控制中心”来根据需要分配空间。也可以使用命令行界面来完成这一任务。

2. 应用修订包 / 补丁集

修订包 / 补丁集是数据库系统供应商定期交付已经过全面测试和集成的产品修订的机制。它们仅提供错误修订而不包括新功能，并且不要请求在目标系统上进行认证。为了避免数据库系统的任何已知问题，在修订包 / 补丁集可用时应用它们十分重要。请与数据库系统供应商联

系以了解更多有关修订的信息。

3. 启动和关闭数据库管理器和数据库

在应用修订以及将数据库从一台服务器移至另一台服务器等过程中，需要关闭数据库管理器 / 数据库。在有需要的时候，将必须启动 / 关闭数据库。

4. 分析数据库模式和收集统计信息

分析数据库模式以收集有关数据库中的表和索引的最新统计信息。成本核算优化方法使用统计信息来确定每项执行计划的预计成本。请定期收集统计信息以便向优化器提供有关模式对象的最佳信息。例如，在将相当大量的行装入表中之后，应该收集新的表统计信息。

要分析数据库模式，请运行 `$TOP/src/db/schema/util` 目录中的 shell 脚本 `analyze_schema.sh`。

5. 重组表和索引

建议您定期重组表和索引以提高性能。

当今，数据库的增长速度比以前快得多，典型的 DBA 必须花费相当大量的时间来执行空间管理和重组才能获得最优的性能。

最优的性能意味着最短的响应时间。但性能会由于众多的空间管理问题而下降。在这些问题上，大部分问题落在三个主要领域，即与表相关的问题、索引和 I/O 平衡停滞问题以及数据分区问题。

与表相关的问题为大部分 DBA 所熟悉。它们包括表块中未充分利用的空间、链接的行、不佳的数据近似度以及零乱的（涵盖范围过大）表。

对性能挑战的第二项主要问题是停滞的索引 — 有 3/4 的索引变得很大并且稀疏填充。

这种情况会严重降低索引范围扫描的性能。并且还会浪费相当大量的磁盘空间。

性能挑战方面的第三项主要问题是 I/O 平衡和数据分区。当频繁访问的对象位于同一个数据文件中时，就会产生 I/O 瓶颈。诸如 DB2 中的 `reorgchk` 之类的工具将提供有关需要重组哪些对象的信息。可使用许多方法和工具来重组数据库对象。请阅读特定于数据库系统供应商的有关重组表和索引的文档。

6. 检查已调度的备份作业的状态

备份是复原和恢复过程的主要部分。请验证所有备份作业的状态以确保它们按照时间表运行。

对备份状态执行的检查取决于定义备份过程的方式以及用来执行备份的工具。请阅读特定于数据库系统供应商的有关备份的文档以了解更

多信息。

7. 复原和恢复数据库

发生数据库故障时，您需要确定故障的类型和范围。执行的分析应该指示恢复系统所需完成的步骤。请使用由您的 IT 支持组定义的复原和恢复过程。

复原物理备份是指重构它并使其可供数据库服务器使用。恢复已复原的数据文件是指使用重做记录（即，在执行备份后对数据库所作的更改的记录）来更新该数据文件。

8. 调整数据库性能

DBA 的其中一项最重要职责是确保正确地调整数据库。任何 RDBMS 都是高度可调整的，并允许对数据库进行监视和调整以提高它的性能。

由于下列原因，应该执行性能调整：

- 计算速度可能会浪费人的宝贵时间（用户等待响应）；
- 使系统能够跟上业务的运作速度；并且

优化硬件的使用以节省资金（公司在硬件上花费了数以百万计的资金）。

请参阅 DB2 产品附带提供的产品文档以了解更多有关可用来调整数据库性能的不同方法的信息。