



提高 DBA、开发、IT 专业人士的工作效率

PowerShell V3——SQL Server 2012 数据库自动化运维权威指南

[加拿大] Donabel Santos 著
许昌永 译



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



PowerShell V3——SQL Server 2012 数据库自动化运维权威指南

[加拿大] Donabel Santos 著
许昌永 译

人 民 邮 电 出 版 社
北 京

图书在版编目 (C I P) 数据

PowerShell V3 : SQL Server 2012数据库自动化运维权威指南 / (加) 桑托斯 (Santos, D.) 著 ; 许昌永译
— 北京 : 人民邮电出版社, 2016. 1
ISBN 978-7-115-40877-8

I. ①P… II. ①桑… ②许… III. ①关系数据库系统—指南 IV. ①TP311.138

中国版本图书馆CIP数据核字 (2015) 第260766号

版 权 声 明

Copyright © 2012 Packt Publishing. First published in the English language under the title SQL Server 2012 with PowerShell V3 Cookbook
All rights reserved.

本书由美国 Packt Publishing 公司授权人民邮电出版社出版。未经出版者书面许可, 对本书的任何部分不得以任何方式或任何手段复制和传播。

版权所有, 侵权必究。

-
- ◆ 著 [加] Donabel Santos
 - 译 许昌永
 - 责任编辑 赵 轩
 - 责任印制 张佳莹 焦志炜
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
 - 邮编 100164 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 北京昌平百善印刷厂印刷
 - ◆ 开本: 800×1000 1/16
 - 印张: 30
 - 字数: 610 千字 2016 年 1 月第 1 版
 - 印数: 1—2 000 册 2016 年 1 月北京第 1 次印刷
 - 著作权合同登记号 图字: 01-2015-7471 号
-

定价: 99.00 元

读者服务热线: (010) 81055410 印装质量热线: (010) 81055316
反盗版热线: (010) 81055315

前言

PowerShell 是微软的新命令行 Shell 和脚本语言，可以简化微软不同的应用程序和组件之间的自动化过程和整合过程。数据库专业人士可以利用它内置的大量 cmdlets 或任意可用的 .NET 类来操作 PowerShell，使数据库任务自动化，简化整合过程，或者探索完成手头工作的新方法。

本书向忙碌的数据库专业人士提供了大量简单易学的实战案例。无论你是正在审核服务器，导出数据，还是部署报表，总有一种方案适合你！

本书从基础的主题开始，简单介绍 SQL Server 和 PowerShell 脚本，然后进入高级的主题，帮助你管理、维护 SQL Server 数据库。

前面几章描述了如何配置 SQL Server 和操作对象，包括导出对象、创建数据库、配置实例参数和部署报表。本书随后深入更多管理主题，例如备份和恢复、凭据、策略和作业。

其他开发和 BI 相关主题也有涉及，包括部署和下载程序集、BLOB 数据、SSIS 包和 SSRS 报表。

附录提供了简单的 PowerShell 入门资料，数据库专业人士可用于温习基础知识或作为参考资料。本书内含 100 多个实战性的可用脚本，可以作为自动化运维和管理 SQL Server 的参考。

本书包含的内容

第 1 章，SQL Server 和 PowerShell 入门，解释了 PowerShell 是什么以及你为什么应该学习 PowerShell。介绍了 PowerShell V3 的新属性，解释了在使用 SQL Server 2012 和 PowerShell 工作时需要哪些准备。

第 2 章，SQL Server 和 PowerShell 的基础任务，展示了使用 PowerShell 完成基本的 SQL Server 任务的脚本和代码片段。我们从执行简单的任务开始，例如列出 SQL Server 实例，创建对象，例如表、索引、存储过程和函数，使你通过编程方式更加轻松地使用 SQL Server。

第 3 章，基本管理，着手处理更多使用 PowerShell 能完成的可管理性任务，提供了将

大量重复任务自动化的方案。有的方案处理实例和数据库属性；有的方案提供检查磁盘空间，创建 WMI 警告，配置数据库邮件以及创建和维护 SQL Server 作业的方法。

第 4 章，安全，提供了简化安全监控的片段，包含如何通过解析事件日志检查失败的登录尝试，或者如何管理角色和许可。

第 5 章，高级管理，介绍了 PowerShell 如何帮助你掌握基于策略的管理（PBM）和加密等属性。本章也包含通过编程方式使用 SQL Server Profiler 跟踪文件和事件。

第 6 章，备份和恢复，深入讲解通过使用 Powershell 以编程方式备份和恢复 SQL Server 数据库的不同方法。

第 7 章，SQL Server 开发，提供了代码片段和引导，教你如何在 PowerShell 和 SQL Server 中使用 XML、XSL、二进制数据和 CLR 程序集。

第 8 章，商业智能，包含 PowerShell 如何管理与 BI 相关的任何任务并实现自动化——从渲染 SQL Server 报表服务（SSRS）报表，到部署新的 SQL Server 集成服务（SSIS）2012 ISPAC 文件，再到备份和恢复 SQL Server 分析服务（SSAS）cubes。

第 9 章，有用的 PowerShell 代码，包括大量并非专用于 SQL Server 的方案，但是你会发现这些方案在 PowerShell 中非常有用。方案包括使用时间戳创建文件，针对当前的系统错误分析时间日志以及将进程列表导出到 CSV 或 XML 文件中。

附录 A，SQL Server 和 PowerShell 的快捷键，提供了当使用 PowerShell 操作 SQL Server 时常用的术语和片段的简明快捷键。

附录 B，PowerShell 入门，简单介绍 PowerShell 原理的入门知识。

附录 C，资源，列出了学习 PowerShell 和 SQL Server 的其他书籍、博客和链接。

附录 D，创建 SQL Server VM，提供了关于如何创建和配置虚拟机用于本书的循序渐进的教程。

运行环境

Windows Server 2008 R2
SQL Server 2012 Developer
Visual Studio 2010 Professional
Windows Management Framework 3.0 (包含 PowerShell 3.0、WMI 和 WinRM)

面向的读者对象

本书适用于旨在使用 PowerShell 自动化、集成和简化数据库任务的数据库专业人士（DBA、开发、BI 开发）。同时适用于没有脚本编程背景的读者。

目录

第 1 章 开始使用 SQL Server 和

PowerShell 1

1.1 简介 1

值得注意的 PowerShell V3 属性 ..1

1.2 在你开始之前：使用 SQL Server 和 PowerShell 工作 3

1.3 使用示例代码工作 4

1.4 浏览 SQL Server PowerShell 层次结构 5

1.5 安装 SMO 9

1.6 导入 SMO 程序集 10

1.7 发现 SQL 相关的 cmdlets 和 模块 12

1.8 创建 SQL Server 实例对象 18

1.9 浏览 SMO 服务器对象 20

第 2 章 SQL Server 和 PowerShell

基础任务 22

2.1 介绍 22

2.2 列出 SQL Server 实例 24

2.3 发现 SQL Server 服务 27

2.4 启动/停止 SQL Server 服务 29

2.5 列出 SQL Server 配置设置 33

2.6 修改 SQL Server 实例配置 36

2.7 查找数据库对象 40

2.8 创建数据库 44

2.9 修改数据库属性 45

2.10 删除数据库 48

2.11 修改数据库所有者 49

2.12 创建表 51

2.13 创建视图 55

2.14 创建存储过程 58

2.15 创建触发器 61

2.16 创建索引 65

2.17 执行查询语句/SQL 脚本 68

2.18 使用 Invoke-Sqlcmd 实施 批量导出 69

2.19 使用 bcp 实施批量导出 71

2.20 使用 BULK INSERT 实施 批量导入 73

2.21 使用 bcp 实施批量导入 77

第 3 章 基本管理 80

3.1 介绍 80

3.2 创建 SQL Server 实例清单 80

3.3 创建 SQL Server 数据库清单 84

3.4	列出安装的热补丁和服务包	86	4.8	列出登录、用户和数据库 匹配	160
3.5	列出运行/阻塞的进程	89	4.9	列出登录/用户角色和许可	162
3.6	杀掉阻塞进程	91	4.10	创建登录	164
3.7	检查磁盘空间使用	93	4.11	给登录分配许可和角色	166
3.8	配置 WMI 服务器事件警告	96	4.12	创建数据库用户	168
3.9	分离数据库	101	4.13	给数据库用户分配许可 权限	171
3.10	附加数据库	102	4.14	创建数据库角色	173
3.11	复制数据库	106	4.15	修复孤立用户	176
3.12	在多台服务器上执行查询	108	4.16	创建凭据	179
3.13	创建文件组	109	4.17	创建代理	181
3.14	将辅助数据文件添加到 文件组	111	第 5 章	高级管理	185
3.15	将索引移动到不同的文件组	113	5.1	介绍	185
3.16	检查索引锁片	116	5.2	列出方面和方面属性	185
3.17	重新组织/重建索引	118	5.3	列出策略	187
3.18	运行 DBCC 命令	120	5.4	导出策略	189
3.19	配置数据库邮件	121	5.5	导入策略	192
3.20	列出 SQL Server 作业	128	5.6	创建条件	195
3.21	添加 SQL Server 操作员	130	5.7	创建策略	198
3.22	创建 SQL Server 作业	132	5.8	评估策略	201
3.23	添加 SQL Server 事件告警	135	5.9	启用/禁用更改跟踪	203
3.24	运行 SQL Server 作业	137	5.10	运行并保存 Profiler 跟踪 事件	205
3.25	调度 SQL Server 作业	139	5.11	提取跟踪文件的内容	211
第 4 章	安全	146	5.12	创建数据库主密钥	215
4.1	介绍	146	5.13	创建证书	217
4.2	列出 SQL Server 服务帐号	146	5.14	创建对称和非对称密钥	219
4.3	修改 SQL Server 服务帐号	148	5.15	配置透明数据加密 (TDE)	224
4.4	列出验证模式	151	第 6 章	备份和恢复	229
4.5	修改验证模式	152	6.1	介绍	229
4.6	列出 SQL Server 日志中的 错误信息	155			
4.7	列出登录失败的尝试信息	159			

6.2	修改数据库恢复模式	229	8.3	列出 SSRS 报表属性	300
6.3	列出备份历史	232	8.4	使用 ReportViewer 查看 SSRS 报表	302
6.4	创建备份设备	233	8.5	下载 SSRS 报表到 Excel 和 PDF	306
6.5	列出备份头部和文件列表 信息	235	8.6	SSRS 目录	309
6.6	创建完整备份	238	8.7	创建 SSRS 数据源	313
6.7	创建一个镜像媒体集合的 备份	243	8.8	修改 SSRS 报表的数据源 参照	316
6.8	创建差异备份	246	8.9	将 SSRS 报表上传到报表 管理器	319
6.9	创建事务日志备份	248	8.10	下载所有 SSRS 报表 RDL 文件	322
6.10	创建文件组备份	250	8.11	将带有角色的用户添加到 SSRS 报表	326
6.11	将数据库恢复到一个时间点	252	8.12	在 SSIS 包存储和 MSDB 创建目录	330
6.12	实施在线逐步恢复	261	8.13	将 SSIS 包部署到包存储	332
第 7 章	SQL Server 开发	268	8.14	执行存储在包存储或文件 系统中的 SSIS 包	334
7.1	介绍	268	8.15	将 SSIS 包下载到文件	336
7.2	插入 XML 到 SQL Server	268	8.16	创建 SSISDB 目录	338
7.3	从 SQL Server 抽取 XML	271	8.17	创建 SSISDB 文件夹	341
7.4	从 SQL Server 内容创建 RSS 种子	273	8.18	将 ISPAC 文件部署到 SSISDB	343
7.5	将 XSL 应用到 RSS 种子	278	8.19	执行存储在 SSISDB 中的 SSIS 包	345
7.6	将二进制数据存储到 SQL Server	281	8.20	列出 SSAS cmdlets	348
7.7	从 SQL Server 抽取二进制 数据	284	8.21	列出 SSAS 实例属性	349
7.8	创建新的程序集	288	8.22	备份 SSAS 数据库	351
7.9	列出用户定义的程序集	291	8.23	恢复 SSAS 数据库	352
7.10	抽取用户定义的程序集	292	8.24	执行 SSAS cube	353
第 8 章	商业智能	298			
8.1	介绍	298			
8.2	列出 SSRS 报表服务器上的 条目	298			

第 9 章 有用的 PowerShell 片段..... 355

- 9.1 介绍 355
- 9.2 为 Get-Help 文档化 PowerShell 脚本 355
- 9.3 获取时间戳 358
- 9.4 获取额外错误信息 359
- 9.5 列出进程 361
- 9.6 获取别名 363
- 9.7 导出到 CSV 和 XML 364
- 9.8 使用 Invoke-Expression 365
- 9.9 测试正则表达式 367
- 9.10 管理文件夹 370
- 9.11 管理文件 372
- 9.12 搜索文件 374

- 9.13 读取事件日志 376
- 9.14 发送电子邮件 377
- 9.15 嵌入 C#代码 379
- 9.16 创建 HTML 报表 381
- 9.17 解析 XML 382
- 9.18 从 Web 服务抽取数据 384
- 9.19 使用 PowerShell 远程 386

附录 A SQL Server 和 PowerShell

- 快捷键 390

附录 B PowerShell 入门..... 406**附录 C 资源**..... 430**附录 D 创建 SQL Server VM**..... 435

第 1 章

开始使用 SQL Server 和 PowerShell

1.1 简介

PowerShell 是一款拥有 Shell 和脚本能力的可管理工具，可以用来调用 Windows 管理规范（Windows Management Instrumentation, WMI）、COM 组件和 .NET 库。PowerShell 在一代一代的微软产品中变得更加杰出。在大量的、新的、即将发布的微软产品中，对于它的支持是捆绑式的，并有所加强。Windows Server、Exchange、ActiveDirectory、SharePoint，甚至 SQL Server，所有这些都添加了 PowerShell 支持和 cmdlets。甚至一些供应商如 VMWare、Citrix、Cisco 和 Quest 提供了允许他们的产品被 PowerShell 访问的方法。

是什么使得 PowerShell 发威？每个系统管理员可能都知道通过脚本集成各种系统的痛苦。曾经用到的解决方案是集成各种脚本，包括 VBScript、一些批处理文件，或者 C# 代码、Perl 脚本。系统管理员要么捆绑不同的语言来工作，要么因为复杂的代码而束手无策。

这就是 PowerShell 的来由。PowerShell 最重要的一点是，它简化了自动化和在不同微软子系统之间的集成。因为大多数产品都支持 PowerShell，让一个系统与另一个系统沟通只是一件探索在脚本中添加哪些 cmdlets、功能或模块的事情。即便产品现在还不支持 PowerShell，但它很有可能已经有了 .NET 或 COM 组件的支持，能够轻松使用 PowerShell。

值得注意的 PowerShell V3 属性

在最新版本中，值得注意的属性如下所示。

- **工作流 (Workflows):** PowerShell V3 包括 Windows PowerShell Workflow (PSWF)，在 MSDN 中有描述 (<http://msdn.microsoft.com/en-us/library/jj134242.aspx>)：帮助自动化分发、业务流程、完成多机任务、解放用户和管理员使他们专注于高级别任务。

PSWF 运用 Windows Workflow Foundation 4.0 作为声明框架，但使用类似 PowerShell 语法和结构。

- **稳健的会话 (Robust sessions):** PowerShell V3 支持更加稳健的会话。会话能在网络中断的过程中被保留。这些会话保留打开状态直到超时。
- **调度作业 (Scheduled jobs):** 对于调度作业有一个增强的支持。在 PSScheduledJob 模块中有新的 cmdlets，允许你创建、激活和管理调度作业。
- **模块自动加载 (Module AutoLoading):** 如果你使用的 cmdlet 所属于的模块还没有被加载，将触发 PowerShell 搜索 PSModulePath，加载第一个包含该 cmdlet 的模块。这些我们很容易测试。

```
#check current modules in session
Get-Module
#use cmdlet from CimCmdlets module, which
#is not loaded yet
Get-CimInstance win32_bios
#note new module loaded CimCmdlets
Get-Module
#use cmdlet from SQLPS module, which
#is not loaded yet
Invoke-Sqlcmd -Query "SELECT GETDATE()" -ServerInstance "KERRIGAN"
#note new modules loaded SQLPS and SQLASCmdlets
Get-Module
```

- **Web 服务支持 (Web service support):** PowerShell V3 介绍了 Invoke-WebRequest cmdlet，它发送 HTTP 或者 HTTPS 请求到 Web 服务，返回基于对象的内容，能够在 PowerShell 中被容易的操作。你可以考虑用 PowerShell 下载完整的站点（看看 Lee Holmes 关于此项服务的文章：<http://www.leeholmes.com/blog/2012/03/31/how-to-download-an-entire-wordpress-blog>）。
- **简化语法 (Simplified language syntax):** 书写 Where-Object 和 Foreach-Object 变得更加整洁。提高包括默认参数值和简化语法。

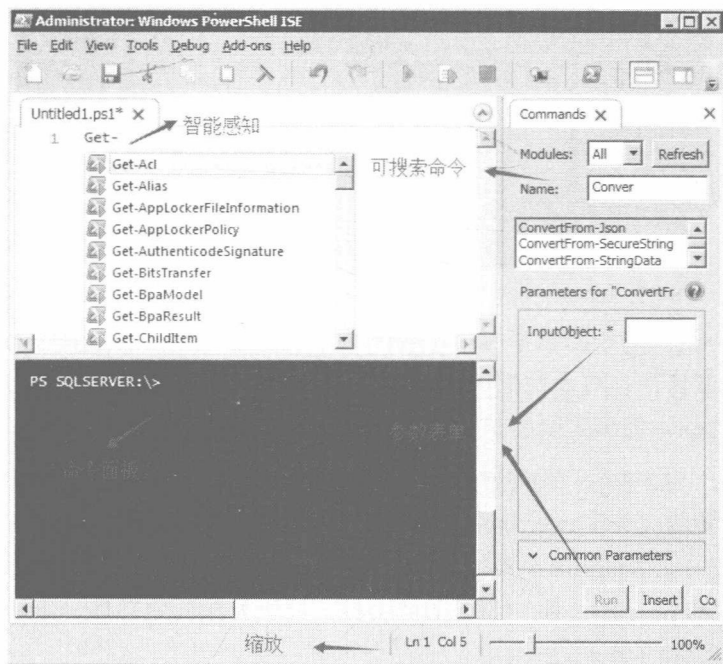
在 V1 和 V2 中必须用大括号和\$_。

```
Get-Service | Where-Object { $_.Status -eq 'Running' }
```

现在能在 V3 中书写为：

```
Get-Service | Where-Object Status -eq 'Running'
```

- **增强集成脚本环境 (Improved Integrated Scripting Enviroment (ISE)):** 新的 ISE 智能感知，侧边栏可搜索命令、参数表单和动态语法检查。



1.2 在你开始之前：使用 SQL Server 和 PowerShell 工作

在我们深入方法之前，重温一些重要的概念和术语，将会帮助我们理解 SQL Server 和 PowerShell 如何一起工作。

- **PSProvider and PSDrive:** PowerShell 允许访问不同的数据存储，无论它们是普通文件还是目录。PSProvider 类似于一个适配器，允许被视为驱动设备，从而被发现。列出支持的 PSProvider 对象，输入：

```
Get-PSProvider
```

你可以看到类似下面的截屏。

Name	Capabilities	Drives
Alias	ShouldProcess	<Alias>
Environment	ShouldProcess	<Env>
FileSystem	Filter, ShouldProcess, Credentials	<C, D, E>
Function	ShouldProcess	<Function>
Registry	ShouldProcess, Transactions	<HKLM, HKCU>
Variable	ShouldProcess	<Variable>

依赖于在你的系统中已经可用的 PSProvider 实例，你的可能稍微有些不同。

- **PSDrive:** 对 C:\ 作为数据存储而不是文件系统。列出在你系统中的 PSDrive 对象，输入：

```
Get-PSDrive
```

你可以看到类似下面的截屏。

Name	Used <GB>	Free <GB>	Provider	Root
Alias			Alias	
C	60.06	19.94	FileSystem	C:\
Cert			Certificate	\
D	74.29	143.79	FileSystem	D:\
E			FileSystem	E:\
Env			Environment	
Function			Function	
HKCU			Registry	HKEY_CURRENT_USER
HKLM			Registry	HKEY_LOCAL_MACHINE
Variable			Variable	
WSMan			WSMan	

注意对于 SQL Server 有一个 PSDrive，可浏览和访问，操作 SQL Server 对象。

- Execution policy: 默认情况下，PowerShell 遵守当前的执行策略，决定什么类型的脚本能被执行。对于我们的方法，假设在你的测试环境中以系统管理员账号运行 PowerShell。你也需要将执行策略设置为 RemoteSigned。

```
Set-ExecutionPolicy RemoteSigned
```

这个设置将允许 PowerShell 运行数字签名的脚本，或者本地未签名的脚本。

- Modules and snap-ins: 模块和插件是扩展 PowerShell 的方法。模块和插件都能将 cmdlets 和 providers 添加到当前的会话。模块能额外将功能、变量、别名和其他工具加载到你的会话。

插件是动态链接库（Dynamically Linked Libraries，DLL），在使用之前需要注册。

插件在 V1、V2 和 V3 版本中可用。例如：

```
Add-PSSnapin SqlServerCmdletSnapin100
```

从另一方面来说，模块更像常规的 PowerShell .ps1 脚本文件。模块在 V2 和 V3 版本中可用。你不必注册一个模块来使用它，只需要导入：

```
Import-Module SQLPS
```

1.3 使用示例代码工作

本书中的示例已经针对在 Windows Server 2008 R2 上的 SQL Server 2012 测试过。

1.3.1 如何做...

如果你想使用当前的机器，而不想创建一个单独的 VM 虚拟机，像附录 D 中描述的那样，创建一个 SQL Server VM 虚拟机，通过以下步骤准备你的机器。

1. 在你当前的操作系统上安装 SQL Server 2012 – Windows 7 或者 Windows Server 2008 R2。查看 SQL Server 2012 支持的操作系统列表。

<http://msdn.microsoft.com/en-us/library/ms143506.aspx>

2. 安装 PowerShell V3。

尽管 PowerShell V3 开始安装在 Windows 8 和 Windows Server 2012 上，在写本书时，

这两个操作系统没有列在 SQL Server 2012 支持的操作系统列表中。

在 Windows 7 SP1、Windows Server 2008 SP2 或 Windows Server 2008 R2 SP1 中安装 PowerShell V3。

安装 Microsoft .NET Framework 4.0，如果还没有安装。

下载并安装 Windows Management Framework 3.0，它包含 PowerShell V3。在写本书时，Release Candidate (RC) 可从这里获得。

<http://www.microsoft.com/en-us/download/details.aspx?id=29939>

3. 开启 PowerShell V3 ISE。在本书中我们将会在很多示例中使用增强集成脚本环境。

- 右键单击任务栏中的“Windows PowerShell”图标，选择以系统管理员身份运行“Run as Administrator”。

- 执行如下语句。

```
PS C:\Users\Administrator>Import-Module ServerManager
PS C:\Users\Administrator>Add-WindowsFeature PowerShell-ISE
```

- 验证是否能加载 ISE。

```
PS C:\Users\Administrator>powershell_ise
```

你也可以通过“Start”、“All Programs”、“Accessories”、“Windows PowerShell”、“Windows PowerShell ISE”来访问。

- 在代码编辑器中通过执行如下语句，设置执行策略。

```
Set-ExecutionPolicy RemoteSigned
```

1.3.2 请参阅

- 看看 PowerShell V3 窥探视频。

<http://technet.microsoft.com/en-us/edge/Video/hh533298>

- 查看 MSDN 上的 SQL Server PowerShell 文档。

[http://msdn.microsoft.com/en-us/library/hh245198\(SQL.110\).aspx](http://msdn.microsoft.com/en-us/library/hh245198(SQL.110).aspx)

1.4 浏览 SQL Server PowerShell 层次结构

在 SQL Server 2012 中，原始的迷你 Shell 已经被废弃，SQLPS 现在作为模块。从 SSMS 进入到 PowerShell，现在是进入了一个 Windows PowerShell 会话，导入 SQLPS 模块，设置当前的安全上下文为 PowerShell 会话进入的源条目。DBA 和开发可以从这里浏览对象架构。

1.4.1 准备

登录 SSMS

1.4.2 如何做...

在这个方案中，我们将从 SSMS 进入 PowerShell 会话来浏览 SQL Server PowerShell 层级结构。

1. 右击实例节点。

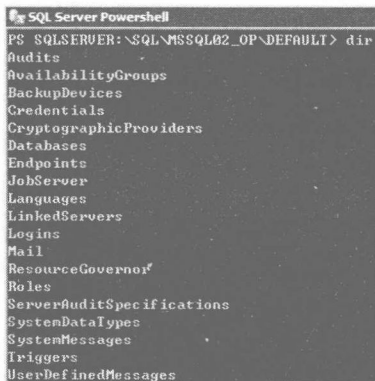
2. 单击“Start PowerShell”。将进入 PowerShell 会话，导入 SQLPS 模块。窗口看起来类似一个命令提示符，有一个当前窗口的 SQL Server 对象源的提示。



```
SQL Server Powershell
PS SQLSERVER:\SQL\MSSQL02_OP\DEFAULT>
```

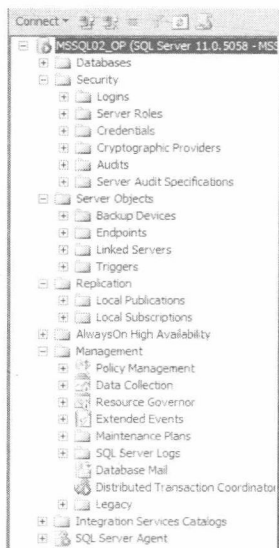
注意窗口的开始路径。

3. 输入 dir。列出当前实例下可访问的所有对象的列表，对于我来说，默认实例是 WIN-LLPKR5BVV6S。注意 dir 是 cmdlet Get-ChildItem 的别名。



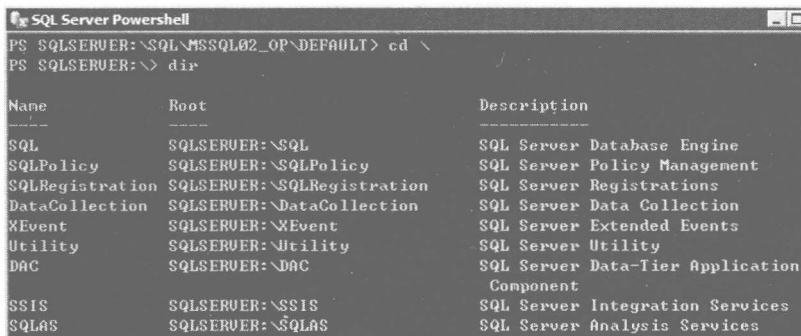
```
SQL Server Powershell
PS SQLSERVER:\SQL\MSSQL02_OP\DEFAULT> dir
Audits
AvailabilityGroups
BackupDevices
Credentials
CryptographicProviders
Databases
Endpoints
JobServer
Languages
LinkedServers
Logins
Mail
ResourceGovernor
Roles
ServerAuditSpecifications
SystemDataTypes
SystemMessages
Triggers
UserDefinedMessages
```

类似于在 SSMS 的“Object Explorer”中，在实例节点下所看到的对象。



4. 当 PowerShell 窗口打开的时候, 浏览 SQL Server PSDrive 或者 SQL Server 数据存储, PowerShell 视为条目序列。输入 `cd \`。到当前驱动器的根目录, 就是 SQL Server PSDrive 的路径将会改变。

5. 输入 `dir`。列出所有 SQL Server PSDrive 根目录下可访问的条目, 将会看到类似以下的截屏。



```

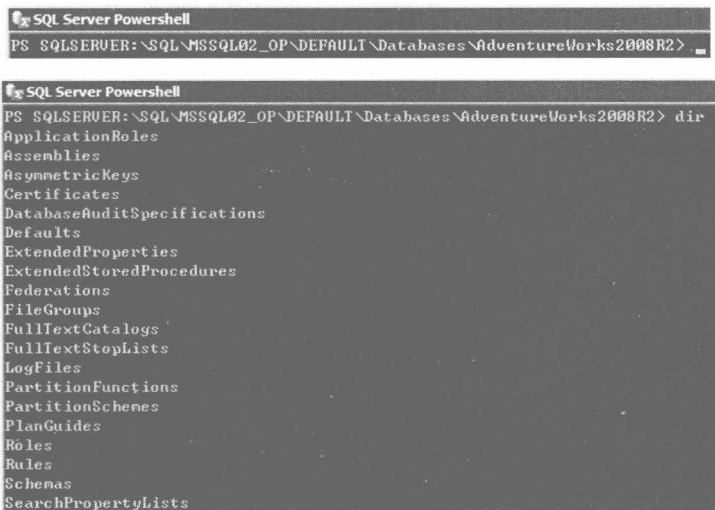
SQL Server PowerShell
PS SQLSERVER:\SQL\MSSQL02_OP\DEFAULT> cd \
PS SQLSERVER:\> dir

Name                Root                Description
-----
SQL                  SQLSERVER:\SQL      SQL Server Database Engine
SQLPolicy            SQLSERVER:\SQLPolicy SQL Server Policy Management
SQLRegistration      SQLSERVER:\SQLRegistration SQL Server Registrations
DataCollection       SQLSERVER:\DataCollection SQL Server Data Collection
XEEvent              SQLSERVER:\XEEvent   SQL Server Extended Events
Utility              SQLSERVER:\Utility   SQL Server Utility
DAC                  SQLSERVER:\DAC       SQL Server Data-Tier Application
                     Component
SSIS                  SQLSERVER:\SSIS      SQL Server Integration Services
SQLAS                SQLSERVER:\SQLAS     SQL Server Analysis Services
  
```

6. 关闭窗口。

7. 返回 SSMS, 右击一个用户数据库。

8. 单击 “Start PowerShell”。注意, 这将会进入另一个 PowerShell 会话, 路径将指向你右击的数据库。



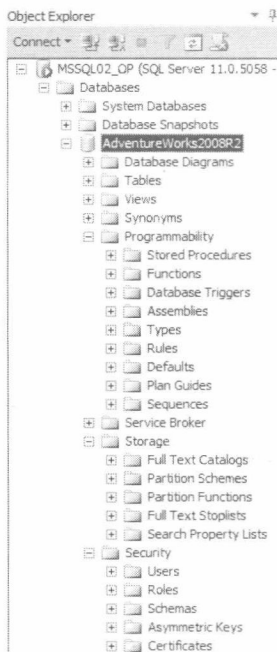
```

SQL Server PowerShell
PS SQLSERVER:\SQL\MSSQL02_OP\DEFAULT\Databases\AdventureWorks2008R2>

SQL Server PowerShell
PS SQLSERVER:\SQL\MSSQL02_OP\DEFAULT\Databases\AdventureWorks2008R2> dir
ApplicationRoles
Assemblies
AsymmetricKeys
Certificates
DatabaseAuditSpecifications
Defaults
ExtendedProperties
ExtendedStoredProcedures
Federations
FileGroups
FullTextCatalogs
FullTextStopLists
LogFiles
PartitionFunctions
PartitionSchemes
PlanGuides
Roles
Rules
Schemas
SearchPropertyLists
  
```

注意, 这个窗口的开始路径不同于第二步中第一次进入 PowerShell 的路径。如果你从当前位置输入 `dir`, 将会看到在 AdventureWorks 2008 R2 数据库下面的所有条目。

如果你展开 AdventureWorks2008R2 数据库节点, 你可以看到在 SSMS 的 “Object Explorer” 中列举的一些条目。



1.4.3 如何实现...

当 PowerShell 从 SSMS 进入时，会创建一个上下文敏感的 PowerShell，它将自动导入 SQLPS 模块。提示符很明显显示了“Start PowerShell”菜单被单击的源对象的路径。

```
PS SQLSERVER:\SQL\MSSQL02_OP\DEFAULT\Databases\AdventureWorks2008R2>
```

SQL Server 2008/2008 R2 作为 SQLPS 迷你 Shell，也作为 SQLPS 工具集。也可以通过从 SSMS 的“Object Explorer”右击一个对象，然后单击“Start PowerShell”进行启动。这个迷你 Shell 被设计为预加载了 SQL Server 扩展的封闭 Shell。这个 Shell 意味着只能用于 SQL Server，被证明是相当有限的，因为 DBA 和开发常常需要通过 PowerShell 导入插件和模块来集成 SQL Server 和其他系统。另一个方法是进入一个成熟的 PowerShell 会话，这依赖于你的 PowerShell 版本要么导入插件，要么导入 SQLPS 模块。

在 SQL Server 2012 中，原始的迷你 Shell 已经被废弃了。当你从 SQL Server 2012 的 SSMS 进入一个 PowerShell 会话时，它进入了一个成熟的 PowerShell 会话，默认导入更新的 SQLPS 模块。

SQL Server 作为一个 PowerShell 驱动设备 (PSDrive)，可以像切换目录和文件一样切换对象。因此，提供者支持熟悉的切换目录命令，如 `dir` 或者 `ls`。注意，这些熟悉的命令常常只是真实的 cmdlet 的别名，这里是 `Get-ChildItem`。

通过 SSMS 进入 PowerShell 时，你能立刻开始浏览 SQL Server PowerShell 的层次结构。