



Solaris 10 10/08 安装指南：自定义 JumpStart 和高级安装



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

文件号码 820-6079-10
2008 年 10 月

版权所有 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 保留所有权利。

对于本文中介绍的产品，Sun Microsystems, Inc. 对其所涉及的技术拥有相关的知识产权。需特别指出的是（但不局限于此），这些知识产权可能包含一项或多项美国专利，或在美国和其他国家/地区申请的待批专利。

美国政府权利—商业软件。政府用户应遵循 Sun Microsystems, Inc. 的标准许可协议，以及 FAR（Federal Acquisition Regulations，即“联邦政府采购法规”）的适用条款及其补充条款。

本发行版可能包含由第三方开发的内容。

本产品的某些部分可能是从 Berkeley BSD 系统衍生出来的，并获得了加利福尼亚大学的许可。UNIX 是 X/Open Company, Ltd. 在美国和其他国家/地区独家许可的注册商标。

Sun、Sun Microsystems、Sun 徽标、Solaris 徽标、Java 咖啡杯徽标、docs.sun.com、Java 和 Solaris 是 Sun Microsystems, Inc. 或其子公司在美国和其他国家/地区的商标或注册商标。所有 SPARC 商标的使用均已获得许可，它们是 SPARC International Inc. 在美国和其他国家/地区的商标或注册商标。标有 SPARC 商标的产品均基于由 Sun Microsystems, Inc. 开发的体系结构。

OPEN LOOK 和 SunTM 图形用户界面是 Sun Microsystems, Inc. 为其用户和许可证持有者开发的。Sun 感谢 Xerox 在研究和开发可视或图形用户界面的概念方面为计算机行业所做的开拓性贡献。Sun 已从 Xerox 获得了对 Xerox 图形用户界面的非独占性许可证，该许可证还适用于实现 OPEN LOOK GUI 和在其他方面遵守 Sun 书面许可协议的 Sun 许可证持有者。

本出版物所介绍的产品以及所包含的信息受美国出口控制法制约，并应遵守其他国家/地区的进出口法律。严禁将本产品直接或间接地用于核设施、导弹、生化武器或海上核设施，也不能直接或间接地出口给核设施、导弹、生化武器或海上核设施的最终用户。严禁出口或转口到美国禁运的国家/地区以及美国禁止出口清单中所包含的实体，包括但不限于被禁止的个人以及特别指定的国家/地区的公民。

本文档按“原样”提供，对于所有明示或默示的条件、陈述和担保，包括对适销性、适用性或非侵权性的默示保证，均不承担任何责任，除非此免责声明的适用范围在法律上无效。

目录

前言	9
第 1 部分 使用自定义 JumpStart	13
1 在何处查找 Solaris 安装规划信息	15
在何处查找规划和系统要求信息	15
2 自定义 JumpStart (概述)	17
自定义 JumpStart 介绍	17
自定义 JumpStart 示例方案	17
JumpStart 程序如何安装 Solaris 软件	18
3 准备自定义 JumpStart 安装 (任务)	23
任务图: 准备自定义 JumpStart 安装	23
为联网系统创建配置文件服务器	24
▼ 在服务器上创建 JumpStart 目录	25
允许所有系统访问配置文件服务器	26
创建独立系统的配置文件软盘	29
▼ SPARC: 创建配置文件软盘	29
▼ x86: 使用 GRUB 创建配置文件软盘	31
创建 rules 文件	32
rules 文件的语法	33
▼ 创建 rules 文件	33
rules 文件示例	34
创建配置文件	35
配置文件语法	36
▼ 创建配置文件	36

配置文件示例	37
测试配置文件	47
▼ 创建用于测试配置文件的临时 Solaris 环境	48
▼ 测试配置文件	48
配置文件测试示例	50
验证 rules 文件	51
▼ 验证 rules 文件	51
4 使用可选的自定义 JumpStart 功能（任务）	53
创建开始脚本	53
关于开始脚本的重要信息	54
用开始脚本创建派生配置文件	54
创建结束脚本	55
关于结束脚本的重要信息	55
▼ 使用结束脚本添加文件	56
使用结束脚本添加软件包或修补程序	56
使用结束脚本自定义根环境	59
使用结束脚本设置系统的超级用户口令	59
使用结束脚本的非交互安装	61
创建压缩的配置文件	61
▼ 创建压缩的配置文件	62
压缩的配置文件示例	62
创建磁盘配置文件	63
▼ SPARC: 创建磁盘配置文件	63
SPARC: 磁盘配置文件示例	64
▼ x86: 创建磁盘配置文件	65
x86: 磁盘配置文件示例	66
使用站点特定的安装程序	68
5 创建自定义规则和探测关键字（任务）	69
探测关键字	69
创建 custom_probes 文件	70
custom_probes 文件的语法	70
custom_probes 中的函数名称的语法	70
▼ 创建 custom_probes 文件	71

custom_probes 文件和关键字的示例	71
验证 custom_probes 文件	72
▼ 验证 custom_probes 文件	73
6 执行自定义 JumpStart 安装 (任务)	75
JumpStart 安装的限制	75
SPARC: 任务图: 针对自定义 JumpStart 安装设置系统	76
SPARC: 执行自定义 JumpStart 安装	77
▼ 准备使用自定义 JumpStart 安装来安装 Solaris Flash 归档文件	77
▼ SPARC: 使用自定义 JumpStart 程序执行安装或升级	80
SPARC: boot 命令的命令参考	81
x86: 任务图: 针对自定义 JumpStart 安装设置系统	82
x86: 执行自定义 JumpStart 安装	83
▼ x86: 使用自定义 JumpStart 程序和 GRUB 执行安装或升级	83
x86: 通过编辑 GRUB boot 命令执行自定义 JumpStart 安装	85
x86: 用于引导系统的命令参考	87
7 使用自定义 JumpStart 进行安装 (示例)	89
样例站点安装	89
创建安装服务器	90
x86: 创建市场营销系统的引导服务器	92
创建 JumpStart 目录	93
共享 JumpStart 目录	93
SPARC: 创建工程设计组的配置文件	93
x86: 创建市场营销组的配置文件	94
更新 rules 文件	94
验证 rules 文件	95
SPARC: 将工程设计系统设置为从网络安装	95
x86: 将市场营销系统设置为从网络安装	96
SPARC: 引导工程设计系统并安装 Solaris 软件	97
x86: 引导市场营销系统并安装 Solaris 软件	97
8 自定义 JumpStart (参考)	99
规则关键字和值	99

配置文件关键字和值	103
配置文件关键字快速参考	103
配置文件关键字描述和示例	105
自定义 JumpStart 环境变量	142
探测关键字和值	144
9 使用 JumpStart 安装 ZFS 根池	147
ZFS 根 (/) 文件系统的 JumpStart 安装 (概述和规划)	147
ZFS 根池的 JumpStart 安装限制	147
ZFS 根池的 JumpStart 配置文件示例	148
用于 ZFS 根 (/) 文件系统的 JumpStart 关键字 (参考)	151
bootenv 配置文件关键字 (ZFS 和 UFS)	152
install_type 关键字 (ZFS 和 UFS)	153
pool 配置文件关键字 (仅 ZFS)	153
root_device 配置文件关键字 (ZFS 和 UFS)	154
其他资源	155
第 2 部分 附录	157
A 疑难解答 (任务)	159
设置网络安装时的问题	159
引导系统时的问题	160
从介质引导: 错误消息	160
从介质引导: 一般问题	161
从网络引导: 错误消息	162
从网络引导: 一般问题	164
Solaris OS 的初始安装	165
▼ x86: 检查 IDE 磁盘上的坏块	166
升级 Solaris 操作系统	167
升级: 错误消息	167
升级: 一般问题	169
▼ 在升级失败后继续进行升级	170
x86: 使用 GRUB 时, Solaris Live Upgrade 出现问题	170
▼ 在运行 Veritas VxVm 时使用 Solaris Live Upgrade 进行升级, 系统将发出警告音 ..	172

x86: 缺省情况下不会在不包含现有服务分区的系统上创建服务分区	174
▼ 从网络安装映像或 Solaris Operating System DVD 安装软件	175
▼ 从 Solaris Software - 1 CD 或从网络安装映像安装	175
B 附加 SVR4 打包要求 (参考)	177
禁止修改当前的 OS	177
使用绝对路径	177
使用 pkgadd -R 命令	178
\$PKG_INSTALL_ROOT 和 \$BASEDIR 之间的差别概述	178
编写脚本的原则	179
维护无盘客户机兼容性	179
验证软件包	179
安装或升级时禁止用户交互	180
为区域设置软件包参数	182
有关背景信息	184
词汇表	185
索引	197

前言

本书介绍如何在联网和非联网的 SPARC® 系统和基于 x86 体系结构的系统中安装和升级 Solaris™ 操作系统 (OS)。本书包括在安装期间如何使用自定义 JumpStart 安装方法以及如何创建 RAID-1 卷。

本书不包括关于如何设置系统硬件或其他外围设备的说明。

注 - 此 Solaris 发行版支持使用以下 SPARC 和 x86 系列处理器体系结构的系统：UltraSPARC®、SPARC64、AMD64、Pentium 和 Xeon EM64T。支持的系统可以在 <http://www.sun.com/bigadmin/hcl> 上的《Solaris OS: Hardware Compatibility Lists》(Solaris OS：硬件兼容性列表) 中找到。本文档列举了在不同类型的平台上进行实现时的所有差别。

在本文档中，这些与 x86 相关的术语表示以下含义：

- "x86" 泛指 64 位和 32 位的 x86 兼容产品系列。
- "x64" 指出了有关 AMD64 或 EM64T 系统的特定 64 位信息。
- "32 位 x86" 指出了有关基于 x86 的系统的特定 32 位信息。

若想了解本发行版支持哪些系统，请参见《Solaris OS: Hardware Compatibility Lists》。

目标读者

本书适用于负责安装 Solaris OS 的系统管理员。本书同时提供了以下两种类型的信息。

- 高级 Solaris 安装信息，适用于在联网环境中管理多台 Solaris 计算机的企业系统管理员
- 基本 Solaris 安装信息，适用于不经常执行 Solaris 安装或升级的系统管理员

相关书籍

表 P-1 列出了适用于系统管理员的文档。

表 P-1 您是否为安装 Solaris 的系统管理员？

说明	信息
是否需要系统要求或较高层面的规划信息？或者，是否需要有关 Solaris ZFS™ 安装、引导、Solaris Zones 分区技术或创建 RAID-1 卷的较高层面的概述？	《Solaris 10 10/08 安装指南：规划安装和升级》
是否需要从 DVD 或 CD 介质安装单系统？Solaris 安装程序将指导您逐步完成安装过程。	《Solaris 10 10/08 安装指南：基本安装》
是否需要几乎不停机的情况下升级或修补系统？如果使用 Solaris Live Upgrade 进行升级，可减少系统停机时间。	《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》
是否需要通过网络或 Internet 执行安全的安装？请使用 WAN Boot 来安装远程客户机。或者，是否需要通过网络从网络安装映像进行安装？Solaris 安装程序将指导您逐步完成安装过程。	《Solaris 10 10/08 安装指南：基于网络的安装》
是否需要快速安装或修补多个系统？请使用 Solaris Flash™ 软件创建 Solaris Flash 归档文件，然后在克隆系统上安装 OS 副本。	《Solaris 10 10/08 安装指南：Solaris Flash 归档文件（创建和安装）》
是否需要备份系统？	《系统管理指南：设备和文件系统》中的第 24 章“备份和恢复文件系统（概述）”
是否需要适用于此发行版的疑难解答信息、已知问题列表或修补程序列表？	Solaris 发行说明
是否需要检验您的系统能否在 Solaris 上正常工作？	SPARC：Solaris Sun 硬件平台指南
是否需要检查在此发行版中添加、删除或更改了哪些软件包？	Solaris Package List
是否需要检验您的系统和设备能否与基于 SPARC 和 x86 的 Solaris 系统及其他第三方供应商产品一起工作？	适用于 x86 平台的 Solaris 硬件兼容性列表

文档、支持和培训

Sun Web 站点提供有关以下附加资源的信息：

- 文档 (<http://www.sun.com/documentation/>)
- 支持 (<http://www.sun.com/support/>)
- 培训 (<http://www.sun.com/training/>)

印刷约定

下表介绍了本书中的印刷约定。

表 P-2 印刷约定

字体或符号	含义	示例
AaBbCc123	命令、文件和目录的名称；计算机屏幕输出	编辑 <code>.login</code> 文件。 使用 <code>ls -a</code> 列出所有文件。 <code>machine_name% you have mail.</code>
AaBbCc123	用户键入的内容，与计算机屏幕输出的显示不同	<code>machine_name% su</code> <code>Password:</code>
<i>aabbcc123</i>	要使用实名或值替换的命令行占位符	删除文件的命令为 <code>rm filename</code> 。
<i>AaBbCc123</i>	保留未译的新词或术语以及要强调的词	这些称为 <i>Class</i> 选项。 注意： 有些强调的项目在联机时以粗体显示。
新词术语强调	新词或术语以及要强调的词	高速缓存 是存储在本地的副本。 请勿保存文件。
《书名》	书名	阅读《用户指南》的第 6 章。

命令中的 shell 提示符示例

下表列出了 C shell、Bourne shell 和 Korn shell 的缺省 UNIX® 系统提示符和超级用户提示符。

表 P-3 shell 提示符

shell	提示符
C shell 提示符	<code>machine_name%</code>
C shell 超级用户提示符	<code>machine_name#</code>
Bourne shell 和 Korn shell 提示符	<code>\$</code>
Bourne shell 和 Korn shell 超级用户提示符	<code>#</code>

第 1 部分

使用自定义 JumpStart

本部分提供有关创建、准备和执行自定义 JumpStart 安装的说明。

在何处查找 Solaris 安装规划信息

本书提供有关如何使用自动 JumpStart 安装程序安装 Solaris 操作系统的信息。本书提供使用 JumpStart 程序进行安装时需要了解的所有信息，但在开始准备 JumpStart 安装之前，阅读安装文档集中的规划书籍可能会很有用。在安装系统之前，以下参考资料可提供有用的信息。

在何处查找规划和系统要求信息

《Solaris 10 10/08 安装指南：规划安装和升级》提供了系统要求和较高层面的规划信息，如文件系统规划原则和升级规划以及其他更多信息。本节对这本书的章节进行了概述。

规划指南中的章节说明	参考
本章介绍 Solaris 安装程序的新增功能。	《Solaris 10 10/08 安装指南：规划安装和升级》中的第 2 章“Solaris 安装的新增功能”
本章提供有关在安装或升级 Solaris OS 之前需要做出哪些决定的信息。例如，决定何时使用网络安装映像或 DVD 介质以及所有 Solaris 安装程序的说明。	《Solaris 10 10/08 安装指南：规划安装和升级》中的第 3 章“Solaris 安装和升级（汇总信息）”
本章介绍安装或升级到 Solaris OS 的系统要求，还提供了关于规划磁盘空间和缺省的交换空间分配的一般原则。此外，还介绍了一些升级限制。	《Solaris 10 10/08 安装指南：规划安装和升级》中的第 4 章“系统要求、原则和升级（规划）”
本章包含一些核对表，可帮助您收集安装或升级系统所需的全部信息。例如，如果您要执行交互式安装，此信息会很有用。您可以在核对表中找到执行交互式安装所需的全部信息。	《Solaris 10 10/08 安装指南：规划安装和升级》中的第 5 章“在安装或升级前收集信息（规划）”
这些章节概述了与 Solaris OS 安装或升级有关的多种技术。同时，还提供了与这些技术相关的原则和要求。这些章节包含有关 ZFS 安装、引导、Solaris Zones 分区技术以及可在安装时创建的 RAID-1 卷的信息。	《Solaris 10 10/08 安装指南：规划安装和升级》中的第 II 部分，“了解与 ZFS、引导、Solaris Zones 和 RAID-1 卷相关的安装”

自定义 JumpStart (概述)

本章对自定义 JumpStart 安装过程进行了简要介绍。

注 - 如果您要安装 Solaris ZFS™ 根池，请参见第 9 章，使用 JumpStart 安装 ZFS 根池以了解限制和配置文件示例。

- 第 17 页中的“自定义 JumpStart 介绍”
- 第 18 页中的“JumpStart 程序如何安装 Solaris 软件”

自定义 JumpStart 介绍

自定义 JumpStart 安装方法是一个命令行界面，它使您可以基于所创建的配置文件来自动安装或升级多个系统。配置文件定义了特定软件安装要求。您还可以结合使用 shell 脚本以包含安装前和安装后任务。选择安装或升级所使用的配置文件和脚本。自定义 JumpStart 安装方法基于所选择的配置文件和脚本来安装或升级系统。此外，还可以使用 sysidcfg 文件指定配置信息，以使自定义 JumpStart 安装完全脱离手动干预。

自定义 JumpStart 示例方案

可以使用示例方案来描述自定义 JumpStart 过程。在此示例方案中，需要使用以下参数设置系统：

- 在 100 个新系统中安装 Solaris。
- 其中，有七十个系统是属于工程设计组的基于 SPARC 的系统，需要作为独立的系统与针对开发者的 Solaris OS 软件组一起安装。
- 其余 30 个系统是属于市场营销组的基于 x86 的系统，需要作为独立的系统与针对最终用户的 Solaris OS 软件组一起安装。

首先，系统管理员必须为每组系统创建一个 `rules` 文件和一个配置文件。`rules` 文件是一个文本文件，包含针对您要安装 Solaris 软件的每组系统或单个系统的规则。每个规则区分一组基于一个或多个系统属性的系统。每个规则还可以将相应的系统组链接到一个配置文件。

配置文件是一个文本文件，它定义了如何在组中的各个系统上安装 Solaris 软件。`rules` 文件和配置文件都必须位于 JumpStart 目录中。

对于该示例方案，系统管理员创建一个包含两个不同规则的 `rules` 文件，其中一个规则用于工程设计组，另一个用于市场营销组。对于每个规则，系统的网络号用于区分工程设计组和市场营销组。

各个规则还包含指向适当配置文件的链接。例如，在工程设计组的规则中，添加一个指向为工程设计组创建的 `eng_profile` 配置文件的链接。在市场营销组的规则中，添加一个指向为市场营销组创建的 `market_profile` 配置文件的链接。

您可以将 `rules` 文件和配置文件保存在软盘上或服务器上。

- 希望在非联网的独立系统上执行自定义 JumpStart 安装时，需要一个配置文件软盘。
- 希望在可访问服务器的联网系统上执行自定义 JumpStart 安装时，需要使用配置文件服务器。

创建 `rules` 文件和配置文件后，使用 `check` 脚本验证这些文件。如果 `check` 脚本运行成功，则创建 `rules.ok` 文件。`rules.ok` 是 JumpStart 程序用来安装 Solaris 软件的 `rules` 文件的生成版本。

JumpStart 程序如何安装 Solaris 软件

在验证 `rules` 文件和配置文件之后，即可开始自定义 JumpStart 安装。JumpStart 程序读取 `rules.ok` 文件。然后，JumpStart 程序搜索第一个具有所定义的系统属性（该系统属性与 JumpStart 程序正尝试在其上安装 Solaris 软件的系统相匹配）的规则。如果存在匹配项，JumpStart 程序将使用在该规则中指定的配置文件在系统上安装 Solaris 软件。

图 2-1 显示了自定义 JumpStart 安装如何在非联网的独立系统上工作。系统管理员在 Pete 的系统上启动自定义 JumpStart 安装。JumpStart 程序访问系统的软盘驱动器中的软盘上的规则文件。JumpStart 程序将 `rule 2` 与系统匹配。`rule 2` 指定 JumpStart 程序使用 Pete's profile 来安装 Solaris 软件。JumpStart 程序读取 Pete's profile，然后根据系统管理员在 Pete's profile 中指定的指令来安装 Solaris 软件。

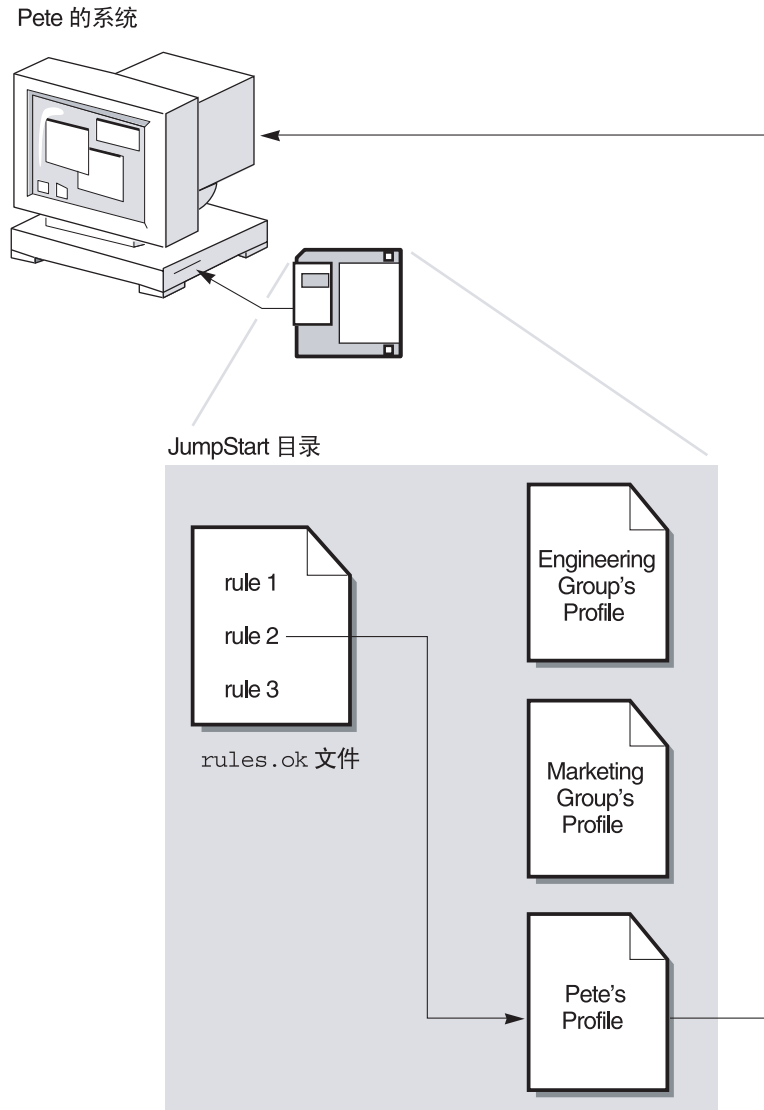


图 2-1 自定义 JumpStart 安装的工作原理：非联网示例

图 2-2 显示了自定义 JumpStart 安装如何在联网的多个系统上工作。之前，系统管理员设置了不同的配置文件，并将这些配置文件保存在单个服务器上。系统管理员在一个工程设计系统上启动自定义 JumpStart 安装。JumpStart 程序访问服务器上 JumpStart/目录中的 rules 文件。JumpStart 程序将工程设计系统与 rule 1 匹配。rule 1 指定

JumpStart 程序使用 Engineering Group's Profile 来安装 Solaris 软件。JumpStart 程序读取 Engineering Group's Profile，然后根据系统管理员在 Engineering Group's Profile 中指定的指令来安装 Solaris 软件。

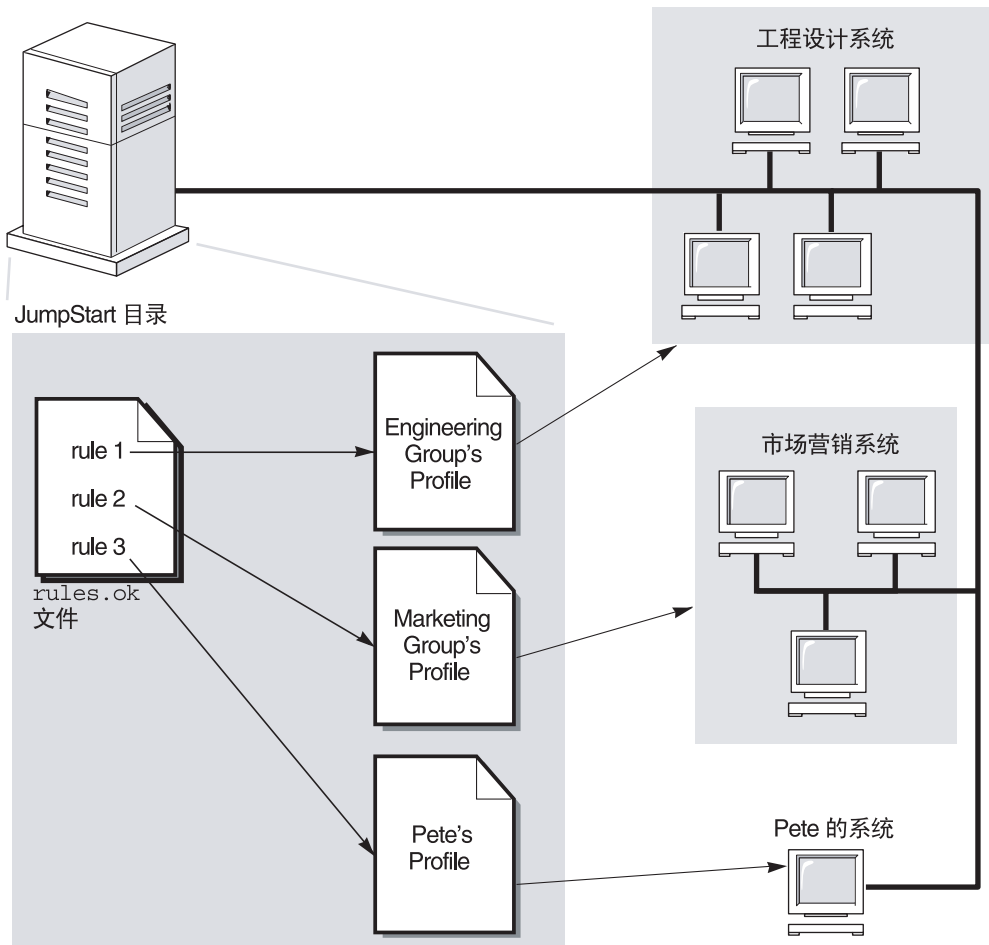


图 2-2 自定义 JumpStart 安装的工作原理：联网示例

图 2-3 说明了 JumpStart 程序搜索自定义 JumpStart 文件的顺序。

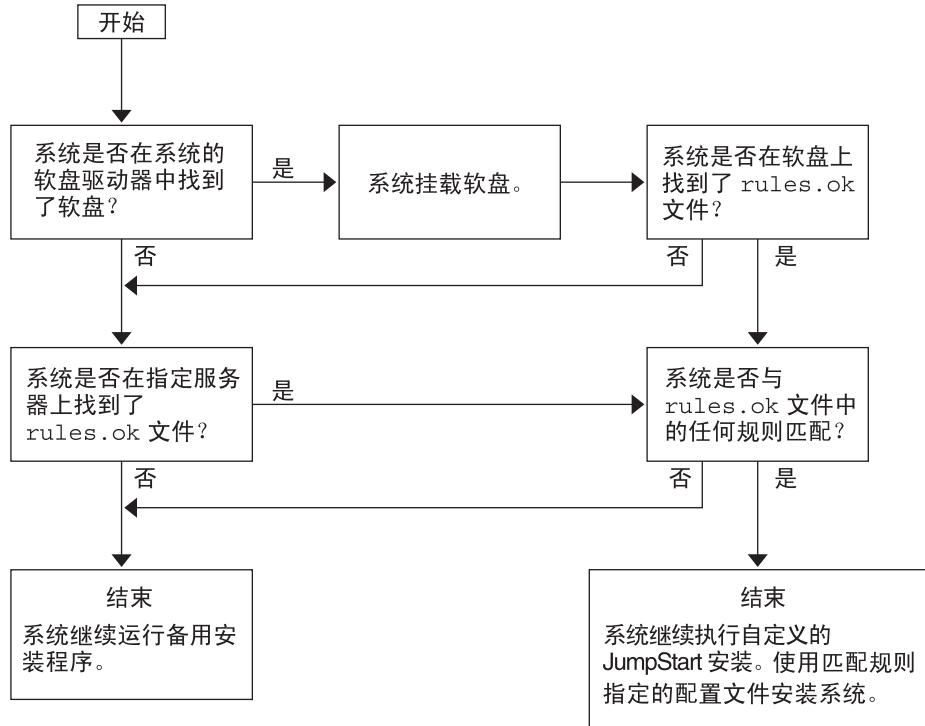


图 2-3 自定义 JumpStart 安装过程中执行的操作

准备自定义 JumpStart 安装（任务）

本章提供逐步说明，介绍使用自定义 JumpStart 安装方法将 Solaris 软件安装到站点上/从站点上安装时，如何准备系统。

注 - 如果您要安装 Solaris ZFS 根池，请参见第 9 章，使用 JumpStart 安装 ZFS 根池以了解限制和配置文件示例。

- 第 23 页中的“任务图：准备自定义 JumpStart 安装”
- 第 24 页中的“为联网系统创建配置文件服务器”
- 第 29 页中的“创建独立系统的配置文件软盘”
- 第 32 页中的“创建 rules 文件”
- 第 35 页中的“创建配置文件”
- 第 47 页中的“测试配置文件”
- 第 51 页中的“验证 rules 文件”

任务图：准备自定义 JumpStart 安装

表 3-1 任务图：准备自定义 JumpStart 安装

任务	说明	参考
在系统上已经安装有 Solaris 软件早期版本的情况下，确定升级系统的方式	如果系统上已经安装了 Solaris 的早期发行版，您需要确定如何升级系统。请务必了解清楚升级系统之前和之后应执行的操作。进行规划有助于您创建配置文件、开始脚本和结束脚本。	《Solaris 10 10/08 安装指南：规划安装和升级》中的“升级规划”

表 3-1 任务图：准备自定义 JumpStart 安装 (续)

任务	说明	参考
创建 JumpStart 目录。	<p>在服务器上</p> <p>如果要在已联网的系统上执行自定义 JumpStart 安装，则必须创建一个配置文件服务器。配置文件服务器包含用于自定义 JumpStart 文件的 JumpStart 目录。</p> <p>在软盘上</p> <p>如果要在未联网的系统上执行自定义 JumpStart 安装，则必须创建一张配置文件软盘。配置文件软盘包含自定义 JumpStart 文件。</p>	<p>第 24 页中的“为联网系统创建配置文件服务器”</p> <p>第 29 页中的“创建独立系统的配置文件软盘”</p>
将规则添加到 rules 文件。	确定各组系统或单组系统的安装方式之后，请为要安装的每个组创建一条规则。每条规则区分一组系统，这些系统基于一个或多个系统属性。规则将对应的组链接到配置文件。	第 32 页中的“创建 rules 文件”
为每条规则创建一个配置文件。	配置文件是一个文本文件，它定义了 Solaris 软件的安装方式。例如，在某个系统上安装哪个软件组。每条规则指定一个配置文件，以定义系统与规则相匹配时，在该系统上安装 Solaris 软件的方式。通常为每个规则创建一个不同的配置文件。但是，同一配置文件可以用于多个规则。	第 35 页中的“创建配置文件”
(可选) 测试配置文件。	创建配置文件之后，请在使用该配置文件来安装或升级系统之前，使用 <code>pfinstall(1M)</code> 命令对其进行测试。	第 47 页中的“测试配置文件”
验证 rules 文件。	rules.ok 文件是 rules 文件的衍生版本。JumpStart 程序使用 rules 文件将要安装的系统与配置文件进行匹配。必须使用 check 脚本验证 rules 文件。	第 51 页中的“验证 rules 文件”

为联网系统创建配置文件服务器

为网络上的系统设置自定义 JumpStart 安装时，需要在服务器上创建一个 JumpStart 目录，该目录包含所有必需的自定义 JumpStart 文件，例如 rules 文件、rules.ok 文件和配置文件。JumpStart 目录必须保存在配置文件服务器的根 (/) 目录中。

包含 JumpStart 目录的服务器称为配置文件服务器。配置文件服务器系统可以与安装服务器或引导服务器相同，也可以与它们完全不同。配置文件服务器可以为不同的平台提供自定义 JumpStart 文件。例如，x86 服务器可以为基于 SPARC 的系统和基于 x86 的系统提供自定义 JumpStart 文件。

注 - 创建配置文件服务器之后，您必须允许系统访问该服务器。有关详细说明，请参见第 27 页中的“允许所有系统访问配置文件服务器”。

▼ 在服务器上创建 JumpStart 目录

注 - 此过程假设系统正在运行卷管理器。如果您未使用卷管理器来管理磁盘，请参阅《系统管理指南：设备和文件系统》，以了解有关在不使用卷管理器时管理可移除介质的详细信息。

- 1 找到要在其上创建 JumpStart 目录的服务器。
- 2 成为超级用户或承担等效角色。
角色包含授权和具有一定权限的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 3 在服务器上的任意位置创建 JumpStart 目录。

```
# mkdir -m 755 jumpstart_dir_path
```

在该命令中，*jumpstart_dir_path* 是 JumpStart 目录的绝对路径。

例如，下面的命令在根 (/) 目录中创建一个名为 *jumpstart* 的目录并将权限设置为 755：

```
# mkdir -m 755 /jumpstart
```
- 4 通过添加以下项，编辑 */etc/dfs/dfstab* 文件。

```
share -F nfs -o ro,anon=0 jumpstart_dir_path
```

例如，以下项共享了 */jumpstart* 目录：

```
share -F nfs -o ro,anon=0 /jumpstart
```
- 5 键入 `shareall`，然后按 **Enter** 键。
- 6 确定是否要将自定义 JumpStart 文件示例复制到 JumpStart 目录中。
 - 如果不复制，请转至步骤 9。
 - 如果复制，请使用以下判定表来确定下一步的操作。

示例位置	参考
适用于您的平台的 Solaris Operating System DVD 或 Solaris Software - 1 CD	将 Solaris Operating System DVD 或 Solaris Software - 1 CD 插入服务器的 CD-ROM 驱动器。 卷管理器会自动挂载 CD 或 DCD。
本地磁盘上适用于您的平台的 Solaris Operating System DVD 映像或 Solaris Software - 1 CD 映像	转到 Solaris Operating System DVD 或 Solaris Software - 1 映像所在的目录。例如，键入以下命令： <code>cd /export/install</code>

7 将自定义 JumpStart 文件示例复制到配置文件服务器上的 JumpStart 目录中。

```
# cp -r media_path/Solaris_10/Misc/jumpstart_sample/* jumpstart_dir_path
```

media_path 指向 CD、DVD 或本地磁盘上的映像的路径

jumpstart_dir_path 配置文件服务器上放置自定义 JumpStart 文件示例的路径

例如，以下命令将 `jumpstart_sample` 目录复制到配置文件服务器上的 `/jumpstart` 目录中：

```
cp -r /cdrom/cdrom0/Solaris_10/Misc/jumpstart_sample/* /jumpstart
```

8 更新 JumpStart 文件示例，以便这些文件可以在您的环境中运行。

9 确保 JumpStart 目录位于 root 中，且权限被设置为 755。

10 允许网络上的系统访问配置文件服务器。

有关详细说明，请参见第 27 页中的“允许所有系统访问配置文件服务器”。

允许所有系统访问配置文件服务器

创建配置文件服务器时，必须确保系统在自定义 JumpStart 安装期间可以访问配置文件服务器上的 JumpStart 目录。请使用以下方法之一来确保该访问权限。

命令或文件	提供访问权限	参考
add_install_client 命令	<p>每次为网络安装添加系统时，请使用 -c 选项和 add_install_client 命令指定配置文件服务器。</p> <p>注 - 如果没有使用 NFS，则必须使用其他方法来提供访问权限。</p> <ul style="list-style-type: none"> ■ 对基于 SPARC 的系统，请使用 boot 命令 ■ 对基于 x86 的系统，请编辑 GRUB 菜单 	<ul style="list-style-type: none"> ■ 对于 DVD 介质，请参见《Solaris 10 10/08 安装指南：基于网络的安装》中的“使用 DVD 映像添加要从网络安装的系统” ■ 对于 CD 介质，请参见《Solaris 10 10/08 安装指南：基于网络的安装》中的“使用 CD 映像添加要从网络安装的系统”
引导系统时指定 JumpStart 目录的位置	<ul style="list-style-type: none"> ■ 对基于 SPARC 的系统，请使用 boot 命令引导系统。引导系统时指定配置文件服务器上 JumpStart 目录的位置。必须将自定义 JumpStart 配置文件压缩为一个文件。然后，将压缩后的配置文件保存在 HTTP 或 HTTPS 服务器上。 ■ 对基于 x86 的系统，当通过编辑 GRUB 菜单上的引导项来引导系统时，请在配置文件服务器上指定 JumpStart 目录的位置。必须将自定义 JumpStart 配置文件压缩为一个文件。然后，将压缩后的配置文件保存在 HTTP 或 HTTPS 服务器上。 编辑 GRUB 菜单项时，请指定压缩文件的位置。 	<ul style="list-style-type: none"> ■ 第 61 页中的“创建压缩的配置文件” ■ 第 80 页中的“SPARC: 使用自定义 JumpStart 程序执行安装或升级”中的步骤 5 ■ 第 61 页中的“创建压缩的配置文件” ■ 第 85 页中的“x86: 通过编辑 GRUB boot 命令执行自定义 JumpStart 安装”
/etc/bootparams 文件	在 /etc/bootparams 文件中添加一个通配符。	第 27 页中的“允许所有系统访问配置文件服务器”

▼ 允许所有系统访问配置文件服务器

只有将网络安装信息存储在下列位置时，才使用以下过程：

- 在 /etc/bootparams 文件中。
- 在命名服务 bootparams 数据库中。要更新 bootparams 数据库，请添加步骤 3 中显示的项。

如果使用以下过程，系统必须是同一类型，例如都是 SPARC 系统。

不要在以下情况中使用该过程：

- 如果将 JumpStart 目录保存在软盘上。
- 如果在引导系统时指定了配置文件服务器的位置。如果有不同体系结构的系统，则在引导系统时必须指定配置文件服务器的位置

如果存在以上情况，请使用 SPARC boot 命令或使用 x86 GRUB 菜单。

注- 还可以将网络安装信息存储在 DHCP 服务器上。

- 对基于 SPARC 的系统，可以使用 `add_install_client` 命令和 `-d` 选项来指定自定义 JumpStart 程序使用 DHCP 服务器。或使用 `boot` 命令和 `dhcp` 选项来指定自定义 JumpStart 程序使用 DHCP 服务器。有关使用该选项的说明，请参见第 81 页中的“SPARC: boot 命令的命令参考”。
 - 对基于 x86 的系统，可以通过以下方式之一使用 `dhcp`：
 - 如果使用安装服务器，可以使用 `add_install_client` 命令和 `-d` 选项来指定自定义 JumpStart 程序使用具有 PXE 的 DHCP 服务器。
 - 可以在 GRUB 菜单上编辑 GRUB 项，并添加 `dhcp` 选项。有关编辑 GRUB 项的说明，请参见第 85 页中的“x86: 通过编辑 GRUB boot 命令执行自定义 JumpStart 安装”。
-

1 在安装服务器或引导服务器上，以超级用户身份登录。

2 使用文本编辑器打开 `/etc/bootparams`。

3 添加此项。

```
* install_config=server:jumpstart_dir_path
```

* 通配符，它指定所有系统都具有访问权限

server JumpStart 目录所在的配置文件服务器的主机名

jumpstart_dir_path JumpStart 目录的绝对路径

例如，以下项允许所有系统访问名为 `sherlock` 的配置文件服务器上的 `/jumpstart` 目录：

```
* install_config=sherlock:/jumpstart
```



注意- 在引导安装客户机时使用此过程可能会产生以下错误消息：

WARNING: getfile: RPC failed: error 5: (RPC Timed out). (警告: getfile: RPC 失败: 错误 5 (RPC 超时)。)

第 162 页中的“从网络引导: 错误消息”包含有关此错误消息的详细信息。

现在，所有系统都可以访问配置文件服务器。

创建独立系统的配置文件软盘

包含 JumpStart 目录的软盘称为配置文件软盘。未联网的系统不具有对配置文件服务器的访问权限。因此，如果系统没有连接到网络，您必须在软盘上创建一个 JumpStart 目录。创建配置文件软盘时所在的系统必须具有软盘驱动器。

JumpStart 目录包含所有基本的自定义 JumpStart 文件，例如 rules 文件、rules.ok 文件和配置文件。该目录必须保存在配置文件软盘的根 (/) 目录中。

请参见以下过程之一：

- 第 29 页中的“SPARC: 创建配置文件软盘”
- 第 31 页中的“x86: 使用 GRUB 创建配置文件软盘”

▼ SPARC: 创建配置文件软盘

注 - 此过程假设系统正在运行卷管理器。如果未使用卷管理器来管理软盘、CD 和 DVD，请参阅《系统管理指南：设备和文件系统》，以获得有关在不使用卷管理器的情况下管理可移除介质的详细信息。

1 找到连接了软盘驱动器的基于 SPARC 的系统。

2 成为超级用户或承担等效角色。

角色包含授权和具有一定权限的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

3 将空白软盘或可覆写的软盘放入软盘驱动器。

4 挂载软盘。

```
# volcheck
```

5 确定该软盘是否包含 UNIX 文件系统 (UFS)。

在系统上检查 /etc/mnttab 文件的内容，确定是否存在与以下内容类似的项：

```
/vol/dev/diskette0/scrap /floppy/scrap ufs suid,rw,largefiles,dev=1740008 927147040
```

- 如果该项存在，请转至步骤 7。
- 如果该项不存在，请执行下一步。

6 在软盘上创建 UFS。

```
# newfs /vol/dev/aliases/floppy0
```

7 确定是否要将自定义 JumpStart 文件示例复制到 JumpStart 目录中。

- 如果不复制，请转至步骤 10。
- 如果复制，请使用以下判定表来确定下一步的操作。

示例位置	参考
Solaris Operating System for SPARC Platforms DVD 或 Solaris Software for SPARC Platforms - 1 CD	将 Solaris Operating System for SPARC Platforms DVD 或 Solaris Software for SPARC Platforms - 1 CD 插入服务器的 CD-ROM 驱动器中。 卷管理器会自动挂载 CD 或 DCD。
本地磁盘上 Solaris Operating System for SPARC Platforms DVD 或 Solaris Software for SPARC Platforms - 1 CD 的映像	将目录更改为 Solaris Operating System for SPARC Platforms DVD 或 Solaris Software for SPARC Platforms - 1 CD 映像的位置。例如，键入以下命令： cd /export/install

8 将示例自定义 JumpStart 文件复制到配置文件软盘上的 JumpStart 目录中。

```
# cp -r media_path/Solaris_10/Misc/jumpstart_sample/* jumpstart_dir_path
```

media_path 指向 CD、DVD 或本地磁盘上的映像的路径

jumpstart_dir_path 指向要放置自定义 JumpStart 文件示例的配置文件软盘的路径

注 - 您必须将所有自定义 JumpStart 安装文件都放在该软盘的根 (/) 目录中。

例如，以下命令将 Solaris Software for SPARC Platforms - 1 CD 上的 `jumpstart_sample` 的内容复制到配置文件软盘 `scrap` 的根 (/) 目录中：

```
cp -r /cdrom/cdrom0/Solaris_10/Misc/jumpstart_sample/* /floppy/scrap
```

9 更新配置文件软盘上的 JumpStart 文件示例，以便这些文件可以在您的环境中运行。

10 确保 JumpStart 目录位于 root 中，且权限被设置为 755。

11 弹出该软盘。

```
# eject floppy
```

配置文件软盘创建完成。现在，您可以更新 `rules` 文件并在配置文件软盘上创建配置文件，以执行自定义 JumpStart 安装。要继续，请转至第 32 页中的“创建 `rules` 文件”。

▼ x86: 使用 GRUB 创建配置文件软盘

通过此过程使用 GRUB 创建配置文件软盘。在启用了引导进程的安装过程中会提供 GRUB 菜单。GRUB 菜单将取代以前的发行版中引导系统时所用的 Solaris Device Configuration Assistant。

注 - 此过程假设系统正在运行卷管理器。如果未使用卷管理器来管理软盘、CD 和 DVD，请参阅《系统管理指南：设备和文件系统》，以获得有关在不使用卷管理器的情况下管理可移除介质的详细信息。

- 1 找到连接了软盘驱动器的基于 x86 的系统。
- 2 成为超级用户或承担等效角色。
角色包含授权和具有一定权限的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 3 将空白软盘或可覆写的软盘插入软盘驱动器中。
- 4 挂载软盘。
`# volcheck`
- 5 确定是否要将自定义 JumpStart 文件示例复制到 JumpStart 目录中。
 - 如果不复制，请转至步骤 8。
 - 如果复制，请使用以下判定表来确定下一步的操作。

示例位置	参考
Solaris Operating System for x86 Platforms DVD 或 Solaris Software for x86 Platforms - 1 CD	将 Solaris Operating System for x86 Platforms DVD 或 Solaris Software for x86 Platforms - 1 CD 插入服务器的 CD-ROM 驱动器中。 卷管理器会自动挂载 CD 或 DVD。
本地磁盘上 Solaris Operating System for x86 Platforms DVD 或 Solaris Software for x86 Platforms - 1 CD 的映像	转到 Solaris Operating System for x86 Platforms DVD 或 Solaris Software for x86 Platforms - 1 CD 映像所在的目录。例如，键入以下命令： <code>cd /export/install</code>

- 6 将示例自定义 JumpStart 文件复制到配置文件软盘上的 JumpStart 目录中。

```
# cp -r media_path/Solaris_10/Misc/jumpstart_sample/* jumpstart_dir_path
```

media_path 指向 CD、DVD 或本地磁盘上的映像的路径

jumpstart_dir_path 指向要放置自定义 JumpStart 文件示例的配置文件软盘的路径

注 – 您必须将所有自定义 JumpStart 安装文件都放在该配置文件软盘上的根 (/) 目录中。

例如，以下命令将 Solaris Software for x86 Platforms - 1 CD 上的 `jumpstart_sample` 的内容复制到名为 `scrap` 的配置文件软盘的根 (/) 目录中：

```
cp -r /cdrom/cdrom0/Solaris_10/Misc/jumpstart_sample/* /floppy/scrap
```

- 7 更新配置文件软盘上的 JumpStart 文件示例，以便这些文件可以在您的环境中运行。
- 8 确保 JumpStart 目录位于 `root` 中，且权限被设置为 `755`。
- 9 单击“文件管理器”窗口中的“弹出磁盘”，或在命令行中键入 `eject floppy`，以弹出该软盘。
- 10 在“可移除介质管理器”对话框中，单击“确定”。
- 11 手动弹出该软盘。

另请参见 配置文件软盘创建完成。现在，您可以更新 `rules` 文件并在配置文件软盘上创建配置文件，以执行自定义 JumpStart 安装。要继续，请转至第 32 页中的“创建 `rules` 文件”。

创建 rules 文件

`rules` 文件是一个文本文件，它包含用于要安装 Solaris OS 的各组系统的规则。每个规则区分一组基于一个或多个系统属性的系统。每个规则还可以将相应的系统组链接到一个配置文件。配置文件是一个文本文件，它定义了如何在组中的各个系统上安装 Solaris 软件。例如，以下规则指定 JumpStart 程序使用 `basic_prof` 配置文件中的信息来安装 `sun4u` 平台组中的系统。

```
karch sun4u - basic_prof -
```

`rules` 文件用于创建自定义 JumpStart 安装所必需的 `rules.ok` 文件。

注 – 如果按照第 29 页中的“创建独立系统的配置文件软盘”或第 24 页中的“为联网系统创建配置文件服务器”中描述的过程设置 JumpStart 目录，则可以在 JumpStart 目录中找到 `rules` 示例文件。该 `rules` 文件样例包含文档和一些规则示例。如果使用 `rules` 文件样例，请确保将不想使用的规则示例注释掉。

rules 文件的语法

rules 文件必须具有以下属性：

- 必须将文件的名称指定为 `rules`。
- 该文件必须至少包含一条规则。

rules 文件可以包含以下任何内容：

- 注释文本

JumpStart 将一行中 `#` 符号后的任何文本都视为注释文本。如果某行以 `#` 符号开头，那么整行将被视为注释。
- 一个或多个空白行
- 一条或多条多行规则

要在新的一行上继续使用单个规则，请在按回车键之前添加反斜杠字符 (`\`)。

▼ 创建 rules 文件

- 1 使用文本编辑器创建一个名为 `rules` 的文本文件。或者，打开已创建的 `JumpStart` 目录中的 `rules` 文件样例。
- 2 在 `rules` 文件中为要安装 Solaris 软件的各组系统添加规则。
有关 `rules` 文件中的关键字和值的列表，请参见第 99 页中的“规则关键字和值”。

rules 文件中的规则必须遵循以下语法：

```
!rule_keyword rule_value && !rule_keyword rule_value ... begin profile finish
```

`!` 在关键字之前使用此符号以表示否定。

`rule_keyword` 一个预定义的词法单元或单词，它描述一般的系统属性，如主机名 `hostname` 或内存大小 `memsize`。`rule_keyword` 与规则值一起使用，以便将系统与指向配置文件的相同属性进行匹配。有关规则关键字的列表，请参见第 99 页中的“规则关键字和值”。

`rule_value` 一个为对应规则关键字提供特定系统属性的值。第 99 页中的“规则关键字和值”对规则值进行了说明。

`&&` 在同一规则中，必须使用此符号来连接规则关键字和规则值对（逻辑 AND）。在自定义 JumpStart 安装过程中，一个系统必须首先匹配规则中的每个对，然后才能匹配该规则。

`begin` 可选 Bourne shell 脚本的名称，该脚本可以在安装开始之前执行。如果没有开始脚本，那么必须在此字段中键入一个减号 (`-`)。所有的开始脚本都必须位于 `JumpStart` 目录中。

有关如何创建开始脚本的信息，请参见第 53 页中的“创建开始脚本”。

profile

一种文本文件的名称，该文件定义了当系统与规则相匹配时，在该系统上安装 Solaris 软件的方式。配置文件中的信息包括配置文件关键字及其相应的配置文件值。所有的配置文件都必须位于 JumpStart 目录中。

注 – 第 68 页中的“使用站点特定的安装程序”和第 54 页中的“用开始脚本创建派生配置文件”中介绍了使用 *profile* 字段的可选方法。

finish

可选 Bourne shell 脚本的名称，该脚本可以在安装完成之后执行。如果没有结束脚本，那么必须在此字段中键入一个减号 (-)。所有的结束脚本都必须位于 JumpStart 目录中。

有关如何创建结束脚本的信息，请参见第 55 页中的“创建结束脚本”。

每条规则至少必须包含以下内容：

- 一个关键字、一个值和一个相应的配置文件
- 如果未指定任何开始或结束脚本，则要在 *begin* 和 *finish* 字段中包含一个减号 (-)

- 3 将 rules 文件保存在 JumpStart 目录中。
- 4 确保 root 拥有 rules 文件，并且权限设置为 644。

rules 文件示例

下面的示例显示了一个 rules 文件中的若干条规则示例。每一行都包含一个规则关键字和该关键字的一个有效值。JumpStart 程序从上到下扫描 rules 文件。

当 JumpStart 程序找到与已知系统相匹配的规则关键字和值时，它将安装 *profile* 字段中列出的配置文件所指定的 Solaris 软件。

有关 rules 文件限制的完整列表，请参见第 33 页中的“rules 文件的语法”。

示例 3-1 rules 文件

```
# rule keywords and rule values      begin script      profile      finish script
# -----
hostname eng-1                       -               basic_prof   -
network 192.168.255.255 && !model \
'SUNW,Sun-Blade-100'                 -               net_prof     -
```

示例 3-1 rules 文件 (续)

```

model SUNW,SPARCstation-LX      -          lx_prof      complete
network 192.168.2.0 && karch i86pc  setup    x86_prof      done
memsize 64-128 && arch i386        -          prog_prof     -
any -                             -          generic_prof  -

```

下表说明了此示例中的一些关键字和值。

hostname	如果系统的主机名为 <code>eng-1</code> ，则系统与此规则相匹配。将使用 <code>basic_prof</code> 配置文件在与此规则匹配的系统上安装 Solaris 软件。
network	如果系统位于子网 <code>192.168.255.255</code> 上，并且系统不是 Sun Blade™ 100 (SUNW, Sun-Blade-100)，则系统与此规则相匹配。 <code>net_prof</code> 配置文件用于在与此规则匹配的系统上安装 Solaris 软件。此规则还提供了一个通过使用反斜杠字符 (\) 在新的一行上继续使用单个规则的示例。
model	如果系统是 SPARCstation LX，则系统与此规则相匹配。 <code>lx_prof</code> 配置文件和 <code>complete</code> 结束脚本用于在与该规则匹配的系统上安装 Solaris 软件。
network	如果系统位于子网 <code>192.168.2.0</code> 上，并且为基于 x86 的 <code>sun4u</code> 系统，则系统与此规则相匹配。 <code>setup</code> 开始脚本、 <code>x864u_prof</code> 配置文件和 <code>done</code> 结束脚本用于在匹配该规则的系统上安装 Solaris 软件。
memsize	如果系统内存介于 64 MB 和 128 MB 之间，且该系统为基于 x86 的系统，则该规则将匹配。将使用 <code>prog_prof</code> 配置文件在与此规则匹配的系统上安装 Solaris 软件。
any	如果系统不与前面的任何规则相匹配，则系统与此规则相匹配。将使用 <code>generic_prof</code> 配置文件在与此规则匹配的系统上安装 Solaris 软件。如果使用了 <code>any</code> ，那么它始终应是 <code>rules</code> 文件中的最后一个规则。

创建配置文件

配置文件是一个文本文件，它定义如何在系统上安装 Solaris 软件。配置文件定义了安装元素，例如要安装的软件组。每条规则指定一个配置文件，配置文件定义了安装系统的方式。您可以为每条规则创建多个不同的配置文件，也可以在多条规则中使用同一个配置文件。

配置文件由一个或多个配置文件关键字及其值组成。每个配置文件关键字都是一个命令，该命令对 JumpStart 程序如何在一个系统上安装 Solaris 软件的某个方面加以控制。例如，以下配置文件关键字和值指定 JumpStart 程序将系统安装为一个服务器：

```
system_type server
```

注 - 如果使用下列任一步骤创建了 JumpStart 目录，则 JumpStart 目录中应该已存在样例配置文件：

- 第 24 页中的 “为联网系统创建配置文件服务器”
 - 第 29 页中的 “创建独立系统的配置文件软盘”
-

配置文件语法

配置文件必须包含以下内容：

- 作为第一项的 `install_type` 配置文件关键字
- 每行一个关键字
- 如果正在通过该配置文件进行升级的系统包含多个可升级的根 (/) 文件系统，则需要包含 `root_device` 关键字

配置文件可以包含以下内容：

- 注释文本
JumpStart 程序将一行中 # 符号之后的任何文本都视为注释文本。如果某行以 # 符号开头，那么整行将被视为注释。
- 一个或多个空白行

▼ 创建配置文件

- 1 使用文本编辑器创建一个文本文件。描述性地命名该文件。或者，打开已创建的 JumpStart 目录中的样例配置文件。

注 - 确保配置文件的名称可以反映您要使用配置文件在系统上安装 Solaris 软件的方式。例如，可以将配置文件命名为 `basic_install`、`eng_profile` 或 `user_profile`。

- 2 在该配置文件中添加配置文件关键字和值。
有关配置文件关键字和值的列表，请参见第 103 页中的 “配置文件关键字和值”。

注 - 配置文件关键字及其值区分大小写。

- 3 将配置文件保存在 JumpStart 目录中。
- 4 确保 root 用户拥有该配置文件，并且权限被设置为 644。

5 测试配置文件（可选）。

第 47 页中的“测试配置文件”包含有关测试配置文件的信息。

配置文件示例

下面的配置文件示例显示了如何使用不同的配置文件关键字和配置文件值控制在系统上安装 Solaris 软件的方式。第 103 页中的“配置文件关键字和值”包含配置文件关键字和值的描述。

注 - 如果您要安装 Solaris ZFS™ 根池，请参见第 9 章，使用 JumpStart 安装 ZFS 根池以了解限制和配置文件示例。

示例 3-2 挂载远程文件系统以及添加和删除软件包

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone
partitioning           default
filesystems            any 512 swap # specify size of /swap
cluster                SUNWCprog
package                SUNWman delete
cluster                SUNWCacc
```

下表说明了此示例中的一些关键字和值。

install_type	install_type 关键字是每个配置文件中都必需的。
system_type	system_type 关键字定义将该系统安装为一个独立系统。
partitioning	文件系统片由将要使用 default 值进行安装的软件来确定。swap 的大小设置为 512 MB，并安装在任何磁盘上（值为 any）。
cluster	开发者 Solaris 软件组 SUNWCprog 将安装在系统上。
package	如果标准手册页从网络上的文件服务器 s_ref 挂载，则手册页软件包将不安装在系统上。而是安装包含 System Accounting 公用程序的软件包。

示例 3-3 挂载远程文件系统和添加第三方软件包

```
# profile keywords      profile values
```

示例 3-3 挂载远程文件系统和添加第三方软件包 (续)

```
# -----
install_type      initial_install
system_type       standalone
partitioning      default
filesystems       any 512 swap # specify size of /swap
cluster           SUNWCprog
cluster           SUNWCacc
package           apache_server \
                  http://package.central/packages/apache timeout 5
```

下表说明了此示例中的一些关键字和值。

install_type	install_type 关键字是每个配置文件中都必需的。
system_type	system_type 关键字定义将该系统安装为一个独立系统。
partitioning	文件系统片由将要使用 default 值进行安装的软件来确定。swap 的大小设置为 512 MB，并安装在任何磁盘上（值为 any）。
cluster	开发者 Solaris 软件组 SUNWCprog 将安装在系统上。
package	第三方软件包安装在 HTTP 服务器中的系统上。

示例 3-4 指定文件系统的安装位置

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone
partitioning           explicit
filesystems            c0t0d0s0 auto /
filesystems            c0t3d0s1 auto swap
filesystems            any auto usr
cluster                SUNWCall
```

下表说明了此示例中的一些关键字和值。

partitioning	文件系统片由 filesystems 关键字和 explicit 值来确定。根 (/) 的大小基于选中的软件，值为 auto，并安装在 c0t0d0s0 上。swap 的大小设置为所需大小，并安装在 c0t3d0s1 上。usr 基于选中的软件；安装程序根据值 any 来确定 usr 的安装位置。
cluster	完整 Solaris 软件组 SUNWCall 将安装在系统上。

示例3-5 升级和安装修补程序

```
# profile keywords      profile values
# -----
install_type           upgrade
root_device            c0t3d0s2
backup_media           remote_filesystem timber:/export/scratch
package                SUNWbcp delete
package               SUNWxwman add
cluster               SUNWCacc add
patch                 patch_list nfs://patch_master/Solaris_10/patches \
                      retry 5
locale                 de
```

下表说明了此示例中的一些关键字和值。

install_type	该配置文件通过重新分配磁盘空间来升级系统。在本示例中，必须重新分配磁盘空间，因为系统上的某些文件系统没有足够的空间可用于升级。
root_device	升级 c0t3d0s2 上的根文件系统。
backup_media	重新分配磁盘空间期间，使用远程系统 timber 来备份数据。有关更多的备份介质关键字值的信息，请参见第 110 页中的“ backup_media 配置文件关键字 ”。
package	升级后，系统上不安装二进制兼容性软件包 SUNWbcp。
package	如果系统尚未安装 X 窗口系统手册页和 System Accounting 实用程序，则此代码可以确保系统安装这些内容。自动升级系统上已有的所有软件包。
patch	升级时安装的修补程序的列表。修补程序列表位于 patch_master NFS 服务器上的 Solaris_10/patches 目录下。如果挂载失败，则会尝试进行五次 NFS 挂载。
locale	在系统上安装德文本地化软件包。

示例3-6 重新分配磁盘空间，以便升级

```
# profile keywords      profile values
# -----
install_type           upgrade
root_device            c0t3d0s2
backup_media           remote_filesystem timber:/export/scratch
layout_constraint      c0t3d0s2 changeable 100
layout_constraint      c0t3d0s4 changeable
layout_constraint      c0t3d0s5 movable
```

示例 3-6 重新分配磁盘空间，以便升级 (续)

```

package      SUNWbcp delete
package      SUNWxwman add
cluster      SUNWCacc add
locale       de

```

下表说明了此示例中的一些关键字和值。

install_type	该配置文件通过重新分配磁盘空间来升级系统。在本示例中，必须重新分配磁盘空间，因为系统上的某些文件系统没有足够的空间可用于升级。
root_device	升级 c0t3d0s2 上的根文件系统。
backup_media	重新分配磁盘空间期间，使用远程系统 timber 来备份数据。有关更多的备份介质关键字值的信息，请参见第 110 页中的“ backup_media 配置文件关键字 ”。
layout_constraint	layout_constraint 关键字指定自动布局在尝试重新分配磁盘空间以便升级时，可以执行以下操作。 <ul style="list-style-type: none"> ■ 更改片 2 和 4。可以将它们移到其他位置，也可以更改其大小。 ■ 移动片 5。可以将该片移到其他位置，但不能更改其大小。
package	升级后，系统上不安装二进制兼容性软件包 SUNWbcp。
package	如果系统尚未安装 X 窗口系统手册页和 System Accounting 实用程序，则此代码可以确保系统安装这些内容。自动升级系统上已有的所有软件包。
locale	在系统上安装德文本地化软件包。

示例 3-7 从 HTTP 服务器检索 Solaris Flash 归档文件

在下面的配置文件示例中，自定义 JumpStart 程序从 HTTP 服务器上检索 Solaris Flash 归档文件。

```

# profile keywords      profile values
# -----
install_type           flash_install
archive_location       http://192.168.255.255/flasharchive/solarisarchive
partitioning           explicit
filesystems            c0t1d0s0 4000 /
filesystems            c0t1d0s1 512 swap
filesystems            c0t1d0s7 free /export/home

```

下表说明了此示例中的一些关键字和值。

示例 3-7 从 HTTP 服务器检索 Solaris Flash 归档文件 (续)

<code>install_type</code>	该配置文件在克隆系统上安装 Solaris Flash 归档文件。在初始安装过程中，所有文件都被覆盖。
<code>archive_location</code>	从 HTTP 服务器检索 Solaris Flash 归档文件。
<code>partitioning</code>	文件系统片由 <code>filesys</code> 关键字和 <code>explicit</code> 值来确定。根目录 (/) 的大小基于 Solaris Flash 归档文件的大小。根文件系统安装在 <code>c0t1d0s0</code> 上。 <code>swap</code> 的大小被设置为所需的大小，安装在 <code>c0t1d0s1</code> 上。 <code>/export/home</code> 基于剩余的磁盘空间。 <code>/export/home</code> 安装在 <code>c0t1d0s7</code> 上。

示例 3-8 从安全 HTTP 服务器检索 Solaris Flash 归档文件

在下面的配置文件示例中，自定义 JumpStart 程序从安全的 HTTP 服务器上检索 Solaris Flash 归档文件。

```
# profile keywords      profile values
# -----
install_type           flash_install
archive_location       https://192.168.255.255/solarisupdate.flar
partitioning           explicit
filesys                c0t1d0s0 4000 /
filesys                c0t1d0s1 512 swap
filesys                c0t1d0s7 free /export/home
```

下表说明了此示例中的一些关键字和值。

<code>install_type</code>	该配置文件在克隆系统上安装 Solaris Flash 归档文件。在初始安装过程中，所有文件都被覆盖。
<code>archive_location</code>	从安全 HTTP 服务器检索已压缩的 Solaris Flash 归档文件。
<code>partitioning</code>	文件系统片由 <code>filesys</code> 关键字和 <code>explicit</code> 值来确定。根目录 (/) 的大小基于 Solaris Flash 归档文件的大小。 <code>swap</code> 的大小被设置为所需的大小，安装在 <code>c0t1d0s1</code> 上。 <code>/export/home</code> 基于剩余的磁盘空间。 <code>/export/home</code> 安装在 <code>c0t1d0s7</code> 上。

示例 3-9 检索 Solaris Flash 归档文件和安装第三方软件包

在下面的配置文件示例中，自定义 JumpStart 程序从 HTTP 服务器上检索 Solaris Flash 归档文件。

```
# profile keywords      profile values
# -----
```

示例 3-9 检索 Solaris Flash 归档文件和安装第三方软件包 (续)

```
install_type          flash_install
archive_location      http://192.168.255.255/flasharchive/solarisarchive
partitioning          explicit
fileys                c0t1d0s0 4000 /
fileys                c0t1d0s1 512 swap
fileys                c0t1d0s7 free /export/home
package              SUNWnew http://192.168.254.255/Solaris_10 timeout 5
```

下表说明了此示例中的一些关键字和值。

install_type	该配置文件在克隆系统上安装 Solaris Flash 归档文件。在初始安装过程中，所有文件都被覆盖。
archive_location	从 HTTP 服务器检索 Solaris Flash 归档文件。
partitioning	文件系统片由 fileys 关键字和 explicit 值来确定。根目录 (/) 的大小基于 Solaris Flash 归档文件的大小。根文件系统安装在 c0t1d0s0 上。swap 的大小被设置为所需的大小，安装在 c0t1d0s1 上。/export/home 基于剩余的磁盘空间。/export/home 安装在 c0t1d0s7 上。
package	SUNWnew 软件包是从 HTTP 服务器 192.168.254.255 上的 Solaris_10 目录添加的。

示例 3-10 从 NFS 服务器检索 Solaris Flash 差别归档文件

在下面的配置文件示例中，自定义 JumpStart 程序从 NFS 服务器上检索 Solaris Flash 归档文件。flash_update 关键字表示这是差别归档文件。差别归档文件仅安装两个系统映像之间不同的部分。

```
# profile keywords          profile values
# -----
install_type                flash_update
archive_location            nfs installserver:/export/solaris/flasharchive \
                             /solarisdiffarchive
no_master_check
```

下表说明了此示例中的一些关键字和值。

install_type	该配置文件在克隆系统上安装 Solaris Flash 差别归档文件。仅安装归档文件指定的文件。
archive_location	从 NFS 服务器检索 Solaris Flash 归档文件。
no_master_check	不会对克隆系统进行系统映像有效性的检查。此类有效系统映像是从初始主系统创建的。

示例 3-11 创建空引导环境

在下面的配置文件示例中，自定义 JumpStart 程序创建空引导环境。空引导环境不包含任何文件系统，也不从当前引导环境进行复制。引导环境可以在以后使用 Solaris Flash 归档文件来总装，然后将其激活。

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone
partitioning           explicit
filesys                c0t0d0s0 auto /
filesys                c0t3d0s1 auto swap
filesys                any auto usr
cluster                SUNWCall
bootenv createbe       bename second_BE \
filesystem /:/dev/dsk/c0t1d0s0:ufs \
filesystem -:/dev/dsk/c0t1d0s0:swap \
filesystem /export:shared:ufs
```

下表说明了此示例中的一些关键字和值。

partitioning	文件系统片由 filesys 关键字和 explicit 值来确定。根 (/) 的大小基于选中的软件，值为 auto，并安装在 c0t0d0s0 上。swap 的大小设置为所需大小，并安装在 c0t3d0s1 上。usr 基于选中的软件；安装程序根据值 any 来确定 usr 的安装位置。
cluster	完整 Solaris 软件组 SUNWCall 将安装在系统上。
bootenv createbe	在磁盘 c0t1d0 上设置未激活的空引导环境。将创建根 (/)、交换和 /export 的文件系统，但是保留为空。此二级引导环境可以在以后使用 Solaris Flash 归档文件来安装。然后可以激活该新引导环境，使之成为当前引导环境。

有关关键字值和使用此关键字的背景，请参见以下参考资料：

- 有关关键字值的描述，请参见第 103 页中的“配置文件关键字和值”。
- 有关使用 Solaris Live Upgrade 创建、升级和激活非活动引导环境的背景，请参见《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 2 章“Solaris Live Upgrade（概述）”。
- 有关使用 Solaris Flash 归档文件的背景，请参见《Solaris 10 10/08 安装指南：Solaris Flash 归档文件（创建和安装）》中的第 1 章“Solaris Flash（概述）”。

示例 3-12 安装 Solaris Flash 归档文件时创建 RAID-1 卷

在下面的配置文件示例中，自定义 JumpStart 程序使用 Solaris 卷管理器技术为根 (/)、swap、/usr 和 /export/home 文件系统创建 RAID-1 卷（镜像）。在引导环境中安装 Solaris Flash 归档文件。

```
# profile keywords      profile values
# -----
install_type           flash_install
archive_location       nfs server:/export/home/export/flash.s10.SUNWCall
partitioning           explicit
fileSYS                mirror:d10 c0t0d0s0 c0t1d0s0 4096 /
fileSYS                mirror c0t0d0s1 2048 swap
fileSYS                mirror:d30 c0t0d0s3 c0t1d0s3 4096 /usr
fileSYS                mirror:d40 c0t0d0s4 c0t1d0s4 4096 /usr
fileSYS                mirror:d50 c0t0d0s5 c0t1d0s5 free /export/home
metadb                 c0t1d0s7 size 8192 count 3
```

下表说明了此示例中的一些关键字和值。

install_type	该配置文件在克隆系统上安装 Solaris Flash 归档文件。在初始安装过程中，所有文件都被覆盖。
archive_location	从 NFS 服务器检索 Solaris Flash 归档文件。
partitioning	文件系统片由 fileSYS 关键字和 explicit 值来确定。
fileSYS	在片 c0t0d0s0 和 c0t1d0s0 上创建根 (/) 文件系统和镜像。根 (/) 文件系统的大小设置为 4096 MB。镜像 c0t0d0s0 和 c0t1d0s0 的 RAID-1 卷名为 d10。
fileSYS	在片 c0t0d0s1 上创建 swap 文件和镜像，共占用 2048 MB。自定义 JumpStart 程序将为镜像指定名称。
fileSYS	在片 c0t1d0s3 和 c0t0d0s3 上创建 /usr 文件系统和镜像。/usr 文件系统的大小设置为 4096 MB。RAID-1 卷名为 d30。
fileSYS	在片 c0t1d0s4 和片 c0t0d0s4 上创建 /usr 文件系统和镜像。/usr 文件系统的大小设置为 4096 MB。RAID-1 卷名为 d40。
metadb	在片 c0t1d0s7 上安装三个状态数据库副本 (metadbs)，共占用 8192 个块 (4 MB)。

- 有关在安装期间如何创建镜像文件系统的概述信息，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的第 9 章“在安装过程中创建 RAID-1 卷（镜像）（概述）”。
- 有关创建镜像文件系统的原则和要求，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的第 10 章“在安装过程中创建 RAID-1 卷（镜像）（规划）”。

示例 3-12 安装 Solaris Flash 归档文件时创建 RAID-1 卷 (续)

- 有关关键字值的描述，请参见第 123 页中的“[filesys 配置文件关键字（创建 RAID-1 卷）](#)”和第 128 页中的“[metadb 配置文件关键字（创建状态数据库副本）](#)”。

示例 3-13 创建 RAID-1 卷，以镜像根文件系统

在下面的配置文件示例中，自定义 JumpStart 程序使用 Solaris 卷管理器技术为根 (/) 文件系统创建 RAID-1 卷（镜像）。

```
# profile keywords      profile values
# -----
install_type          initial_install
cluster               SUNWCXall
filesys               mirror:d30 c0t1d0s0 c0t0d0s0 /
filesys               c0t0d0s3 512 swap
metadb                c0t0d0s4 size 8192 count 4
metadb                c0t1d0s4 size 8192 count 4
```

下表说明了此示例中的一些关键字和值。

cluster	完整 Solaris 软件加 OEM 支持软件组 (SUNWCXall) 将安装在系统上。
filesys	在片 c0t1d0s0 和片 c0t0d0s0 上创建根 (/) 文件系统和镜像。镜像 c0t1d0s0 和 c0t0d0s0 的 RAID-1 卷名为 d30。自定义 JumpStart 程序将为两个子镜像指定名称。
filesys	在片 c0t0d0s3 上创建 swap 文件系统和镜像，该系统大小为 512 MB。
metadb	在片 c0t0d0s4 上安装四个状态数据库副本 (metadbs)，共占用 8192 个块 (4 MB)
metadb	在片 c0t1d0s4 上安装四个状态数据库副本 (metadbs)，共占用 8192 个块 (4 MB)

- 有关在安装期间如何创建 RAID-1 卷的概述信息，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的第 9 章“[在安装过程中创建 RAID-1 卷（镜像）（概述）](#)”。
- 有关创建 RAID-1 卷的原则和要求，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的第 10 章“[在安装过程中创建 RAID-1 卷（镜像）（规划）](#)”。
- 有关关键字值的描述，请参见第 123 页中的“[filesys 配置文件关键字（创建 RAID-1 卷）](#)”和第 128 页中的“[metadb 配置文件关键字（创建状态数据库副本）](#)”。

示例 3-14 创建 RAID-1 卷，以镜像多个文件系统

在下面的配置文件示例中，自定义 JumpStart 程序使用 Solaris 卷管理器技术为根 (/) 文件系统、swap 文件系统和 /usr 文件系统创建 RAID-1 卷（镜像）。

```
# profile keywords      profile values
# -----
install_type          initial_install
cluster              SUNWCXall
filesystems          mirror:d100 c0t1d0s0 c0t0d0s0 200 /
filesystems          c0t1d0s5 500 /var
filesystems          c0t0d0s5 500
filesystems          mirror c0t0d0s1 512 swap
metadb               c0t0d0s3 size 8192 count 5
filesystems          mirror c0t1d0s4 c0t0d0s4 2000 /usr
filesystems          c0t1d0s7 free /export/home
filesystems          c0t0d0s7 free
```

下表说明了此示例中的一些关键字和值。

- | | |
|-------------|--|
| cluster | 完整 Solaris 软件加 OEM 支持软件组 (SUNWCXall) 将安装在系统上。 |
| filesystems | 在片 c0t1d0s0 和片 c0t0d0s0 上创建根 (/) 文件系统和镜像。根 (/) 文件系统的大小被设置成 200 MB。镜像 c0t1d0s0 和 c0t0d0s0 的 RAID-1 卷名为 d100。 |
| filesystems | 在片 c0t1d0s5 上安装 /var 文件系统，共占用 500 MB。在片 c0t1d0s0 和片 c0t0d0s0 上创建根 (/) 文件系统和镜像。根 (/) 文件系统的大小被设置成 200 MB。镜像 c0t1d0s0 和 c0t0d0s0 的 RAID-1 卷名为 d100。 |
| filesystems | 在片 c0t0d0s1 上创建 swap 文件系统和镜像，并为该文件系统分配 512 MB 的磁盘空间。自定义 JumpStart 程序将为镜像指定名称。 |
| metadb | 在片 c0t0d0s3 上安装五个状态数据库副本 (metadbs)，共占用 8192 个块 (4 MB)。 |
| filesystems | 在片 c0t1d0s4 和片 c0t0d0s4 上创建 /usr 文件系统和镜像。/usr 文件系统的大小被设置成 2000 MB。自定义 JumpStart 程序将为镜像指定名称。 |
- 有关在安装期间如何创建镜像文件系统的概述信息，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的第 9 章“在安装过程中创建 RAID-1 卷（镜像）（概述）”。
 - 有关创建镜像文件系统的原则和要求，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的第 10 章“在安装过程中创建 RAID-1 卷（镜像）（规划）”。
 - 有关关键字值的描述，请参见第 123 页中的“filesystems 配置文件关键字（创建 RAID-1 卷）”和第 128 页中的“metadb 配置文件关键字（创建状态数据库副本）”。

示例 3-15 x86: 使用 fdisk 关键字

```
# profile keywords      profile values
# -----
install_type          initial_install
system_type           standalone

fdisk                 c0t0d0 0x04 delete
fdisk                 c0t0d0 solaris maxfree
cluster               SUNWCall
cluster               SUNWCacc delete
```

下表说明了此示例中的一些关键字和值。

fdisk	从 c0t0d0 磁盘上删除所有 DOSOS16 类型（十六进制为 04）的 fdisk 分区。
fdisk	在磁盘 c0t0d0 上的最大连续空闲空间上创建 Solaris fdisk 分区。
cluster	在系统上安装完整分发软件组 SUNWCall。
cluster	不在系统上安装 System Accounting 实用程序 SUNWCacc。

测试配置文件

创建配置文件后，使用 `pfinstall(1M)` 命令测试配置文件。请在使用配置文件安装或升级一个系统之前，测试该配置文件。在创建用于重新分配磁盘空间的升级配置文件时，测试配置文件尤为重要。

通过查看 `pfinstall` 生成的安装输出，可以快速确定配置文件是否在按预期的方式工作。例如，在一个系统上执行升级之前，使用配置文件可以确定该系统是否有足够的磁盘空间，以升级到 Solaris 软件的新发行版。

`pfinstall` 允许您针对以下方面来测试一个配置文件：

- 运行 `pfinstall` 的系统的磁盘配置。
- 其他磁盘配置。磁盘配置文件用于表示磁盘的结构，例如磁盘每扇区的字节数、标志和片。第 63 页中的“创建磁盘配置文件”和第 65 页中的“x86: 创建磁盘配置文件”说明了如何创建磁盘配置文件。

注 - 您不能使用磁盘配置文件测试要用于升级系统的配置文件，而必须根据系统的实际磁盘配置以及该系统上当前已安装的软件来测试配置文件。

▼ 创建用于测试配置文件的临时 Solaris 环境

要成功而准确地测试某一特定 Solaris 发行版的配置文件，您必须在同一发行版的 Solaris 环境中测试配置文件。例如，如果要测试 Solaris 初始安装配置文件，请在运行 Solaris OS 的系统上运行 `pfinstall` 命令。

如果要在以下某种情况下测试配置文件，则需要创建一个临时安装环境：

- 希望在运行 Solaris 软件早期版本的系统上测试 Solaris 10 10/08 升级配置文件。
- 尚未安装用于测试 Solaris 10 10/08 初始安装配置文件的 Solaris 10 10/08 系统。

1 从以下介质之一的映像引导系统：

对于基于 SPARC 的系统：

- Solaris Operating System for SPARC Platforms DVD
- Solaris Software for SPARC Platforms - 1 CD

对于基于 x86 的系统：

- Solaris Operating System for x86 Platforms DVD
- Solaris Software for x86 Platforms - 1 CD

注 - 如果要测试升级配置文件，请引导要升级的系统。

2 响应系统标识问题。

3 要从安装程序退出，请在以下提示符后键入！。

```
The Solaris installation program will assist you in installing software for Solaris.  
<Press ENTER to continue> {"!" exits}
```

4 从 shell 执行 `pfinstall` 命令。有关使用 `pfinstall` 命令的详细信息，请参见第 48 页中的“测试配置文件”中的步骤 7。

▼ 测试配置文件

仅适用于 x86 - 如果使用 `locale` 关键字，则 `pfinstall -D` 命令无法测试配置文件。有关解决方法，请参见第 167 页中的“升级 Solaris 操作系统”一节中的错误消息“无法选择语言环境”。

1 找到要进行配置文件测试的系统，该平台的类型与创建配置文件的平台的类型相同（SPARC 或 x86）。

如果要测试升级配置文件，那么您必须在要升级的实际系统上测试该配置文件。

2 使用以下判定表来确定下一步的操作。

测试方案	参考
测试初始安装配置文件，并具有运行 Solaris 10 10/08 软件的系统。	成为系统的超级用户，然后转至步骤 5。
测试升级配置文件，或没有运行用于测试初始安装配置文件的 Solaris 10 10/08 系统。	创建一个临时 Solaris 10 10/08 环境，以测试配置文件。有关详细信息，请参见第 48 页中的“创建用于测试配置文件的临时 Solaris 环境”。然后，请转至步骤 3。

3 创建一个临时挂载点。

```
# mkdir /tmp/mnt
```

4 挂载包含要测试的配置文件的目录。

挂载方案	键入指令
挂载网络系统的远程 NFS 文件系统。	<code>mount -F nfs server_name:path /tmp/mnt</code>
SPARC：挂载 UFS 格式的软盘。	<code>mount -F ufs /dev/diskette /tmp/mnt</code>
挂载 PCFS 格式的软盘。	<code>mount -F pcfs /dev/diskette /tmp/mnt</code>

5 要使用特定的系统内存大小测试配置文件，请将 SYS_MEMSIZE 设置为特定的内存大小（以 MB 为单位）。

```
# SYS_MEMSIZE=memory_size
# export SYS_MEMSIZE
```

6 是否已在步骤 4 中挂载了目录？

- 如果是，请将该目录更改为 /tmp/mnt。

```
# cd /tmp/mnt
```

- 如果否，请将该目录更改为配置文件所在的目录（通常为 JumpStart 目录）。

```
# cd jumpstart_dir_path
```

7 使用 `pfinstall(1M)` 命令测试配置文件。

```
# /usr/sbin/install.d/pfinstall -D:-d disk_config_file -c path profile
```



注意 - 必须包含 `-d` 或 `-D` 选项。如果这两个选项都未包括，则 `pfinstall` 将使用所指定的配置文件来安装 Solaris 软件。系统上的所有数据都会被覆写。

`-D` `pfinstall` 使用当前系统的磁盘配置来测试配置文件。测试升级配置文件时必须使用 `-D` 选项。

`-d disk_config_file` `pfinstall` 使用磁盘配置文件 `disk_config_file` 来测试配置文件。如果 `disk_config_file` 不在运行 `pfinstall` 的目录中，您必须指定路径。

有关如何创建磁盘配置文件的说明，请参见第 63 页中的“创建磁盘配置文件”。

注 - 不能将 `-d disk_config_file` 选项与升级配置文件 `install_type upgrade` 一起使用。而必须始终根据系统的磁盘配置来测试升级配置文件，也就是说，您必须使用 `-D` 选项。

`-c path` 指向 Solaris 软件映像的路径。例如，如果系统正在使用卷管理器来挂载您的平台的 Solaris Software - 1 CD，则使用此选项。

注 - 如果您从平台的 Solaris Operating System DVD 或 Solaris Software - 1 CD 映像进行引导，那么不需要使用 `-c` 选项。此时，DVD 或 CD 映像将作为引导过程的一部分挂载在 `/cdrom` 上。

`profile` 要测试的配置文件的名称。如果 `profile` 不在 `pfinstall` 的运行目录中，那么您必须指定路径。

配置文件测试示例

以下示例显示了如何使用 `pfinstall` 来测试配置文件 `basic_prof`。对该配置文件的测试是针对安装有 Solaris 10 10/08 软件的系统上的磁盘配置而进行的。`basic_prof` 配置文件位于 `/jumpstart` 目录中。同时，因为使用了卷管理器，所以指向 Solaris Operating System DVD 映像的路径已经指定。

示例 3-16 使用 Solaris 10 10/08 系统来测试配置文件

```
# cd /jumpstart
# /usr/sbin/install.d/pfinstall -D -c /cdrom/pathname basic_prof
```

以下示例显示了如何使用 `pinstall` 在 Solaris 10 10/08 系统上测试名为 `basic_prof` 的配置文件。该测试是针对磁盘配置文件 `535_test` 而进行的，目的是检查系统是否具有 64 MB 内存。此示例使用了 `/export/install` 目录中的 Solaris Software for SPARC Platforms - 1 CD 或 Solaris Software for x86 Platforms - 1 CD 映像。

示例 3-17 使用磁盘配置文件来测试配置文件

```
# SYS_MEMSIZE=64
# export SYS_MEMSIZE
# /usr/sbin/install.d/pinstall -d 535_test -c /export/install basic_prof
```

验证 rules 文件

在能够使用配置文件和 `rules` 文件之前，您必须运行 `check` 脚本来验证是否已正确设置这些文件。如果所有的规则和配置文件都已正确设置，那么将创建 `rules.ok` 文件，自定义 JumpStart 安装软件需要使用该文件将系统与配置文件进行匹配。

表 3-2 描述了 `check` 脚本的作用。

表 3-2 使用 `check` 脚本时发生的情况

阶段	说明
1	检查 <code>rules</code> 文件的语法。 <code>check</code> 将验证规则关键字是否合法，以及是否为每条规则都指定了 <code>begin</code> 、 <code>class</code> 和 <code>finish</code> 字段。 <code>begin</code> 和 <code>finish</code> 字段可以包含减号 (-) 而不是文件名。
2	如果 <code>rules</code> 文件中没有错误，将检查规则中指定的各个配置文件的语法。
3	如果未发现错误，则 <code>check</code> 会根据 <code>rules</code> 文件创建 <code>rules.ok</code> 文件，删除所有注释和空白行，保留所有规则并在结尾添加以下注释行： <code># version=2 checksum=num</code>

▼ 验证 rules 文件

- 1 确保 `check` 脚本位于 JumpStart 目录中。

注 - `check` 脚本位于 Solaris Operating System DVD 或 Solaris Software - 1 CD 上的 `Solaris_10/Misc/jumpstart_sample` 目录中。

- 2 转到 JumpStart 目录。

3 运行 check 脚本以验证 rules 文件：

```
$ ./check -p path -r file_name
```

-p path 使用 Solaris 软件映像中的 check 脚本（而不是所用系统中的 check 脚本）来验证 rules。path 是本地磁盘上的映像或已挂载的 Solaris Operating System DVD 或 Solaris Software - 1 CD。

如果系统运行的是 Solaris 的早期版本，则使用此选项可运行 check 的最新版本。

-r file_name 指定一个名称不是 rules 的规则文件。使用此选项，您可以在将某条规则集成到 rules 文件之前，测试该规则的有效性。

check 脚本运行时，它会报告对 rules 文件和每个配置文件的有效性的检查情况。如果未遇到错误，该脚本将报告以下信息。

```
The custom JumpStart configuration is ok
```

4 确保 root 拥有 rules.ok 文件，并且权限被设置为 644。

另请参见 验证 rules 文件后，您可在第 4 章，使用可选的自定义 JumpStart 功能（任务）中了解有关可选自定义 JumpStart 功能的更多信息。可在第 6 章，执行自定义 JumpStart 安装（任务）中了解有关如何执行自定义 JumpStart 安装的信息。

使用可选的自定义 JumpStart 功能（任务）

本章介绍可用于创建附加的自定义 JumpStart 安装工具的可选功能。

注 - 如果您要安装 Solaris ZFS 根池，请参见第 9 章，使用 JumpStart 安装 ZFS 根池以了解限制和配置文件示例。

- 第 53 页中的“创建开始脚本”
- 第 55 页中的“创建结束脚本”
- 第 61 页中的“创建压缩的配置文件”
- 第 63 页中的“创建磁盘配置文件”
- 第 68 页中的“使用站点特定的安装程序”

注 - 本章中的说明对用于提供自定义 JumpStart 文件的 SPARC 服务器或 x86 服务器（称为配置文件服务器）都是有效的。配置文件服务器可以为不同类型的平台提供自定义 JumpStart 文件。例如，SPARC 服务器可以为基于 SPARC 的系统和基于 x86 的系统提供自定义 JumpStart 文件。

创建开始脚本

开始脚本是在 `rules` 文件中指定的用户定义的 Bourne shell 脚本。开始脚本可以在将 Solaris 软件安装到系统之前执行任务。只有在使用自定义 JumpStart 安装 Solaris 软件时才可以开始脚本。

使用开始脚本可以执行以下任务之一：

- 创建派生的配置文件
- 升级前备份文件

关于开始脚本的重要信息

- 不要在脚本中指定将会在初始或升级安装过程中阻止文件系统挂载到 /a 的内容。如果 JumpStart 程序无法将文件系统挂载到 /a 上，将发生错误且安装失败。
- 安装期间，开始脚本的输出存储在 /tmp/begin.log 中。安装结束后，该日志文件被重定向到 /var/sadm/system/logs/begin.log。
- 确保 root 拥有开始脚本，并且权限被设置为 644。
- 可以在开始脚本中使用自定义 JumpStart 环境变量。有关环境变量的列表，请参见第 142 页中的“自定义 JumpStart 环境变量”。
- 将开始脚本保存在 JumpStart 目录中。

注 - 对于 Solaris 10 发行版，在介质上提供了 JumpStart 样例脚本 `set_nfs4_domain`，以防止在 JumpStart 安装期间出现提示。此脚本可在安装期间抑制 NFSv4 提示。现在已不再需要此脚本，从 **Solaris 10 10/08 发行版** 开始，使用 `sysidcfg` 关键字 `nfs4_domain` 来抑制提示。`set_nfs4_domain` 脚本不再用于抑制提示。

如果安装了非全局区域，而且 `sysidcfg` 文件中存在新的 `nfs4_domain` 关键字，则首次引导非全局区域时会设置域。否则，在引导过程完成之前，将出现 Solaris 交互式安装程序，并提示您提供域名。

请参见《Solaris 10 10/08 安装指南：基于网络的安装》中的“`nfs4_domain` 关键字”

用开始脚本创建派生配置文件

派生配置文件是在自定义 JumpStart 安装期间由开始脚本动态创建的配置文件。当不能设置 `rules` 文件以使特定的系统与配置文件相匹配时，就需要派生配置文件。例如，可能需要将派生配置文件用于具有不同硬件组件的相同系统型号，如包含不同的帧缓存器的系统。

要设置使用派生配置文件的规则，必须执行以下任务：

- 将 `profile` 字段设置为等号 (=) 而不是配置文件。
- 将 `begin` 字段设为创建派生配置文件的开始脚本，该派生配置文件取决于打算安装 Solaris 的系统。

当系统匹配一个 `profile` 字段为等号 (=) 的规则时，开始脚本将创建用于在系统上安装 Solaris 软件的派生配置文件。

下面是一个开始脚本的示例，该脚本每次都创建相同的派生配置文件。您可以编写一个开始脚本，根据规则的计算创建不同的派生配置文件。

示例 4-1 创建派生配置文件的开始脚本

```
#!/bin/sh
echo "install_type      initial_install"    > ${SI_PROFILE}
echo "system_type      standalone"      >> ${SI_PROFILE}
echo "partitioning     default"         >> ${SI_PROFILE}
echo "cluster          SUNWCprog"       >> ${SI_PROFILE}
echo "package          SUNWman   delete" >> ${SI_PROFILE}
echo "package          SUNWolman delete" >> ${SI_PROFILE}
echo "package          SUNWxwman delete" >> ${SI_PROFILE}
```

在本示例中，开始脚本必须将 `SI_PROFILE` 环境变量用于派生配置文件的名称，缺省情况下将名称设置为 `/tmp/install.input`。

注 - 如果使用开始脚本创建派生配置文件，请确保该脚本没有任何错误。派生配置文件不由 `check` 脚本验证，这是因为派生配置文件直到开始脚本执行后才创建。

创建结束脚本

结束脚本是在 `rules` 文件中指定的用户定义的 Bourne shell 脚本。结束脚本可以在将 Solaris 软件安装到系统之后但在系统重新引导之前执行任务。只有在使用自定义 JumpStart 安装 Solaris 时才可以使⤵用结束脚本。

可以使用结束脚本执行的任务包括：

- 添加文件
- 添加个别软件包或修补程序，以补充安装在特定软件组中的软件包或修补程序
- 自定义根环境
- 设置系统的超级用户口令
- 安装附加软件

关于结束脚本的重要信息

- Solaris 安装程序在 `/a` 上挂载系统的文件系统。文件系统保持挂载在 `/a` 上，直到系统重新引导。通过修改与 `/a` 相对应的文件系统，可以使用结束脚本添加、更改或删除新安装的文件系统结构中的文件。
- 安装期间，结束脚本的输出存储在 `/tmp/finish.log` 中。安装结束后，该日志文件被重定向到 `/var/sadm/system/logs/finish.log`。
- 确保 `root` 拥有结束脚本，并且权限被设置为 `644`。
- 您可以在自己的结束脚本中使用自定义 JumpStart 环境变量。有关环境变量的列表，请参见第 142 页中的“自定义 JumpStart 环境变量”。

- 将结束脚本保存在 JumpStart 目录中。

▼ 使用结束脚本添加文件

通过结束脚本，可以将文件从 JumpStart 目录添加到已安装的系统。可以添加文件的原因在于，JumpStart 目录挂载在 `SI_CONFIG_DIR` 变量指定的目录中。缺省情况下，该目录设置为 `/tmp/install_config`。

注 - 也可以替换文件，方法是将文件从 JumpStart 目录复制到已安装系统中已存在的文件。

- 1 将所有要添加到已安装系统的文件复制到 JumpStart 目录中。
- 2 对于希望复制到新安装的文件系统结构中的每个文件，在结束脚本中插入以下行：

```
cp ${SI_CONFIG_DIR}/file_name /a/path_name
```

示例 4-2 使用结束脚本添加文件

例如，假设您有一个为站点上所有用户开发的特殊应用程序 `site_prog`。如果您将 `site_prog` 的副本放到 JumpStart 目录中，则结束脚本中的以下行会将 `site_prog` 从 JumpStart 目录复制到系统的 `/usr/bin` 目录中：

```
cp ${SI_CONFIG_DIR}/site_prog /a/usr/bin
```

使用结束脚本添加软件包或修补程序

Solaris 软件在一个系统上安装完成之后，您可以创建一个结束脚本来自动添加软件包或修补程序。通过使用结束脚本添加软件包，可减少时间并确保安装在站点上不同系统中的软件包和修补程序的一致性。

在结束脚本中使用 `pkgadd(1M)` 或 `patchadd(1M)` 命令时，请使用 `-R` 选项将 `/a` 指定为根路径。

- 示例 4-3 显示了添加软件包的结束脚本示例。
- 示例 4-4 显示了添加修补程序的结束脚本示例。

示例 4-3 使用结束脚本添加软件包

```
#!/bin/sh

BASE=/a
MNT=/a/mnt
```


示例 4-3 使用结束脚本添加软件包 (续)

```

ADMIN_FILE=/a/tmp/admin

mkdir ${MNT}
mount -f nfs sherlock:/export/package ${MNT}
cat >${ADMIN_FILE} <<DONT_ASK
mail=root
instance=overwrite
partial=nocheck
runlevel=nocheck
idepend=nocheck
rdepend=nocheck
space=ask
setuid=nocheck
conflict=nocheck
action=nocheck
basedir=default
DONT_ASK

/usr/sbin/pkgadd -a ${ADMIN_FILE} -d ${MNT} -R ${BASE} SUNwxyz
umount ${MNT}
rmdir ${MNT}

```

以下说明了用于此示例的一些命令。

- 以下命令在包含要安装的软件包的服务器上挂载目录。

```
mount -f nfs sherlock:/export/package ${MNT}
```

- 以下命令创建一个临时软件包管理文件 `admin`，强制使 `pkgadd(1M)` 命令在安装软件包时不执行检查或提示问题。添加软件包时，请使用临时软件包管理文件以便不用干预安装。

```
cat >${ADMIN_FILE} <<DONT_ASK
```

- 下面的 `pkgadd` 命令通过使用 `-a` 选项指定软件包管理文件，并使用 `-R` 选项指定根路径来添加软件包。

```
/usr/sbin/pkgadd -a ${ADMIN_FILE} -d ${MNT} -R ${BASE} SUNwxyz
```

示例 4-4 使用结束脚本添加修补程序

```

#!/bin/sh

#####
#
# USER-CONFIGURABLE OPTIONS
#

```

示例 4-4 使用结束脚本添加修补程序 (续)

```
#####

# The location of the patches to add to the system after it's installed.
# The OS rev (5.x) and the architecture ('mach') will be added to the
# root. For example, /foo on a 8 SPARC would turn into /foo/5.8/sparc
LUPATCHHOST=ins3525-svr
LUPATCHPATHROOT=/export/solaris/patchdb
#####
#
# NO USER-SERVICEABLE PARTS PAST THIS POINT
#
#####

BASEDIR=/a

# Figure out the source and target OS versions
echo Determining OS revisions...
SRCREV='uname -r'
echo Source $SRCREV

LUPATCHPATH=$LUPATCHPATHROOT/$SRCREV/'mach'

#
# Add the patches needed
#
echo Adding OS patches
mount $LUPATCHHOST:$LUPATCHPATH /mnt >/dev/null 2>&1
if [ $? = 0 ] ; then
    for patch in `cat /mnt/*Recommended/patch_order` ; do
        (cd /mnt/*Recommended/$patch ; echo yes | patchadd -u -d -R $BASEDIR .)
    done
    cd /tmp
    umount /mnt
else
    echo "No patches found"
if
```

注 – 以前，`chroot(1M)` 命令与结束脚本环境中的 `pkgadd` 和 `patchadd` 命令一起使用。在极少数情况下，一些软件包或修补程序无法与 `-R` 选项一起工作。在使用 `chroot` 命令前，必须在 `/a` 根路径中创建一个伪 `/etc/mnttab` 文件。

要创建伪 `/etc/mnttab` 文件，请将以下行添加到您的结束脚本中：

```
cp /etc/mnttab /a/etc/mnttab
```

使用结束脚本自定义根环境

您还可以使用结束脚本来自定义已经安装在系统上的文件。例如，[示例 4-5](#) 中的结束脚本通过将信息附加到根 (`/`) 目录中的 `.cshrc` 文件来自定义根环境。

示例 4-5 使用结束脚本自定义根环境

```
#!/bin/sh
#
# Customize root's environment
#
echo "***adding customizations in /.cshrc"
test -f a/.cshrc || {
cat >> a/.cshrc <<EOF
set history=100 savehist=200 filec ignoreeof prompt="\$user@'uname -n'> "
alias cp cp -i
alias mv mv -i
alias rm rm -i
alias ls ls -FC
alias h history
alias c clear
unset autologout
EOF
}
```

使用结束脚本设置系统的超级用户口令

Solaris 软件安装到系统上之后，系统会重新引导。在引导过程完成前，系统提示输入超级用户口令。除非有人键入口令，否则系统无法完成引导。

名为 `set_root_pw` 的结束脚本保存在 `auto_install_sample` 目录中。结束脚本显示如何自动设置超级用户口令，而无需提示。[示例 4-6](#) 中显示了 `set_root_pw`。

注 – 如果使用结束脚本设置系统的超级用户口令，则用户可以尝试从结束脚本的加密口令中搜索超级用户口令。确保有相应的措施防止用户试图确定超级用户口令。

示例 4-6 使用结束脚本设置系统的超级用户口令

```
#!/bin/sh
#
#      @(#)set_root_pw 1.4 93/12/23 SMI
#
# This is an example Bourne shell script to be run after installation.
# It sets the system's root password to the entry defined in PASSWD.
# The encrypted password is obtained from an existing root password entry
# in /etc/shadow from an installed machine.

echo "setting password for root"

# set the root password
PASSWD=dK05IBkSF42lw
#create a temporary input file
cp /a/etc/shadow /a/etc/shadow.orig

mv /a/etc/shadow /a/etc/shadow.orig
nawk -F: '{
    if ( $1 == "root" )
        printf"%s:%s:%s:%s:%s:%s:%s:%s:%s\n",$1,passwd,$3,$4,$5,$6,$7,$8,$9
    else
        printf"%s:%s:%s:%s:%s:%s:%s:%s:%s\n",$1,$2,$3,$4,$5,$6,$7,$8,$9
    }' passwd="$PASSWD" /a/etc/shadow.orig > /a/etc/shadow
#remove the temporary file
rm -f /a/etc/shadow.orig
# set the flag so sysidroot won't prompt for the root password
sed -e 's/0 # root/1 # root/' ${SI_SYS_STATE} > /tmp/state.$$
mv /tmp/state.$$ ${SI_SYS_STATE}
```

以下说明此例中的一些命令。

- 以下命令将变量 PASSWD 设置为从系统 /etc/shadow 文件中的现有项获取的加密超级用户口令。

```
#create a temporary input file
```

- 以下命令创建一个临时输入文件 /a/etc/shadow。

```
cp /a/etc/shadow /a/etc/shadow.orig
```

- 以下命令在 /etc/shadow 文件中更改新安装的系统的超级用户项，使用 \$PASSWD 作为口令字段。

示例 4-6 使用结束脚本设置系统的超级用户口令 (续)

```
if ( $1 == "root" )
```

- 以下命令删除临时文件 `/a/etc/shadow`。

```
rm -f /a/etc/shadow.orig
```

- 以下命令在状态文件中将项从 0 更改为 1，这样将不提示用户输入超级用户口令。状态文件通过变量 `SI_SYS_STATE` 访问，该变量的当前值为 `/a/etc/.sysIDtool.state`。为了避免在此值更改时脚本出现问题，请使用 `$SI_SYS_STATE` 以便始终引用此文件。此处显示的 `sed` 命令在 0 之后和 1 之后包含一个制表符。

```
sed -e 's/0 # root/1 # root/' ${SI_SYS_STATE} > /tmp/state.$$
```

使用结束脚本的非交互安装

安装了 Solaris OS 后，可使用结束脚本来安装附加软件。Solaris 安装程序在安装过程中提示您输入信息。要使安装不受干扰，可以使用 `-nodisplay` 或 `-noconsole` 选项来运行 Solaris 安装程序。

表 4-1 Solaris 安装选项

选项	说明
<code>-nodisplay</code>	运行安装程序，但没有图形用户界面。使用缺省产品安装，除非安装已被 <code>-locales</code> 选项修改。
<code>-noconsole</code>	运行安装，但没有任何交互式文本控制台设备。当与 <code>-nodisplay</code> 一起用于 UNIX 脚本时很有用。

有关更多信息，请参见手册页 [installer\(1M\)](#)。

创建压缩的配置文件

可以在引导系统时指定自定义 JumpStart 配置文件的位置，而不是使用 `add_install_client` 命令来指定其位置。但是，只能指定一个文件的名称。因此，必须将所有的自定义 JumpStart 配置文件压缩为一个文件。

- 对基于 SPARC 的系统，可以在 `boot` 命令中指定文件的位置
- 对基于 x86 的系统，可以通过编辑 GRUB 菜单中的 GRUB 项来指定文件的位置

压缩的配置文件可以是以下类型之一：

- `tar`

- 压缩的 tar
- zip
- bzip tar

▼ 创建压缩的配置文件

- 1 将目录更改到配置文件服务器上的 JumpStart 目录。

```
# cd jumpstart_dir_path
```

- 2 使用压缩工具将自定义 JumpStart 配置文件压缩为一个文件。

注 - 压缩的配置文件不能包含相对路径。自定义 JumpStart 配置文件必须与压缩文件位于同一目录中。

压缩的配置文件必须包含以下文件：

- 配置文件
- rules
- rules.ok

也可以将 sysidcfg 文件包含在压缩的配置文件中。

- 3 将压缩的配置文件保存在 NFS 服务器、HTTP 服务器或本地硬盘中。

压缩的配置文件示例

下面的示例说明如何使用 tar 命令创建一个名为 config.tar 的压缩配置文件。自定义 JumpStart 配置文件位于 /jumpstart 目录中。

示例 4-7 创建压缩的配置文件

```
# cd /jumpstart
# tar -cvf config.tar *
a profile 1K
a rules 1K
a rules.ok 1K
a sysidcfg 1K
```

创建磁盘配置文件

本节介绍如何创建单磁盘配置文件和多磁盘配置文件。利用磁盘配置文件，您可以在单个系统中使用 `pfinstall(1M)` 针对不同磁盘配置测试配置文件。

▼ SPARC: 创建磁盘配置文件

- 1 找到包含要测试的磁盘的基于 SPARC 的系统。

- 2 成为超级用户或承担等效角色。

角色包含授权和具有一定权限的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 3 通过将 `prtvtoc(1M)` 命令的输出重定向到文件，创建单磁盘配置文件。

```
# prtvtoc /dev/rdisk/device_name >disk_config_file
/dev/rdisk/device_name    系统磁盘的设备名称。device_name 的格式必须是 cwt xdys2
                           或 cxdy s2。
disk_config_file          磁盘配置文件的名称。
```

- 4 确定是否在多个磁盘上测试 Solaris 软件的安装。

- 如果否，则停止。操作完成。
- 如果是，串联单磁盘配置文件并将输出保存到一个新文件中。

```
# cat disk_file1 disk_file2 >multi_disk_config
```

此新文件即成为多磁盘配置文件，如下例所示：

```
# cat 104_disk2 104_disk3 104_disk5 >multi_disk_test
```

- 5 确定磁盘设备名称中的目标号码在前一个步骤中创建的多磁盘配置文件中是否唯一。

- 如果是，则停止。操作完成。
- 如果否，使用文本编辑器打开此文件并使目标号码在磁盘设备名称中是唯一的。例如，假设对于不同的磁盘设备名称，此文件包含同一目标号码 `t0`，如下所示：

```
* /dev/rdisk/c0t0d0s2 partition map
...
* /dev/rdisk/c0t0d0s2 partition map
```

将第二个目标号码更改为 `t2`，如下所示：

```
* /dev/rdisk/c0t0d0s2 partition map
...
* /dev/rdisk/c0t2d0s2 partition map
```

SPARC: 磁盘配置文件示例

下面的示例显示如何在具有 104 MB 磁盘的基于 SPARC 的系统中创建单磁盘配置文件 `104_test`。

示例 4-8 SPARC: 创建磁盘配置文件

将 `prtvtoc` 命令的输出重定向到名为 `104_test` 的单磁盘配置文件：

```
# prtvtoc /dev/rdisk/c0t3d0s2 >104_test
```

`104_test` 文件的内容与下面的内容相似：

```
* /dev/rdisk/c0t3d0s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   72 sectors/track
*   14 tracks/cylinder
*  1008 sectors/cylinder
*  2038 cylinders*   2036 accessible cylinders
* Flags:
*  1: unmountable
* 10: read-only
*
*
* Partition  Tag  Flags   First   Sector   Last      Mount Directory
*           1    2    00       0    164304   164303   /
*           2    5    00       0   2052288  2052287
*           3    0    00    164304   823536   987839   /disk2/b298
*           5    0    00    987840   614880   1602719  /install/298/sparc/work
*           7    0    00   1602720   449568   2052287  /space
```

您已为基于 SPARC 的系统创建了磁盘配置文件。第 47 页中的“测试配置文件”包含有关使用磁盘配置文件测试配置文件的的信息。

▼ x86: 创建磁盘配置文件

1 找到包含要测试的磁盘的基于 x86 的系统。

2 成为超级用户或承担等效角色。

角色包含授权和具有一定权限的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

3 通过将 `fdisk(1M)` 命令的输出保存在文件中，创建单磁盘配置文件的一部分。

```
# fdisk -R -W disk_config_file -h /dev/rdisk/device_name
```

`disk_config_file` 磁盘配置文件的名称。

`/dev/rdisk/device_name` 整个磁盘的 `fdisk` 布局的设备名称。`device_name` 的格式必须是 `cwtx dys0` 或 `cxdys0`。

4 将 `prtvtoc(1M)` 命令的输出附加到磁盘配置文件：

```
# prtvtoc /dev/rdisk/device_name >>disk_config
```

`/dev/rdisk/device_name` 系统磁盘的设备名称。`device_name` 的格式必须是 `cwt xdy2` 或 `cxdy s2`。

`disk_config` 磁盘配置文件的名称。

5 确定是否在多个磁盘上测试 Solaris 软件的安装。

- 如果否，则停止。操作完成。
- 如果是，串联单磁盘配置文件并将输出保存到一个新文件中。

```
# cat disk_file1 disk_file2 >multi_disk_config
```

此新文件即成为多磁盘配置文件，如下例所示：

```
# cat 104_disk2 104_disk3 104_disk5 >multi_disk_test
```

6 确定磁盘设备名称中的目标号码在前一个步骤中创建的多磁盘配置文件中是否唯一。

- 如果是，则停止。操作完成。
- 如果否，用文本编辑器打开该文件并使目标号码唯一。

例如，对于不同的磁盘设备名称，此文件可能包含同一目标号码 `t0`，如下所示：

```
* /dev/rdisk/c0t0d0s2 partition map
...
* /dev/rdisk/c0t0d0s2 partition map
```

将第二个目标号码更改为 t2，如下所示：

```
* /dev/rdisk/c0t0d0s2 partition map
...
* /dev/rdisk/c0t2d0s2 partition map
```

x86: 磁盘配置文件示例

下面的示例说明如何在具有 500 MB 磁盘的基于 x86 的系统中创建单磁盘配置文件 500_test。

示例 4-9 x86: 创建磁盘配置文件

首先，将 fdisk 命令的输出保存到名为 500_test 的文件中：

```
# fdisk -R -W 500_test -h /dev/rdisk/c0t0d0p0
```

500_test 文件如下所示：

```
* /dev/rdisk/c0t0d0p0 default fdisk table
* Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
*   1455 cylinders
*
* HBA Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
*   1455 cylinders
*
* systid:
* 1:  DOSOS12
* 2:  PCIXOS
* 4:  DOSOS16
* 5:  EXTDOS
* 6:  DOSBIG
* 86: DOSDATA
* 98: OTHEROS
* 99: UNIXOS
* 130: SUNIXOS
*
* Id  Act  Bhead  Bsect   Bcyl  Ehead  Esect   Ectl  Rsect  Numsect
130  128  44     3       0     46     30     1001  1410   2050140
```

示例 4-9 x86: 创建磁盘配置文件 (续)

第二步, 将 prtvtoc 命令的输出附加到 500_test 文件:

```
# prtvtoc /dev/rdisk/c0t0d0s2 >>500_test
```

现在, 500_test 文件是一个完整的磁盘配置文件:

```
* /dev/rdisk/c0t0d0p0 default fdisk table
* Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
*   1455 cylinders
*
* HBA Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
*   1455 cylinders
*
* systid:
* 1:  DOSOS12
* 2:  PCIXOS
* 4:  DOSOS16
* 5:  EXTDOS
* 6:  DOSBIG
* 86: DOSDATA
* 98: OTHEROS
* 99: UNIXOS
* 130: SUNIXOS
*
* Id Act Bhead Bsect Bcyl Ehead Esec Ectl Rsect Numsect
130 128 44 3 0 46 30 1001 1410 2050140
* /dev/rdisk/c0t0d0s2 partition map
*
* Dimensions:
*   512 bytes/sector
*   94 sectors/track
*   15 tracks/cylinder
*   1110 sectors/cylinder
*   1454 cylinders
*   1452 accessible cylinders
*
* Flags:
* 1: unmountable
* 10: read-only
```

示例 4-9 x86: 创建磁盘配置文件 (续)

```
*
* Partition Tag  Flags      First Sector   Last
* Partition Tag  Flags      Sector   Count   Sector  Mount Directory
      2      5    01      1410   2045910 2047319
      7      6    00      4230   2043090 2047319 /space
      8      1    01         0     1410    1409
      9      9    01      1410     2820   422987
```

您已为基于 x86 的系统创建了磁盘配置文件。第 47 页中的“测试配置文件”包含有关使用磁盘配置文件测试配置文件的信息。

使用站点特定的安装程序

您也可以使用开始脚本和结束脚本创建自己的安装程序来安装 Solaris 软件。

在 `profile` 字段中指定减号 (-) 时，由开始脚本和结束脚本控制如何在系统上安装 Solaris 软件，而不是由配置文件和 Solaris 安装程序控制。

例如，如果下面的规则匹配系统，则 `x_install.beg` 开始脚本和 `x_install.fin` 结束脚本将 Solaris 软件安装在名为 `clover` 的系统上：

```
hostname clover x_install.beg - x_install.fin
```

创建自定义规则和探测关键字（任务）

本章提供创建自定义规则和探测关键字的信息和步骤。

注 – 如果您要安装 Solaris ZFS 根池，请参见第 9 章，使用 [JumpStart 安装 ZFS 根池](#) 以了解限制和配置文件示例。

- 第 69 页中的 “探测关键字”
- 第 70 页中的 “创建 `custom_probes` 文件”
- 第 72 页中的 “验证 `custom_probes` 文件”

探测关键字

要理解什么是探测关键字，首先需要回顾一下什么是规则关键字。规则关键字是一个预定义的词法单元或单词，它说明一般的系统属性，如主机名 `hostname` 或内存大小 `memsize`。规则关键字及其关联值使您能够将一个具有相同属性的系统与一个配置文件匹配。这种系统属性的匹配定义了如何在组中的每个系统上安装 Solaris 软件。

自定义 `JumpStart` 环境变量用在开始脚本和结束脚本中，它们是根据需要进行设置的。例如，只有在使用了 `installed` 规则关键字之后，有关系统上安装了哪个操作系统的信息才会在 `SI_INSTALLED` 中提供。

在某些情况下，出于匹配系统和运行配置文件以外的其他目的，也可能需要在开始脚本或结束脚本中提取相同的信息。探测关键字提供了解决方法。探测关键字提取属性信息并消除了设置匹配条件和运行配置文件的需要。

有关探测关键字和值的列表，请参见第 144 页中的 “探测关键字和值”。

创建 custom_probes 文件

第 99 页中的“规则关键字和值”和第 144 页中的“探测关键字和值”描述的规则关键字和探测关键字可能无法完全满足您的需求。可以通过创建 custom_probes 文件，自定义规则关键字或探测关键字。

custom_probes 文件是一个 Bourne shell 脚本，它包含两种函数。您必须将 custom_probes 文件保存在保存了 rules 文件的同一 JumpStart 目录中。您可以在 custom_probes 文件中定义两种函数如下：

- 探测函数—收集您需要的信息或进行实际的工作，并设置您定义的相应 SI_ 环境变量。探测函数成为探测关键字。
- 比较函数—调用相应的探测函数，比较探测函数的输出，如果关键字匹配则返回 0，如果关键字不匹配则返回 1。比较函数成为规则关键字。

custom_probes 文件的语法

custom_probes 文件可以包含任何有效的 Bourne shell 命令、变量或算法。

注—您可以在 custom_probes 文件中定义需要单个变量的探测函数和比较函数。当您在 rules 文件中使用相应的自定义探测关键字时，关键字后面的变量将被解释（为 \$1）。

在 rules 文件中使用相应的自定义规则关键字时，将依次解释这些变量。该序列在关键字之后开始，并在下一个 && 或开始脚本（先出现的项）之前结束。

custom_probes 文件必须满足以下要求：

- 具有名称 custom_probes
- 将 root 作为其属主
- 是可执行文件并将权限设置为 755
- 包含至少一个探测函数和一个相应的比较函数

为提高明确性和改善结构，请首先在文件的顶部定义所有的探测函数，后跟所有的比较函数。

custom_probes 中的函数名称的语法

探测函数的名称必须以 probe_ 开头。比较函数的名称必须以 cmp_ 开头。

以 probe_ 开头的函数定义新的探测关键字。例如，函数 probe_tcx 定义新的探测关键字 tcx。以 cmp_ 开头的函数定义新的规则关键字。例如，cmp_tcx 定义新的规则关键字 tcx。

▼ 创建 custom_probes 文件

- 1 使用文本编辑器创建 **Bourne shell** 脚本文本文件。将该文件命名为 `custom_probes`。
- 2 在 `custom_probes` 文本文件中，定义您的探测函数和比较函数。

注 - 您可以在 `custom_probes` 文件中定义需要变量的探测函数和比较函数。当您在 `rules` 文件中使用相应的自定义探测关键字时，关键字后面的变量将依次被解释（为 `$1`、`$2`，依此类推）。

在 `rules` 文件中使用相应的自定义规则关键字时，将依次解释这些变量。该序列在关键字之后开始，并在下一个 `&&` 或开始脚本（先出现的项）之前结束。

- 3 将 `custom_probes` 文件保存在 **JumpStart** 目录中的 `rules` 文件旁边。
- 4 确保 `root` 拥有 `rules` 文件，并且权限设置为 `644`。

custom_probes 文件和关键字的示例

在以下目录中可以找到探测函数和比较函数的其他示例：

- `/usr/sbin/install.d/chkprobe`（在安装了 Solaris 软件的系统上）
- Solaris Operating System DVD 或 Solaris Software - 1 CD 中的 `/Solaris_10/Tools/Boot/usr/sbin/install.d/chkprobe`

下面的 `custom_probes` 文件包含测试 TCX 图形卡是否存在的探测函数和比较函数。

示例 5-1 custom_probes 文件

```
#!/bin/sh
#
# custom_probe script to test for the presence of a TCX graphics card.
#
#
# PROBE FUNCTIONS
#
probe_tcx() {
    SI_TCX='modinfo | grep tcx | nawk '{print $6}''
    export SI_TCX
}
#
# COMPARISON FUNCTIONS
```

示例 5-1 custom_probes 文件 (续)

```
#
cmp_tcx() {
    probe_tcx

    if [ "X${SI_TCX}" = "X${1}" ]; then
        return 0
    else
        return 1
    fi
}
```

下面的示例 `rules` 文件显示如何使用前面示例中定义的探测关键字 `tcx`。如果 TCX 图形卡已安装并在系统中被找到，则运行 `profile_tcx`。否则，运行 `profile`。

注 - 总是将探测关键字放置在 `rules` 文件的开头或开头附近。这样，可确保在其他可能依赖于这些探测关键字的规则关键字之前读取和运行这些关键字。

示例 5-2 自定义在 `rules` 文件中使用的探测关键字

```
probe tcx
tcx    tcx    -    profile_tcx    -
any    any    -    profile        -
```

验证 custom_probes 文件

在使用配置文件、`rules` 和 `custom_probes` 文件之前，必须运行 `check` 脚本来验证是否已正确设置这些文件。如果所有的配置文件、规则以及探测函数和比较函数均已正确设置，则创建 `rules.ok` 和 `custom_probes.ok` 文件。[表 5-1](#) 介绍了 `check` 脚本的作用。

表 5-1 使用 `check` 脚本时发生的情况

阶段	说明
1	check 搜索 <code>custom_probes</code> 文件。
2	如果文件存在， <code>check</code> 将基于 <code>custom_probes</code> 文件创建 <code>custom_probes.ok</code> 文件，删除所有注释和空白行，并保留所有 Bourne shell 命令、变量和算法。然后， <code>check</code> 在结尾添加以下注释行： # version=2 checksum=num

▼ 验证 custom_probes 文件

- 1 验证 check 脚本是否位于 JumpStart 目录中。

注 - check 脚本位于 Solaris Operating System DVD 或 Solaris Software - 1 CD 上的 Solaris_10/Misc/jumpstart_sample 目录中。

- 2 转到 JumpStart 目录。
- 3 运行 check 脚本可验证 rules 和 custom_probes 文件。

```
$ ./check -p path -r file_name
```

-p path 使用来自平台的 Solaris 软件映像的 check 脚本代替您正在使用的系统中的 check 脚本来验证 custom_probes 文件。path 是本地磁盘或已挂载的 Solaris Operating System DVD 或 Solaris Software - 1 CD 上的映像。

如果您的系统运行的是 Solaris 的早期版本，则使用此选项可运行最新版本的 check。

-r file_name 指定一个文件名（名称 custom_probes 除外）。通过使用 -r 选项，可以在将一组函数集成到 custom_probes 文件中之前验证这些函数的有效性。

check 脚本运行时，它会报告 rules 文件和 custom_probes 文件以及每个配置文件的有效性。如果没有遇到任何错误，该脚本将报告：“自定义 JumpStart 配置正常”并在 JumpStart 目录中创建 rules.ok 和 custom_probes.ok 文件。

- 4 确定 custom_probes.ok 文件是否为可执行文件。
 - 如果是，请转至步骤 5。
 - 如果不是，请键入以下命令：

```
# chmod +x custom_probes
```

- 5 确保 root 拥有 custom_probes.ok 文件，并且权限设置为 755。

执行自定义 JumpStart 安装（任务）

本章介绍如何在基于 SPARC 或基于 x86 的系统中执行自定义 JumpStart 安装。在要安装 Solaris 软件的系统中您需要遵循这些过程。

注 - 如果您要安装 Solaris ZFS 根池，请参见第 9 章，使用 JumpStart 安装 ZFS 根池以了解限制和配置文件示例。

- 第 80 页中的“SPARC: 使用自定义 JumpStart 程序执行安装或升级”
- 第 83 页中的“x86: 使用自定义 JumpStart 程序和 GRUB 执行安装或升级”

JumpStart 安装的限制

许多问题可能会导致 JumpStart 安装期间出错。有关特定信息，请查看下表。

表 6-1 JumpStart 安装限制

问题	说明	更多信息
已不再需要使用 JumpStart 样例脚本来抑制 NFSv4 提示	<p>对于 Solaris 10 发行版，在介质上提供了 JumpStart 样例脚本 <code>set_nfs4_domain</code>，以防止在 JumpStart 安装期间出现提示。此脚本可在安装期间抑制 NFSv4 提示。现在已不再需要此脚本，从 Solaris 10 8/07 发行版开始，使用 <code>sysidcfg</code> 关键字 <code>nfs4_domain</code> 来抑制提示。<code>set_nfs4_domain</code> 脚本不再用于抑制提示。</p> <p>如果安装了非全局区域，而且 <code>sysidcfg</code> 文件中存在新的 <code>nfs4_domain</code> 关键字，则首次引导非全局区域时会设置域。否则，在引导过程完成之前，将显示 Solaris 交互式安装程序，并提示您提供域名。</p>	《Solaris 10 10/08 安装指南：基于网络的安装》中的“ <code>nfs4_domain</code> 关键字”

表 6-1 JumpStart 安装限制 (续)

问题	说明	更多信息
在 <code>sysidcfg</code> 文件中选择键盘语言可防止提示	如果您的键盘不是自识别键盘，而且您希望防止在 JumpStart 安装期间出现提示，请在 <code>sysidcfg</code> 文件中选择键盘语言。对于 JumpStart 安装，缺省设置为美国英语。要选择其他语言及其相应的键盘布局，请在 <code>sysidcfg</code> 文件中设置键盘关键字。	<ul style="list-style-type: none"> ■ 《Solaris 10 10/08 安装指南：基于网络的安装》中的“<code>sysidcfg</code> 文件关键字” ■ 有关手册页，请参见： <ul style="list-style-type: none"> ■ <code>sysidtool(1M)</code> ■ <code>sysidcfg(4)</code>
如果具有非全局区域，请使用 Solaris Live Upgrade 进行升级	可以使用 JumpStart 升级安装有非全局区域的系统，但是建议使用 Solaris Live Upgrade 程序进行升级。JumpStart 可能需要很长的升级时间，因为完成升级所需的时间会随安装的非全局区域数线性增长。	《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》
Solaris Flash 归档文件中不能包含非全局区域	如果使用 Solaris Flash 归档文件进行安装，则不会在系统上正确安装包含非全局区域的归档文件。	有关创建非全局区域的一般信息，请参见《系统管理指南：Solaris Containers—资源管理和 Solaris Zones》。
只能从 UFS 文件系统创建和安装 Solaris Flash 归档文件。	无法从 ZFS 根池创建 Solaris Flash 归档文件，也无法在 ZFS 根池中安装 Solaris Flash 归档文件。	
SPARC：其他硬件要求	有关在平台上完成 JumpStart 安装可能需要的任何其他要求，请参阅硬件文档。	

SPARC: 任务图：针对自定义 JumpStart 安装设置系统

表 6-2 任务图：针对自定义 JumpStart 安装设置系统

任务	说明	参考
检查系统是否受支持。	检查在 Solaris 环境中系统支持的硬件文档。	http://docs.sun.com 上的 Solaris Sun 硬件平台指南
检查系统是否有足够的磁盘空间用于 Solaris 软件。	验证是否已规划了足够的空间在系统上安装 Solaris 软件。	《Solaris 10 10/08 安装指南：规划安装和升级》中的第 4 章“系统要求、原则和升级（规划）”
（可选）设置系统参数。	您可以预配置系统信息，以避免在安装或升级时被提示输入该信息。	《Solaris 10 10/08 安装指南：基于网络的安装》中的第 2 章“预配置系统配置信息（任务）”
针对自定义 JumpStart 安装准备系统。	创建并验证 <code>rules</code> 文件和配置文件。	第 3 章，准备自定义 JumpStart 安装（任务）

表 6-2 任务图：针对自定义 JumpStart 安装设置系统 (续)

任务	说明	参考
(可选) 准备可选的自定义 JumpStart 功能。	如果使用了开始脚本、结束脚本或其他可选功能，请准备这些脚本或文件。	第 4 章，使用可选的自定义 JumpStart 功能 (任务) 和第 5 章，创建自定义规则和探测关键字 (任务)
(可选) 准备从网络安装 Solaris 软件。	要从远程 Solaris Operating System DVD 或 Solaris Software for SPARC Platforms CD 映像安装系统，您需要对系统进行设置，以从安装服务器或引导服务器引导和安装。	《Solaris 10 10/08 安装指南：基于网络的安装》中的第 5 章“使用 DVD 介质从网络进行安装 (任务)” 《Solaris 10 10/08 安装指南：基于网络的安装》中的第 6 章“使用 CD 介质从网络进行安装 (任务)”
(可选) 准备 Solaris Flash 归档文件安装。	设置 Solaris Flash 归档文件安装的具体信息。	第 77 页中的“准备使用自定义 JumpStart 安装来安装 Solaris Flash 归档文件”
执行安装或升级。	引导系统以便初始化安装或升级。	第 80 页中的“SPARC: 使用自定义 JumpStart 程序执行安装或升级”

SPARC: 执行自定义 JumpStart 安装

在自定义 JumpStart 安装过程中，JumpStart 程序尝试将要安装的系统与 `rules.ok` 文件中的规则进行匹配。JumpStart 程序从第一条规则到最后一规则通读这些规则。当安装的系统与规则中定义的所有系统属性一致时发生匹配。如果系统与规则匹配，JumpStart 程序即停止读取 `rules.ok` 文件，然后根据该匹配规则的配置文件开始安装系统。

▼ 准备使用自定义 JumpStart 安装来安装 Solaris Flash 归档文件

对于初始安装，可安装完全归档文件；如果您已安装了归档文件，则可安装差别归档文件以进行更新。使用自定义 JumpStart 安装方法或使用 Solaris Live Upgrade，可以在不活动的引导环境中安装归档文件。本步骤提供使用自定义 JumpStart 来安装归档文件的说明。

- 有关完全归档文件或差别归档文件概述，请参见《Solaris 10 10/08 安装指南：Solaris Flash 归档文件 (创建和安装)》中的第 1 章“Solaris Flash (概述)”。

- 有关使用 Solaris Live Upgrade 在非活动的引导环境中安装归档文件的过程，请参见《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的“使用配置文件安装 Solaris Flash 归档文件”。

1 检查以下限制。

说明	示例
<p>注意：使用 <code>archive_location</code> 关键字安装 Solaris Flash 归档文件时，归档文件和安装介质必须包含相同的操作系统版本。</p> <p>注意-当安装了非全局区域时，将无法正确创建 Solaris Flash 归档文件。Solaris Flash 功能与 Solaris Zone 分区技术不兼容。如果创建了 Solaris Flash 归档文件，当在下列条件下部署归档文件时，所生成的归档文件无法正确安装：</p> <ul style="list-style-type: none"> 归档文件是在非全局区域中创建的 归档文件是在安装了非全局区域的全局区域中创建的 	<p>例如，如果归档文件是 Solaris 10 10/08 操作系统且您使用的是 DVD 介质，则必须使用 Solaris 10 10/08 DVD 介质来安装归档文件。如果操作系统的版本不匹配，将无法在克隆系统中安装归档文件。</p>

2 在安装服务器上，创建自定义 JumpStart rules 文件。

有关创建自定义 JumpStart 文件的详细说明，请参阅第 3 章，准备自定义 JumpStart 安装（任务）。

3 在安装服务器上，创建自定义 JumpStart 配置文件。

有关 Solaris Flash 归档文件配置文件的示例，请参见第 37 页中的“配置文件示例”。

基于表 8-2 中自定义 JumpStart 关键字的现有列表，只有以下关键字在安装 Solaris Flash 归档文件时有效：

关键字	初始安装	差别归档文件
(必需) <code>archive_location</code>	X	X
<code>fdisk</code> (仅适用于 x86)	X	X
<code>filesys</code>	X	
注- 不能将关键字 <code>filesys</code> 的值设置为 <code>auto</code> 。		
<code>forced_deployment</code>		X
(必需) <code>install_type</code>	X	X

关键字	初始安装	差别归档文件
local_customization	X	X
no_content_check		X
no_master_check		X
package (软件包)	X	
root_device	X	X

- a. 将 `install_type` 关键字的值设置为下列类型之一。
 - 对于完全归档文件安装，将该值设置为 `flash_install`。
 - 对于差别归档文件安装，将该值设置为 `flash_update`。
 - b. 使用 `archive_location` 关键字来添加 Solaris Flash 归档文件的路径。
有关 `archive_location` 关键字的详细信息，请参阅第 105 页中的“[archive_location 关键字](#)”。
 - c. 指定文件系统配置。
Solaris Flash 归档文件提取过程不支持分区的自动布局。
 - d. (可选) 如果希望在安装归档文件的同时安装附加软件包，请使用 `package` 关键字。有关更多信息，请参见第 130 页中的“[package 配置文件关键字 \(UFS 和 ZFS \)](#)”。
 - e. (可选) 如果要在克隆系统上安装一个附加的 Solaris Flash 归档文件，请为要安装的每个归档文件添加一个 `archive_location` 行。
- 4 在安装服务器上，添加要与 Solaris Flash 归档文件一起安装的客户程序。
有关详细说明，请参阅以下章节：
 - 《Solaris 10 10/08 安装指南：基于网络的安装》中的“使用 DVD 映像添加要从网络安装的系统”
 - 《Solaris 10 10/08 安装指南：基于网络的安装》中的“使用 CD 映像添加要从网络安装的系统”
 - 5 在克隆系统上执行自定义 JumpStart 安装。
有关详细说明，请参阅第 80 页中的“[SPARC: 使用自定义 JumpStart 程序执行安装或升级](#)”。

▼ SPARC: 使用自定义 JumpStart 程序执行安装或升级

- 1 如果系统是网络的一部分，请确保已将以太网连接器或类似的网络适配器连接到了系统上。
- 2 如果要安装通过 `tip(1)` 行连接的系统，请确保所显示的窗口宽度和长度至少分别为 80 列和 24 行。
要确定 `tip` 窗口的当前尺寸，请使用 `stty(1)` 命令。
- 3 如果使用系统的 DVD-ROM 或 CD-ROM 驱动器安装 Solaris 软件，请将 Solaris Operating System for SPARC Platforms DVD 或 Solaris Software for SPARC Platforms - 1 CD 插入驱动器。
- 4 如果使用了配置文件软盘，请将此配置文件软盘放入系统的软盘驱动器。
- 5 引导系统。
 - 如果系统是新的，刚从包装箱中拿出来，请打开系统。
 - 如果要安装或升级现有系统，则关闭该系统。在 `ok` 提示符下，键入 `boot` 命令的适当选项。`boot` 命令的语法如下。

```
ok boot [cd-dvd|net] - install [url|ask] options
```

例如，如果键入以下命令，将使用 JumpStart 配置文件通过网络安装 OS。

```
ok boot net - install http://131.141.2.32/jumpstart/config.tar
```

有关 `boot` 命令选项的描述，请参见下表。

仅适用于 SPARC – 系统将检查硬件、系统组件以及基于 SPARC 的系统引导。引导过程将持续几分钟。

- 6 如果没有预配置 `sysidcfg` 文件中的系统信息，则在收到系统提示时，请回答有关系统配置的问题。
- 7 按照屏幕上的说明来安装软件。
当 JumpStart 程序完成 Solaris 软件的安装后，系统将自动重新引导。
完成安装后，安装日志将保存在文件中。您可以在以下目录中找到安装日志：

- `/var/sadm/system/logs`
- `/var/sadm/install/logs`

SPARC: boot 命令的命令参考

boot 命令的语法如下。

```
ok boot [cd-dvd|net] - install [url|ask] options
```

下表描述了适用于 JumpStart 安装的 boot 命令的命令行选项。

选项	说明
<code>[<i>cd-dvd net</i>]</code>	<p>指定从 CD 或 DVD 引导，或者从网络上的安装服务器引导。</p> <ul style="list-style-type: none"> ■ <code>cd-dvd</code>—使用 <code>cdrom</code> 从 CD 或 DVD 引导。 ■ <code>net</code>—指定从网络上的安装服务器引导。
<code>[<i>url ask</i>]</code>	<p>指定自定义 JumpStart 文件的位置或提示您指定位置。</p> <ul style="list-style-type: none"> ■ <code>url</code>—指定文件的路径。可以为位于 HTTP 或 HTTPS 服务器中的文件指定 URL。 HTTP 服务器 <code>http://server_name:IP_address/jumpstart_dir_path/compressed_config_file&proxy_info</code> <ul style="list-style-type: none"> ■ 如果将一个 <code>sysidcfg</code> 文件放置在压缩配置文件中，必须指定包含该文件的服务器的 IP 地址，如下例所示： <code>http://131.141.2.32/jumpstart/config.tar</code> ■ 如果将压缩配置文件保存在位于防火墙后面的 HTTP 服务器上，则在引导过程中必须使用代理说明符。不需要为包含该文件的服务器指定 IP 地址。必须指定代理服务器的 IP 地址，如下例所示： <code>http://www.shadow.com/jumpstart/config.tar&proxy=131.141.6.151</code> ■ <code>ask</code>—指定在系统引导并连接到网络后，安装程序提示您键入压缩的配置文件的位置。如果使用该选项，您将无法执行完全无人参与的 JumpStart 安装。如果通过按 <code>Return</code> 来跳过提示，Solaris 安装程序将交互式地配置网络参数。然后，安装程序提示您键入压缩配置文件的位置。
<code>options</code>	<ul style="list-style-type: none"> ■ <code>dhcp</code>—指定使用 DHCP 服务器获取引导系统所需的网络安装信息。该选项对于 JumpStart 安装不是必需的。如果未通过键入 <code>dhcp</code> 来指定使用 DHCP 服务器，则系统将使用 <code>/etc/bootparams</code> 文件或命名服务 <code>bootparams</code> 数据库。例如，如果您希望保留静态 IP 地址，您将不会指定 <code>dhcp</code>。 ■ 选项 <code>nowin</code> 和 <code>text</code> 不适用于 JumpStart 安装。这些选项对于交互式安装很有帮助。有关更多信息，请参见《Solaris 10 10/08 安装指南：基本安装》中的“使用 Solaris 安装程序安装或升级”。

x86: 任务图：针对自定义 JumpStart 安装设置系统

表 6-3 x86: 任务图：针对自定义 JumpStart 安装设置系统

任务	说明	参考
确定是否需要保留现有的操作系统和用户数据。	如果系统中的现有操作系统使用整个磁盘，则您必须保留现有操作系统以使其可以与 Solaris 10 10/08 软件共存。这一决定确定了如何指定系统配置文件中的 <code>fdisk(1M)</code> 关键字。	第 117 页中的“x86: <code>fdisk</code> 配置文件关键字 (UFS 和 ZFS)”
检查系统是否受支持。	检查在 Solaris 环境中系统支持的硬件文档。	硬件生产商文档
检查系统是否有足够的磁盘空间用于 Solaris 软件。	验证是否已规划了足够的空间在系统上安装 Solaris 软件。	《Solaris 10 10/08 安装指南：规划安装和升级》中的第 4 章“系统要求、原则和升级 (规划)”
(可选) 设置系统参数。	您可以预配置系统信息，以避免在安装或升级时被提示输入该信息。	《Solaris 10 10/08 安装指南：基于网络的安装》中的第 2 章“预配置系统配置信息 (任务)”
针对自定义 JumpStart 安装准备系统。	创建并验证 <code>rules</code> 文件和配置文件。	第 3 章，准备自定义 JumpStart 安装 (任务)
(可选) 准备可选的自定义 JumpStart 功能。	如果使用了开始脚本、结束脚本或其他可选功能，请准备这些脚本或文件。	第 4 章，使用可选的自定义 JumpStart 功能 (任务) 和第 5 章，创建自定义规则和探测关键字 (任务)
(可选) 准备从网络安装 Solaris 软件。	要从远程 Solaris Operating System for x86 Platforms DVD 或 Solaris Software For x86 Platforms CD 映像安装系统，您需要设置系统以从安装服务器或引导服务器引导和安装。	《Solaris 10 10/08 安装指南：基于网络的安装》中的第 6 章“使用 CD 介质从网络进行安装 (任务)”
(可选) 准备 Solaris Flash 归档文件安装。	设置 Solaris Flash 归档文件安装的具体信息。	第 77 页中的“准备使用自定义 JumpStart 安装来安装 Solaris Flash 归档文件”
执行安装或升级。	引导系统以便初始化安装或升级。	第 83 页中的“x86: 使用自定义 JumpStart 程序和 GRUB 执行安装或升级”

x86: 执行自定义 JumpStart 安装

在自定义 JumpStart 安装过程中，JumpStart 程序尝试将要安装的系统与 `rules.ok` 文件中的规则进行匹配。JumpStart 程序从第一条规则到最后一条规则通读这些规则。当安装的系统与规则中定义的所有系统属性一致时发生匹配。一旦系统与规则匹配，JumpStart 程序即停止读取 `rules.ok` 文件，然后根据匹配规则的配置文件开始安装系统。

可以使用自定义 JumpStart 安装 Solaris Flash 归档文件。有关说明，请参见第 77 页中的“准备使用自定义 JumpStart 安装来安装 Solaris Flash 归档文件”。

请选择以下过程之一：

- 对于标准的自定义 JumpStart 过程，请参见第 83 页中的“x86: 使用自定义 JumpStart 程序和 GRUB 执行安装或升级”。
- 要通过编辑 GRUB 命令来执行自定义 JumpStart 安装，请参见第 85 页中的“x86: 通过编辑 GRUB boot 命令执行自定义 JumpStart 安装”。

▼ x86: 使用自定义 JumpStart 程序和 GRUB 执行安装或升级

通过本过程可以使用 GRUB 菜单为基于 x86 的系统安装 Solaris OS。

- 1 如果系统是网络的一部分，请确保已将以太网连接器或类似的网络适配器连接到了系统上。
- 2 如果要安装通过 [tip\(1\)](#) 行连接的系统，请确保所显示的窗口宽度和长度至少分别为 80 列和 24 行。
要确定 tip 窗口的当前尺寸，请使用 `stty(1)` 命令。
- 3 确定是否要使用配置文件软盘。
配置文件软盘不再用于引导系统，但可以准备一张只包含 JumpStart 目录的软盘。以后在执行 JumpStart 安装并从 CD-ROM 引导时可以使用该软盘。
 - 如果使用配置文件软盘，请将此配置文件软盘插入系统的软盘驱动器。
 - 如果不使用配置文件软盘，则继续执行[步骤 4](#)。
- 4 确定如何引导系统。
 - 如果从 Solaris Operating System DVD 或 Solaris Software - 1 CD 引导，请插入磁盘。系统的 BIOS 必须支持从 DVD 或 CD 引导。
 - 如果从网络引导，请使用引导前执行环境 (PXE) 网络引导。系统必须支持 PXE。通过使用系统 BIOS 设置工具或网络适配器的配置设置工具，使系统能够使用 PXE。

- 5 (可选) 如果希望从 DVD 或 CD 引导, 请在系统的 BIOS 中更改引导设置, 将其设置为从 DVD 或 CD 介质引导。请参见硬件文档获取相关说明。
- 6 如果系统关闭, 请打开系统。如果系统已打开, 请重新引导系统。
此时将显示 GRUB 菜单。此菜单提供了引导项的列表。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
|Solaris 10 10/08 image_directory |
|Solaris Serial Console ttya      |
|Solaris Serial Console ttyb (for lx50, v60x and v65x |
+-----+
Use the ^ and v keys to select which entry is highlighted. Press
enter to boot the selected OS, 'e' to edit the commands before
booting, or 'c' for a command-line.
```

image_directory 是安装映像所在目录的名称。JumpStart 文件的路径是使用 `add_install_client` 命令和 `-c` 选项进行定义的。

注 - 现在无需从 GRUB 项引导, 即可编辑引导项。编辑 GRUB 项后, 再执行 JumpStart 安装。有关如何编辑 GRUB 项和安装选项列表的说明, 请参见第 85 页中的“[x86: 通过编辑 GRUB boot 命令执行自定义 JumpStart 安装](#)”。

- 7 出现提示时, 执行以下指令之一:

```
Select the type of installation you want to perform:
```

```
 1 Solaris Interactive
 2 Custom JumpStart
 3 Solaris Interactive Text (Desktop session)
 4 Solaris Interactive Text (Console session)
 5. Apply driver updates
 6. Single User Shell
```

```
Enter the number of your choice.
```

```
Please make a selection (1-6).
```

要选择自定义 JumpStart 方法, 则键入 2 并按 Enter 键。

JumpStart 安装随即开始。

注 -

- 如果 30 秒内未进行选择，则 Solaris 交互式安装程序将启动。可以在命令行中键入任何键来停止计时器。
- 如果选择 1、3 或 4 项，将使用交互式安装进行安装。有关交互式安装的信息，请参见《Solaris 10 10/08 安装指南：基本安装》。
- 如果选择第 5 项，则将安装驱动程序更新。
- 如果选择第 6 项，则可以执行维护任务。

8 如果没有预配置 `sysidcfg` 文件中的系统信息，则在收到系统提示时，请回答有关系统配置的问题。

9 按照屏幕上的说明来安装软件。

当 JumpStart 程序完成 Solaris 软件的安装后，系统将自动重新引导。同时，`GRUB menu.lst` 文件将自动更新。然后，下一次使用 GRUB 菜单时，将显示所安装的 Solaris 实例。

完成安装后，安装日志将保存在文件中。您可以在以下目录中找到安装日志：

- `/var/sadm/system/logs`
- `/var/sadm/install/logs`

x86: 通过编辑 GRUB boot 命令执行自定义 JumpStart 安装

在某些情况下（如为了进行调试），可能需要修改 GRUB boot 命令。以下过程描述了执行自定义 JumpStart 安装之前编辑 GRUB boot 命令的步骤。

▼ x86: 修改 GRUB boot 命令

1 要开始安装，请执行前面的第 83 页中的“x86: 使用自定义 JumpStart 程序和 GRUB 执行安装或升级”过程中的步骤 1 至步骤 5。

2 如果系统关闭，请打开系统。如果系统已打开，请重新引导系统。

此时将显示 GRUB 菜单。此菜单提供了引导项的列表。所提供的项是要安装的 Solaris 实例。

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-----+
|Solaris 10 10/08 image_directory |
|Solaris Serial Console ttya      |
|Solaris Serial Console ttyb (lx50, v60x and v68) |
```

```
+-----+
Use the ^ and v keys to select which entry is highlighted. Press
enter to boot the selected OS, 'e' to edit the commands before
booting, or 'c' for a command-line.
```

image_directory 是安装映像所在目录的名称。

注 -

- 如果使用了 NFS 以及 `add_install_client` 命令和 `-c` 选项设置 JumpStart 目录的路径，则不需要将该路径包含在引导项中。
 - 如果未使用 NFS，则必须记录包含 JumpStart 目录的压缩配置文件的路径。
-

3 要停止引导进程并使用菜单项编辑器，请键入 **e**。

此时将显示 GRUB 编辑菜单。

```
kernel /I86PC.Solaris_11-8/multiboot kernel/unix -B console=ttyb,\
install_media=131.141.2.32:/export/mary/v11 \
module /I86PC.Solaris_11-8/x86.new
```

4 使用方向键选择引导项。

5 要编辑所选的命令，请键入 **e**。

将显示与以下示例相似的命令。

```
grub edit>kernel /I86PC.Solaris_11-8/multiboot kernel/unix -B \
console=ttyb,install_media=131.141.2.32:/export/mary/\
module /I86PC.Solaris_11-8/x86.new
```

6 通过键入所需的选项来编辑命令。

用于 JumpStart 安装的语法如下所示：

```
grub edit>kernel /I86PC.Solaris_11-image_directory/multiboot kernel/unix/ \
- install [url]ask options -B install_media=media_type
```

有关 JumpStart 选项的描述，请参见第 87 页中的“x86: 用于引导系统的命令参考”。

在以下示例中，OS 是通过网络以及自定义 JumpStart 配置文件进行安装的。

```
kernel /I86PC.Solaris_11-8/multiboot kernel/unix/ - install \
-B install_media=131.141.2.32:/export/mary/v11 \
module /I86PC.Solaris_11-8/x86.new
```

7 要接受编辑，请按 **Enter** 键。

系统将保存所做更改并显示 GRUB 主菜单。

注 - 按 Esc 键返回 GRUB 主菜单，而不保存所做更改。

- 8 要开始安装，请键入 **b**。

x86: 用于引导系统的命令参考

下表介绍了 GRUB 菜单引导命令的命令行选项。所列选项适用于 JumpStart 安装。

boot 命令的语法如下。

```
kernel /I86PC.Solaris_11-image_directory/multiboot kernel/unix/ - install \
[url|ask] options -B install_media=media_type
```

表 6-4 GRUB 菜单引导命令参考

选项	说明
- install	<p>执行自定义 JumpStart 安装。</p> <p>在以下示例中，系统将从 DVD 介质引导并使用下列选项：</p> <ul style="list-style-type: none"> ■ - install 执行自定义 JumpStart 安装 ■ file://jumpstart/config.tar 查找本地磁盘上的 JumpStart 配置文件 <pre>kernel /I86pc.Solaris_11.8/multiboot - install file://jumpstart/config.tar \ -B install_media=dvdrom module /I86Solaris_11.8/x86.new</pre>

表 6-4 GRUB 菜单引导命令参考 (续)

选项	说明
<code>[url ask]</code>	<p>指定自定义 JumpStart 文件的位置或提示您指定位置。</p> <ul style="list-style-type: none"> url—指定文件的路径。可以为位于 HTTP 或 HTTPS 服务器上的文件指定 URL。HTTP 服务器的语法如下： <pre data-bbox="349 347 835 399">http://server_name:IP_address/jumpstart_dir_path/ compressed_config_file&proxy_info</pre> <ul style="list-style-type: none"> 如果将一个 <code>sysidcfg</code> 文件放置在压缩配置文件中，必须指定包含该文件的服务器的 IP 地址，如下例所示： <pre data-bbox="385 494 992 598">kernel /I86pc.Solaris_11.8/multiboot install \ http://192.168.2.1/jumpstart/config.tar \ -B install_media=192.168.2.1/export/Solaris_11.8/boot \ module /I86PC.Solaris_11.8/x86.new</pre> 如果将压缩配置文件保存在位于防火墙后面的 HTTP 服务器上，则在引导过程中必须使用代理说明符。不需要为包含该文件的服务器指定 IP 地址。必须指定代理服务器的 IP 地址，如下例所示： <pre data-bbox="385 729 1078 833">kernel /I86pc.Solaris_11.8/multiboot install \ http://www.shadow.com/jumpstart/config.tar&proxy=131.141.6.151 \ -B install_media=192.168.2.1/export/Solaris_11.8/boot \ module /I86PC.Solaris_11.8/x86.new</pre> ask—指定在系统引导并连接到网络后，安装程序提示您键入压缩的配置文件的位。如果使用该选项，您将无法执行完全无人参与的 JumpStart 安装。如果通过按 <code>Return</code> 来跳过提示，Solaris 安装程序将交互式地配置网络参数。然后，安装程序提示您键入压缩配置文件的位置。以下示例将执行自定义 JumpStart 安装并从 DVD 介质引导。在系统连接到网络后，将提示您键入配置文件的位置。 <pre data-bbox="349 1058 963 1137">kernel /boot/multiboot kernel/unix install ask -B \ install_media=192.168.2.1:export/sol_11_x86/boot module \ /I86PC.Solaris_11.8_</pre>
<code>options</code>	<ul style="list-style-type: none"> dhcp—指定使用 DHCP 服务器获取引导系统所需的网络安装信息。该选项对于 JumpStart 安装不是必需的。如果未通过键入 <code>dhcp</code> 来指定使用 DHCP 服务器，则系统将使用 <code>/etc/bootparams</code> 文件或命名服务 <code>bootparams</code> 数据库。例如，如果您希望保留静态 IP 地址，您将不会指定 <code>dhcp</code>。例如： <pre data-bbox="349 1336 963 1414">kernel /I86pc.Solaris_11.8/multiboot install \ dhcp -B install_media=192.168.2.1:/export/Solaris_11.8/ \ boot module /I86PC.Solaris_11.8/x86.new</pre> 选项 <code>nowin</code> 和 <code>text</code> 不适用于 JumpStart 安装。这些选项对于交互式安装很有帮助。有关更多信息，请参见《Solaris 10 10/08 安装指南：基本安装》中的“通过 GRUB 使用 Solaris 安装程序执行安装或升级”。

使用自定义 JumpStart 进行安装（示例）

本章提供了一个示例，该示例使用自定义 JumpStart 安装程序在基于 SPARC 和基于 x86 的系统上设置和安装 Solaris 软件。

注 - 如果您要安装 Solaris ZFS 根池，请参见第 9 章，使用 JumpStart 安装 ZFS 根池以了解限制和配置文件示例。

- 第 89 页中的 “样例站点安装”
- 第 90 页中的 “创建安装服务器”
- 第 92 页中的 “x86: 创建市场营销系统的引导服务器”
- 第 93 页中的 “创建 JumpStart 目录”
- 第 93 页中的 “共享 JumpStart 目录”
- 第 93 页中的 “SPARC: 创建工程设计组的配置文件”
- 第 94 页中的 “x86: 创建市场营销组的配置文件”
- 第 94 页中的 “更新 rules 文件”
- 第 95 页中的 “验证 rules 文件”
- 第 95 页中的 “SPARC: 将工程设计系统设置为从网络安装”
- 第 96 页中的 “x86: 将市场营销系统设置为从网络安装”
- 第 97 页中的 “SPARC: 引导工程设计系统并安装 Solaris 软件”
- 第 97 页中的 “x86: 引导市场营销系统并安装 Solaris 软件”

样例站点安装

图 7-1 显示了本示例的站点安装。

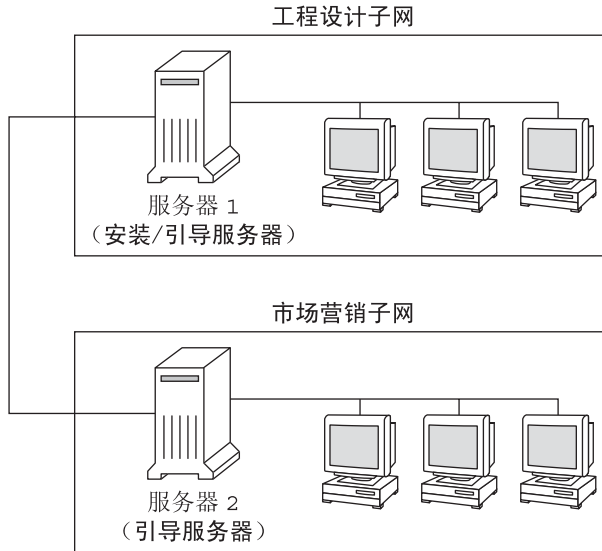


图 7-1 样例站点安装

在此样例站点上，条件如下：

- SPARC：工程设计组位于各自的子网上。该组使用 SPARCstation™ 系统进行软件开发。
- x86：市场营销组位于各自的子网上。该组使用基于 x86 的系统运行文字处理程序、电子表格和其他办公生产工具。
- 该站点使用 NIS。系统的以太网地址、IP 地址和主机名已在 NIS 映射中进行了预配置。该站点的子网掩码、日期和时间以及地理范围也已在 NIS 映射中进行了预配置。

注 - 市场营销系统的外围设备已在 `sysidcfg` 文件中进行了预配置。

- 工程设计系统和市场营销系统都要通过网络安装 Solaris 10 10/08 软件。

创建安装服务器

由于这两个组均需要从网络安装 Solaris 10 10/08 软件，因此使用 `server-1` 作为这两个组的安装服务器。使用 `setup_install_server(1M)` 命令将映像复制到 `server-1` 本地磁盘的 `/export/install` 目录中。从下列任一介质复制映像。

- Solaris Software CD 和 Solaris Languages CD
- Solaris Operating System DVD

必须将映像从光盘复制到一个空目录中，在这些示例中为 `sparc_10` 目录和 `x86_10` 目录。

示例 7-1 SPARC: 复制 Solaris 10 10/08 CD

将 Solaris Software for SPARC Platforms - 1 CD 放入连接到 `server-1` 的 CD-ROM 驱动器，然后键入以下命令：

```
server-1# mkdir -p /export/install/sparc_10
server-1# cd /CD_mount_point/Solaris_10/Tools
server-1# ./setup_install_server /export/install/sparc_10
```

将 Solaris Software for SPARC Platforms - 2 CD 放入连接到 `server-1` 的 CD-ROM 驱动器，然后键入以下命令：

```
server-1# cd /CD_mount_point/Solaris_10/Tools
server-1# ./add_to_install_server /export/install/sparc_10
```

为要安装的每张 Solaris Software CD 重复前一命令。

将第一张 SPARC: Solaris Languages for SPARC Platforms CD 插入连接到 `server-1` 的 CD-ROM 驱动器，然后键入以下命令：

```
server-1# cd /CD_mount_point/Solaris_10/Tools
server-1# ./add_to_install_server /export/install/sparc_10
```

对每张 SPARC: Solaris Languages for SPARC Platforms CD 重复上面的命令。

示例 7-2 x86: 复制 Solaris 10 10/08 CD

将 Solaris Software for x86 Platforms - 1 CD 放入连接到 `server-1` 的 CD-ROM 驱动器，然后键入以下命令：

```
server-1# mkdir -p /export/install/x86_10
server-1# cd /CD_mount_point/Solaris_10/Tools
server-1# ./setup_install_server /export/install/x86_10
```

将 Solaris Software for x86 Platforms - 2 CD 放入连接到 `server-1` 的 CD-ROM 驱动器，然后键入以下命令：

```
server-1# cd /CD_mount_point/Solaris_10/Tools
server-1# ./add_to_install_server /export/install/x86_10
```

为要安装的每张 Solaris Software CD 重复前一命令。

将第一张 Solaris Languages for x86 Platforms CD 插入连接到 `server-1` 的 CD-ROM 驱动器，然后键入以下命令：

示例 7-2 x86: 复制 Solaris 10 10/08 CD (续)

```
server-1# cd /CD_mount_point/Solaris_10/Tools
server-1# ./add_to_install_server /export/install/x86_10
```

对每张 Solaris Languages for x86 Platforms CD 重复上面的命令。

示例 7-3 SPARC: 复制 Solaris 10 10/08 DVD

将 Solaris Operating System for SPARC Platforms DVD 插入连接到 server-1 的 DVD-ROM 驱动器，然后键入以下命令：

```
server-1# mkdir -p /export/install/sparc_10
server-1# cd /DVD_mount_point/Solaris_10/Tools
server-1# ./setup_install_server /export/install/sparc_10
```

示例 7-4 x86: 复制 Solaris Operating System for x86 Platforms DVD

将 Solaris Operating System for x86 Platforms DVD 放入连接到 server-1 的 DVD-ROM 驱动器，然后键入以下命令：

```
server-1# mkdir -p /export/install/x86_10
server-1# cd /DVD_mount_point/Solaris_10/Tools
server-1# ./setup_install_server /export/install/x86_10
```

x86: 创建市场营销系统的引导服务器

系统无法从位于其他子网中的安装服务器进行引导，因此将 server-2 作为市场营销组所在子网的引导服务器。使用 `setup_install_server(1M)` 命令从 Solaris Operating System for x86 Platforms DVD 或 Solaris Software for x86 Platforms - 1 CD 复制引导软件。引导软件将被复制到 server-2 本地磁盘的 `/export/boot` 目录中。

选择介质并将引导软件安装到本地磁盘上。

- 如果将 Solaris Software for x86 Platforms - 1 CD 放入连接到 server-2 的 CD-ROM 驱动器，请键入以下命令：

```
server-2# cd /CD_mount_point/Solaris_10/Tools
server-2# ./setup_install_server -b /export/boot
```

- 如果将 Solaris Operating System for x86 Platforms DVD 放入连接到 server-2 的 DVD-ROM，请键入以下命令：

```
server-2# cd /DVD_mount_point/Solaris_10/Tools
server-2# ./setup_install_server -b /export/boot
```

在 `setup_install_server` 命令中，`-b` 指定 `setup_install_server` 将引导信息复制到名为 `/export/boot` 的目录中。

创建 JumpStart 目录

至此，您已设置了安装服务器和引导服务器，接下来将在 `server-1` 中创建一个 JumpStart 目录。您可以使用网络上的任何系统。此目录保存 Solaris 软件的自定义 JumpStart 安装所需要的文件。设置此目录的方法是从 Solaris Operating System DVD 映像或从已复制到 `/export/install` 的 Solaris Software - 1 CD 映像复制样例目录：

```
server-1# mkdir /jumpstart
server-1# cp -r /export/install/sparc_10/Solaris_10/Misc/jumpstart_sample /jumpstart
```

共享 JumpStart 目录

要使网络中的系统能够访问 `rules` 文件和配置文件，您需要共享 `/jumpstart` 目录。要使目录能够共享，需要将下行添加到 `/etc/dfs/dfstab` 文件：

```
share -F nfs -o ro,anon=0 /jumpstart
```

然后，在命令行中键入 `shareall` 命令：

```
server-1# shareall
```

SPARC: 创建工程设计组的配置文件

对于工程设计系统，在 `/jumpstart` 目录中创建名为 `eng_prof` 的文件。`eng_prof` 文件包含以下各项，这些项定义了要在工程设计组的系统中安装的 Solaris 10 10/08 软件：

```
install_type  initial_install
system_type   standalone
partitioning  default
cluster       SUNWCprog
filesystems   any 512 swap
```

以上示例配置文件指定下列安装信息。

<code>install_type</code>	该安装将被视为初始安装而不是升级。
<code>system_type</code>	工程设计系统是独立系统。
<code>partitioning</code>	JumpStart 软件使用缺省的磁盘分区在工程设计系统上安装 Solaris 软件。

cluster	将安装开发者系统支持软件组。
filesys	工程设计组中的每个系统都将有 512 MB 的交换空间。

x86: 创建市场营销组的配置文件

对于市场营销系统，在 /jumpstart 目录中创建名为 marketing_prof 的文件。marketing_prof 文件包含以下各项，这些项定义了要在市场营销组的系统中安装的 Solaris 10 10/08 软件：

```
install_type  initial_install
system_type   standalone
partitioning  default
cluster       SUNWCuser
package       SUNWaudio
```

以上示例配置文件指定下列安装信息。

install_type	该安装将被视为初始安装而不是升级。
system_type	市场营销系统是独立系统。
partitioning	JumpStart 软件将使用缺省的磁盘分区在市场营销系统上安装 Solaris。
cluster	将安装最终用户 Solaris 软件组。
package	音频演示教程软件包将被添加到每个系统。

更新 rules 文件

现在，必须将规则添加到 rules 文件中。Solaris 安装程序使用这些规则在自定义 JumpStart 安装过程中为每个系统选择正确的安装（配置文件）。

在该站点上，每个部门都位于各自的子网上并有各自的网络地址。工程设计部门位于 255.222.43.0 子网中。市场营销部门位于 255.222.44.0 子网中。您可以使用此信息控制为工程设计系统和市场营销系统安装 Solaris 10 10/08 软件的方式。在 /jumpstart 目录中，编辑 rules 文件，删除所有的示例规则，并将以下行添加到该文件中：

```
network 255.222.43.0 - eng_prof -
network 255.222.44.0 - marketing_prof -
```

基本上，这些规则说明位于 255.222.43.0 网络中的系统将通过使用 eng_prof 配置文件来安装 Solaris 10 10/08 软件。而位于 255.222.44.0 网络中的系统将通过使用 marketing_prof 配置文件来安装 Solaris 10 10/08 软件。

注 - 可以使用样例规则，以便使用网络地址标识分别要通过使用 `eng_prof` 和 `marketing_prof` 来安装 Solaris 10 10/08 软件的系统。还可以使用主机名、内存大小或模型类型作为规则关键字。表 8-1 包含一个可在 `rules` 文件中使用的关键字的完整列表。

验证 rules 文件

设置 `rules` 和配置文件后，运行 `check` 脚本来验证文件是否正确：

```
server-1# cd /jumpstart
server-1# ./check
```

如果 `check` 脚本没有找到任何错误，该脚本将创建 `rules.ok` 文件。

SPARC: 将工程设计系统设置为从网络安装

设置 `/jumpstart` 目录和文件后，在安装服务器 `server-1` 中使用 `add_install_client` 命令设置工程设计系统，以便从安装服务器安装 Solaris 软件。`server-1` 也是工程设计组的子网的引导服务器。

```
server-1# cd /export/install/sparc_10/Solaris_10/Tools
server-1# ./add_install_client -c server-1:/jumpstart host-eng1 sun4u
server-1# ./add_install_client -c server-1:/jumpstart host-eng2 sun4u
```

在 `add_install_client` 命令中，所使用的选项具有以下含义：

-c 指定服务器 (`server-1`) 和 JumpStart 目录的路径 (`/jumpstart`)。使用 NFS 时使用此选项。

注 - 如果未使用 NFS，则使用以下命令指定 JumpStart 目录的路径：

- 对基于 SPARC 的系统，在 `boot` 命令中指定路径
 - 对基于 x86 的系统，通过编辑 GRUB 菜单项指定路径
-

host-eng1 工程设计组中的某个系统的名称。

host-eng2 工程设计组中的另一个系统的名称。

sun4u 指定将 `server-1` 用作安装服务器的系统的平台组。该平台组是面向 Ultra 5 系统的。

x86: 将市场营销系统设置为从网络安装

接下来，将在引导服务器 (server-2) 上使用 `add_install_client` 命令。该命令将市场营销系统设置为从引导服务器进行引导，并从安装服务器 (server-1) 安装 Solaris 软件：

```
server-2# cd /marketing/boot-dir/Solaris_10/Tools
server-2# ./add_install_client -s server-1:/export/install/x86_10 \
-c server-1:/jumpstart host-mkt1 i86pc
server-2# ./add_install_client -s server-1:/export/install/x86_10 \
-c server-1:/jumpstart host-mkt2 i86pc
server-2# ./add_install_client -d -s server-1:/export/install/x86_10 \
-c server-1:/jumpstart SUNW.i86pc i86pc
server-2# ./add_install_client -c server-1:/jumpstart host-mkt1 sun4u
server-2# ./add_install_client -c server-1:/jumpstart host-mkt2 sun4u
```

在 `add_install_client` 命令中，所使用的选项具有下列含义：

- d 指定客户机使用 DHCP 获取网络安装参数。此选项对使用 PXE 网络引导从网络引导的客户机是必需的。对于不使用 PXE 网络引导的网络引导客户机，-d 是可选的。
- s 指定安装服务器 (server-1) 和 Solaris 软件的路径 (/export/install/x86_10)。
- c 指定服务器 (server-1) 和 JumpStart 目录的路径 (/jumpstart)。使用 NFS 时使用此选项。

注 - 如果未使用 NFS，则使用以下命令指定 JumpStart 目录的路径：

- 对基于 SPARC 的系统，在 boot 命令中指定路径
 - 对基于 x86 的系统，通过编辑 GRUB 菜单项指定路径
-

- host-mkt1 市场营销组中的某个系统的名称。
- host-mkt2 市场营销组中的另一个系统的名称。
- sun4u 指定将 server-1 用作安装服务器的系统的平台组。该平台组是面向 Ultra 5 系统的。
- SUNW.i86pc 用于所有 Solaris x86 客户机的 DHCP 类名称。如果希望用一个命令来配置所有的 Solaris x86 DHCP 客户机，请使用此类名称。
- i86pc 指定使用该引导服务器的系统的平台组。平台名称代表基于 x86 的系统。

SPARC: 引导工程设计系统并安装 Solaris 软件

设置服务器和文件后，可通过在每个系统的 ok (PROM) 提示符下使用以下 boot 命令引导工程设计系统：

```
ok boot net - install
```

Solaris OS 会自动安装到工程设计组的系统上。

x86: 引导市场营销系统并安装 Solaris 软件

可以从以下位置之一引导系统：

- Solaris Software for x86 Platforms - 1 CD
- Solaris Operating System for x86 Platforms DVD
- 使用 PXE 网络引导的网络

Solaris 软件会自动安装在市场营销组的系统中。

自定义 JumpStart (参考)

本章列出您可以在 `rules` 文件、配置文件、开始脚本和结束脚本中使用的关键字和值。

注 - 如果您要安装 Solaris ZFS 根池，请参见第 9 章，使用 [JumpStart 安装 ZFS 根池](#) 以了解限制和配置文件示例。此外，有关特定于 ZFS 的关键字和可以在配置文件中使用的关键字列表，请参见表 8-2。

- 第 99 页中的“规则关键字和值”
- 第 103 页中的“配置文件关键字和值”
- 第 142 页中的“自定义 JumpStart 环境变量”
- 第 144 页中的“探测关键字和值”

规则关键字和值

表 8-1 介绍了可在 `rules` 文件中使用的关键字和值。有关创建 `rules` 文件的详细说明，请参见第 32 页中的“创建 `rules` 文件”。

表 8-1 规则关键字和值的描述

关键字	值	匹配项
any	减号 (-)	任意项。使用 any 关键字总会成功。
arch	<code>processor_type</code> <code>processor_type</code> 的有效值包括： <ul style="list-style-type: none"> ■ SPARC: <code>sparc</code> ■ x86: <code>i386</code> 	系统的处理器类型。 <code>uname -p</code> 命令报告系统的处理器类型。

表 8-1 规则关键字和值的描述 (续)

关键字	值	匹配项
disksize	<p><i>actual_disk_name</i> <i>size_range</i></p> <p><i>actual_disk_name</i>—磁盘名称，格式为 <i>cxt ydz</i>，如 <i>c0t3d0</i> 或 <i>c0d0</i>，或者特殊字 <i>rootdisk</i>。如果使用 <i>rootdisk</i>，那么将按以下顺序确定要匹配的磁盘：</p> <ul style="list-style-type: none"> ■ SPARC：包含预安装的引导映像的磁盘，这种磁盘是一种基于 SPARC 的新系统，出厂时安装了 JumpStart ■ <i>c0t3d0s0</i> 磁盘（如果该磁盘存在） ■ 按照内核探测顺序搜索到的第一个可用磁盘 <p><i>size_range</i>—磁盘的大小，必须指定为一个范围 (<i>x-x</i>)，以 MB 为单位。</p> <p>注—计算 <i>size_range</i> 时，请记住 1 MB 等于 1,048,576 字节。被宣传为 "535 MB" 的磁盘可能仅包含 510 MB 的磁盘空间。JumpStart 程序会将 "535 MB" 的磁盘视为 510 MB 的磁盘，原因是 $535,000,000 / 1,048,576 = 510$。"535 MB" 的磁盘与等于 530-550 的 <i>size_range</i> 不匹配。</p>	<p>系统磁盘的名称和大小（以 MB 为单位）。</p> <p>示例：</p> <p><code>disksize c0t3d0 250-300</code></p> <p>在该示例中，JumpStart 程序尝试匹配名为 <i>c0t3d0</i> 的系统磁盘。该磁盘可以保存 250 MB 到 300 MB 的信息。</p> <p>示例：</p> <p><code>disksize rootdisk 750-1000</code></p> <p>在该示例中，JumpStart 程序尝试按下列顺序匹配磁盘：</p> <ol style="list-style-type: none"> 1. 包含预安装引导映像的系统磁盘 2. <i>c0t3d0s0</i> 磁盘（如果该磁盘存在） 3. 可包含 750 MB 到 1 GB 信息的第一个可用磁盘
domainname	<i>actual_domain_name</i>	<p>系统的域名，它控制着命名服务确定信息的方式。</p> <p>安装系统后，<code>domainname</code> 命令就可以报告系统的域名。</p>
hostaddress	<i>actual_IP_address</i>	系统的 IP 地址。
hostname	<i>actual_host_name</i>	<p>系统的主机名。</p> <p>安装系统后，<code>uname -n</code> 命令就可以报告系统的主机名。</p>

表 8-1 规则关键字和值的描述 (续)

关键字	值	匹配项
installed	<p><i>slice version</i></p> <p><i>slice</i>—磁盘片名称，格式为 <i>cwtxdysz</i>，如 <i>c0t3d0s5</i>，或特殊字 <i>any</i> 或 <i>rootdisk</i>。如果使用 <i>any</i>，JumpStart 程序将尝试按照内核探测顺序来匹配所有的系统磁盘。如果使用了 <i>rootdisk</i>，则按以下顺序确定要匹配的磁盘：</p> <ul style="list-style-type: none"> ■ SPARC：包含预安装的引导映像的磁盘。这种磁盘是一种基于 SPARC 的新系统，出厂时安装了 JumpStart ■ <i>c0t3d0s0</i> 磁盘（如果该磁盘存在） ■ 按照内核探测顺序搜索到的第一个可用磁盘 <p><i>version</i>—版本名称或特殊字 <i>any</i> 或 <i>upgrade</i>。如果使用 <i>any</i>，那么将匹配任何一个 Solaris 发行版或 SunOS 发行版。如果使用 <i>upgrade</i>，那么将匹配任何支持和可升级的 Solaris 发行版。</p> <p>如果 JumpStart 程序找到一个 Solaris 发行版但无法确定版本，则返回的版本为 <i>SystemV</i>。</p>	<p>包含与特定 Solaris 软件版本相对应的根 (/) 文件系统的磁盘。</p> <p>示例：</p> <pre>installed c0t3d0s1 Solaris 10</pre> <p>在该示例中，JumpStart 程序尝试匹配在 <i>c0t3d0s1</i> 上包含 Solaris 根 (/) 文件系统的系统。</p>
karch	<p><i>actual_platform_group</i></p> <p>有效值为 <i>sun4u</i>、<i>i86pc</i> 和 <i>prep</i>。 http://docs.sun.com 上的 Solaris Sun 硬件平台指南中提供了系统及其相应平台组的列表。</p>	<p>系统的平台组。</p> <p>安装系统后，<i>arch -k</i> 命令或 <i>uname -m</i> 命令就可以报告该系统的平台组。</p>
memsize	<p><i>physical_mem</i></p> <p>该值必须是 MB 值的范围 <i>x-x</i> 或单个 MB 值。</p>	<p>系统物理内存的大小（以 MB 为单位）。</p> <p>示例：</p> <pre>memsize 64-128</pre> <p>该示例尝试匹配物理内存大小在 64 MB 到 128 MB 之间的系统。</p> <p>如果已经安装了系统，<i>prtconf</i> 命令输出的第 2 行将报告系统的物理内存大小。</p>

表 8-1 规则关键字和值的描述 (续)

关键字	值	匹配项
model	<i>actual_platform_name</i>	<p>系统的平台名称。有关有效平台名称的列表，请参见 http://docs.sun.com 上的 Solaris Sun 硬件平台指南。</p> <p>要查找已安装的系统的平台名称，请使用 <code>uname -i</code> 命令或 <code>prtconf</code> 命令输出的第 5 行。</p> <p>注 - 如果 <i>actual_platform_name</i> 包含空格，那么您必须使用下划线 () 来代替空格。</p> <p>示例：</p> <pre>SUNW,Sun_4_50</pre>
network	<i>network_num</i>	<p>系统的网络号。JumpStart 程序通过在系统的 IP 地址和子网掩码之间执行逻辑“与”来确定该值。</p> <p>示例：</p> <pre>network 192.168.2.0</pre> <p>如果子网掩码是 255.255.255.0，该示例尝试将系统与 192.168.2.8 IP 地址匹配。</p>
osname	Solaris_x	<p>已经安装在系统上的某个版本的 Solaris 软件。</p> <p>示例：</p> <pre>osname Solaris 10</pre> <p>在本示例中，JumpStart 程序尝试匹配已安装了 Solaris 10 10/08 OS 的系统。</p>

表 8-1 规则关键字和值的描述 (续)

关键字	值	匹配项
probe	<i>probe_keyword</i>	<p>有效的探测关键字或有效的自定义探测关键字。</p> <p>示例：</p> <pre>probe disks</pre> <p>本示例按内核探测顺序（如 <code>c0t3d0s1</code>、<code>c0t4d0s0</code>）返回基于 SPARC 的系统的磁盘大小，以 MB 为单位。JumpStart 程序会设置 <code>SI_DISKLIST</code>、<code>SI_DISKSIZE</code>、<code>SI_NUMDISKS</code> 和 <code>SI_TOTALDISK</code> 环境变量。</p> <p>注 - <code>probe</code> 关键字的独特之处在于，它既不尝试匹配属性，也不尝试运行配置文件，<code>probe</code> 关键字将返回一个值。因此，您不能使用 <code>probe</code> 规则关键字指定开始脚本、配置文件和结束脚本。</p> <p>第 5 章，创建自定义规则和探测关键字（任务） 对探测关键字进行了介绍。</p>
totaldisk	<p><i>size_range</i></p> <p>该值必须指定为 MB 范围 (<i>x-x</i>)。</p> <p>注 - 计算 <i>size_range</i> 时，请记住 1 MB 等于 1,048,576 字节。被宣传为 "535 MB" 的磁盘可能仅有 510 MB 的磁盘空间。JumpStart 程序会将 "535 MB" 的磁盘视为 510 MB 的磁盘，原因是 $535,000,000 / 1,048,576 = 510$。"535 MB" 的磁盘与等于 530-550 的 <i>size_range</i> 不匹配。</p>	<p>系统上的总磁盘空间（以 MB 为单位）。总磁盘空间包括连接到系统上的所有可操作磁盘的空间。</p> <p>示例：</p> <pre>totaldisk 300-500</pre> <p>在本示例中，JumpStart 程序尝试匹配总磁盘空间在 300 MB 到 500 MB 之间的系统。</p>

配置文件关键字和值

本节说明了可在配置文件中使用的配置文件关键字和值。有关创建配置文件的详细说明，请参见第 35 页中的“创建配置文件”。这些关键字可用于安装 UFS 和 ZFS 文件系统。如果关键字可在 ZFS 配置文件中使用，则会注明“ZFS”术语。

配置文件关键字快速参考

借助表 8-2，您可以根据安装方案快速确定可以使用的关键字。除非在关键字描述中另有说明，否则该关键字只能用于初始安装选项。此外，除非注明了关键字可在 ZFS 根池配置文件中使用的，否则，这些关键字用于 UFS 文件系统。

表 8-2 配置文件关键字概述

配置文件关键字	安装方案					
	独立系统（非联网的）	独立系统（联网的）或服务器	OS 服务器	升级	升级时重新分配磁盘空间	可用于 ZFS 根池
archive_location（安装 Solaris Flash 归档文件）	X	X				
backup_media					X	
boot_device（UFS 和 ZFS）	X	X	X			X
bootenv（UFS 和 ZFS）	X	X	X			X
client_arch			X			
client_root			X			
client_swap			X			
cluster（添加软件组）（UFS 和 ZFS）	X	X	X			X
cluster（添加或删除簇）（UFS 和 ZFS）	X	X	X	X	X	X
dontuse（UFS 和 ZFS）	X	X	X			X
fdisk（仅 x86）（UFS 和 ZFS）	X	X	X			X
filesystem（安装远程文件系统）（UFS 和 ZFS）		X	X			X
filesystem（创建本地文件系统）	X	X	X			
filesystem（创建镜像的文件系统）	X	X	X			
forced_deployment（安装 Solaris Flash 差别归档文件）	X	X				
geo（UFS 和 ZFS）	X	X	X	X	X	X
install_type（UFS 和 ZFS）	X	X	X	X	X	X
layout_constraint					X	
local_customization（安装 Solaris Flash 归档文件）	X	X				

表 8-2 配置文件关键字概述 (续)

配置文件关键字	安装方案					
	独立系统 (非联网的)	独立系统 (联网的) 或服务器	OS 服务器	升级	升级时重新分配磁盘空间	可用于 ZFS 根池
locale (UFS 和 ZFS)	X	X	X	X	X	X
metadb (创建状态数据库副本)	X	X	X			
no_master_check (安装 Solaris Flash 差别归档文件)	X	X				
no_content_check (安装 Solaris Flash 差别归档文件)	X	X				
num_clients			X			
package (UFS 和 ZFS)	X	X	X	X	X	X
partitioning	X	X	X			
patch (修补程序)	X	X	X	X	X	
pool (仅 ZFS 根池)	X	X	X			X
root_device (UFS 和 ZFS)	X	X	X	X	X	X
system_type	X	X	X			
usedisk (UFS 和 ZFS)	X	X	X			X

配置文件关键字描述和示例

archive_location 关键字

archive_location *retrieval_type location*

retrieval_type *retrieval_type* 和 *location* 的值取决于 Solaris Flash 归档文件的存储位置。以下部分包含可用于 *retrieval_type* 和 *location* 的值以及如何使用 archive_location 关键字的示例。

- 第 106 页中的 “NFS 服务器上存储的归档文件”
- 第 106 页中的 “HTTP 或 HTTPS 服务器上存储的归档文件”
- 第 107 页中的 “FTP 服务器上存储的归档文件”
- 第 108 页中的 “本地磁带上存储的归档文件”
- 第 109 页中的 “本地设备上存储的归档文件”
- 第 110 页中的 “本地文件上存储的归档文件”

location 以下各节对位置进行了详细说明。



注意 - 安装了非全局区域后，将无法正确创建 Solaris Flash 归档文件。Solaris Flash 功能与 Solaris Zone 分区技术不兼容。如果创建了 Solaris Flash 归档文件，当在下列条件下部署归档文件时，所生成的归档文件无法正确安装：

- 归档文件是在非全局区域中创建的
- 归档文件是在安装了非全局区域的全局区域中创建的

NFS 服务器上存储的归档文件

如果归档文件存储在 NFS 服务器上，则对 `archive_location` 关键字使用以下语法。

```
archive_location nfs server_name:/path/filename retry n
```

server_name 存储归档文件的服务器的名称。

path 要从指定服务器检索的归档文件的位置。如果该路径含有 \$HOST，那么 Solaris Flash 安装实用程序将使用要安装的克隆系统的名称替换 \$HOST。

filename Solaris Flash 归档文件的名称。

retry n 可选的关键字。*n* 是 Solaris Flash 实用程序尝试挂载归档文件的最大次数。

示例 8-1 NFS 服务器上存储的归档文件

```
archive_location nfs golden:/archives/usrarchive
```

```
archive_location nfs://golden/archives/usrarchive
```

HTTP 或 HTTPS 服务器上存储的归档文件

如果归档文件存储在 HTTP 服务器上，则对 `archive_location` 关键字使用以下语法。

```
archive_location http://server_name:port/path/filename optional_keywords
```

如果归档文件存储在 HTTPS 服务器上，则对 `archive_location` 关键字使用以下语法。

```
archive_location https://server_name:port/path/filename optional_keywords
```

server_name 存储归档文件的服务器的名称。

端口 可选的端口。*port* 可以是一个端口号，也可以是在运行时确定其端口号的 TCP 服务的名称。

	如果不指定端口，那么 Solaris Flash 安装实用程序将使用缺省的 HTTP 端口号 80。
<i>path</i>	要从指定服务器检索的归档文件的位置。如果该路径含有 \$HOST，那么 Solaris Flash 安装实用程序将使用要安装的克隆系统的名称替换 \$HOST。
<i>filename</i>	Solaris Flash 归档文件的名称。
<i>optional_keywords</i>	可在从 HTTP 服务器检索 Solaris Flash 归档文件时指定的可选关键字。

表 8-3 用于 archive_location HTTP 的可选关键字

关键字	值定义
<code>auth basic user_name password</code>	<p>如果归档文件位于有口令保护的 HTTP 服务器上，那么您必须在配置文件中包含访问 HTTP 服务器所需的用户名和口令。</p> <p>注 - 在准备与自定义 JumpStart 一起使用的配置文件中使用时此认证方法具有一定的风险。未经授权的用户可能会拥有对包含口令的配置文件的访问权限。</p>
<code>timeout min</code>	<p>使用 <code>timeout</code> 关键字，可以以分钟为单位，指定在无法接收到来自 HTTP 服务器的数据的情形下允许等待的最大时间长度。如果发生超时，则连接将被关闭、重新打开并恢复。如果指定 <code>timeout</code> 值为 0（零），则不重新打开连接。</p> <ul style="list-style-type: none"> 如果重新连接超时，那么 Solaris Flash 安装实用程序将尝试在归档文件中的最后一个已知位置处恢复安装。如果 Solaris Flash 安装实用程序无法在最后一个已知位置处恢复安装，那么将从归档文件的开头重新开始检索，并且将放弃超时前检索到的数据。 如果安装软件包时发生超时重新连接，那么将从软件包开头重试该软件包，并且将放弃超时前检索到的数据。
<code>proxy host:port</code>	<code>proxy</code> 关键字允许您指定代理主机和代理端口。您可以使用代理主机从防火墙的另一侧检索 Solaris Flash 归档文件。指定 <code>proxy</code> 关键字时必须提供一个代理端口。

示例 8-2 HTTP 或 HTTPS 服务器上存储的归档文件

```
archive_location http://silver/archives/usrarchive.flar timeout 5
```

`auth basic user_name password` 关键字的示例：

```
archive_location http://silver/archives/usrarchive.flar timeout 5 user1 secret
```

FTP 服务器上存储的归档文件

如果归档文件存储在 FTP 服务器上，则对 `archive_location` 关键字使用以下语法。

```
archive_location ftp://user_name:password@server_name:port/path/filename optional_keywords
```

<i>user_name:password</i>	配置文件中访问 FTP 服务器所需的用户名和口令。
<i>server_name</i>	存储归档文件的服务器的名称。
<i>port</i>	端口是可选项。 <i>port</i> 可以是一个端口号，也可以是在运行时确定其端口号的 TCP 服务的名称。 如果不指定端口，那么 Solaris Flash 安装实用程序将使用缺省的 FTP 端口号 21。
<i>path</i>	要从指定服务器检索的归档文件的位置。如果该路径含有 \$HOST，那么 Solaris Flash 安装实用程序将使用要安装的克隆系统的名称替换 \$HOST。
<i>filename</i>	Solaris Flash 归档文件的名称。
<i>optional_keywords</i>	可在从 FTP 服务器检索 Solaris Flash 归档文件时指定的可选关键字。

表 8-4 用于 archive_location FTP 的可选关键字

关键字	值定义
<i>timeout min</i>	<p>使用 <i>timeout</i> 关键字，可以以分钟为单位，指定在无法接收到来自 HTTP 服务器的数据的情形下允许等待的最大时间长度。如果发生超时，则连接将被关闭、重新打开并恢复。如果指定 <i>timeout</i> 值为 0（零），则不重新打开连接。</p> <ul style="list-style-type: none"> ■ 如果重新连接超时，那么 Solaris Flash 安装实用程序将尝试在归档文件中的最后一个已知位置处恢复安装。如果 Solaris Flash 安装实用程序无法在最后一个已知位置处恢复安装，那么将从归档文件的开头重新开始检索，并且将放弃超时前检索到的数据。 ■ 如果安装软件包时发生超时重新连接，那么将从软件包开头重试该软件包，并且将放弃超时前检索到的数据。
<i>proxy host:port</i>	<i>proxy</i> 关键字允许您指定代理主机和代理端口。您可以使用代理主机从防火墙的另一侧检索 Solaris Flash 归档文件。指定 <i>proxy</i> 关键字时必须提供一个代理端口。

示例 8-3 FTP 服务器上存储的归档文件

```
archive_location ftp://user1:secret@silver/archives/usrarchive.flar timeout 5
```

本地磁带上存储的归档文件

如果归档文件存储在磁带上，则对 *archive_location* 关键字使用以下语法。

```
archive_location local_tape device position
```

<i>device</i>	存储 Solaris Flash 归档文件的磁带机的名称。如果设备名称使用了规则路径，那么 Solaris Flash 安装实用程序将从设备节点的路径检索归档文件。如果设备名称未使用规则路径，那么 Solaris Flash 安装实用程序会将 <code>/dev/rmt/</code> 添加到其路径中。
<i>position</i>	指定磁带上存储归档文件的位置。如果未提供位置，那么 Solaris Flash 安装实用程序将从磁带机上的当前位置检索归档文件。通过指定 <i>position</i> ，您可以将开始脚本或 <code>sysidcfg</code> 文件放置于磁带上归档文件前面的位置上。

示例 8-4 本地磁带上存储的归档文件

```
archive_location local_tape /dev/rmt/0n 5
```

```
archive_location local_tape 0n 5
```

本地设备上存储的归档文件

如果将 Solaris Flash 归档文件存储在面向文件系统的、可随机访问的设备（如软盘或 DVD）上，那么您可以从本地设备检索 Solaris Flash 归档文件。对于 `archive_location` 关键字，使用以下语法。

注 - 通过使用本地磁带语法，您可以从面向流的设备（如磁带）中检索归档文件。

```
archive_location local_device device path/filename file_system_type
```

<i>device</i>	存储 Solaris Flash 归档文件的驱动器的名称。如果设备名称使用了规则路径，那么将直接挂载该设备。如果设备名称未使用规则路径，那么 Solaris Flash 安装实用程序会将 <code>/dev/dsk/</code> 添加到其路径中。
<i>path</i>	Solaris Flash 归档文件的路径，相对于指定设备上的文件系统的根。如果该路径含有 <code>\$HOST</code> ，那么 Solaris Flash 安装实用程序将使用要安装的克隆系统的名称替换 <code>\$HOST</code> 。
<i>filename</i>	Solaris Flash 归档文件的名称。
<i>file_system_type</i>	指定设备上文件系统的类型。如果未提供文件系统类型，那么 Solaris Flash 安装实用程序将尝试挂载 UFS 文件系统。如果 UFS 挂载失败，那么 Solaris Flash 安装实用程序将尝试挂载 HSFS 文件系统。

示例 8-5 本地设备上存储的归档文件

要从 UFS 文件系统格式的本地硬盘驱动器上检索归档文件，请使用以下命令：

```
archive_location local_device c0t0d0s0 /archives/$HOST
```

要从包含 HSFS 文件系统的本地 CD-ROM 上检索归档文件，请使用以下命令：

示例 8-5 本地设备上存储的归档文件 (续)

```
archive_location local_device c0t0d0s0 /archives/usrarchive
```

本地文件上存储的归档文件

您可以检索作为本地文件存储在从中引导克隆系统的 `miniroot` 中的归档文件。执行自定义 JumpStart 安装时，从 DVD、CD 或基于 NFS 的 `miniroot` 引导系统。安装软件从该 `miniroot` 装入和运行。因而，存储在 DVD、CD 或基于 NFS 的 `miniroot` 中的 Solaris Flash 归档文件可以作为本地文件进行访问。对于 `archive_location` 关键字，请使用以下语法。

```
archive_location local_file path/filename
```

path 归档文件的位置。从 Solaris Software - 1 CD 或者从 Solaris Operating System DVD 引导系统时，路径必须可由该系统作为本地文件访问。在从 Solaris Software - 1 CD 或从 Solaris Operating System DVD 引导时，系统不能访问 `/net` 或其他任何自动挂载的目录。

filename Solaris Flash 归档文件的名称。

示例 8-6 本地文件上存储的归档文件

```
archive_location local_file /archives/usrarchive
```

backup_media 配置文件关键字

```
backup_media type path
```

需要重新分配磁盘空间时，只能将 `backup_media` 用于升级选项。

`backup_media` 定义在升级过程中，由于空间不足而需要重新分配空间时，用于备份文件系统的介质。如果备份需要多个磁带或软盘，那么在升级过程中会提示您插入磁带或软盘。

有效的 <i>type</i> 值	有效的 <i>path</i> 值	规格
<code>local_tape</code>	<code>/dev/rmt/n</code>	将被升级的系统上的本地磁带机。 <i>path</i> 必须是该磁带机的字符（原始）设备路径。 <i>n</i> 是磁带机的编号。
<code>local_diskette</code>	<code>/dev/rdisketten</code>	将被升级的系统上的本地软盘驱动器。 <i>path</i> 必须是该软盘驱动器的字符（原始）设备路径。 <i>n</i> 是软盘驱动器的编号。 用于备份的软盘必须已经格式化。

有效的 <i>type</i> 值	有效的 <i>path</i> 值	规格
local_filesystem	<i>/dev/dsk/cwt xdyz z</i> <i>/file_system</i>	将被升级的系统上的本地文件系统。您不能指定升级过程中会被更改的本地文件系统。 <i>path</i> 可以是磁盘片的块设备路径。例如， <i>/dev/dsk/c wtxd ysz</i> 中的 <i>tx</i> 可能是不需要的。或者， <i>path</i> 可以是指向 <i>/etc/vfstab</i> 文件挂载的文件系统的绝对路径。
remote_filesystem	<i>host:/file_system</i>	远程系统上的 NFS 文件系统。 <i>path</i> 必须包含远程系统 <i>host</i> 的名称或 IP 地址，以及指向 NFS 文件系统 <i>file_system</i> 的绝对路径。NFS 文件系统必须具有读/写访问权限。
remote_system	<i>user@host :/directory</i>	远程系统上，可通过远程 shell <i>rsh</i> 访问的目录。将被升级的系统必须能够通过远程系统的 <i>.rhosts</i> 文件访问该远程系统。 <i>path</i> 必须包含远程系统 <i>host</i> 的名称和指向目录 <i>directory</i> 的绝对路径。缺省情况下，如果未指定用户登录 ID <i>user</i> ，那么将使用 <i>root</i> 。

示例 8-7 backup_media 配置文件关键字

```

backup_media local_tape /dev/rmt/0

backup_media local_diskette /dev/rdiskette1

backup_media local_filesystem /dev/dsk/c0t3d0s4

backup_media local_filesystem /export

backup_media remote_filesystem system1:/export/temp

backup_media remote_system user1@system1:/export/temp

```

boot_device 配置文件关键字 (UFS 和 ZFS)

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 *boot_device* 关键字。在 UFS 和 ZFS 安装中，此关键字的用法是相同的。

- 有关可在 UFS 和 ZFS 安装中使用的关键字列表，请参见表 8-2
- 有关执行 ZFS 安装的信息，请参见第 9 章，使用 [JumpStart 安装 ZFS 根池](#)

```
boot_device device eeprom
```

boot_device 指定 JumpStart 程序安装根 (*/*) 文件系统的设备和系统的引导设备。*boot_device* 必须匹配任何指定根 (*/*) 文件系统的 *filesys* 关键字和 *root_device* 关键字。

如果没有在配置文件中指定 `boot_device` 关键字，则缺省情况下，在安装过程中将指定以下 `boot_device` 关键字：

`boot_device any update`

device 请使用以下值之一。

SPARC: `cw txdy sz` 或 `cx dysz` JumpStart 程序放置根 (/) 文件系统的磁盘片，例如 `c0t0d0s0`。

x86: `cw txdy` 或 `cx dy` JumpStart 程序放置根 (/) 文件系统的磁盘，例如 `c0d0`。

`existing` JumpStart 程序将根 (/) 文件系统放置在系统的现有引导设备上。

`any` JumpStart 程序选择放置根 (/) 文件系统的位置。JumpStart 程序将尝试使用系统的现有引导设备。如有必要，JumpStart 程序可能会选择其他引导设备。

eeprom 选择以便更新或保存系统的 EEPROM。

如果更改系统的当前引导设备，*eeprom* 值还允许您更新系统的 EEPROM。通过更新系统的 EEPROM，系统可以从新引导设备自动引导。

注 - x86: 必须指定 `preserve` 值。

`update` JumpStart 程序将系统的 EEPROM 更新到指定的引导设备，以便已安装的系统自动从中引导。

`preserve` 未更改系统的 EEPROM 中的引导设备值。如果指定了新引导设备而没有更改系统的 EEPROM，那么您需要手动更改系统的 EEPROM，以便它能够从新引导设备自动引导。

示例 8-8 `boot_device` 配置文件关键字

```
boot_device c0t0d0s2 update
```


bootenv 配置文件关键字 (UFS 和 ZFS)

注 – bootenv 关键字可用于 UFS 文件系统安装或 ZFS 根池安装。对于 ZFS 安装，用法有所不同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字的完整列表，请参见表 8-2
- 有关在安装 ZFS 根池时如何使用 bootenv 关键字的说明，请参见第 151 页中的“用于 ZFS 根 (/) 文件系统的 JumpStart 关键字 (参考)”

```
bootenv createbe bename new_BE_name filesystem mountpoint:device:fs_options
[filesystem...]
```

bootenv createbe 关键字允许您在安装 Solaris OS 的同时快速创建空且非活动的引导环境。至少，您必须创建根 (/) 文件系统。系统会保留指定文件系统的片，但是不会复制该文件系统；会命名引导环境，但在使用 Solaris Flash 归档文件安装之前不会真正创建该引导环境。使用归档文件安装空引导环境时，文件系统会被安装在保留的片上。下面列出了 *bename* 和 *filename* 的值。

bename new_BE_name

bename 指定了要创建的新引导环境的名称。*new_BE_name* 不能超过 30 个字符，只能包含字母数字字符，不能包含多字节字符。名称在系统中必须是唯一的。

filesystem mountpoint: device:fs_options

filesystem 确定将在新引导环境中创建的文件系统的类型和数量。至少必须定义一个包含根 (/) 文件系统的片。文件系统可位于同一个磁盘上，也可以分布在多个磁盘上。

- *mountpoint* 可以是任何有效的挂载点，也可以是表示交换片的 - (连字符)。
- 首次引导正在安装的操作系统的时，*device* 必须处于可用状态。此设备与 JumpStart 特殊存储设备 (例如 *free*) 无关，并且不能是 Solaris 卷管理器卷或 Veritas 卷管理器卷。*device* 是磁盘设备的名称，格式为 */dev/dsk/cwt:xd ysz*。
- *fs_options* 可以是以下内容之一：
 - *ufs*，表示 UFS 文件系统。
 - *swap*，表示交换文件系统。交换挂载点必须为 - (连字符)。

有关配置文件示例和使用此关键字的背景，请参见以下参考资料：

有关配置文件的示例

示例 3-11

有关使用能够创建、升级和激活非活动引导环境的 Solaris Live Upgrade 的背景

《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 2 章“Solaris Live Upgrade (概述)”

有关使用 Solaris Flash 归档文件的背景

《Solaris 10 10/08 安装指南：Solaris Flash 归档文件（创建和安装）》中的第 1 章“Solaris Flash（概述）”

client_arch 配置文件关键字

`client_arch karch_value ...`

`client_arch` 指定操作系统服务器将支持与该服务器所用平台组不同的平台组。如果未在配置文件中指定 `client_arch`，那么使用此操作系统服务器的任何无盘客户机都必须包含与该服务器所使用的平台组相同的平台组。您必须指定希望操作系统服务器支持的每个平台组。

`karch_value` 的有效值是 `sun4u` 和 `i86pc`。有关平台名称和各种系统的详细列表，请参见 <http://docs.sun.com> 上的 Solaris Sun 硬件平台指南。

注 - 只有当 `system_type` 被指定为 `server` 时，才可以使用 `client_arch`。

client_root 配置文件关键字

`client_root root_size`

`client_root` 定义为每个客户机分配的根空间量 `root_size`（以 MB 为单位）。如果未在服务器的配置文件中指定 `client_root`，那么安装软件将为每个客户机分配 15 MB 的根空间。客户机根空间的大小与 `num_clients` 关键字组合使用，以确定要为 `/export/root` 文件系统保留多少空间。

注 - 只有当 `system_type` 被指定为 `server` 时，才可以使用 `client_root`。

client_swap 配置文件关键字

`client_swap swap_size`

`client_swap` 定义为每个无盘客户机分配的交换空间量 `swap_size`（以 MB 为单位）。如果未在配置文件中指定 `client_swap`，则缺省情况下，将分配 32 MB 的交换空间。

注 - 仅当 `system_type` 被指定为 `server` 时，您才可以使用 `client_swap`。

示例 8-9 `client_swap` 配置文件关键字

以下示例指定每个无盘客户机将具有 64 MB 的交换空间。

```
client_swap 64
```

确定 swap 大小的方式

如果配置文件没有指定 swap 的大小，则 JumpStart 程序将根据系统的物理内存确定交换空间的大小。表 8-5 显示如何在自定义 JumpStart 安装过程中确定 swap 的大小。

表 8-5 确定 swap 的大小

物理内存 (以 MB 为单位)	交换空间 (以 MB 为单位)
16-64	32
64-128	64
128-512	128
大于 512	256

JumpStart 程序将 swap 的大小设置为不超过 swap 所在磁盘的 20%。如果放置其他文件系统后，该磁盘上包含空闲空间，则分配结果将有所不同。如果有空闲空间，则 JumpStart 程序会将空闲空间分配给 swap，如果可能，将分配表 8-5 中显示的大小。

注 - 物理内存和交换空间之和至少应为 32 MB。

cluster 配置文件关键字 (添加软件组) (UFS 和 ZFS)

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 cluster 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关如何在 ZFS 根池安装的配置文件中使用 cluster 关键字的说明，请参见第 148 页中的“ZFS 根池的 JumpStart 配置文件示例”

`cluster group_name`

cluster 指定要添加到系统中的软件组。

软件组是包含簇和软件包集合的元簇。通过使用 cluster 关键字和 group_name 变量安装软件组。只能在初始安装中安装 cluster 关键字。该 cluster 关键字指向位于 clustertoc(4) 文件中的元簇。

簇是名为 SUNWname 的软件包的集合。使用 cluster 关键字和 cluster_name 变量可以安装簇。可以在初始安装或升级时从软件组 (元簇) 添加或删除簇。

下表列出了每个软件组的 group_name。

软件组	<i>group_name</i>
精简网络支持软件组	SUNWCrnet
核心系统支持软件组	SUNWCreq
最终用户 Solaris 软件组	SUNWCuser
开发者 Solaris 软件组	SUNWCprog
完整 Solaris 软件组	SUNWCall
完整 Solaris 软件组加 OEM 支持	SUNWCxall

将应用以下限制：

- 在一个配置文件中只能指定一个软件组。
- 软件组必须在其他 `cluster` 项和 `package` 项之前指定。
- 如果未在配置文件中使用 `cluster` 指定软件组，那么将在系统上安装最终用户软件组 `SUNWCuser`。

有关软件组的更多信息，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的“软件组的磁盘空间建议”。

cluster 配置文件关键字（添加或删除簇）（UFS 和 ZFS）

`cluster cluster_name add_delete_switch`

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 `cluster` 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2。
- 有关如何在 ZFS 根池安装的配置文件中使用的 `cluster` 关键字的说明，请参见第 148 页中的“ZFS 根池的 JumpStart 配置文件示例”

`cluster` 指定了在系统上要安装的软件组中是添加簇还是删除簇。

cluster_name 簇名称的格式必须为 `SUNWCname`。

add_delete_switch 可选关键字，用于指明是添加还是删除指定的簇。使用 `add` 或 `delete` 值。如果您没有指定 `add` 或 `delete`，则在缺省情况下使用 `add`。

在升级期间使用 `cluster` 时，将会出现下列情况：

- 系统上已有的所有簇都将被自动升级。
- 如果指定 `cluster_name add`，且 `cluster_name` 未安装在系统上，那么将安装该簇。

- 如果指定 `cluster_name delete`，且 `cluster_name` 已经安装在系统上，那么将在升级开始之前删除该软件包。

注 - 软件组是包含簇和软件包集合的元簇。通过使用 `cluster` 关键字和 `group_name` 变量安装软件组。只能在初始安装中安装 `cluster` 关键字。该 `cluster` 关键字指向位于 `clustertoc(4)` 文件中的元簇。

簇是软件包的集合。可以将簇组合到一起形成软件组（元簇）。簇的名称始终采用 `SUNW<name>` 的格式。使用 `cluster` 关键字和 `cluster_name` 变量可以安装簇。可以在初始安装或升级时从软件组（元簇）添加或删除簇。

dontuse 配置文件关键字（UFS 和 ZFS）

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 `dontuse` 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关执行 ZFS 安装的信息，请参见第 9 章，使用 [JumpStart 安装 ZFS 根池](#)

`dontuse disk_name ...`

缺省情况下，当指定 `partitioning default` 时，JumpStart 程序使用系统上所有可操作的磁盘。`dontuse` 用于指定不希望 JumpStart 程序使用的一块或多块磁盘。必须按以下格式指定 `disk_name`：`cxt ydz` 或 `cydz`，例如，`c0t0d0`。

注 - 不能在同一配置文件中同时指定 `dontuse` 关键字和 `usedisk` 关键字。

x86: fdisk 配置文件关键字（UFS 和 ZFS）

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 `fdisk` 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关执行 ZFS 安装的信息，请参见第 9 章，使用 [JumpStart 安装 ZFS 根池](#)

`fdisk disk_name type size`

`fdisk` 定义了如何在基于 x86 的系统上设置 `fdisk` 分区。您可以多次指定 `fdisk`。当 `fdisk` 对基于 x86 的系统进行分区时，将发生以下情况：

- 保留磁盘上的所有 `fdisk` 分区，除非您通过将 `size` 的值指定为 `delete` 或 `0` 来删除具有 `fdisk` 关键字的分区。同时，如果 `size` 被设置为 `all`，则所有现有的 `fdisk` 分区将被删除。
- 包含根 (`/`) 文件系统的 Solaris `fdisk` 分区始终被指定为磁盘上的活动分区。

注-缺省情况下，系统从活动分区进行引导。

- 如果未在配置文件中指定 `fdisk` 关键字，则缺省情况下，安装期间将使用以下 `fdisk` 关键字。

```
fdisk all solaris maxfree
```

- 系统按各个 `fdisk` 项在配置文件中的先后顺序对其进行处理。

`disk_name` 使用以下值可指定在哪个位置创建或删除 `fdisk` 分区：

- `cxt ydz` 或 `c ydz`—特定磁盘，例如 `0t3d0`。
- `rootdisk`—包含系统根磁盘（进行安装的位置）的值的变量。根磁盘由 JumpStart 程序确定，如第 140 页中的“确定系统根磁盘的方式”中所述。
- `all`—所有选中的磁盘。

`type` 使用以下值可指定要在指定磁盘上创建或删除的 `fdisk` 分区的类型：

- `solaris`—Solaris `fdisk` 分区（SUNIXOS `fdisk` 类型）。
- `dosprimary`—主 DOS `fdisk` 分区的别名，不适用于为数据 DOS 扩展或保留的 `fdisk` 分区。通过将 `size` 的值指定为 `delete` 来删除 `fdisk` 分区时，`dosprimary` 是 `DOSHUGE`、`DOSOS12` 和 `DOSOS16` `fdisk` 类型的别名。创建 `fdisk` 分区时，`dosprimary` 是 `DOSHUGE` `fdisk` 分区的别名。
- `DDD`—整数 `fdisk` 分区。`DDD` 是 1 和 255（含 1 和 255）之间的一个整数。

注-仅当 `size` 为 `delete` 时，您才可以指定此值。

- `0xHH`—十六进制 `fdisk` 分区。`HH` 是 01 到 FF 之间的十六进制数。

注-仅当 `size` 为 `delete` 时，您才可以指定此值。

下表显示了用于某些 `fdisk` 类型的整数和十六进制数。

<i>fdisk</i> 类型	<i>DDD</i>	<i>HH</i>
DOSOS12	1	01
PCIXOS	2	02
DOSOS16	4	04
EXTDOS	5	05
DOSHUGE	6	06
DOSDATA	86	56
OTHEROS	98	62
UNIXOS	99	63

size

请使用以下值之一：

- *DDD*—在指定磁盘上创建大小为 *DDD* 的 *fdisk* 分区，以 MB 为单位。*DDD* 必须是一个整数，JumpStart 程序自动将其向上舍入为最接近的柱面界限。指定 0 与指定 *delete* 的效果是相同的。
- *all*—在整个磁盘上创建 *fdisk* 分区，所有的现有 *fdisk* 分区都将被删除。

仅适用于 x86—仅当 *type* 为 *solaris* 时，您才可以指定 *all* 值。

- *maxfree*—在指定磁盘上的最大连续可用空间中创建 *fdisk* 分区。如果磁盘上已存在指定 *type* 的 *fdisk* 分区，则使用现有的 *fdisk* 分区。而不是在磁盘上创建新的 *fdisk* 分区。

仅适用于 x86—磁盘必须包含至少一个未使用的 *fdisk* 分区。此外，磁盘必须有空闲空间，否则安装将失败。只有当 *type* 为 *solaris* 或 *dosprimary* 时，才能指定 *maxfree* 值。

- *delete*—删除指定磁盘上所有指定 *type* 的 *fdisk* 分区。

filesystems 配置文件关键字（挂载远程文件系统）（UFS 和 ZFS）

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 `filesystems` 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关执行 ZFS 安装的信息，请参见第 9 章，使用 `JumpStart` 安装 ZFS 根池

```
filesystems server:path server_address mount_pt_name mount_options
```

通过将 `filesystems` 和所列的值一起使用，`JumpStart` 程序将已安装的系统设置为在引导系统时自动挂载远程文件系统。您可以多次指定 `filesystems`。

<code>server</code>	远程文件系统所在的服务器的名称，后面为一个冒号。
<code>path</code>	远程文件系统的挂载点名称。例如， <code>/usr</code> 或 <code>/export/home</code> 。
<code>server_address</code>	在 <code>server:path</code> 中指定的服务器的 IP 地址。如果网络上未运行命名服务，那么可以使用 <code>server_address</code> 值来填充带有服务器的主机名和 IP 地址的 <code>/etc/hosts</code> 文件。如果未指定服务器的 IP 地址，那么您必须指定一个减号 (-)。例如，如果网络上已经运行命名服务，那么您不需要指定服务器的 IP 地址。
<code>mount_pt_name</code>	要挂载远程文件系统的挂载点的名称。
<code>mount_options</code>	一个或多个挂载选项，与 <code>mount(1M)</code> 命令的 <code>-o</code> 选项相同。挂载选项被添加到指定的 <code>mount_pt_name</code> 的 <code>/etc/vfstab</code> 项。

注 - 如果需要指定多个挂载选项，这些挂载选项必须用逗号分隔，并且各选项之间不留空格，例如 `ro,quota`。

示例 8-10 `filesystems` 配置文件关键字

```
filesystems sherlock:/export/home/user2 - /home
```

filesystems 配置文件关键字（创建本地文件系统）

```
filesystems slice size file_system optional_parameters
```

通过将 `filesystems` 和所列的值一起使用，`JumpStart` 程序在安装期间创建本地文件系统。您可以多次指定 `filesystems`。

<code>slice</code>	请使用以下值之一：
<code>any</code>	<code>JumpStart</code> 程序将文件系统放置于任意磁盘。

注 - 当 `size` 的值为 `existing`、`all`、`free`、`start:size` 或 `ignore` 时，无法指定 `any`。

`cw txdy sz` 或 `cx dysz` JumpStart 程序放置文件系统的磁盘片，例如 `c0t0d0s0` 或 `c0d0s0`。

`rootdisk.sn` 包含系统根磁盘的值的变量，根磁盘由 JumpStart 程序确定，如第 140 页中的“确定系统根磁盘的方式”中所述。`sn` 后缀表示该磁盘上的特定片。

注 - 根磁盘由 JumpStart 程序确定，根磁盘可确定将要安装 OS 的位置。`rules` 文件使用探测关键字 "rootdisk"，但是此关键字在 `rules` 文件中的使用方式与在 JumpStart 配置文件中的使用方式不同。在 `rules` 文件中无法使用探测关键字 "rootdisk" 来设置安装位置。探测关键字 `rootdisk` 确定安装时的引导位置。请参见表 8-10。

size

请使用以下值之一：

`num` 将文件系统的大小设置为 `num`，以 MB 为单位。

`existing` 使用现有文件系统的当前大小。

注 - 使用 `existing` 值时，您可以通过将 `file_system` 指定为不同的 `mount_pt_name` 来更改现有片的名称。

`auto` 根据所选软件，自动确定文件系统的大小。

`all` 指定的 `slice` 将整个磁盘用于该文件系统。指定 `all` 值时，系统无法将其他文件系统放置于指定的磁盘上。

`free` 将磁盘上其余的未用空间用于该文件系统。

注 - 如果将 `free` 用作 `filesys` 的值，那么 `filesys` 项必须是配置文件中的最后一项。

	<i>start:size</i>	对文件系统进行明确地分区。 <i>start</i> 是片的起始柱面。 <i>size</i> 是片的柱面数。
<i>file_system</i>		<i>file_system</i> 值是可选的，并且在 <i>slice</i> 被指定为 <code>any</code> 或 <code>cwtxdysz</code> 的情况下使用。缺省情况下，如果未指定 <i>file_system</i> ，那么将设置为 <code>unnamed</code> 。如果设置了 <code>unnamed</code> ，则不能指定 <i>optional_parameters</i> 值。请使用以下值之一：
	<i>mount_pt_name</i>	文件系统的挂载点名称，例如 <code>/var</code> 。
	<code>swap</code>	将指定的 <i>slice</i> 用作 <code>swap</code> 。
	<code>overlap</code>	将指定的 <i>slice</i> 定义为表示磁盘区域。VTOC 值是 <code>V_BACKUP</code> 。缺省情况下，片 2 是重叠片，表示整个磁盘。

注 - 仅当 *size* 为 `existing`、`all` 或 *start:size* 时，您才可以指定 `overlap`。

	<code>unnamed</code>	指定的 <i>slice</i> 被定义为原始片，因此该 <i>slice</i> 没有挂载点名称。缺省情况下，如果未指定 <i>file_system</i> ，那么将使用 <code>unnamed</code> 。
	<code>ignore</code>	指定的 <i>slice</i> 不由 <code>JumpStart</code> 程序来使用或识别。您可以使用此选项以指定希望在安装期间忽略的磁盘上的某个文件系统。 <code>JumpStart</code> 程序将在同一块磁盘上创建一个新的同名文件系统。仅当指定了 <code>partitioning existing</code> 时，您才可以使使用 <code>ignore</code> 。
<i>optional_parameters</i>		请使用以下值之一：
	<code>preserve</code>	保存指定的 <i>slice</i> 上的文件系统。

注 - 只有当 *size* 为 `existing` 并且 *slice* 为 `cwtxdysz` 时，才可以指定 `preserve`。

mount_options 一个或多个挂载选项，与 `mount(1M)` 命令的 `-o` 选项相同。挂载选项被添加到指定的 *mount_pt_name* 的 `/etc/vfstab` 项。

注 - 如果需要指定多个挂载选项，这些挂载选项必须用逗号分隔，并且各选项之间不留空格，例如 `ro,quota`。

filesystems 配置文件关键字（创建 RAID-1 卷）

`filesystems mirror[:name]slice [slice] size file_system optional_parameters`

通过将 `filesystems mirror` 关键字和所列的值一起使用，JumpStart 程序创建了创建镜像的文件系统所需的 RAID-1 卷和 RAID-0 卷。您可以多次指定 `filesystems mirror`，为不同的文件系统创建 RAID-1 卷（镜像）。

注 - `filesystems mirror` 关键字仅用于初始安装。

<i>name</i>	此可选关键字允许您命名 RAID-1 卷（镜像）。镜像名称必须以字母 "d" 开头，后接 0 到 127 之间的数字，例如 <code>d100</code> 。如果您未指定镜像名称，那么自定义 JumpStart 程序将指定一个镜像名称。有关如何命名镜像的原则，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的“自定义 JumpStart 和 Solaris Live Upgrade 的 RAID 卷名要求和原则”。
<i>slice</i>	此值指定磁盘片，自定义 JumpStart 程序将您要复制的文件系统放置在该磁盘片中。该值必须遵循以下格式： <code>cwtxdysz</code> ，例如 <code>c0t0d0s0</code> 或 <code>c0t0d0s5</code> 。自定义 JumpStart 程序将在该片上创建 RAID-0 卷（单片串联），然后创建 RAID-1 卷以镜像该串联。最多可以为两个 RAID-0 卷指定两个片。
<i>size</i>	此值指定文件系统的大小，以 MB 为单位。
<i>file_system</i>	此值指定正在复制的文件系统的大小。自定义 JumpStart 程序使用指定片创建 RAID-1 卷，并在指定文件系统中挂载 RAID-1 卷。除了关键文件系统，如根 (<code>/</code>)、 <code>/usr</code> 和 <code>/var</code> ，您还可以将 <code>swap</code> 指定为文件系统。
<i>optional_parameters</i>	一个或多个挂载选项，与 <code>mount(1M)</code> 命令的 <code>-o</code> 选项相同。挂载选项被添加到指定的 <code>file_system</code> 的 <code>/etc/vfstab</code> 项。如果需要指定多个挂载选项，这些挂载选项必须用逗号分隔，并且各选项之间不留空格，例如 <code>ro,quota</code> 。

有关在安装期间创建镜像文件系统的更多信息，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的第 9 章“在安装过程中创建 RAID-1 卷（镜像）（概述）”。

forced_deployment 配置文件关键字（安装 Solaris Flash 差别归档文件）

forced_deployment

forced_deployment 强制在某个克隆系统（不同于软件所需的系统）上安装 Solaris Flash 差别归档文件。



注意 - 如果使用了 forced_deployment，那么将删除所有的新文件，以便使克隆系统达到预期的状态。如果不能确定是否要删除文件，请使用缺省设置来终止安装以保护新文件。

geo 配置文件关键字（UFS 和 ZFS）

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 geo 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关执行 ZFS 安装的信息，请参见第 9 章，使用 JumpStart 安装 ZFS 根池

geo region

geo 指定升级系统时要在系统上安装或添加的一个或多个区域语言环境。region 指定包含要安装的语言环境的地理区域。下表列出了可为 region 指定的值。

值	说明
N_Africa	北非，包括埃及
C_America	中美洲，包括哥斯达黎加、萨尔瓦多、危地马拉、墨西哥、尼加拉瓜和巴拿马
N_America	北美洲，包括加拿大和美国
S_America	南美洲，包括阿根廷、玻利维亚、巴西、智利、哥伦比亚、厄瓜多尔、巴拉圭、秘鲁、乌拉圭和委内瑞拉
Asia	亚洲，包括日本、韩国、中华人民共和国、台湾地区和泰国
Ausi	澳洲，包括澳大利亚和新西兰
C_Europe	中欧，包括奥地利、捷克共和国、德国、匈牙利、波兰、斯洛伐克和瑞士

值	说明
E_Europe	东欧，包括阿尔巴尼亚、波斯尼亚、保加利亚、克罗地亚、爱沙尼亚、拉脱维亚、立陶宛、马其顿、罗马尼亚、俄罗斯、塞尔维亚、斯洛文尼亚和土耳其
N_Europe	北欧，包括丹麦、芬兰、冰岛、挪威和瑞典
S_Europe	南欧，包括希腊、意大利、葡萄牙和西班牙
W_Europe	西欧，包括比利时、法国、英国、爱尔兰和荷兰
M_East	中东，包括以色列

《国际语言环境指南》中给出了语言环境组成部分的值的完整列表，这些语言环境值构成了以前列出的每个区域语言环境。

注 - 您可以为需要添加到系统中的每种语言环境指定一个 geo 关键字。

install_type 配置文件关键字 (UFS 和 ZFS)

注 - install_type 关键字可以用于 UFS 文件系统安装或 ZFS 根池安装。对于 ZFS 安装，其用法有一定的限制。对于 ZFS 安装，只能使用 initial_install 选项。

- 如果要将 UFS 文件系统迁移到 ZFS 根池或升级 ZFS 根池，则必须使用 Solaris Live Upgrade。请参见《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 11 章“Solaris Live Upgrade 和 ZFS（概述）”。
- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关在安装 ZFS 根池时如何使用 install_type 关键字的描述，请参见第 151 页中的“用于 ZFS 根 (/) 文件系统的 JumpStart 关键字（参考）”。

install_type initial_upgrade_flash_switch

install_type 定义了是要在系统上清除然后安装新的 Solaris OS、升级系统上的现有 Solaris OS，还是要在系统上安装 Solaris Flash 归档文件。

注 - 您必须在配置文件中指定 install_type，并且 install_type 必须是每个配置文件中的第一个配置文件关键字。

必须为 initial_upgrade_flash_switch 使用下列选项之一：

initial_install 指定执行 Solaris OS 的初始安装

upgrade 指定执行 Solaris OS 的升级

<code>flash_install</code>	指定安装将覆写所有文件的 Solaris Flash 归档文件
<code>flash_update</code>	指定安装仅覆写指定文件的 Solaris Flash 差别归档文件

注 - 某些配置文件关键字只能用于 `initial_install` 选项。某些配置文件关键字只能用于 `upgrade` 选项。某些配置文件关键字只能用于 `flash_install` 选项。

layout_constraint 配置文件关键字

`layout_constraint slice constraint minimum_size`

`layout_constraint` 指定了在升级期间，自动布局由于空间问题而需要重新分配空间时，该自动布局对文件系统的约束。

限制	说明
该关键字仅与升级选项一起使用。	需要重新分配磁盘空间时，只可将 <code>layout_constraint</code> 用于升级选项。
如果未指定 <code>layout_constraint</code> 关键字	JumpStart 程序将按照以下方式设置磁盘布局： <ul style="list-style-type: none"> ■ 将需要更多升级空间的文件系统标记为可更改的。 ■ 将与需要更多空间的文件系统位于同一块磁盘上的文件系统且由 <code>/etc/vfstab</code> 文件挂载的文件系统标记为可更改的。 ■ 将剩余的文件系统标记为固定的，因为自动布局不能更改这些文件系统。
如果指定一个或多个 <code>layout_constraint</code> 关键字	JumpStart 程序将按照以下方式设置磁盘布局： <ul style="list-style-type: none"> ■ 将需要更多升级空间的文件系统标记为可更改的。 ■ 使用指定的约束标记为其指定了 <code>layout_constraint</code> 关键字的文件系统。 ■ 将其余的文件系统标记为固定的。
如果未将文件系统标记为可更改的	您不能更改需要更多升级空间的文件系统上的约束，因为这些文件系统一定会被标记为可更改的。您可以使用 <code>layout_constraint</code> 关键字更改需要更多升级空间的文件系统上的 <code>minimum_size</code> 值。
如果文件系统需要更多升级空间	为帮助自动布局重新分配空间，请将更多文件选择为可更改的或可移动的，尤其是那些与需要更多升级空间的文件系统位于同一块磁盘上的文件系统。

`slice` 指定在其上指定约束的文件系统的磁盘片。必须按以下格式指定系统的磁盘片：`c wtxd ysz` 或 `c xdyz z`。

`constraint` 请为指定的文件系统使用以下约束之一：

<code>changeable</code>	<p>自动布局可将文件系统移动到另一个位置，并且可以更改该文件系统的大小。只能在由 <code>/etc/vfstab</code> 文件挂载的文件系统上指定 <code>changeable</code> 约束。可以通过指定 <code>minimum_size</code> 值来更改文件系统的大小。</p> <p>将一个文件系统标记为可更改的，并且未指定 <code>minimum_size</code> 时，文件系统的最小值将被设置为比所需的最小值大 10%。例如，如果文件系统的最小值是 100 MB，那么更改后的大小则是 110 MB。如果指定了 <code>minimum_size</code>，那么会将剩余的所有空闲空间（原始大小减去最小大小）用于其他文件系统。</p>
<code>movable</code>	<p>自动布局可以将该文件系统移动到同一块磁盘或其他磁盘上的另一个片中。文件系统的大小保持不变。</p>
<code>available</code>	<p>自动布局可以使用文件系统上的所有空间来重新分配空间。文件系统中的所有数据都将丢失。<code>available</code> 约束只能在不是通过 <code>/etc/vfstab</code> 文件挂载的文件系统上指定。</p>
<code>collapse</code>	<p>自动布局将指定的文件系统移动和折叠到父文件系统中。作为升级的一部分，您可以使用 <code>collapse</code> 选项来减少系统上的文件系统数。例如，如果系统上有 <code>/usr</code> 和 <code>/usr/share</code> 两个文件系统，那么折叠 <code>/usr/share</code> 文件系统可以将其移动到父文件系统 <code>/usr</code>。您只能在由 <code>/etc/vfstab</code> 文件挂载的文件系统上指定 <code>collapse</code> 约束。</p>
<code>minimum_size</code>	<p>指定自动布局重新分配空间后，文件系统的大小。<code>minimum_size</code> 选项允许您更改文件系统的大小。如果将未分配的空间添加到文件系统中，那么文件系统的大小可能会更大。但是，该大小永远不会小于您指定的值。<code>minimum_size</code> 值是可选的。仅当已将文件系统标记为“可更改的”时才能使用该值，而且最小值不能小于文件系统用于现有文件系统内容所需的大小。</p>

示例 8-11 `layout_constraint` 配置文件关键字

```
layout_constraint c0t3d0s1 changeable 200

layout_constraint c0t3d0s4 movable

layout_constraint c0t3d1s3 available

layout_constraint c0t2d0s1 collapse
```

local_customization 配置文件关键字（安装 Solaris Flash 归档文件）

local_customization *local_directory*

在克隆系统上安装 Solaris Flash 归档文件之前，可以创建自定义脚本以保留克隆系统上的本地配置。local_customization 关键字指定了保存这些脚本的目录。local_directory 是克隆系统上指向脚本的路径。

有关 predeployment 脚本和 postdeployment 脚本的信息，请参见《Solaris 10 10/08 安装指南：Solaris Flash 归档文件（创建和安装）》中的“创建自定义脚本”。

locale 配置文件关键字（UFS 和 ZFS）

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 locale 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关执行 ZFS 安装的信息，请参见第 9 章，使用 JumpStart 安装 ZFS 根池

locale *locale_name*

注 - locale 既可以与初始安装选项一起使用，也可以与升级选项一起使用。

locale 指定了在升级时针对指定的 *locale_name* 安装或添加的语言环境软件包。locale_name 值与那些用于 \$LANG 环境变量的值相同。《国际语言环境指南》中包含有效语言环境值的列表。

当使用 locale 关键字时，请考虑以下情况：

- 如果已经预配置了一种缺省语言环境，那么系统将自动安装该语言环境。缺省情况下，将安装语言环境为英文的软件包。
- 您可以为需要添加到系统中的每种语言环境指定一个 locale 关键字。

metadb 配置文件关键字（创建状态数据库副本）

metadb *slice* [*size size-in-blocks*] [*count number-of-replicas*]

metadb 关键字允许您在自定义 JumpStart 安装期间创建 Solaris 卷管理器状态数据库副本 (mediates)。您可以在配置文件中多次使用 metadb 关键字，以在不同的磁盘片上创建状态数据库副本。

<i>slice</i>	您必须指定希望自定义 JumpStart 程序在其上放置状态数据库副本的磁盘片。 <i>slice</i> 的值必须遵循以下格式 <i>cwtxdysz</i> 。
<i>size size-in-blocks</i>	<i>size</i> 可选关键字允许您指定要创建的状态数据库副本的大小，以块为单位。如果未指定 <i>size</i> 值，自定义 JumpStart 程序将使用缺省大小 8192 块作为状态数据库副本的大小。
<i>count number-of-replicas</i>	您可以指定要创建的状态数据库副本的数量，方法是在配置文件中设置可选的 <i>count</i> 关键字值。缺省情况下，如果未指定 <i>count</i> 值，那么自定义 JumpStart 程序将创建三个状态数据库副本。

有关在安装期间创建 Solaris 卷管理器状态数据库副本的更多信息，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的“状态数据库副本原则和要求”。

no_content_check 配置文件关键字（安装 Solaris Flash 归档文件）

`no_content_check`

使用 Solaris Flash 差别归档文件安装克隆系统时，您可以使用 `no_content_check` 关键字以忽略逐个文件验证的过程。逐个文件验证的过程可以确保克隆系统是主系统的副本。除非能够确定克隆系统是初始主系统的副本，否则请避免使用该关键字。



注意 – 如果使用了 `no_content_check`，那么所有的新文件都将被删除，以便使克隆系统达到期望的状态。如果不能确定是否要删除文件，请使用缺省设置来终止安装以保护新文件。

有关安装 Solaris Flash 差别归档文件的信息，请参见第 77 页中的“准备使用自定义 JumpStart 安装来安装 Solaris Flash 归档文件”。

no_master_check 配置文件关键字（安装 Solaris Flash 归档文件）

`no_master_check`

使用 Solaris Flash 差别归档文件安装克隆系统时，可以使用 `no_master_check` 关键字来忽略旨在确定克隆系统是从原始主系统生成的检查过程。除非能够确定克隆系统是初始主系统的副本，否则请避免使用该关键字。

有关安装 Solaris Flash 差别归档文件的信息，请参见第 77 页中的“准备使用自定义 JumpStart 安装来安装 Solaris Flash 归档文件”。

num_clients 配置文件关键字

`num_clients client_num`

安装服务器时，会为每个无盘客户机的根 (/) 和 swap 文件系统分配空间。num_clients 定义了服务器支持的无盘客户机的数目 client_num。如果未在配置文件中指定 num_clients，则缺省情况下，将分配五个无盘客户机。

注 - 只有当 system_type 被指定为 server 时，才可以使用 num_clients。

package 配置文件关键字 (UFS 和 ZFS)

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 package 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关执行 ZFS 安装的信息，请参见第 9 章，使用 JumpStart 安装 ZFS 根池

```
package package_name [add [retrieval_type location] | delete]
```

package 既可以与初始安装选项一起使用，也可以与升级选项一起使用。使用 package 关键字可执行下列操作：

- 从要安装的 Solaris 分发将软件包添加到软件组。
- 从正在安装的分发之外的其他位置将软件包添加到软件组。
- 从要安装或升级的软件组中排除或删除软件包。
- 在安装 Solaris Flash 归档文件时，从正在安装的分发之外的其他位置添加软件包。

package_name 以 SUNWname 格式指定软件包名称。要查看有关已安装的系统上的软件包及其名称的详细信息，请使用 pkginfo -l 命令。

add | delete 指定要添加还是删除指定软件包。如果不指定 add 或 delete，则在缺省情况下使用 add。

注 - 通过将另一软件包项添加到配置文件并省略位置，可添加多个软件包。如果将位置留空，则所有后续软件包都使用前一个软件包的位置。

[*retrieval_type location*] 指定从正在安装的 Solaris 分发之外的其他位置添加一个或多个软件包。*retrieval_type* 和 *location* 的值取决于软件包存储的位置。以下各节介绍了可用于 *retrieval_type* 和 *location* 的值，以及如何使用 package_name 关键字的示例。

NFS 服务器上存储的软件包

如果软件包存储在 NFS 服务器上，请为 package 关键字使用下列语法之一。

```
package package_name add nfs server_name:/path [retry n]
package package_name add nfs://server_name:/path [retry n]
```

package_name 以 *SUNWname* 格式指定软件包名称。要查看有关已安装的系统上的软件包及其名称的详细信息，请使用 `pkginfo -l` 命令。

server_name 指定存储软件包的服务器的名称。

path 指定所指定的服务器上的软件包目录的位置。如果路径包含 `$HOST`，则 `$HOST` 由正在安装的主机系统名称替换。

retry n 是可选关键字。*n* 是安装过程尝试挂载目录的最大次数。

示例 8-12 使用 NFS 添加软件包

在本示例中，`package` 配置文件关键字将添加 NFS 位置 `nfs://golden/packages/Solaris_10/` 的 `SUNWnew` 软件包。如果挂载失败，将尝试进行五次 NFS 挂载。

```
package SUNWnew add nfs golden:/packages/Solaris_10 retry 5
```

HTTP 服务器上存储的软件包

如果软件包存储在 HTTP 服务器上，请为 `package` 关键字使用下列语法之一。

```
package package_name add http://server_name[:port] path optional_keywords
package package_name add http server_name[:port] path optional_keywords
```

package_name 以 *SUNWname* 格式指定软件包名称。要查看有关已安装的系统上的软件包及其名称的详细信息，请使用 `pkginfo -l` 命令。

server_name 指定存储软件包的服务器的名称。

port 指定可选端口。*port* 可以是一个端口号，也可以是在运行时确定其端口号的 TCP 服务的名称。

如果不指定端口，将使用缺省的 HTTP 端口号 80。

path 指定要从指定服务器检索的软件包的位置。使用 HTTP 服务器时，软件包必须为软件包数据流格式。

optional_keywords 指定从 HTTP 服务器检索软件包时使用的可选关键字。

表 8-6 与 HTTP 一起使用的可选 package 关键字

关键字	值定义
<code>timeout min</code>	<p>使用 <code>timeout</code> 关键字，可以以分钟为单位，指定在无法接收到来自 HTTP 服务器的数据的情形下允许等待的最大时间长度。如果发生超时，则连接将被关闭、重新打开并恢复。如果指定 <code>timeout</code> 值为 0（零），则不重新打开连接。</p> <p>如果重新连接发生超时，那么将从软件包开头重试该软件包，并且将放弃在超时时检索到的数据。</p>
<code>proxy host:port</code>	<p><code>proxy</code> 关键字允许您指定代理主机和代理端口。您可以使用代理主机从防火墙的另一侧检索 Solaris 软件包。指定 <code>proxy</code> 关键字时必须提供一个代理端口。</p>

示例 8-13 使用 HTTP 添加软件包

在本示例中，`package` 配置文件关键字将添加 HTTP 位置 `http://package.central/Solaris_10` 的 Solaris 10 目录中列出的所有软件包。如果五分钟后未检索到数据，将重新检索软件包数据。前一个软件包数据被放弃。可使用以下任一格式。

```
package SUNWnew add http package.central/Solaris_10 timeout 5
```

```
package SUNWnew add http://package.central/Solaris_10 timeout 5
```

示例 8-14 通过将 HTTP 与代理端口一起使用添加软件包

在本示例中，`package` 配置文件关键字将添加 HTTP 位置 `http://package.central/Solaris_10` 的 Solaris 10 目录中列出的所有软件包。通过使用 `proxy` 关键字，跨防火墙检索软件包。

```
package SUNWnew add http://package.central/Solaris_10 proxy webcache.east:8080
```

本地设备上存储的软件包

如果在面向文件系统的随机访问设备（例如，软盘或 DVD-ROM）上存储软件包，则可以从本地设备检索 Solaris 软件包。请按照以下语法使用 `package` 关键字。

```
package package_name add local_device device path file_system_type
```

package_name 以 `SUNWname` 格式指定软件包名称。要查看有关已安装的系统上的软件包及其名称的详细信息，请使用 `pkginfo -l` 命令。

device 指定 Solaris 软件包驻留的驱动器的名称。如果设备名称使用了规则路径，那么将直接挂载该设备。如果设备名称未使用规则路径，那么安装实用程序会将 `/dev/dsk/` 添加到其路径中。

<i>path</i>	指定 Solaris 软件包的路径（相对于指定的设备上的根 (/) 文件系统）。
<i>file_system_type</i>	指定设备上文件系统的类型。如果未提供文件系统类型，那么安装实用程序将尝试挂载 UFS 文件系统。如果 UFS 挂载失败，那么安装实用程序将尝试挂载 HSFS 文件系统。

示例 8-15 通过将本地设备与 UFS 文件系统一起使用来添加软件包

在本示例中，`package` 配置文件关键字将添加本地设备 `c0t6d0s0` 的 `/Solaris_10/Product` 目录中的 `SUNWnew` 软件包。这是 UFS 文件系统。

```
package SUNWnew add local_device c0t6d0s0 /Solaris_10/Product ufs
```

示例 8-16 使用本地设备从 HSFS 文件系统添加软件包

在本示例中，`package` 配置文件关键字将添加本地设备 `c0t6d0s0` 的 `/Solaris_10/Product` 目录中的 `SUNWnew` 软件包。这是 HSFS 文件系统。

```
package SUNWnew add local_device c0t6d0s0 /Solaris_10/Product hsfs
```

本地文件上存储的软件包

可从用于引导系统的 `miniroot` 安装软件包。执行自定义 `JumpStart` 安装时，从 DVD、CD 或基于 NFS 的 `miniroot` 引导系统。安装软件从该 `miniroot` 装入和运行。因而，存储在 DVD、CD 或基于 NFS 的 `miniroot` 中的软件包可作为本地文件进行访问。请按照以下语法使用 `package` 关键字。

```
package package_name add local_file path
```

package_name 以 `SUNWname` 格式指定软件包名称。要查看有关已安装的系统上的软件包及其名称的详细信息，请使用 `pkginfo -l` 命令。

path 指定软件包的位置。从 Solaris Software - 1 CD 或者从 Solaris Operating System DVD 引导系统时，路径必须可由该系统作为本地文件访问。从 Solaris Software - 1 CD 或者从 Solaris Operating System DVD 引导系统时，系统不能访问 `/net`。

示例 8-17 使用本地文件添加软件包

在本示例中，`package` 配置文件关键字将添加 `/Solaris_10/Product` 目录中的 `SUNWnew` 软件包。

```
package SUNWnew add local_file /Solaris_10/Product
```

使用 package 关键字时的限制

使用 package 关键字时，请注意下列限制：

- 某些软件包是必需的，不能删除。
- 您不能使用 package 配置文件关键字来单独添加或删除本地化软件包。要添加本地化软件包，请使用 locale 配置文件关键字。
- 不能从 FTP 服务器位置或本地备份（如磁带）检索软件包。
- 不能从备用位置添加正在安装的 Solaris 分发中的软件包。如果指定了 Solaris 分发中的软件包，则该软件包不能有备用位置，以保证与最后安装的系统一致。
- 要在安装时不介入手动操作，必须能够使用 pkgadd 命令安装该软件包。必须使用同一 admin 文件来安装软件组软件包和驻留在另一位置的软件包。
 - 如果 retrieval_type 为 HTTP，那么该软件包必须为流格式。
 - 如果 retrieval_type 是 NFS 服务器、本地设备或本地文件，那么软件包必须遵循标准打包格式，并且目录名与将安装的软件包的名称相同。
 - 如果正在从单独位置安装一个软件包，并且该软件包与当前未安装的另一软件包相关，则不安装该软件包。安装或升级日志文件中将记录一条错误消息。
- 如果正在使用 Solaris Flash 归档文件安装该软件包，请遵循下列指南。
 - 安装的所有软件包必须与归档文件兼容。
 - 如果归档文件中包含一个软件包，那么 JumpStart 将覆写现有软件包。

使用 package 关键字时的升级行为

将 package 用于升级时，JumpStart 程序将执行以下操作：

- 自动升级系统上已有的所有软件包。
- 如果指定 *package_name* add，并且 *package_name* 未安装在系统上，则安装该软件包。
- 如果指定 *package_name* delete，并且 *package_name* 已安装在系统上，则在升级开始之前删除该软件包。
- 如果指定 *package_name* delete，并且 *package_name* 未安装在系统上，同时该软件包是指定要安装的簇的一部分，那么将不安装该软件包。

partitioning 配置文件关键字

partitioning type

partitioning 定义如何在安装过程中将磁盘拆分为片以用于文件系统。

如果未在配置文件中指定 *partitioning*，则缺省情况下，将使用 default 类型的分区。

type 请使用以下值之一：

<code>default</code>	JumpStart 程序选择磁盘并创建要安装指定软件的文件系统（由 <code>filesys</code> 关键字指定的文件系统除外）。首先选择 <code>rootdisk</code> 。如果指定的软件不适合 <code>rootdisk</code> ，那么 JumpStart 程序将使用其他磁盘。
<code>existing</code>	JumpStart 使用系统磁盘上的现有文件系统。保留除 <code>/</code> 、 <code>/usr</code> 、 <code>/usr/openwin</code> 、 <code>/opt</code> 和 <code>/var</code> 以外的所有文件系统。JumpStart 程序使用文件系统超级块中的上一个挂载点字段来确定片所代表的文件系统挂载点。

注 - 同时使用 `filesys` 和 `partitioning existing` 配置文件关键字时，您必须将大小 `size` 设置为 `existing`。

<code>explicit</code>	JumpStart 使用磁盘并创建由 <code>filesys</code> 关键字指定的文件系统。如果只使用 <code>filesys</code> 关键字指定根 (<code>/</code>) 文件系统，那么将在根 (<code>/</code>) 文件系统中安装所有 Solaris 软件。
-----------------------	---

注 - 如果使用了 `explicit` 配置文件值，那么您必须使用 `filesys` 关键字指定要使用的磁盘和要创建的文件系统。

patch 配置文件关键字

`patch patch_id_list | patch_file patch_location optional_keywords]`

<code>patch_id_list</code>	指定要安装的修补程序 ID 号。该列表应为 Solaris 修补程序 ID 的逗号分隔列表。按列表中指定的顺序安装修补程序。不要在逗号后添加空格，例如：112467-01,112765-02。
<code>patch_file</code>	位于 <code>patch_location</code> 中的包含修补程序列表的文件。修补程序按文件中指定的顺序安装。
<code>patch_location</code>	指定修补程序所在的位置。允许的位置如下： <ul style="list-style-type: none"> ▪ NFS 服务器 ▪ HTTP 服务器 ▪ 本地设备 ▪ 本地文件
<code>optional_keywords</code>	可选关键字与存储修补程序的位置相关。以下部分介绍了可用位置和可选关键字。

NFS 服务器上存储的修补程序

如果修补程序存储在 NFS 服务器上，请为 `patch` 关键字使用下列语法之一。

```
patch patch_id_list | patch_file nfs server_name:/patch_directory [retry n]
patch patch_id_list | patch_file nfs://server_name/patch_director [retry n]
```

<i>patch_id_list</i>	指定要安装的修补程序 ID 号。该列表应为 Solaris 修补程序 ID 的逗号分隔列表。按列表中指定的顺序安装修补程序。
<i>patch_file</i>	位于 <i>patch_location</i> 中的包含修补程序列表的文件。修补程序按文件中指定的顺序安装。
<i>server_name</i>	指定存储修补程序的服务器的名称。
<i>patch_directory</i>	指定所指定的服务器上的修补程序目录的位置。修补程序必须为单独的修补程序格式。
<i>retry n</i>	是可选关键字。 <i>n</i> 是安装实用程序尝试挂载目录的最大次数。

示例 8-18 使用 NFS 从有序列表添加修补程序

在本示例中，`patch` 配置文件关键字将添加 NFS 修补程序目录 `nfs://patch_master/Solaris/v10/patches` 的 `patch` 文件中列出的所有修补程序。按 `patch` 中列出的顺序安装修补程序。如果挂载失败，将尝试进行五次 NFS 挂载。

```
patch patch_file nfs://patch_master/Solaris/v10/patches retry 5
```

示例 8-19 使用 NFS 添加修补程序

在本示例中，`patch` 配置文件关键字将添加 `patch_master` 服务器上的修补程序目录 `/Solaris/v10/patches` 中的修补程序 `112467-01` 和 `112765-02`。

```
patch 112467-01,112765-02 nfs patch_master:/Solaris/v10/patches
```

HTTP 服务器上存储的修补程序

如果修补程序存储在 HTTP 服务器上，请为 `patch` 关键字使用下列语法之一。

```
patch patch_id_list | patch_file http://server_name [:port] patch_directory optional_http_keywords
```

```
patch patch_id_list | patch_file http server_name [:port] patch_directory optional_http_keywords
```

<i>patch_id_list</i>	指定要安装的修补程序 ID 号。该列表应为 Solaris 修补程序 ID 的逗号分隔列表。按列表中指定的顺序安装修补程序。不要在逗号后添加空格，例如： <code>112467-01,112765-02</code> 。
<i>patch_file</i>	位于 <i>patch_location</i> 中的包含修补程序列表的文件。修补程序按文件中指定的顺序安装。
<i>server_name</i>	指定存储修补程序的服务器的名称。

<i>port</i>	指定可选端口。 <i>port</i> 可以是一个端口号，也可以是在运行时确定其端口号的 TCP 服务的名称。
	如果不指定端口，将使用缺省的 HTTP 端口号 80。
<i>patch_directory</i>	指定要从指定服务器检索的修补程序目录的位置。在使用 HTTP 服务器时，修补程序必须为 JAR 格式。
<i>optional_keywords</i>	指定从 HTTP 服务器检索修补程序时使用的可选关键字。

表 8-7 与 HTTP 一起使用的可选 patch 关键字

关键字	值定义
<i>timeout min</i>	使用 <i>timeout</i> 关键字，可以以分钟为单位，指定在无法接收到来自 HTTP 服务器的数据的情形下允许等待的最大时间长度。如果发生超时，则连接将被关闭、重新打开并恢复。如果指定 <i>timeout</i> 值为 0（零），则不重新打开连接。 如果重新连接发生超时，那么将从软件包开头重试该软件包，并且将放弃在超时前检索到的数据。
<i>proxy host:port</i>	<i>proxy</i> 关键字允许您指定代理主机和代理端口。您可以使用代理主机从防火墙的另一侧检索 Solaris 软件包。指定 <i>proxy</i> 关键字时必须提供一个代理端口。

示例 8-20 使用 HTTP 从有序列表添加修补程序

在本示例中，*patch* 配置文件关键字添加 HTTP 位置

`http://patch.central/Solaris/v10/patches` 的 *patch_file* 文件中列出的所有修补程序。按照 *patch* 文件中指定的顺序安装修补程序。如果五分钟后未检索到任何数据，那么将重新检索修补程序数据，并且将放弃此前的修补程序数据。

```
patch patch_file http://patch.central/Solaris/v10/patches timeout 5
```

示例 8-21 使用 HTTP 添加修补程序

在本示例中，*patch* 配置文件关键字项将添加修补程序位置

`http://patch_master/Solaris/v10/patches` 中的修补程序 112467-01 和 112765-02。

```
patch 112467-01,112765-02 http://patch.central/Solaris/v10/patches
```

本地设备上存储的修补程序

如果在面向文件系统的随机访问设备（例如，软盘或 DVD-ROM）上存储软件包，则可以从本地设备检索 Solaris 软件包。请按照以下语法使用 *patch* 关键字。

```
patch patch_id_list | patch_file local_device \
device path file_system_type
```

<i>patch_id_list</i>	指定要安装的修补程序 ID 号。该列表应为 Solaris 修补程序 ID 的逗号分隔列表。按列表中指定的顺序安装修补程序。不要在逗号后添加空格，例如：112467-01,112765-02。
<i>patch_file</i>	位于 <i>patch_location</i> 中的包含修补程序列表的文件。修补程序按文件中指定的顺序安装。
<i>device</i>	指定 Solaris 软件包驻留的驱动器的名称。如果设备名称使用了规则路径，那么将直接挂载该设备。如果设备名称未使用规则路径，那么安装实用程序会将 <i>/dev/dsk/</i> 添加到其路径中。
<i>path</i>	指定 Solaris 修补程序的路径（相对于指定的设备上的根 (<i>/</i>) 文件系统）。
<i>file_system_type</i>	指定设备上文件系统的类型。如果未提供文件系统类型，那么安装实用程序将尝试挂载 UFS 文件系统。如果 UFS 挂载失败，那么安装实用程序将尝试挂载 HSFS 文件系统。

示例 8-22 通过使用本地设备按有序列表添加修补程序

在本示例中，*patch* 配置文件关键字将添加本地设备 *c0t6d0s0* 的 */Solaris_10/patches* 目录中 *patch_file* 文件列出的所有修补程序。*patch* 文件确定要安装的修补程序的顺序。

```
patch patch_file c0t6d0s0 /Solaris_10/patches
```

示例 8-23 使用本地设备添加修补程序

在本示例中，*patch* 配置文件关键字将添加本地设备 *c0t6d0s0* 中的修补程序目录 */Solaris_10/patches* 中的修补程序 112467-01 和 112765-02。

```
patch 112467-01,112765-02 local_device c0t6d0s0 /Solaris_10/patches
```

本地文件上存储的修补程序

可从用于引导系统的 *miniroot* 安装修补程序。执行自定义 *JumpStart* 安装时，从 DVD、CD 或基于 NFS 的 *miniroot* 引导系统。安装软件从该 *miniroot* 装入和运行。因而，可将存储在 DVD、CD 或基于 NFS 的 *miniroot* 中的修补程序作为本地文件进行访问。请按照以下语法使用 *patch* 关键字。

```
patch patch_id_list | patch_file local_file patch_directory
```

patch_id_list 指定要安装的修补程序 ID 号。该列表应为 Solaris 修补程序 ID 的逗号分隔列表。按列表中指定的顺序安装修补程序。不要在逗号后添加空格，例如：112467-01,112765-02。

<i>patch_file</i>	位于 <i>patch_location</i> 中的包含修补程序列表的文件。修补程序按文件中指定的顺序安装。
<i>patch_directory</i>	指定修补程序目录的位置。从 Solaris Software - 1 CD 或者从 Solaris Operating System DVD 引导系统时，修补程序目录必须可由该系统作为本地文件访问。从 Solaris Software - 1 CD 或者从 Solaris Operating System DVD 引导系统时，系统不能访问 /net。

示例 8-24 通过使用本地文件按有序列表添加修补程序

在本示例中，`patch` 配置文件关键字将添加 `/Solaris_10/patches` 目录中 `patch_file` 文件列出的所有修补程序。`patch` 文件确定要安装的修补程序的顺序。

```
patch patch_cal_file /Solaris_10/patches
```

示例 8-25 使用本地文件添加修补程序

在本示例中，`patch` 配置文件关键字将添加修补程序目录 `/Solaris_10/patches` 中的修补程序 `112467-01` 和 `112765-02`。

```
patch 112467-01,112765-02 local_file /Solaris_10/patches
```

使用 `patch` 关键字的限制

使用 `patch` 关键字时，请注意下列限制：

- 不能从 FTP 位置或本地备份（如磁带）检索修补程序。
- 不能添加签名的修补程序。
- 必须使用 `patchadd` 命令安装修补程序。
- 如果一个修补程序与当前未安装的修补程序相关，那么将不安装该修补程序。安装或升级日志文件中将记录一条错误消息。
- 必须确定修补程序的正确顺序，才能正确安装修补程序。

`pool` 配置文件关键字（仅 ZFS）

`pool` 关键字定义 ZFS 根池的安装。该池随使用 `cluster` 关键字指定的软件组一起安装。创建新的根池时需要使用 `poolsize`、`swapsize`、`dumpsize` 和 `vdevlist` 选项。

有关 `pool` 关键字和可用于 ZFS 根池的其他关键字的完整描述，请参见第 153 页中的“`pool` 配置文件关键字（仅 ZFS）”。

root_device 配置文件关键字 (UFS 和 ZFS)

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 `root_device` 关键字。对于 ZFS 安装，此关键字的使用仅限于单个系统。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
 - 有关在安装 ZFS 根池时如何使用 `root_device` 关键字的描述，请参见第 151 页中的“用于 ZFS 根 (/) 文件系统的 JumpStart 关键字 (参考)”
-

`root_device slice`

`root_device` 指定系统的根磁盘。第 140 页中的“确定系统根磁盘的方式”中提供了其他信息。

注 - 根磁盘由 JumpStart 程序确定，根磁盘可确定将要安装 OS 的位置。rules 文件使用探测关键字 "rootdisk"，但此关键字的用法与 JumpStart 配置文件中 "rootdisk" 关键字的用法不同。在 rules 文件中无法使用探测关键字 "rootdisk" 来设置安装位置。探测关键字 `rootdisk` 确定安装时的引导位置。请参见表 8-10。

升级系统时，`root_device` 指定了要升级的根 (/) 文件系统和由其 `/etc/vfstab` 文件挂载的文件系统。如果系统上有多个根 (/) 文件系统可以升级，那么必须指定 `root_device`。必须按以下格式指定 `slice`：`cwt xdys z` 或 `cxd ysz`。

使用 `root_device` 关键字时，请考虑以下情况：

- 如果在只有一个磁盘的系统上指定 `root_device`，那么该 `root_device` 必须与磁盘相匹配。此外，所有指定根 (/) 文件系统的 `filesys` 关键字也必须与 `root_device` 相匹配。
- 如果要升级 RAID-1 卷（镜像），那么为 `root_device` 指定的值应该是该镜像的一个面，该镜像的另一面将自动升级。

示例 8-26 `root_device` 配置文件关键字

```
root_device c0t0d0s2
```

确定系统根磁盘的方式

系统的根磁盘是系统上包含根 (/) 文件系统的磁盘。在配置文件中，可以使用 `rootdisk` 变量来代替磁盘名，JumpStart 程序会将其设置为系统的根磁盘。表 8-8 说明 JumpStart 程序如何确定用于安装的系统根磁盘。

注 - JumpStart 程序只在初始安装期间确定系统根磁盘的大小。您不能在升级期间更改系统根磁盘的大小。

表 8-8 JumpStart 确定系统根磁盘的方式（初始安装）

阶段	操作
1	如果在配置文件中指定了 <code>root_device</code> 关键字，那么 JumpStart 程序会将 <code>rootdisk</code> 设置为根设备。
2	如果没有设置 <code>rootdisk</code> ，并且在配置文件中指定了 <code>boot_device</code> 关键字，那么 JumpStart 程序会将 <code>rootdisk</code> 设置为根设备。
3	如果没有设置 <code>rootdisk</code> ，并且在配置文件中指定了 <code>filesys c wtxd ysz size /</code> 项，那么 JumpStart 程序会将 <code>rootdisk</code> 设置为该项中指定的磁盘。
4	如果没有设置 <code>rootdisk</code> ，并且在配置文件中指定了 <code>rootdisk.s n</code> 项，那么 JumpStart 程序将按照内核探测顺序，在系统磁盘的指定片上搜索现有根文件系统。如果找到了一个磁盘，那么 JumpStart 程序会将 <code>rootdisk</code> 设置为找到的磁盘。
5	如果没有设置 <code>rootdisk</code> 并且在配置文件中指定了 <code>partitioning existing</code> ，则 JumpStart 程序按内核探测顺序，在系统磁盘上搜索现有的根文件系统。如果找不到根文件系统或找到多个根文件系统，那么将发生错误。如果找到一个根文件系统，那么 JumpStart 程序会将 <code>rootdisk</code> 设置为找到的磁盘。
6	如果没有设置 <code>rootdisk</code> ，那么 JumpStart 程序会将 <code>rootdisk</code> 设置为安装根 (/) 文件系统的磁盘。

system_type 配置文件关键字

`system_type type_switch`

`system_type` 定义要安装 Solaris OS 的系统的类型。

`type_switch` 代表选项 `standalone` 或 `server`，它用于指示要安装 Solaris 软件的系统类型。如果没有在配置文件中指定 `system_type`，则在缺省情况下，使用 `standalone`。

usedisk 配置文件关键字（UFS 和 ZFS）

注 - 在安装 UFS 文件系统或 ZFS 根池时，可以使用 `usedisk` 关键字。在 UFS 和 ZFS 安装中，此关键字的用法相同。

- 有关可在 UFS 或 ZFS 安装中使用的关键字完整列表，请参见表 8-2
- 有关执行 ZFS 安装的信息，请参见第 9 章，使用 JumpStart 安装 ZFS 根池

`usedisk disk_name ...`

缺省情况下，如果指定了 `partitioning default`，那么 JumpStart 程序将使用系统上所有可操作的磁盘。`usedisk` 配置文件关键字指定了希望 JumpStart 程序使用的一个或多个磁盘。必须按以下格式指定 `disk_name`：`cxt ydz` 或 `c ydz`，例如 `c0t0d0` 或 `c0d0s0`。

如果在配置文件中指定 `usedisk`，则 JumpStart 程序仅使用在 `usedisk` 关键字后面指定的磁盘。

注 - 您不能在同一配置文件中同时指定 `dontuse` 关键字和 `usedisk` 关键字。

自定义 JumpStart 环境变量

您可以在开始脚本和结束脚本中使用环境变量。例如，您可以使用开始脚本提取磁盘大小 `SI_DISKSIZES`，然后根据该脚本提取的实际磁盘大小在系统上安装或不安装特定的软件包。

所收集的关于系统的信息存储在这些环境变量中，通常根据在 `rules` 文件中使用的规则关键字和值来确定是否设置这些变量。

例如，关于系统上已经安装哪个操作系统的信息只有在使用了 `installed` 关键字之后，才能在 `SI_INSTALLED` 中获得。

表 8-9 对这些变量及其值进行了说明。

表 8-9 安装环境变量

环境变量	值
<code>SI_ARCH</code>	安装客户机的硬件体系结构。当在 <code>rules</code> 文件中使用了 <code>arch</code> 关键字时，将设置 <code>SI_ARCH</code> 变量。
<code>SI_BEGIN</code>	开始脚本的名称（如果使用了开始脚本）。
<code>SI_CLASS</code>	用于安装“安装客户机”的配置文件的名称。
<code>SI_DISKLIST</code>	安装客户机上以逗号分隔的磁盘名称列表。当在 <code>rules</code> 文件中使用并匹配了 <code>disksize</code> 关键字时，将设置 <code>SI_DISKLIST</code> 变量。 <code>SI_DISKLIST</code> 和 <code>SI_NUMDISKS</code> 变量用于确定 <code>rootdisk</code> 所使用的物理磁盘。第 140 页中的“确定系统根磁盘的方式”中介绍了 <code>rootdisk</code> 。
<code>SI_DISKSIZES</code>	安装客户机上以逗号分隔的磁盘大小列表。当在 <code>rules</code> 文件中使用并匹配了 <code>disksize</code> 关键字时，将设置 <code>SI_DISKSIZES</code> 变量。
<code>SI_DOMAINNAME</code>	域名。当在 <code>rules</code> 文件中使用并匹配了 <code>domainname</code> 关键字时，将设置 <code>SI_DOMAINNAME</code> 变量。
<code>SI_FINISH</code>	结束脚本的名称（如果使用了结束脚本）。

表 8-9 安装环境变量 (续)

环境变量	值
SI_HOSTADDRESS	安装客户机的 IP 地址。
SI_HOSTNAME	安装客户机的主机名。当在 rules 文件中使用并匹配了 hostname 关键字时，将设置 SI_HOSTNAME 变量。
SI_INSTALLED	具有特定操作系统（例如，Solaris、SunOS 或 System V）的磁盘的设备名称。在 rules 文件中使用和匹配 installed 关键字时，将设置 SI_INSTALLED 变量。SI_INST_OS 和 SI_INST_VER 用于确定 SI_INSTALLED 的值。
SI_INST_OS	操作系统的名称。SI_INST_OS 和 SI_INST_VER 用于确定 SI_INSTALLED 的值。
SI_INST_VER	操作系统的版本。SI_INST_OS 和 SI_INST_VER 用于确定 SI_INSTALLED 的值。
SI_KARCH	安装客户机的内核体系结构。当在 rules 文件中使用并匹配了 karch 关键字时，将设置 SI_KARCH 变量。
SI_MEMSIZE	安装客户机上的物理内存数量。当在 rules 文件中使用并匹配了 memsize 关键字时，将设置 SI_MEMSIZE 变量。
SI_MODEL	安装客户机的模型名。当在 rules 文件中使用并匹配了 model 关键字时，将设置 SI_MODEL 变量。
SI_NETWORK	安装客户机的网络号。当在 rules 文件中使用并匹配了 network 关键字时，将设置 SI_NETWORK 变量。
SI_NUMDISKS	安装客户机上的磁盘数目。当在 rules 文件中使用并匹配了 disksize 关键字时，将设置 SI_NUMDISKS 变量。SI_NUMDISKS 和 SI_DISKLIST 变量用于确定 rootdisk 所使用的物理磁盘。第 140 页中的“确定系统根磁盘的方式”中介绍了 rootdisk。
SI_OSNAME	Solaris 软件映像上的操作系统发行版。例如，如果您是在基于 Solaris Operating System DVD 或 Solaris Software-1 CD 映像上的操作系统版本的系统上安装 Solaris 软件，那么可以在脚本中使用 SI_OSNAME 变量。
SI_ROOTDISK	由逻辑名称 rootdisk 代表的磁盘的设备名。当在 rules 文件中将 disksize 或 installed 关键字设置为 rootdisk 时，将设置 SI_ROOTDISK 变量。SI_ROOTDISK 变量设置安装时从其进行引导的设备。 注 - 在 rules 文件中无法使用探测关键字 "rootdisk" 来设置安装位置。有关在 JumpStart 配置文件中设置的 "rootdisk" 变量的信息，请参见第 140 页中的“确定系统根磁盘的方式”。
SI_ROOTDISKSIZE	由逻辑名称 rootdisk 代表的磁盘的大小。当在 rules 文件中将 disksize 或 installed 关键字设置为 rootdisk 时，将设置 SI_ROOTDISKSIZE 变量。
SI_TOTALDISK	安装客户机上的磁盘空间总量。当在 rules 文件中使用并匹配了 totaldisk 关键字时，将设置 SI_TOTALDISK 变量。

探测关键字和值

表 8-10 说明了每个规则关键字及其等效的探测关键字。

注 - 总是将探测关键字放置在 `rules` 文件的开头或开头附近。

表 8-10 探测关键字的描述

规则关键字	等效的探测关键字	探测关键字的描述
any	无	
arch	arch	确定内核体系结构 (i386 或 SPARC) 并设置 <code>SI_ARCH</code> 。
disksize	disks	以 MB 为单位、按内核探测顺序 <code>c0t3d0s0</code> 、 <code>c0t3d0s1</code> 、 <code>c0t4d0s0</code> 返回系统磁盘的大小。 <code>disksize</code> 设置 <code>SI_DISKLIST</code> 、 <code>SI_DISKSIZE</code> 、 <code>SI_NUMDISKS</code> 和 <code>SI_TOTALDISK</code> 。
domainname	domainname	返回系统的 NIS/NIS+ 域名或空白，并设置 <code>SI_DOMAINNAME</code> 。 <code>domainname</code> 关键字返回 <code>domainname(1M)</code> 的输出。
hostaddress	hostaddress	返回系统的 IP 地址，即 <code>ifconfig(1M) -a</code> 的输出中列出的第一个不是 <code>lo0</code> 的地址，并且设置 <code>SI_HOSTADDRESS</code> 。
hostname	hostname	返回系统的主机名 (它是 <code>uname(1) -n</code> 的输出) 并设置 <code>SI_HOSTNAME</code> 。
installed	installed	返回在系统上安装的 Solaris OS 的版本名称，并设置 <code>SI_ROOTDISK</code> 和 <code>SI_INSTALLED</code> 。 如果 JumpStart 程序找到一个 Solaris 发行版但无法确定版本，则返回的版本为 <code>SystemV</code> 。
karch	karch	返回系统的平台组 (例如 <code>i86pc</code> 、 <code>sun4m</code> 和 <code>sun4</code>)，并设置 <code>SI_KARCH</code> 。有关平台名称的列表，请参见 http://docs.sun.com 上的 Solaris Sun 硬件平台指南。
memsize	memsize	返回系统上的物理内存的大小 (以 MB 为单位)，并设置 <code>SI_MEMSIZE</code> 。
model	model	返回系统的平台名称并设置 <code>SI_MODEL</code> 。有关平台名称的列表，请参见 http://docs.sun.com 上的 Solaris Sun 硬件平台指南。
network	network	返回系统的网络号，它由 JumpStart 程序通过在系统的 IP 地址和子网掩码之间执行逻辑“与”来确定。系统的 IP 地址和子网掩码是从 <code>ifconfig(1M) -a</code> 的输出中列出的第一个不是 <code>lo0</code> 的地址中提取的。 <code>network</code> 关键字设置 <code>SI_NETWORK</code> 。
osname	osname	返回在 CD 或 CD 组上找到的 Solaris OS 的版本和操作系统名称，并设置 <code>SI_OSNAME</code> 。 如果 JumpStart 程序找到一个 Solaris 发行版但无法确定版本，则返回的版本为 <code>SystemV</code> 。
	rootdisk	返回系统根磁盘的名称和大小 (以 MB 为单位)，并设置 <code>SI_ROOTDISK</code> 。

表 8-10 探测关键字的描述 (续)

规则关键字	等效的探测关键字	探测关键字的描述
totaldisk	totaldisk	返回系统上的总磁盘空间（以 MB 为单位），并设置 SI_TOTALDISK。总磁盘空间包括连接到系统上的所有可操作磁盘。

使用 JumpStart 安装 ZFS 根池

本章提供执行 ZFS 根池的 JumpStart 安装时所必需的信息。以下各节分别提供规划信息、配置文件示例和配置文件关键字描述。

- 第 147 页中的“ZFS 根 (/) 文件系统的 JumpStart 安装（概述和规划）”
- 第 148 页中的“ZFS 根池的 JumpStart 配置文件示例”
- 第 151 页中的“用于 ZFS 根 (/) 文件系统的 JumpStart 关键字（参考）”

ZFS 根 (/) 文件系统的 JumpStart 安装（概述和规划）

本章提供用于创建 JumpStart 配置文件以安装 ZFS 根池的信息。

注 - 如果要安装 UFS 根 (/) 文件系统，则所有现有配置文件关键字的作用与在以前版本的 Solaris 中一样。有关 UFS 配置文件关键字的列表，请参见第 8 章，[自定义 JumpStart（参考）](#)。

特定于 ZFS 的配置文件必须包含 pool 关键字。pool 关键字可安装一个新的根池，并在缺省情况下创建一个新的引导环境。您可以提供引导环境的名称，并可以使用现有的 bootenv installbe 关键字和新的 bename 和 dataset 选项来创建单独的 /var 数据集。有些允许在特定于 UFS 的配置文件中使用的关键字不允许在特定于 ZFS 的配置文件中使用，例如，指定创建 UFS 挂载点的那些关键字。

有关 ZFS 总体规划的信息，请参《Solaris 10 10/08 安装指南：规划安装和升级》中的第 6 章“ZFS 根文件系统安装（规划）”。

ZFS 根池的 JumpStart 安装限制

在考虑可引导 ZFS 根池的 JumpStart 安装之前，请记住以下问题。

表 9-1 ZFS 根池的 JumpStart 限制

限制	说明	更多信息
对于 JumpStart 安装，您无法使用现有的 ZFS 存储池来创建一个可引导的 ZFS 根池。	<p>您必须使用类似于以下内容的语法来创建一个新 ZFS 存储池：</p> <pre>pool rpool 20G 4G 4G c0t0d0s0</pre> <p>由于无法使用现有池，因此需要使用完整的 pool 关键字行。bootenv 关键字行是可选的。如果您没有使用 bootenv，则会为您创建缺省的引导环境。例如：</p> <pre>install_type initial_install cluster SUNWCall pool rpool 20G 4g 4g any bootenv installbe bename newBE</pre>	第 153 页中的“pool 配置文件关键字（仅 ZFS）”
您不能使用整个磁盘创建一个池。	<p>必须使用磁盘片而不是整个磁盘来创建池。</p> <p>如果在配置文件中使用时使用整个磁盘（例如 c0t0d0）创建池，则安装将会失败。您将收到一条类似于以下内容的错误消息。</p> <pre>Invalid disk name (c0t0d0)</pre>	
有些允许在特定于 UFS 的配置文件中使用的关键字不允许在特定于 ZFS 的配置文件中使用时，例如，指定创建 UFS 挂载点的那些关键字。		第 103 页中的“配置文件关键字快速参考”
无法使用 JumpStart 进行升级。必须使用 Solaris Live Upgrade	使用 Solaris Live Upgrade，您可以创建当前正在运行的系统的副本。可以升级此副本，然后激活它，使其成为当前运行的系统。	《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 11 章“Solaris Live Upgrade 和 ZFS（概述）”

ZFS 根池的 JumpStart 配置文件示例

本节提供特定于 ZFS 的 JumpStart 配置文件示例。

注 - 为了使 ZFS 根池可升级且可引导，您必须使用磁盘片而不是整个磁盘来创建池。如果在配置文件中用整个磁盘（例如 `c0t0d0`）创建池，则您将收到类似于以下内容的错误消息。

```
Invalid disk name (c0t0d0)
```

示例 9-1 安装镜像的 ZFS 根池

```
install_type initial_install
cluster SUNWCall
pool newpool auto auto auto mirror c0t0d0s0 c0t1d0s0
bootenv installbe bename solaris10_6
```

下表说明了此示例中的一些关键字和值。

<code>install_type initial_install</code>	<code>install_type</code> 关键字是每个配置文件中都必需的。 <code>initial_install</code> 关键字可执行在新的 ZFS 池中安装新 Solaris OS 的初始安装。
<code>cluster</code>	在系统上安装完整分发软件组 <code>SUNWCall</code> 。有关软件组的更多信息，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的“软件组的磁盘空间建议”。
<code>pool</code>	<code>pool</code> 关键字可定义新 ZFS 根池的特征。 <ul style="list-style-type: none"> <code>newpool</code> 定义根池的名称。 <code>auto</code> 自动指定磁盘的大小。此大小由指定磁盘的大小确定。 <code>auto</code> 交换区域的大小由 <code>auto</code> 关键字自动确定。缺省大小为物理内存大小的 1/2，但不小于 512 MB 且不大于 2 GB。您可以使用 <code>size</code> 选项将大小设置成超出此范围的某个值。 <code>auto</code> 转储设备的大小是自动确定的。 <code>mirror</code> 磁盘的镜像配置具有 <code>mirror</code> 关键字和指定为 <code>c0t0d0s0</code> 和 <code>c0t1d0s0</code> 的磁盘片。
<code>bootenv</code>	<code>installbe</code> 可更改在安装期间创建的缺省引导环境的特征。 <ul style="list-style-type: none"> <code>bename</code> 将新的引导环境命名为 <code>solaris10_6</code>。

示例 9-2 自定义 ZFS 根池的磁盘大小

```
install_type initial_install
cluster SUNWCall
pool newpool 80g 2g 2g mirror any any
bootenv installbe bename solaris10_6
```

下表说明了此示例中的一些关键字和值。

<code>install_type initial_install</code>	<code>install_type</code> 关键字是每个配置文件中都必需的。 <code>initial_install</code> 关键字可执行在新的 ZFS 池中安装新 Solaris OS 的初始安装。。
<code>cluster</code>	在系统上安装完整分发软件组 SUNWCall。有关软件组的更多信息，请参见《Solaris 10 10/08 安装指南：规划安装和升级》中的“软件组的磁盘空间建议”。
<code>pool</code>	<code>pool</code> 关键字可定义新 ZFS 根池的特征。 <i>newpool</i> 指定根池的名称。 80g 指定磁盘片的大小。 2g 交换区域和转储卷为 2 G。 mirror 磁盘的镜像配置具有 <code>mirror</code> 关键字和指定为 <code>c0t0d0s0</code> 和 <code>c0t1d0s0</code> 的磁盘片。 镜像配置中的 <code>any</code> 选项查找可足够创建一个 80 G 池的任何两个可用设备。如果找不到这样的两个设备，则安装将会失败。
<code>bootenv</code>	<code>installbe</code> 可更改在安装期间创建的缺省引导环境的特征。 bename 将新的引导环境命名为 <code>solaris10_6</code> 。

示例 9-3 指定操作系统的安装位置

```
install_type initial_install
cluster SUNWCall
root_device c0t0d0s0
pool nrpool auto auto auto rootdisk.s0
bootenv installbe bename bnv dataset /var
```

下表说明了此示例中的一些关键字和值。

示例 9-3 指定操作系统的安装位置	(续)										
<code>install_type initial_install</code>	<code>install_type</code> 关键字是每个配置文件中都必需的。 <code>initial_install</code> 关键字可执行在新的 ZFS 根池中安装新 Solaris OS 的初始安装。										
<code>cluster</code>	在系统上安装完整分发软件组 SUNWCall。有关软件组的更多信息, 请参见《Solaris 10 10/08 安装指南: 规划安装和升级》中的“软件组的磁盘空间建议”。										
<code>root_device</code>	指定要安装操作系统的磁盘片。 <code>c0t0d0s0</code> 可定义用于 OS 的特定磁盘和片。										
<code>pool</code>	<code>pool</code> 关键字可定义新 ZFS 根池的特征。 <table border="0" style="margin-left: 2em;"> <tr> <td><code>nrpool</code></td> <td>定义根池的名称。</td> </tr> <tr> <td><code>auto</code></td> <td>自动指定磁盘大小。此大小由指定磁盘的大小确定。</td> </tr> <tr> <td><code>auto</code></td> <td>交换区域的大小是由 <code>auto</code> 关键字自动确定的。缺省大小为物理内存大小的 1/2, 但不小于 512 MB 且不大于 2 GB。您可以使用 <code>size</code> 选项将大小设置成超出此范围的某个值。</td> </tr> <tr> <td><code>auto</code></td> <td>转储设备的大小是自动确定的。</td> </tr> <tr> <td><code>rootdisk.s0</code></td> <td>用于创建根池的设备被指定为片 0。</td> </tr> </table>	<code>nrpool</code>	定义根池的名称。	<code>auto</code>	自动指定磁盘大小。此大小由指定磁盘的大小确定。	<code>auto</code>	交换区域的大小是由 <code>auto</code> 关键字自动确定的。缺省大小为物理内存大小的 1/2, 但不小于 512 MB 且不大于 2 GB。您可以使用 <code>size</code> 选项将大小设置成超出此范围的某个值。	<code>auto</code>	转储设备的大小是自动确定的。	<code>rootdisk.s0</code>	用于创建根池的设备被指定为片 0。
<code>nrpool</code>	定义根池的名称。										
<code>auto</code>	自动指定磁盘大小。此大小由指定磁盘的大小确定。										
<code>auto</code>	交换区域的大小是由 <code>auto</code> 关键字自动确定的。缺省大小为物理内存大小的 1/2, 但不小于 512 MB 且不大于 2 GB。您可以使用 <code>size</code> 选项将大小设置成超出此范围的某个值。										
<code>auto</code>	转储设备的大小是自动确定的。										
<code>rootdisk.s0</code>	用于创建根池的设备被指定为片 0。										
<code>bootenv</code>	<code>installbe</code> 可更改在安装期间创建的缺省引导环境的特征。 <table border="0" style="margin-left: 2em;"> <tr> <td><code>bename</code></td> <td>将新的引导环境命名为 <code>bnv</code>。</td> </tr> <tr> <td><code>dataset</code></td> <td>创建独立于 ROOT 数据集的 <code>/var</code> 数据集。<code>/var</code> 是适用于 <code>dataset</code> 的唯一值。</td> </tr> </table>	<code>bename</code>	将新的引导环境命名为 <code>bnv</code> 。	<code>dataset</code>	创建独立于 ROOT 数据集的 <code>/var</code> 数据集。 <code>/var</code> 是适用于 <code>dataset</code> 的唯一值。						
<code>bename</code>	将新的引导环境命名为 <code>bnv</code> 。										
<code>dataset</code>	创建独立于 ROOT 数据集的 <code>/var</code> 数据集。 <code>/var</code> 是适用于 <code>dataset</code> 的唯一值。										

用于 ZFS 根 (/) 文件系统的 JumpStart 关键字 (参考)

本节提供一些可在 JumpStart 配置文件中使用的特定于 ZFS 关键字的描述。本节中的这些关键字的用法与它们在 UFS 配置文件中的用法不同, 或者仅在 ZFS 配置文件中使用。

- 有关 UFS 和 ZFS 配置文件关键字的快速参考, 请参见第 103 页中的“配置文件关键字快速参考”。

- 下列关键字可在 ZFS 配置文件中使用。对于 UFS 配置文件和 ZFS 配置文件而言，用法是相同的。有关这些关键字的描述，请参见第 105 页中的“配置文件关键字描述和示例”。
 - boot_device
 - cluster
 - dontuse
 - fdisk
 - filesystem (挂载远程文件系统)
 - geo
 - locale
 - package (软件包)
 - usedisk

bootenv 配置文件关键字 (ZFS 和 UFS)

bootenv 关键字可标识引导环境特征。缺省情况下，引导环境是在安装期间由 pool 关键字创建的。如果您结合使用 bootenv 关键字和 installbe 选项，则可以命名新的引导环境，并在引导环境内创建 /var 数据集。

可以在配置文件中使用此关键字来安装 UFS 文件系统或 ZFS 根池。

- 在 UFS 文件系统中，可使用此关键字为将来的 Solaris Flash 归档文件安装创建一个空引导环境。有关适用于 UFS 的 bootenv 关键字的完整描述，请参见第 113 页中的“bootenv 配置文件关键字 (UFS 和 ZFS)”。
- 对于 ZFS 根池，bootenv 关键字可更改在安装时创建的缺省引导环境的特征。此引导环境是您所安装的根文件系统的副本。

bootenv 关键字可以与 installbe、bename 和 dataset 选项结合使用。这些选项可命名引导环境，并创建独立的 /var 数据集。

```
bootenv installbe bename new-BE-name [dataset mount-point]
```

installbe 更改在安装期间创建的缺省引导环境的特征。

bename 指定要创建的新引导环境的名称 *new_BE_name*。此名称不能超过 30 个字符，只能包含字母数字字符，不能包含多字节字符。名称在系统中必须是唯一的。

dataset *mount-point* 使用可选的 dataset 关键字可标识独立于 ROOT 数据集的 /var 数据集。*mount-point* 值限制为 /var。例如，用于独立的 /var 数据集的 bootenv 语法行类似于以下内容：

```
bootenv installbe bename zfsroot dataset /var
```


有关升级和激活引导环境的更多信息，请参见《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 11 章“Solaris Live Upgrade 和 ZFS（概述）”。

install_type 关键字 (ZFS 和 UFS)

`install_type` 关键字在每个配置文件中都是必需的。对于 UFS 安装，有几个可用选项。对于 ZFS 安装，唯一可用的选项是 `initial_install` 关键字。此选项可在系统上安装新的 Solaris OS。配置文件语法如下所示：

```
install_type initial_install
```

注 - 对于 ZFS 安装，以下 UFS 选项不可用。

- `upgrade` - 必须使用 Solaris Live Upgrade 才能升级 ZFS 根池。请参见《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 11 章“Solaris Live Upgrade 和 ZFS（概述）”。
 - `flash_install` - 无法安装 Solaris Flash 归档文件。
 - `flash_update` - 无法安装 Solaris Flash 归档文件。
-

pool 配置文件关键字 (仅 ZFS)

`pool` 关键字可定义要创建的新根池。然后，该池随使用 `cluster` 关键字指定的软件组一起安装。在创建新的根池时，需要使用 `poolsize`、`swapsize`、`dumpsizesize` 和 `vdevlist` 选项。

```
pool poolname poolsize swapsize dumpsizesize vdevlist
```

`poolname` 指定要创建的新池的名称。新池是使用指定的 `size` 和指定的设备 `vdevlist` 创建的。

`poolsize` 要创建的新池的大小。如果指定了空间大小，则此大小的单位假定为 MB，除非指定了 g (GB)。您还可以使用 `auto` 选项。

`auto` 在给定的约束条件下分配最大的可能池大小，例如磁盘和保留的片的大小。

注 - `poolsize` 关键字的 `auto` 含义与在 UFS 文件系统中使用的 `filesys` 关键字的 `auto` 含义不同。在 ZFS 中，将检查磁盘大小，以验证是否可提供最小大小。如果可以提供最小大小，则在给定的约束条件下分配最大的可能池大小，例如磁盘和保留的片的大小。

<i>swapsize</i>	要在新根池中创建的交换卷 (zvol) 的大小。选项可以是 <i>auto</i> 或 <i>size</i> 。 <i>auto</i> 交换区域的大小是自动确定的。缺省大小为物理内存大小的 1/2，但不小于 512 MB 且不大于 2 GB。您可以使用 <i>size</i> 选项将大小设置成超出此范围的某个值。 <i>size</i> 可用于指定一个量。大小的单位假定为 MB，除非指定了 g (GB)。
<i>dumpsize</i>	要在新根池中创建的转储卷的大小。 <i>auto</i> 使用缺省的交换大小。 <i>size</i> 可用于指定一个量。大小的单位假定为 MB，除非指定了 g (GB)。
<i>vdevlist</i>	用于创建池的一个或多个设备。 对于根池， <i>vdevlist</i> 中的设备必须是片。 <i>vdevlist</i> 可以是格式为 <i>cwtxdysz</i> 的 <i>single-device-name</i> ，也可以是 <i>mirror</i> 或 <i>any</i> 选项。

注 - *vdevlist* 的格式与 *zpool create* 命令的格式相同。

<i>single-device-name</i>	格式为 <i>cwtxdysz</i> 的磁盘片，例如 <i>c0t0d0s0</i> 。
<i>mirror</i> [<i>device-names</i>] <i>any</i>	指定磁盘镜像。 此时，如果指定了多个设备，则仅支持镜像配置。您可以镜像任意多的磁盘，但创建的池的大小由指定的最小磁盘确定。有关创建镜像存储池的更多信息，请参见《Solaris ZFS 管理指南》中的“镜像存储池配置”。 <ul style="list-style-type: none">▪ <i>device-names</i> 可列出要镜像的设备。这些名称的格式是 <i>cwtxdysz</i>，例如，<i>c0t0d0s0</i> 和 <i>c0t0d1s5</i>。▪ <i>any</i> 选项可使安装程序选择设备。
<i>any</i>	使安装程序选择适当的设备。

root_device 配置文件关键字 (ZFS 和 UFS)

root_device *cwtxdysz*

`root_device` 可指定用于根池的设备。`root_device` 关键字可确定操作系统的安装位置。此关键字在 ZFS 和 UFS 文件系统中的用法相同，都具有一些限制。对于 ZFS 根池，根设备限制为单个系统。此关键字对于镜像池没有用。

`cwtxd ysz` 标识用来安装操作系统的根磁盘。

其他资源

有关本章中包含的主题的其他信息，请参见表 9-2 中列出的资源。

表 9-2 其他资源

资源	位置
有关 ZFS 的信息，包括概述、规划和逐步说明	《Solaris ZFS 管理指南》
有关所有 JumpStart 关键字的列表	第 8 章，自定义 JumpStart（参考）
有关使用 Solaris Live Upgrade 从 UFS 迁移到 ZFS 或在 ZFS 根池中创建新的引导环境的信息	《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 11 章“Solaris Live Upgrade 和 ZFS（概述）”

第 2 部分

附录

本部分包含疑难解答和参考信息。

疑难解答（任务）

本章包含您在安装 Solaris 10 10/08 软件时可能遇到的特定错误消息和一般问题的列表。本章还说明了如何解决这些问题。首先使用以下章节列表确定安装过程中出现问题的地方。

- 第 159 页中的“设置网络安装时的问题”
- 第 160 页中的“引导系统时的问题”
- 第 165 页中的“Solaris OS 的初始安装”
- 第 167 页中的“升级 Solaris 操作系统”

注 - “可引导介质”这一短语指的是 Solaris 安装程序和 JumpStart 安装方法。

设置网络安装时的问题

Unknown client "*host_name*" (未知的客户机 "*host_name*")

原因: `add_install_client` 命令中的 *host_name* 参数不是命名服务中的主机。

解决方法: 将主机 *host_name* 添加到命名服务, 并再次执行 `add_install_client` 命令。

Error: <system name> does not exist in the NIS ethers map

Add it, and rerun the `add_install_client` command (错误: NIS 以太网映射中不存在 <系统名>。请添加它, 并重新运行 `add_install_client` 命令)

描述: 当您运行 `add_install_client` 命令时, 该命令会失败并出现上述错误。

原因: 您要添加至安装服务器的客户机在该服务器的 `/etc/ethers` 文件中不存在。

解决方法: 将所需信息添加至安装服务器的 `/etc/ethers` 文件, 然后再次运行 `add_install_client` 命令。

1. 成为超级用户或承担等效角色。

2. 在客户机上，找到以太网地址。

```
# ifconfig -a grep ethers
ether 8:0:20:b3:39:1d
```

3. 在安装服务器上，在编辑器中打开 `/etc/ethers` 文件。将地址添加到列表中。
4. 在客户机上，再次运行 `add_install_client`，如以下示例所示。

```
# ./add_install_client bluegill sun4u
```

引导系统时的问题

从介质引导：错误消息

`le0: No carrier - transceiver cable problem (le0 : 无载体—收发器电缆问题)`

原因: 系统未连接到网络。

解决方法: 如果这是非联网的系统，请忽略该消息。如果这是联网系统，请确保安全地连接了以太网电缆。

`The file just loaded does not appear to be executable (刚装入的文件不是可执行文件)`

原因: 系统无法找到合适的引导介质。

解决方法: 验证是否已经正确地设置系统，以使用安装服务器通过网络来安装 Solaris 10 10/08 软件。下面是可执行的检查示例。

- 如果已将 Solaris Operating System DVD 或 Solaris Software CD 的映像复制到安装服务器，请确保在设置系统时为其指定了正确的平台组。
- 如果要使用 DVD 或 CD 介质，请确保已在安装服务器上挂载了 Solaris Operating System DVD 或 Solaris Software - 1 CD，并且可以对其进行访问。

`boot: cannot open <filename> (引导：无法打开 <filename>) (仅限于基于 SPARC 的系统)`

原因: 通过显式地设置 `boot -file` 来覆写该位置时，将发生此错误。

注 - `filename` 是一个变量，指代受影响文件的名称。

解决方法: 请按照以下说明操作：

- 将 PROM 中的 `boot -file` 重置为“”（空白）。

- 确保将 diag-switch 设置为 off 和 true。

Can't boot from file/device (无法从 file/device 引导)

原因: 安装介质找不到可引导介质。

解决方法: 确保满足以下条件:

- 正确地安装 DVD-ROM 或 CD-ROM 驱动器, 并将其打开。
- 将 Solaris Operating System DVD 或 Solaris Software - 1 CD 放入到驱动器中。
- 光盘未损坏或未弄脏。

WARNING: clock gained xxx days -- CHECK AND RESET DATE! (警告: 时钟快 xxx 天 - 检查并重置日期!) (仅限于基于 SPARC 的系统)

描述: 这是提示性消息。

解决方法: 忽略此消息, 并继续进行安装。

Not a UFS file system (不是 UFS 文件系统) (仅限于基于 x86 的系统)

原因: 安装 Solaris 10 10/08 软件时 (通过 Solaris 安装程序或自定义 JumpStart), 未选中任何引导磁盘。现在, 必须编辑 BIOS 才能引导系统。

解决方法: 选择 BIOS 以进行引导。有关说明, 请参见 BIOS 文档。

从介质引导: 一般问题

系统不引导。

描述: 在最初设置自定义 JumpStart 服务器时, 您可能会遇到并不返回错误消息的引导问题。要验证有关系统以及系统引导方式的信息, 请运行带有 -v 选项的引导命令。当您使用 -v 选项时, 引导命令将在屏幕上显示详细调试信息。

注 - 如果未给出此标志, 这些消息仍然会打印出来, 但输出将被重定向到系统日志文件。有关更多信息, 请参见 [syslogd\(1M\)](#)。

解决方法: 对于基于 SPARC 的系统, 请在 ok 提示符下键入以下命令。

```
ok boot net -v - install
```

在带有 Toshiba SD-M 1401 DVD-ROM 的系统上, 从 DVD 介质引导失败

描述: 如果系统的 Toshiba SD-M1401 DVD-ROM 的固件修订版为 1007, 则无法从 Solaris Operating System DVD 引导系统。

解决方法: 应用修补程序 111649-03 或更高版本以更新 Toshiba SD-M1401 DVD-ROM 驱动器的固件。sunsolve.sun.com 上提供了修补程序 111649-03。

插入无内存 PC 卡时，系统挂起或发出警告音。（仅限于基于 x86 的系统）

原因: 无内存 PC 卡不能与其他设备使用相同的内存资源。

解决方法: 要修正这个问题，请查看您的 PC 卡说明并检查地址范围。

在显示系统提示符之前，系统挂起。（仅限于基于 x86 的系统）

解决方法: 系统中有不支持的硬件。查看您的硬件制造商文档。

从网络引导：错误消息

WARNING: getfile: RPC failed: error 5 (RPC Timed out). (警告：getfile：RPC 失败：错误 5 (RPC 超时)。)

描述: 当网络上有两个或更多个服务器响应安装客户机的引导请求时，将发生此错误。安装客户机与错误的引导服务器连接，安装挂起。以下特定原因可能导致发生该错误：

原因: 原因 1：/etc/bootparams 文件可能位于不同的服务器上，这些服务器均含有此安装客户机的项。

解决方法: 原因 1：确保网络上的服务器没有多个用于该安装客户机的 /etc/bootparams 项。如果它们有多个项，请删除所有安装服务器和引导服务器（安装客户机要使用的服务器除外）上 /etc/bootparams 文件中的重复客户机项。

原因: 原因 2：可能存在多个用于该安装客户机的 /tftpboot 或 /rplboot 目录项。

解决方法: 原因 2：确保网络上的服务器没有多个用于该安装客户机的 /tftpboot 或 /rplboot 目录项。如果它们有多个项，请删除所有安装服务器和引导服务器（安装客户机要使用的服务器除外）上 /tftpboot 或 /rplboot 目录中的重复客户机项。

原因: 原因 3：服务器上的 /etc/bootparams 文件中可能存在一个安装客户机项，并且另一个 /etc/bootparams 文件中可能存在使所有系统能够访问配置文件服务器的项。这样的项与以下内容类似：

```
* install_config=profile_server:path
```

类似于 NIS 或 NIS+ bootparams 表中的前一个项的行也可引起该错误。

解决方法: 原因 3：如果通配符项在命名服务 bootparams 映射或表中（例如，* install_config=），请将其删除然后添加到引导服务器的 /etc/bootparams 文件中。

No network boot server. Unable to install the system. See installation instructions. (没有网络引导服务器。无法安装系统。请参见安装说明。)(仅限于基于 SPARC 的系统)

原因: 此错误发生在您试图从网络安装的系统上。系统设置不正确。

解决方法: 确保正确地将系统设置为从网络安装。请参见《Solaris 10 10/08 安装指南：基于网络的安装》中的“使用 CD 映像添加要从网络安装的系统”。

prom_panic: Could not mount file system (prom_panic : 无法挂载文件系统) (仅限于基于 SPARC 的系统)

原因: 当您从网络安装 Solaris，但引导软件无法找到以下内容时将发生此错误：

- Solaris Operating System DVD（此 DVD 或安装服务器上 DVD 映像的副本）
- Solaris Software - 1 CD 映像（Solaris Software - 1 CD 或安装服务器上的 CD 映像的副本）

解决方法: 确保安装并挂载了安装软件。

- 如果从安装服务器的 DVD-ROM 或 CD-ROM 驱动器安装 Solaris，请确保 Solaris Operating System DVD 或 Solaris Software - 1 CD 已插入 CD-ROM 驱动器中，已挂载并已在 `/etc/dfs/dfstab` 文件中共享。
- 如果从安装服务器磁盘上的 Solaris Operating System DVD 映像或 Solaris Software - 1 CD 映像的副本进行安装，确保在 `/etc/dfs/dfstab` 文件中共享该副本的目录路径。

Timeout waiting for ARP/RARP packet... (等待 ARP/RARP 包超时...) (仅限于基于 SPARC 的系统)

原因: **原因 1：** 客户机试图从网络引导，但它找不到知道该客户机的系统。

解决方法: **原因 1：** 验证系统的主机名是否在 NIS 或 NIS+ 命名服务中。此外，还要验证引导服务器的 `/etc/nsswitch.conf` 文件中的 `bootparams` 搜索顺序。

例如，`/etc/nsswitch.conf` 文件中的以下行指示 `JumpStart` 或 `Solaris` 安装程序首先在 NIS 映射中查找 `bootparams` 信息。如果此程序未找到任何信息，则安装程序将在引导服务器的 `/etc/bootparams` 文件中查找。

`bootparams: nis files`

原因: **原因 2：** 客户机的以太网地址不正确。

解决方法: **原因 2：** 验证安装服务器的 `/etc/ethers` 文件中的客户机以太网地址是否正确。

原因: **原因 3：** 在自定义 `JumpStart` 安装中，`add_install_client` 命令指定将指定的服务器作为安装服务器的平台组。如果在使用 `add_install_client` 时使用了错误的体系结构值，则会发生此问题。例如，要安装的计算机是 `sun4u`，但您却改用了 `i86pc`。

解决方法: **原因 3：** 使用正确的体系结构值，重新运行 `add_install_client`。

ip: joining multicasts failed on tr0 - will use link layer broadcasts for multicast (ip : 在 tr0 上连接多路广播失败—多路广播将使用链路层广播) (仅限于基于 x86 的系统)

原因:当使用令牌环卡引导系统时会显示此错误消息。以太网多址广播和令牌环多址广播的工作方式不同。由于向驱动器提供了无效的多址广播地址,因此驱动器返回此错误消息。

解决方法:忽略此错误消息。如果不能使用多址广播,IP将改用链路层广播,不会造成安装失败。

Requesting Internet address for Ethernet_Address (请求 Ethernet_Address 的 Internet 地址) (仅限于基于 x86 的系统)

原因:客户机试图从网络引导,但它找不到知道该客户机的系统。

解决方法:验证系统的主机名是否列在命名服务中。如果系统主机名列在NIS或NIS+命名服务中,并且系统继续显示此错误消息,请尝试重新引导系统。

RPC: Timed out No bootparams (whoami) server responding; still trying... (RPC : 超时,没有 bootparams (whoami) 服务器响应;仍在尝试... (仅限于基于 x86 的系统)

原因:客户机试图从网络引导,但它无法找到具有安装服务器上的/etc/bootparams文件项的系统。

解决方法:在安装服务器上使用add_install_client。使用该命令向/etc/bootparams文件中添加适当的项,使客户机可以从网络引导。

仍在尝试查找 RPL 服务器... (仅基于 x86 系统)

原因:系统试图从网络引导,但服务器没有设置为引导此系统。

解决方法:在安装服务器上,为要安装的系统执行add_install_client。add_install_client命令建立一个/rplboot目录,其中包含需要的网络引导程序。

CLIENT MAC ADDR: FF FF FF FF FF FF (仅存在于使用 DHCP 执行的网络安装)

原因:DHCP服务器未正确配置。如果未在DHCP管理器软件中正确定义选项或宏,可能会发生这种错误。

解决方法:在DHCP管理器软件中,验证是否正确定义了选项和宏。确认定义了“路由器”选项,并且对于用来进行网络安装的子网,“路由器”选项的值是正确的。

从网络引导：一般问题

统从网络引导,但从其他系统(而非指定的安装服务器)进行引导。

原因:客户机的另一个系统上可能存在/etc/bootparams或/etc/ethers项。

解决方法: 在名称服务器上，为要安装的系统更新 `/etc/bootparams` 项。该项应符合以下语法：

```
install_system root=boot_server:path install=install_server:path
```

此外，确保安装客户机的子网中只有一个 `bootparams` 项。

系统没有从网络引导（仅限于使用 **DHCP** 执行的网络安装）。

原因: DHCP 服务器未正确配置。如果未在 DHCP 服务器上将系统配置为安装客户机，则可能发生这种错误。

解决方法: 在 DHCP 管理器软件中，验证是否为客户机系统定义了安装选项和宏。有关更多信息，请参见《Solaris 10 10/08 安装指南：基于网络的安装》中的“使用 DHCP 服务预配置系统配置信息（任务）”。

Solaris OS 的初始安装

Initial installation fails（初始安装失败）

解决方法: 如果 Solaris 安装失败，则您必须重新启动安装。要重新启动安装，请从 Solaris Operating System DVD、Solaris Software - 1 CD 或网络引导系统。

不能在安装部分 Solaris 软件之后卸载该软件，必须从备份中恢复系统或重新开始 Solaris 安装过程。

```
/cdrom/cdrom0/SUNW xxxx/reloc.cpio: Broken pipe ( /cdrom/cdrom0/SUNW
xxxx/reloc.cpio : 中断的管道 )
```

描述: 此错误消息是提示性消息，不影响安装。在没有读取进程的管道上写入时会出现此情况。

解决方法: 忽略此消息，并继续进行安装。

WARNING: CHANGE DEFAULT BOOT DEVICE（警告：更改缺省引导设备）（仅限于基于 x86 的系统）

原因: 这是提示性消息。系统 BIOS 中设置的缺省引导设备可能被设置成需要使用 Solaris Device Configuration Assistant 来引导系统。

解决方法: 继续安装，如有必要，在将 Solaris 软件安装到无需 Solaris Device Configuration Assistant 的设备后，请更改在 BIOS 中指定的系统的缺省引导设备。

仅适用于 x86 - 如果使用 `locale` 关键字测试初始安装的自定义 JumpStart 配置文件，则 `pfinstall -D` 命令无法测试配置文件。有关解决方法，请参见第 167 页中的“升级 Solaris 操作系统”一节中的错误消息“无法选择语言环境”。

▼ x86: 检查 IDE 磁盘上的坏块

IDE 磁盘驱动器不像其他 Solaris 软件所支持的驱动器那样可以自动映射出坏块。在 IDE 磁盘上安装 Solaris 之前，您可能需要执行磁盘表面分析。要在 IDE 磁盘上进行表面分析，请执行以下过程。

1 成为超级用户或承担等效角色。

角色包含授权和具有一定权限的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

2 引导到安装介质。

3 当提示您选择安装类型时，选择选项 6，单用户 shell。

4 启动 `format(1M)` 程序。

```
# format
```

5 指定要在其上执行表面分析的 IDE 磁盘驱动器。

```
# cxdy
```

`cx` 是控制器编号

`dy` 是设备编号

6 确定是否存在 `fdisk` 分区。

- 如果已经存在 Solaris `fdisk` 分区，请继续执行步骤 7。
- 如果没有 Solaris `fdisk` 分区，请使用 `fdisk` 命令在磁盘上创建一个 Solaris 分区。

```
format> fdisk
```

7 要开始表面分析，请键入：

```
format> analyze
```

8 要确定当前设置，请键入：

```
analyze> config
```

9 （可选）要更改设置，请键入：

```
analyze> setup
```

10 要查找坏块，请键入：

```
analyze> type_of_surface_analysis
```

`type_of_surface_analysis` 是读、写或者比较

如果 format 发现坏块，则将重新映射这些坏块。

- 11 要退出分析，请键入：

```
analyze> quit
```

- 12 确定是否要指定重映射的块。

- 如果不指定，请转至步骤 13。
- 如果需要，请键入：

```
format> repair
```

- 13 要退出格式化程序，请键入：

```
quit
```

- 14 通过键入以下命令在多用户模式下重新启动介质。

```
# exit
```

升级 Solaris 操作系统

升级：错误消息

No upgradable disks (没有可升级磁盘)

原因: /etc/vfstab 文件中的交换项导致升级失败。

解决方法: 注释掉 /etc/vfstab 文件中的下列行：

- 不升级磁盘上的所有交换文件和片
- 不再存在的交换文件
- 任何未使用的交换片

usr/bin/bzcat not found (未找到 usr/bin/bzcat)

原因: Solaris Live Upgrade 因需要修补程序簇而失败。

解决方法: 安装 Solaris Live Upgrade 需要修补程序。请访问 <http://sunsolve.sun.com>，以确保具有最近更新的修补程序列表。在 SunSolve Web 站点上搜索信息文档 72099。

Upgradeable Solaris root devices were found, however, no suitable partitions to hold the Solaris install software were found. Upgrading using the Solaris Installer is not possible. It might be possible to upgrade using the Solaris Software 1 CDRom. (找到可升级的 Solaris 根设备, 但没有找到合适的用于保存 Solaris 安装软件的分区。无法使用 Solaris 安装程序升级。可以使用 Solaris Software 1 CDRom 进行升级。)(仅限于基于 x86 的系统)

原因: 由于没有足够空间, 因此无法使用 Solaris Software - 1 CD 升级。

解决方法: 要进行升级, 可以创建大于或等于 512 MB 的交换片, 或者使用另一种升级方法, 例如, 通过 Solaris Operating System DVD 上的 Solaris 安装程序、网络安装映像或 JumpStart。

ERROR: Could not select locale (错误: 无法选择语言环境)(仅限于基于 x86 的系统)

原因: 通过使用 `pfinstall -D` 命令测试 JumpStart 配置文件时, 预运行测试在以下情况下会失败:

- 该配置文件包含 locale 关键字。
- 所测试的发行版包含 GRUB 软件。从 Solaris 10 1/06 发行版开始, GRUB 引导装载机通过 GRUB 菜单简化了对安装在系统上的不同操作系统的引导。

引入 GRUB 软件后, 会压缩 miniroot。该软件无法再从压缩的 miniroot 中找到语言环境列表。miniroot 是可能具有的最小的 Solaris 根 (/) 文件系统, 可以在 Solaris 安装介质中找到。

解决方法: 请执行以下步骤。请使用以下值。

- MEDIA_DIR 为 /cdrom/cdrom0/
- MINIROOT_DIR 为 \$MEDIA_DIR/Solaris_10/Tools/Boot
- MINIROOT_ARCHIVE 为 \$MEDIA_DIR/boot/x86.miniroot
- TEMP_FILE_NAME 为 /tmp/test

1. 成为超级用户或承担等效角色。

角色包含授权和具有一定权限的命令。有关角色的更多信息, 请参见《系统管理指南: 安全性服务》中的“配置 RBAC (任务列表)”。

2. 解压缩 miniroot 归档文件。

```
# /usr/bin/gzcat $MINIROOT_ARCHIVE > $TEMP_FILE_NAME
```

3. 使用 `lofiadm` 命令创建 miniroot 设备。

```
# LOFI_DEVICE=/usr/sbin/lofiadm -a $TEMP_FILE_NAME
# echo $LOFI_DEVICE
/dev/lofi/1
```

4. 使用 `lofi` 命令在 Miniroot 目录下挂载 miniroot。


```
# /usr/sbin/mount -F ufs $LOFI_DEVICE $MINIROOT_DIR
```

5. 测试配置文件。

```
# /usr/sbin/install.d/pfinstall -D -c $MEDIA_DIR $path-to-jumpstart_profile
```

6. 测试完成后，卸载 lofi 设备。

```
# umount $LOFI_DEVICE
```

7. 删除 lofi 设备。

```
# lofiadm -d $TEMP_FILE_NAME
```

升级：一般问题

即使存在一个可在系统升级的 Solaris 软件版本，也不显示升级选项。

原因:原因 1：/var/sadm 目录是符号链接，或是从另一个文件系统挂载的。

解决方法:原因 1：将 /var/sadm 目录移动到根 (/) 或 /var 文件系统中。

原因:原因 2：缺少 /var/sadm/softinfo/INST_RELEASE 文件。

解决方法:原因 2：使用以下模板创建新的 INST_RELEASE 文件：

```
OS=Solaris
VERSION=x
REV=0
```

x 是系统上的 Solaris 软件的版本

原因:原因 3：/var/sadm/softinfo 中缺少 SUNWusr。

解决方法:解决方案 3：需要执行初始安装。Solaris 软件是不可升级的。

无法关闭或初始化 md 驱动程序

解决方法:请按照以下说明操作：

- 如果该文件系统不是 RAID-1 卷，请在 vsftab 文件中注释掉。
- 如果该文件系统是 RAID-1 卷，请中断镜像并重新安装。有关取消镜像的信息，请参见《Solaris Volume Manager 管理指南》中的“删除 RAID-1 卷（取消镜像）”。

因 Solaris 安装程序无法挂载文件系统而导致升级失败。

原因: 在升级期间, 该脚本会在要升级的根 (/) 文件系统中尝试挂载系统的 /etc/vfstab 文件中列出的所有文件系统。如果安装脚本无法挂载文件系统, 则它将失败并退出。

解决方法: 确保系统的 /etc/vfstab 文件中的所有文件系统都可进行挂载。在 /etc/vfstab 文件中注释掉任何无法挂载或可能引起问题的文件系统, 以便 Solaris 安装程序在升级过程中不会尝试挂载它们。不能注释掉那些包含要升级软件的基于系统的文件系统 (例如 /usr)。

升级失败

描述: 系统没有足够的升级空间。

原因: 有关空间问题, 请参阅《Solaris 10 10/08 安装指南: 规划安装和升级》中的“[通过重新分配磁盘空间进行升级](#)”, 查明是否能在不使用自动布局重新分配空间的情况下修复此问题。

升级 RAID-1 卷根 (/) 文件系统时遇到问题

解决方法: 如果在使用作为根 (/) 文件系统的 Solaris 卷管理器 RAID-1 卷进行升级时遇到问题, 请参见《Solaris Volume Manager 管理指南》中的第 25 章“[对 Solaris Volume Manager 进行疑难解答 \(任务\)](#)”。

▼ 在升级失败后继续进行升级

升级失败, 系统无法进行软引导。失败是由于您无法控制的原因 (例如断电或网络连接失败) 造成的。

- 1 从 Solaris Operating System DVD、Solaris Software - 1 CD 或网络重新引导系统。
- 2 选择安装的升级选项。
Solaris 安装程序确定系统是否已经部分升级并继续进行升级。

x86: 使用 GRUB 时, Solaris Live Upgrade 出现问题

在基于 x86 的系统上使用 Solaris Live Upgrade 和 GRUB 引导装载器时, 可能会发生以下错误。

ERROR: 介质产品工具安装目录 *path-to-installation-directory* 不存在。

ERROR: 介质目录不包含操作系统升级映像。

描述: 当使用 luupgrade 命令升级新的引导环境时可以看到这些错误消息。

原因:使用的是旧版本的 Solaris Live Upgrade。您在系统上安装的 Solaris Live Upgrade 软件包与介质以及介质上的软件包发行版不兼容。

解决方法:确保所使用的 Solaris Live Upgrade 软件包的版本始终为将要升级到的版本。

示例:在以下示例中，错误消息表示系统上的 Solaris Live Upgrade 软件包与介质上的软件包的版本不同。

```
# luupgrade -u -n s10u1 -s /mnt
Validating the contents of the media </mnt>.
The media is a standard Solaris media.
ERROR: The media product tools installation directory
</mnt/Solaris_10/Tools/Boot/usr/sbin/install.d/install_config> does
not exist.
ERROR: The media </mnt> does not contain an operating system upgrade
image.
```

ERROR: 找不到或不可执行： </sbin/biosdev>。

ERROR: Solaris Live Upgrade 所需的一个或多个修补程序未安装。

原因:Solaris Live Upgrade 所需的一个或多个修补程序未安装在系统上。请注意该错误消息没有捕捉到所有缺少的修补程序。

解决方法:使用 Solaris Live Upgrade 之前，始终安装所有必需的修补程序。请访问 <http://sunsolve.sun.com>，以确保具有最近更新的修补程序列表。在 SunSolve Web 站点上搜索信息文档 72099。

ERROR: 设备映射命令 </sbin/biosdev> 失败。请重新引导，然后重试。

原因:原因 1：由于以前的管理任务，Solaris Live Upgrade 无法映射设备。

解决方法:原因 1：重新引导系统，然后使用 Solaris Live Upgrade 重试

原因:原因 2：如果重新引导系统后出现相同的错误消息，表明您有两个或多个相同的磁盘。设备映射命令无法区分这些磁盘。

解决方法:原因 2：在这些磁盘中的某一个磁盘上创建一个新的伪 fdisk 分区。请参见 [fdisk\(1M\)](#) 手册页。然后重新引导系统。

无法删除包含 GRUB 菜单的引导环境

原因:Solaris Live Upgrade 强加了一条限制，即如果引导环境包含 GRUB 菜单，则无法删除该引导环境。

解决方法:使用 [lumake\(1M\)](#) 或 [luupgrade\(1M\)](#) 命令重新使用该引导环境。

无意中重新创建了包含 GRUB 菜单的文件系统。但是，磁盘具有和以前相同的片。例如，磁盘没有重新分片。

原因: 包含 GRUB 菜单的文件系统对保持系统的可引导性非常重要。Solaris Live Upgrade 命令不会破坏 GRUB 菜单。但是，如果使用 Solaris Live Upgrade 命令之外的命令无意中重新创建或破坏了包含 GRUB 菜单的文件系统，恢复软件会尝试重新安装 GRUB 菜单。在下次重新引导时恢复软件会将 GRUB 菜单放回到相同的文件系统中。例如，您可能在文件系统上使用了 `newfs` 或 `mkfs` 命令，并无意中破坏了 GRUB 菜单。要正确恢复 GRUB 菜单，片必须符合以下条件：

- 包含一个可挂载的文件系统
- 保留片以前所驻留的同一 Solaris Live Upgrade 引导环境的一部分

重新引导系统之前，对片进行一些必要的更正操作。

解决方法: 重新引导系统。会自动安装 GRUB 菜单的副本。

GRUB 菜单的 `menu.lst` 文件被无意中删除。

解决方法: 重新引导系统。会自动安装 GRUB 菜单的副本。

▼ 在运行 Veritas VxVm 时使用 Solaris Live Upgrade 进行升级，系统将发出警告音

如果在升级并运行 Veritas VxVM 的同时使用 Solaris Live Upgrade，除非您使用以下过程进行升级，否则系统在重新引导时将发出警告音。如果软件包不符合 Solaris 高级打包指南，将出现问题。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有一定权限的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 创建非活动的引导环境。请参见《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的“创建新的引导环境”。
- 3 在升级非活动的引导环境之前，必须禁用非活动的引导环境中的现有 Veritas 软件。

- a. 挂载非活动的引导环境。

```
# lumount inactive_boot_environment_name mount_point
```

例如：

```
# lumount solaris8 /mnt
```

- b. 转到包含 `vfstab` 的目录，例如：

```
# cd /mnt/etc
```

- c. 生成非活动引导环境的 `vfstab` 文件的副本，例如：

```
# cp vfstab vfstab.501
```

- d. 在复制的 `vfstab` 中注释掉所有 Veritas 文件系统项，例如：

```
# sed '/vx\dsk/s/^/#/g' < vfstab > vfstab.novxfs
```

将每行的第一个字符更改为 `#`，这使该行成为注释行。请注意，该注释行不同于系统文件注释行。

- e. 复制更改后的 `vfstab` 文件，例如：

```
# cp vfstab.novxfs vfstab
```

- f. 转到非活动引导环境的系统文件所在的目录，例如：

```
# cd /mnt/etc
```

- g. 生成非活动的引导环境的系统文件副本，例如：

```
# cp system system.501
```

- h. 注释掉所有包含 `drv/vx` 的 "forceload:" 项。

```
# sed '/forceload: drv\/vx/s/^/*/' <system> system.novxfs
```

将每行的第一个字符更改为 `*`，这使该行成为注释行。请注意，该注释行不同于 `vfstab` 文件注释行。

- i. 创建 Veritas `install-db` 文件，例如：

```
# touch vx/reconfig.d/state.d/install-db
```

- j. 卸载非活动的引导环境。

```
# luumount inactive_boot_environment_name
```

- 4 升级非活动的引导环境。请参见《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 5 章“使用 Solaris Live Upgrade 进行升级（任务）”。
- 5 激活非活动的引导环境。请参见《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的“激活引导环境”。
- 6 关闭系统。
init 0

7 在单用户模式下引导非活动的引导环境：

```
OK boot -s
```

将显示若干个包含 "vxvm" 或 "VXVM" 的消息和错误消息，可以忽略这些消息。非活动的引导环境成为活动的引导环境。

8 升级 Veritas。**a. 将 Veritas VRTSvmsa 软件包从系统中删除，例如：**

```
# pkgrm VRTSvmsa
```

b. 转到 Veritas 软件包所在的目录。

```
# cd /location_of_Veritas_software
```

c. 将最新的 Veritas 软件包添加到系统中：

```
# pkgadd -d 'pwd' VRTSvxvm VRTSvmsa VRTSvmdoc VRTSvmman VRTSvmdev
```

9 恢复初始的 vfstab 和系统文件：

```
# cp /etc/vfstab.original /etc/vfstab
```

```
# cp /etc/system.original /etc/system
```

10 重新引导系统。

```
# init 6
```

x86: 缺省情况下不会在不包含现有服务分区的系统上创建服务分区

如果在当前不包含服务分区或诊断分区的系统上安装 Solaris 10 10/08 OS，则在缺省情况下，安装程序可能不会创建服务分区。如果要在同一个磁盘上包含服务分区和 Solaris 分区，则必须在安装 Solaris 10 10/08 OS 之前重新创建服务分区。

如果您将 Solaris 8 2/02 OS 安装在具有服务分区的系统上，则安装程序可能不会保留服务分区。如果您没有手动编辑 `fdisk` 引导分区布局以保留服务分区，安装程序会在安装过程中删除服务分区。

注 - 如果安装 Solaris 8 2/02 OS 时未特别地保留服务分区，则可能无法重新创建服务分区并升级到 Solaris 10 10/08 OS。

如果要使包含 Solaris 分区的磁盘包含服务分区，请选择以下解决方法之一。

▼ 从网络安装映像或 Solaris Operating System DVD 安装软件

要通过网络安装映像安装软件或通过网络从 Solaris Operating System DVD 进行安装，请执行以下步骤。

- 1 删除磁盘中的内容。
- 2 安装前，使用诊断 CD 为系统创建服务分区。
有关如何创建服务分区的信息，请参见您的硬件文档。
- 3 通过网络引导系统。
将显示“自定义 fdisk 分区”屏幕。
- 4 要装入缺省引导磁盘分区布局，请单击“缺省”。
安装程序将保留服务分区并创建 Solaris 分区。

▼ 从 Solaris Software - 1 CD 或从网络安装映像安装

要使用 Solaris 安装程序通过 Solaris Software - 1 CD 进行安装，或通过引导服务器上的网络安装映像进行安装，请执行以下步骤。

- 1 删除磁盘中的内容。
- 2 安装前，使用诊断 CD 为系统创建服务分区。
有关如何创建服务分区的信息，请参见您的硬件文档。
- 3 安装程序将提示您选择创建 Solaris 分区的方法。
- 4 引导系统。
- 5 选择“将剩余磁盘空间用于 Solaris 分区”选项。
安装程序将保留服务分区并创建 Solaris 分区。
- 6 完成安装。

附加 SVR4 打包要求（参考）

本附录适用于安装或删除软件包（特别是第三方软件包）的系统管理员。遵循这些打包要求可实现以下目的：

- 避免修改当前运行的系统，以便使用 Solaris Live Upgrade 进行升级，并创建和维护非全局区域和无盘客户机
- 在使用安装程序（如自定义 JumpStart）时，禁止软件包进行交互式自动安装

本章包含以下几节：

- 第 177 页中的“禁止修改当前的 OS”。
- 第 180 页中的“安装或升级时禁止用户交互”。
- 第 182 页中的“为区域设置软件包参数”

禁止修改当前的 OS

遵守本节中的要求可确保当前运行的 OS 不被更改。

使用绝对路径

为了成功地安装操作系统，软件包必须能识别和正确地遵守备用根 (*/*) 文件系统，如 Solaris Live Upgrade 非活动引导环境。

软件包可以在其 `pkgmap` 文件（软件包映射）中包括绝对路径。如果这些文件存在，则将相对于 `pkgadd` 命令的 `-R` 选项来写它们。还可以将同时包含绝对和相对（可重定位）路径的软件包安装到备用根 (*/*) 文件系统上。`$PKG_INSTALL_ROOT` 可放在绝对和可重定位文件之前，这样在使用 `pkgadd` 安装时，可正确解析所有路径。

使用 pkgadd -R 命令

使用 pkgadd -R 选项安装或使用 pkgrm -R 选项删除的软件包不得更改当前正在运行的系统。此功能由自定义 JumpStart、Solaris Live Upgrade、非全局区域和无盘客户机使用。

使用 pkgadd 命令 -R 选项安装的或使用 pkgrm 命令 -R 选项删除的软件包中包括的任何过程脚本均不得更改当前正在运行的系统。您提供的任何安装脚本必须引用前缀为 \$PKG_INSTALL_ROOT 变量的目录或文件。软件包必须对所有带有 \$PKG_INSTALL_ROOT 前缀的目录和文件进行写操作。软件包不得删除不带 \$PKG_INSTALL_ROOT 前缀的目录。

表 B-1 提供了脚本语法的示例。

表 B-1 安装脚本语法示例

脚本类型	正确的语法	错误的语法
Bourne shell “if”语句段	<pre>if [-f \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf] ; then</pre>	<pre>if [-f /etc/myproduct.conf] ; \ then</pre>
删除文件	<pre>/bin/rm -f \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf</pre>	<pre>/bin/rm -f /etc/myproduct.conf</pre>
更改文件	<pre>echo "test=no" > \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf</pre>	<pre>echo "test=no" > \ /etc/myproduct.conf</pre>

\$PKG_INSTALL_ROOT 和 \$BASEDIR 之间的差别概述

\$PKG_INSTALL_ROOT 表示您要向其添加软件包的计算机的根 (/) 文件系统的位置。该位置被设置为 pkgadd 命令的 -R 参数。例如，如果调用以下命令，则在软件包安装过程中 \$PKG_INSTALL_ROOT 将成为 /a。

```
# pkgadd -R /a SUNWvxvm
```

\$BASEDIR 指向安装可重定位的软件包对象的可重定位基目录。在此仅安装可重定位的对象。不可重定位的对象（即那些在 pkgmap 文件中具有绝对路径的对象）的安装实际上始终相对于非活动的引导环境，而不是相对于 \$BASEDIR。如果一个软件包没有可重定位的对象，则该软件包被称为绝对软件包（或不可重定位的软件包），系统将不定义 \$BASEDIR，因而不可用于软件包过程脚本。

例如，假定一个软件包的 pkgmap 文件包含以下两项：

```
l f none sbin/ls 0555 root sys 3541 12322 1002918510
l f none /sbin/ls2 0555 root sys 3541 12322 2342423332
```

pkginfo 文件对 \$BASEDIR 进行了说明：

```
BASEDIR=/opt
```

如果使用以下命令安装软件包，则 `ls` 安装在 `/a/opt/sbin/ls` 中，而 `ls2` 安装为 `/a/sbin/ls2`。

```
# pkgadd -R /a SUNWtest
```

编写脚本的原则

软件包过程脚本必须独立于当前正在运行的 OS，以防止修改 OS。过程脚本定义在软件包安装和删除过程中的特定时刻所发生的操作。可以使用以下预定义的名称创建四个过程脚本：`preinstall`、`postinstall`、`preremove` 和 `postremove`。

表 B-2 创建脚本的原则

原则	影响 Solaris Live Upgrade	影响非全局区域
脚本必须在 Bourne shell (<code>/bin/sh</code>) 中编写。Bourne shell 是 <code>pkgadd</code> 命令用来执行过程脚本的解释程序。	X	X
脚本不得启动或停止任何进程，或者依赖于某些命令（例如 <code>ps</code> 或 <code>truss</code> ）的输出，因为这些进程或命令都与操作系统有关，并且会报告关于当前正在运行的系统的信息。	X	X
脚本可自由使用其他标准 UNIX 命令，例如 <code>expr</code> 、 <code>cp</code> 和 <code>ls</code> 以及其他有助于编写 shell 脚本的命令。	X	X
脚本调用的任何命令必须在所有支持的发行版中可用，因为软件包必须在所有这些发行版上运行。所以，不能使用在 Solaris 8 发行版之后添加或删除的命令。	X	
要验证在 Solaris 8、9 或 10 发行版中是否支持某个特定的命令或选项，请参见 http://docs.sun.com 上特定版本的 Solaris 参考手册资料。		

维护无盘客户机兼容性

软件包不得执行由软件包本身提供的命令。这是为了维护无盘客户机的兼容性，同时避免运行那些可能需要尚未安装的共享库的命令。

验证软件包

所有软件包都必须通过 `pkgchk` 验证。在创建软件包之后以及安装它之前，必须使用以下命令对其进行检查。

```
# pkgchk -d dir_name pkg_name
```

`dir_name` 指定该软件包所驻留的目录的名称

`pkg_name` 指定该软件包的名称

示例 B-1 测试软件包

创建软件包后，必须通过使用 `pkgadd` 的 `-R dir_name` 选项将该软件包安装到备用根 (/) 文件系统位置，以便对该软件包进行测试。安装完软件包后，必须使用 `pkgchk` 命令检查软件包的正确性，如本例所示。

```
# pkgadd -d . -R /a SUNWvxxm
# pkgchk -R /a SUNWvxxm
```

不应显示任何错误。

示例 B-2 在 /export/SUNWvxxm 中测试软件包

如果软件包位于 /export/SUNWvxxm，则可发出以下命令。

```
# pkgchk -d /export SUNWvxxm
```

不应显示任何错误。

在创建、修改和删除文件时，其他命令可检查软件包。以下是一些命令示例。

- 例如，`dircmp` 或 `fssnap` 命令可用于检验软件包是否运行正常。
- 同时，`ps` 命令可用于测试守护进程的适合性，方法是确保守护进程未被该软件包停止或启动。
- `truss`、`pkgadd -v` 和 `pkgrm` 命令可测试运行时软件包安装的适合性，但可能不会在所有情况下都有效。在以下示例中，`truss` 命令去除了所有只读的非 `$TMPDIR` 访问，而对于没有位于指定的非活动引导环境中的那些路径，该命令仅显示对它们的非只读访问。

```
# TMPDIR=/a; export TMPDIR
# truss -t open /usr/sbin/pkgadd -R ${TMPDIR} SUNWvxxm \
2>&1 > /dev/null | grep -v O_RDONLY | grep -v \
'open("${TMPDIR}
```

安装或升级时禁止用户交互

使用下列标准的 Solaris 实用程序时，必须在无用户输入信息提示的情况下，添加或删除软件包。

- 自定义 JumpStart 程序
- Solaris Live Upgrade
- Solaris 安装程序程序
- Solaris Zones

要测试软件包以确保该软件包能在无用户交互的情况下安装，可使用 `pkgadd` 命令的 `-a` 选项创建一个新管理文件。`-a` 选项将定义一个安装管理文件，该文件用来替代缺省的管理文件。使用缺省文件时可能会导致提示用户输入更多信息。您可以创建一个管理文件，该文件指示 `pkgadd` 应绕过这些检查并安装软件包，无需用户确认。有关详细信息，请参见手册页 [admin\(4\)](#) 或 [pkgadd\(1M\)](#)。

以下示例说明了 `pkgadd` 命令如何使用管理文件。

- 如果未提供任何管理文件，则 `pkgadd` 将使用 `/var/sadm/install/admin/default`。使用此文件可能会导致发生用户交互。

```
# pkgadd
```

- 如果在命令行上提供了一个相对管理文件，则 `pkgadd` 将在 `/var/sadm/install/admin` 中查找文件名并使用该文件。在本示例中，相对管理文件名为 `nocheck`，因此 `pkgadd` 将查找 `/var/sadm/install/admin/nocheck`。

```
# pkgadd -a nocheck
```

- 如果提供了绝对文件，则 `pkgadd` 将使用该文件。在此示例中，`pkgadd` 在 `/tmp` 中查找 `nocheck` 管理文件。

```
# pkgadd -a /tmp/nocheck
```

示例 B-3 安装管理文件

以下是一个安装管理文件示例，该安装管理文件几乎不需要与 `pkgadd` 实用程序进行用户交互。除非软件包需要的空间大于系统中的可用空间，否则 `pkgadd` 实用程序将使用此文件并安装软件包，而不会提示用户输入更多的信息。

```
mail=
instance=overwrite
partial=nocheck
runlevel=nocheck
idepend=nocheck
space=ask
setuid=nocheck
conflict=nocheck
action=nocheck
basedir=default
```

为区域设置软件包参数

软件包的参数可以控制软件包内容的分发方式，并使这些内容在安装于非全局区域的系统上可见。SUNW_PKG_ALLZONES、SUNW_PKG_HOLLOW和SUNW_PKG_THISZONE软件包参数定义安装了区域的系统上的软件包的特征。必须对这些参数进行设置，才能在安装了非全局区域的系统中管理软件包。

下表列出了设置软件包参数的四种有效组合。如果所选的设置组合不是表中所列的设置组合，则该设置无效，并将导致无法安装软件包。

注 - 请确保您已设置了全部三个软件包参数。可以将这三个软件包参数保留为空。如果不设置这些参数，软件包工具会将缺少区域软件包参数视为该设置设为"false"，但强烈建议您设置这些参数。通过设置全部三个软件包参数，可以指定安装或删除软件包时，软件包工具应当表现的确切行为。

表 B-3 区域的有效软件包参数设置

SUNW_PKG_ALLZONES 设置	SUNW_PKG_HOLLOW 设置	SUNW_PKG_THISZONE 设置	软件包说明
false	false	false	<p>此为软件包的缺省设置，该设置不会指定所有区域软件包参数的值。</p> <p>具有这些设置的软件包既可安装在全局区域中，也可安装在非全局区域中。</p> <ul style="list-style-type: none"> ■ 如果在全局区域中运行 pkgadd 命令，则会将软件包安装在全局区域和所有非全局区域中。 ■ 如果在非全局区域中运行 pkgadd 命令，则仅将软件包安装在非全局区域中。 <p>在这两种情况下，软件包的所有内容都会在其安装所在的所有区域中可见。</p>
false	false	true	<p>具有这些设置的软件包既可安装在全局区域中，也可安装在非全局区域中。如果在安装软件包之后创建新的非全局区域，则软件包不会传播到这些新的非全局区域。</p> <ul style="list-style-type: none"> ■ 如果在全局区域中运行 pkgadd 命令，则仅将软件包安装在全局区域中。 ■ 如果在非全局区域中运行 pkgadd 命令，则仅将软件包安装在非全局区域中。 <p>在这两种情况下，软件包的所有内容都会在其安装所在的区域中可见。</p>

表 B-3 区域的有效软件包参数设置 (续)

SUNW_PKG_ALLZONES 设置	SUNW_PKG_HOLLOW 设置	SUNW_PKG_THISZONE 设置	软件包说明
true	false	false	<p>具有这些设置的软件包只能安装在全局区域中。运行 <code>pkgadd</code> 命令时，会将软件包安装在全局区域和所有非全局区域中。软件包的所有内容在所有区域中可见。</p> <p>注- 任何将软件包安装在非全局区域中的尝试都会失败。</p>
true	true	false	<p>具有这些设置的软件包只能由全局管理员安装在全局区域中。运行 <code>pkgadd</code> 命令时，软件包的内容会全部安装在全局区域中。如果软件包的软件包参数设置为这些值，则不会在任何非全局区域中提供软件包内容本身。非全局区域中仅会安装使软件包显示为已安装状态所必需的软件包安装信息。这将安装依赖于该软件包的要安装的其他软件包。有关 "hollow" 软件包的更多信息，请参见《系统管理指南：Solaris Containers—资源管理和 Solaris Zones》中的第 24 章“关于安装了区域的 Solaris 系统上的软件包和修补程序（概述）”。</p> <p>为了检查软件包的相关性，该软件包显示为已安装在所有区域中。</p> <ul style="list-style-type: none"> ■ 在全局区域中，该软件包的所有内容均可见。 ■ 在完全根非全局区域中，该软件包的所有内容均不可见。 ■ 当非全局区域从全局区域中继承文件系统时，安装在该文件系统中的软件包在非全局区域中可见，而该软件包所提供的所有其他文件在非全局区域中均不可见。 <p>例如，稀疏根非全局区域 (sparse root non-global zone) 与全局区域共享某些目录。这些目录为只读目录。稀疏根非全局区域与其他区域共享 <code>/platform</code> 文件系统。另一个示例为软件包提供仅与引导硬件有关的文件。</p> <p>注- 任何将软件包安装在非全局区域中的尝试都会失败。</p>
说明			更多信息
有关软件包和区域的更多详细信息			《系统管理指南：Solaris Containers—资源管理和 Solaris Zones》中的第 24 章“关于安装了区域的 Solaris 系统上的软件包和修补程序（概述）”

说明	更多信息
有关稀疏根区域和完全根区域的概述	《系统管理指南：Solaris Containers—资源管理和 Solaris Zones》中的第 16 章“Solaris Zones 介绍”
有关软件包特征和参数的信息	<code>pkginfo(4)</code>
有关显示软件包参数值的信息	<code>pkgparam(1)</code>

有关背景信息

下列参考提供有关打包要求和特定命令语法的背景信息。

有关打包要求和术语定义的详细信息	《应用程序包开发者指南》中的第 6 章“创建软件包的高级技术”信息
有关添加和删除软件包以及安装管理文件的基本信息	《系统管理指南：基本管理》中的第 18 章“管理软件（概述）”
有关本附录中引用的特定命令的详细信息，请参见手册页	<code>dircmp(1)</code> , <code>fssnap(1M)</code> , <code>ps(1)</code> 、 <code>truss(1)</code> <code>pkgadd(1M)</code> , <code>pkgchk(1M)</code> 或 <code>pkgrm(1M)</code>
有关 Solaris Live Upgrade 的概述	《Solaris 10 10/08 安装指南：Solaris Live Upgrade 和升级规划》中的第 2 章“Solaris Live Upgrade（概述）”
有关自定义 JumpStart 的概述	第 2 章，自定义 JumpStart（概述）
有关 Solaris Zones 的概述	《系统管理指南：Solaris Containers—资源管理和 Solaris Zones》中的第 16 章“Solaris Zones 介绍”

词汇表

3DES	([三重 DES] 三重数据加密标准)。一种对称密钥加密方法，可提供 168 位密钥长度。
AES	(高级加密标准) 一种对称加密方法，可提供 128 位块数据加密技术。美国政府在 2000 年 10 月采用该种算法的 Rijndael 变体作为其加密标准。AES 代替 DES 加密作为政府标准。
bootlog-cgi 程序	一种 CGI 程序，使 Web 服务器可以在 WAN Boot 安装期间收集和存储远程客户机的引导和安装控制台的消息。
certstore 文件	一种文件，包含用于特定客户机系统的数字证书。在 SSL 协商过程中，可能会要求客户机向服务器提供证书文件。服务器使用该文件来验证客户机的身份。
CGI	(公共网关接口) 一种接口，外部程序通过该接口与 HTTP 服务器进行通信。为使用 CGI 而编写的程序称为 CGI 程序或 CGI 脚本。CGI 程序可以处理格式或分析输出，而服务器通常不能执行这些操作。
DES	(数据加密标准) 一种对称密钥加密方法，开发于 1975 年，并在 1981 年由 ANSI 标准化为 ANSI X.3.92。DES 使用 56 位密钥。
DHCP	(动态主机配置协议) 一种应用层协议。它使 TCP/IP 网络上的每台计算机或客户机可以从一台或多台指定的和集中维护的 DHCP 服务器上提取 IP 地址和其他网络配置信息。此工具减少了维护和管理大型 IP 网络的额外费用。
disk	一个或一套磁化介质的圆盘，形成同心的磁轨和扇区，用于存储文件等数据。另请参见光盘。
加密	通过使信息变得不可理解来防止未经授权的信息使用的过程。加密基于称为密钥的密码，密钥用于解密信息。另请参见 解密 。
/etc/netboot 目录	此目录位于 WAN Boot 服务器上，包含 WAN Boot 安装所需的客户机配置信息和安全数据。
/etc 目录	包含关键系统配置文件和维护命令的目录。
/export 文件系统	OS 服务器上与网络上的其他系统共享的文件系统。例如， <code>/export</code> 文件系统可以包含无盘客户机的根 (<code>/</code>) 文件系统和交换空间以及网络用户的起始目录。无盘客户机的引导和运行依赖于 OS 服务器上的 <code>/export</code> 文件系统。

fdisk 分区	磁盘驱动器的一个逻辑分区，专用于基于 x86 的系统上的特定操作系统。要安装 Solaris 软件，必须在基于 x86 的系统上至少设置一个 Solaris fdisk 分区。基于 x86 的系统允许在一个磁盘上最多设置四个不同的 fdisk 分区。这些分区可用于保留单独的操作系统。每个操作系统必须位于唯一的 fdisk 分区上。每个系统在每个磁盘上只能有一个 Solaris fdisk 分区。
文件系统	在 SunOS™ 操作系统中，您可以访问的文件和目录的树状网络。
format	将数据放入结构或将磁盘分成若干扇区以接收数据。
global zone (全局区域)	在 Solaris Zones 中，全局区域既是系统的缺省区域，也是用于系统范围内管理控制的区域。只能从全局区域配置、安装、管理或卸载非全局区域。只能在全局区域中进行系统基础设施（如物理设备）的管理、路由或动态重新配置 (dynamic reconfiguration, DR)。全局区域中运行的具有适当权限的进程可以访问与其他区域关联的对象。另请参见 <i>Solaris Zones</i> 和 <i>非全局区域</i> 。
GRUB	仅适用于 x86： GNU GRand Unified Bootloader (GRUB) 是具有简单菜单界面的开放源代码引导装载器。该菜单显示系统中已安装的操作系统列表。使用 GRUB 可以轻松地从各种不同的操作系统，如 Solaris OS、Linux 或 Microsoft Windows。
GRUB 编辑菜单	仅适用于 x86： 引导菜单，GRUB 主菜单的子菜单。该菜单中显示了 GRUB 命令。可以编辑这些命令以更改引导行为。
GRUB 主菜单	仅适用于 x86： 引导菜单，列出系统中已安装的操作系统。使用该菜单，无需修改 BIOS 或 fdisk 分区设置即可轻松地引导操作系统。
HMAC	用于进行消息验证的加密散列方法。HMAC 与重复加密散列函数（例如 MD5 或 SHA-1）以及机密共享密钥配合使用。HMAC 的加密能力取决于基础散列函数的特性。
HTTP	（超文本传输协议）(n)。从远程主机获取超文本对象的 Internet 协议。该协议基于 TCP/IP。
HTTPS	HTTP 的安全版本，通过使用安全套接字层 (SSL) 实现。
IPv6	IPv6 是 Internet 协议 (IP) 的一个版本（版本 6），是从当前版本 IPv4（版本 4）演化而来的版本。使用定义的转换机制部署 IPv6 不会破坏当前的操作。此外，IPv6 还提供了一个平台以便应用新的 Internet 功能。
JumpStart 安装	一种安装类型，在此类安装中，通过使用工厂安装的 JumpStart 软件自动将 Solaris 软件安装在系统上。
JumpStart 目录	当对自定义 JumpStart 安装使用配置文件软盘时，JumpStart 目录是软盘上包含全部基本自定义 JumpStart 文件的根目录。当对自定义 JumpStart 安装使用配置文件服务器时，JumpStart 目录是服务器上包含全部基本自定义 JumpStart 文件的目录。
Kerberos	一种网络验证协议，它使用强大的密钥加密技术，使客户机和服务器能够通过不安全的网络连接向对方证明自己的身份。
keystore 文件	一种文件，包含由客户机和服务器共享的密钥。在 WAN Boot 安装期间，客户机系统使用密钥验证服务器的完整性，或解密从服务器传输来的数据和文件。
LAN	（局域网）一组邻近的计算机系统，通过某种连接硬件和软件的方式可以进行通信。

LDAP	(轻量目录访问协议) 一种标准的、可扩展的目录访问协议，由 LDAP 命名服务客户机和服务器用于进行相互通信。
Manifest 段	Solaris Flash 归档文件的一个段，用于验证克隆系统。Manifest 段列出了系统上要保留的、要添加到克隆系统的或者要从克隆系统中删除的文件。该段仅用于提供信息，它以内部格式列出文件，不能用于脚本。
MD5	(消息摘要 5) 一种迭代式加密散列函数，用于进行消息验证 (包含数字签名)。该函数于 1991 年由 Rivest 开发。
menu.lst 文件	仅适用于 x86 ：列出系统中安装的所有操作系统的文件。该文件的内容指定在 GRUB 菜单上显示的操作系统列表。使用该菜单，无需修改 BIOS 或 fdisk 分区设置即可轻松地引导操作系统。
Miniroot	Solaris 安装介质中所包含的最小可引导根 (/) 文件系统。miniroot 包含安装和升级系统所需的 Solaris 软件。在基于 x86 的系统上，miniroot 被复制到系统中，用作故障安全引导归档文件。请参见 故障安全引导归档文件 。
NIS	SunOS 4.0 (最低版本) 网络信息服务。一种分布式网络数据库，包含有关网络上的系统和用户的关键信息。NIS 数据库存储在主服务器和全部从属服务器上。
NIS+	SunOS 5.0 (最低版本) 网络信息服务。NIS+ 取代了 NIS (即 SunOS 4.0 [最低版本] 网络信息服务)。
/opt 文件系统	包含第三方和非捆绑软件的挂载点的文件系统。
OS 服务器	向网络上的系统提供服务的系统。要为无盘客户机提供服务，OS 服务器必须为每个无盘客户机的根 (/) 文件系统和交换空间 (/export/root 和 /export/swap) 留出磁盘空间。
Power Management	自动保存系统状态并在系统空闲 30 分钟后将其关闭的软件。当您在符合美国环保署的能源之星标准的版本 2 的系统中安装 Solaris 软件时，系统将缺省安装 Power Management 软件。例如，基于 sun4u SPARC 的系统即是一种缺省安装 Power Management 的系统。接下来重新引导之后，系统会提示您启用或禁用 Power Management 软件。 能源之星标准要求系统或显示器在不活动之后自动进入“休眠状态” (功耗 30 瓦或更低)。
RAID-0 卷	一类可以是条状或串联的卷。这些组件也称为子镜像。条状和串联是镜像的基本生成块。
RAID-1 卷	一类通过保留多个副本复制数据的卷。RAID-1 卷由一个或多个称为子镜像的 RAID-0 卷组成。RAID-1 卷有时称为 镜像 。
RAID-Z 存储池	在可以用作 ZFS 存储池的多个磁盘上存储数据和奇偶校验的虚拟设备。RAID-Z 类似于 RAID-5。
root	项分层结构中的顶层项。根是其他所有项均由此向下派生的唯一项。请参见 根目录 或 根 (/) 文件系统 。
rules.ok 文件	生成的 rules 文件的版本。自定义 JumpStart 安装软件需要使用 rules.ok 文件将系统与配置文件进行匹配。要创建 rules.ok 文件， 必须使用 check 脚本 。

rules 文件	一种文本文件，包含用于您希望自动安装的每组系统或单个系统的规则。每个规则根据一个或多个系统属性区分一组系统。rules 文件将每组系统链接到一个配置文件，该配置文件是定义如何在该组中的每个系统上安装 Solaris 软件的文本文件。rules 文件在自定义 JumpStart 安装中使用。另请参见 配置文件 。
server	管理资源并向客户机提供服务的网络设备。
SHA1	（安全散列算法）可以在长度小于 2^{64} 的任意输入上运行以生成消息摘要的算法。
Solaris DVD 或 CD 映像	安装在系统中的 Solaris 软件，该软件可以通过 Solaris DVD 或 CD 访问，也可以通过复制了 Solaris DVD 或 CD 映像的安装服务器的硬盘访问。
Solaris Flash	一种 Solaris 安装功能，利用该功能，您可以在一个系统（称为 主系统 ）上创建文件的归档文件。然后您可以使用此归档文件安装其他系统，使其他系统的配置与主系统相同。另请参见 归档文件 。
Solaris Live Upgrade	一种升级方法，使得当活动引导环境仍在运行时可以升级复制的引导环境，从而消除了生产环境的停机时间。
Solaris Zones	用于虚拟化操作系统服务的软件分区技术，提供安全的隔离环境以便运行应用程序。在创建非全局区域时，会构建一个应用程序执行环境，其中的进程与所有其他区域隔离。该隔离禁止一个区域中运行的进程监视或影响任何其他区域中运行的进程。另请参见 全局区域和非全局区域 。
Solaris 安装程序	一个图形用户界面 (GUI) 或命令行界面 (CLI) 安装程序，它使用向导面板指导您逐步安装 Solaris 软件和第三方软件。
sysidcfg 文件	一种文件，在其中可以指定一套预先配置系统的特殊系统配置关键字。
truststore 文件	一种文件，包含一个或多个数字证书。在 WAN Boot 安装期间，客户机系统将通过查看 truststore 文件中的数据来验证尝试执行安装的服务器的标识。
URL	（统一资源定位器）服务器和客户机用于请求文档的寻址系统。URL 通常被称为位置。URL 的格式是 <i>protocol://machine:port/document</i> 。 URL 样例为 <code>http://www.example.com/index.html</code> 。
/usr 文件系统	独立系统或服务器上包含许多标准 UNIX 程序的文件系统。与服务器共享大型 /usr 文件系统而不是维护本地副本，从而最大限度地减少在系统上安装和运行 Solaris 软件所需的总磁盘空间。
/var 文件系统	一种文件系统或目录（位于独立系统上），包含可能在系统的生命周期内不断更改或增长的系统文件。这些文件包括系统日志、vi 文件、邮件文件和 UUCP 文件。
WAN	（广域网）一种网络，通过使用电话、光纤或卫星链接连接位于不同地点的多个局域网 (LAN) 或系统。
WAN Boot Miniroot	已被修改为执行 WAN Boot 安装的一种 Miniroot。WAN Boot Miniroot 包含 Solaris Miniroot 中的软件的子集。另请参见 Miniroot 。

WAN Boot 安装	一种安装类型，使您可以使用 HTTP 或 HTTPS 通过广域网 (WAN) 引导和安装软件。WAN Boot 安装方法使您可以通过公共网络传输加密的 Solaris Flash 归档文件，以及在远程客户机上执行自定义 JumpStart 安装。
WAN Boot 服务器	一种 Web 服务器，提供在 WAN Boot 安装期间使用的配置文件和安全文件。
wanboot-cgi 程序	CGI 程序，用于检索和传输在 WAN Boot 安装中使用的数据和文件。
wanboot.conf 文件	一种文本文件，在其中可以指定执行 WAN Boot 安装所需的配置信息和安全设置。
wanboot 程序	二级引导程序，用于装入执行 WAN 引导安装所需的 WAN Boot Miniroot、客户机配置文件和安装文件。对于 WAN Boot 安装，wanboot 二进制程序执行类似 ufsboot 或 inetboot 二级引导程序的任务。
ZFS	使用存储池管理物理存储的文件系统。
安全套接字层	(SSL) 在双方（客户机和服务器）之间建立安全连接的软件库，用于实现 HTTPS，即 HTTP 的安全版本。
安装服务器	提供 Solaris DVD 或 CD 映像的服务器，网络上的其他系统可以通过该服务器（也称为 介质服务器 ）安装 Solaris。通过将 Solaris DVD 或 CD 映像复制到服务器的硬盘上可以创建安装服务器。
差别归档文件	一种 Solaris Flash 归档文件，仅包含两个系统映像（未更改的主映像和已更新的主映像）之间的差别。差别归档文件包含要在克隆系统上保留、更改或删除的文件。差别更新只更改指定的文件，并仅限于所包含的软件与未更改的主映像一致的系统。
超级用户	特殊用户，具有执行所有管理任务的权限。该超级用户可以读取和写入任何文件，执行任何程序，以及给任何程序发出中止信号。
池	设备的逻辑组，用于说明可用 ZFS 存储的布局 and 物理特征。数据集的空间是从池中分配的。
初始安装	覆写当前运行的软件或初始化空白磁盘的安装。 Solaris OS 的初始安装将使用新版本的 Solaris OS 覆写一个或多个系统磁盘。如果您的系统没有运行 Solaris OS，则必须执行初始安装。如果您的系统正在运行可升级的 Solaris OS 版本，则初始安装会覆写磁盘并且不保留 OS 或本地修改。
串联	RAID-0 卷。如果片被串联，则数据将被写入第一个可用片，直到该片被写满。该片写满后，数据将依次写入下一个片。串联不提供数据冗余，除非它包含在镜像中。另请参见 RAID-0 卷。
磁盘配置文件	表示磁盘结构（例如，字节/扇区、标志、片）的文件。利用磁盘配置文件，您可以在单个系统中使用 pfinstall 命令来测试不同大小磁盘上的配置文件。
簇	软件包（软件模块）的逻辑集合。Solaris 软件分为多个 软件组 ，其中每一个软件组都由 簇 和 软件包 组成。
独立	不需要其他任何计算机支持的计算机。
方向键	数字小键盘上的四个方向键之一。

非联网系统	未连接到网络或不依赖于其他系统的系统。
非全局区域	在 Solaris 操作系统的单个实例中创建的虚拟操作系统环境。一个或多个应用程序可在非全局区域中运行，不与系统的其余部分交互。非全局区域也称为区域。另请参见 <i>Solaris Zones</i> 和 <i>全局区域</i> 。
根 (/) 文件系统	顶层文件系统，其他所有文件系统均由此向下派生。根 (/) 文件系统是挂载其他所有文件系统的基础，并且永远不会卸载。根 (/) 文件系统包含对系统操作至关重要的目录和文件，例如内核、设备驱动器和用于启动（引导）系统的程序。
根目录	顶层目录，其他所有目录均由此向下派生。
更新	一种安装过程或执行安装的过程，更改系统上同一类型软件。与升级不同，更新可能会使系统降级。与初始安装不同，系统上必须存在要安装的同一种类型软件，才能进行更新。
公钥	在公钥加密中使用的加密密钥。
公钥密码学	一种密码系统，它使用两种密钥：所有人都知道的公钥和只有消息接收者知道的私钥。
功能键	标有 F1、F2、F3 等的 10 个或更多个键盘键之一，这些键映射到特定的任务。
故障安全引导归档文件	仅适用于 x86： 当主引导归档文件被破坏时，用于进行恢复的引导归档文件。该引导归档文件用于在没有挂载根 (/) 文件系统的情况下启动系统。该引导归档文件在 GRUB 菜单中被称为故障安全。该归档文件的主要作用是重新生成通常用于引导系统的主引导归档文件。请参见 引导归档文件 。
挂载	访问某个磁盘上的目录的过程，该磁盘可以是正在提出挂载请求的计算机上的磁盘，也可以是网络上的远程磁盘。要挂载文件系统，您需要本地系统上的挂载点和要挂载的文件系统的名称（例如 /usr）。
挂载点	一种工作站目录，可在此目录下挂载远程计算机上的文件系统。
关键文件系统	Solaris OS 需要的文件系统。使用 Solaris Live Upgrade 时，这些文件系统在活动和非活动引导环境的 <code>vfstab</code> 文件中是独立的挂载点。示例文件系统有 <code>root (/)</code> 、 <code>/usr</code> 、 <code>/var</code> 以及 <code>/opt</code> 。这些文件系统总是从源环境被复制到非活动的引导环境。
光盘	光盘，与磁盘相对，与压缩光盘 (CD) 市场上的常用拼写一致。例如，CD-ROM 或 DVD-ROM 就是光盘。
归档文件	一种文件，包含从主系统复制的文件的集合。该文件还包含有关归档文件的标识信息，例如名称和归档文件的创建日期。当您在系统上安装归档文件后，该系统就将包含主系统的确切配置信息。 归档文件可以是差别归档文件，即仅包含两种系统映像（未更改的主映像和已更新的主映像）之间的差别的 Solaris Flash 归档文件。差别归档文件包含要在克隆系统上保留、更改或删除的文件。差别更新只更改指定的文件，并仅限于所包含的软件与未更改的主映像一致的系统。
规则	为配置文件指定一个或多个系统属性的一系列值。规则在自定义 JumpStart 安装中使用。

核心软件组	一种软件组，包含在系统中引导和运行 Solaris OS 所需的最低数量的软件。核心软件组包括一些运行公用桌面环境 (CDE) 桌面所需的联网软件和驱动程序。核心软件组不包括 CDE 软件。
回退	返回到先前运行的环境。当您正在激活一个环境，而指定用于引导的引导环境失败或出现不良行为时，可以使用回退。
交换空间	临时保留内存区内容（直至它被重新装入内存）的片或文件。也称为 <code>/swap</code> 或 <code>swap</code> 卷。
结束脚本	一种用户定义的 Bourne shell 脚本，在 <code>rules</code> 文件中指定，该脚本在 Solaris 软件安装在系统上之后、系统重新引导之前执行任务。可以对自定义 JumpStart 安装使用结束脚本。
解密	将编码数据转换为纯文本的过程。另请参见 加密 。
介质服务器	请参见 安装服务器 。
精简网络支持软件组	一种软件组，包含在有限的网络服务支持下引导和运行 Solaris 系统所需的最少数量的代码。精简网络软件组提供基于多用户文本的控制台和系统管理实用程序。该软件组还使系统能够识别网络接口，但不能激活网络服务。
镜像	请参见 <i>RAID-1</i> 卷。
卷	一组物理片或其他卷，在系统中显示为单个逻辑设备。对应用程序或文件系统来说，卷在功能上等同于物理磁盘。 在某些命令行公用程序中，卷称作元设备。在标准 UNIX 术语中，卷也称为 伪设备 或 虚拟设备 。
卷管理器	一种程序，提供管理 DVD-ROM、CD-ROM 和软盘上的数据和获得对这些数据的访问的机制。
开发者 Solaris 软件组	一种软件组，包含最终用户 Solaris 软件组以及用于开发软件的库（包括文件、手册页和编程工具）。
开始脚本	用户定义的 Bourne shell 脚本，在 <code>rules</code> 文件中指定，该脚本在 Solaris 软件安装在系统上之前执行任务。只能对自定义 JumpStart 安装使用开始脚本。
可共享文件系统	像 <code>/export/home</code> 和 <code>/swap</code> 这样的用户自定义文件的文件系统。当您使用 Solaris Live Upgrade 时，这些文件系统在活动和非活动的引导环境之间可以共享。可共享文件系统在活动和非活动引导环境的 <code>vfstab</code> 文件中包含同一挂载点。更新活动引导环境中的可共享文件也会更新非活动引导环境中的数据。缺省情况下，可共享文件系统是共享的，但是您可以指定目标盘片，然后复制该文件系统。
克隆系统	通过使用 Solaris Flash 归档文件安装的系统。克隆系统与主系统具有相同的安装配置。
客户机	在用于通信的客户机/服务器模型中，客户机是远程访问计算服务器资源（例如计算能力和大容量内存）的处理机。
快照	ZFS 文件系统或卷在指定时间点的只读映像。
联网系统	为了通信和共享信息而通过硬件和软件连接起来的一组系统（称为主机）。称为局域网 (LAN)。系统联网时通常需要一个或多个服务器。

逻辑设备	位于一个或多个磁盘上的一组物理片，在系统中显示为单个设备。逻辑设备在 Solaris 卷管理器中称为卷。对应用程序或文件系统来说，卷在功能上等同于物理磁盘。
密钥	用于加密或解密数据的密码。另请参见 加密 。
面板	用于组织窗口、对话框或 applet 中内容的容器。面板可以收集并确认用户输入。面板可由向导使用，并按照一定的顺序来完成指定的任务。
名称服务器	为网络上的系统提供命名服务的服务器。
命令行	一种字符串，以命令开始，其后常跟参数（包括选项、文件名和其他表达式），以行结束符结束。
命名服务	一个分布式网络数据库，它包含网络上所有系统的关键系统信息，以便系统能够彼此通信。使用命名服务，可以在网络范围的基础上维护、管理和访问系统信息。如果不使用命名服务，则每个系统必须在本地 <code>/etc</code> 文件中维护各自的系统信息副本。Sun 支持以下命名服务：LDAP、NIS 和 NIS+。
派生的配置文件	一种在自定义 JumpStart 安装过程中由开始脚本动态创建的配置文件。
配置文件	一种文本文件，用于定义在使用自定义 JumpStart 方法时如何安装 Solaris 软件。例如，配置文件可以定义要安装的软件组。每个规则指定一个配置文件，如果系统与此规则相匹配，该文件就可以定义系统的安装方式。通常为每个规则创建一个不同的配置文件。但是，同一配置文件可以用于多个规则。另请参见 <i>rules</i> 文件。
配置文件服务器	在 JumpStart 目录中包含全部基本自定义 JumpStart 文件的服务器。
配置文件软盘	在根目录（JumpStart 目录）中包含全部基本自定义 JumpStart 文件的软盘。
片	软件将磁盘空间分成的单元。
平台名称	<code>uname -i</code> 命令的输出。例如，Ultra 60 的平台名称是 SUNW,Ultra-60。
平台组	供应商出于发行特定软件的目的定义的硬件平台组。i86pc 和 sun4u 都属于有效的平台组。
区域	请参见 非全局区域
软件包	组合成单个实体以进行模块化安装的软件集合。Solaris 软件分为多个软件组，其中每一个软件组都由簇和软件包组成。
软件组	Solaris 软件的逻辑分组（簇和软件包）。在 Solaris 安装期间，您可以安装以下软件组之一：核心软件组、最终用户 Solaris 软件组、开发者 Solaris 软件组或完整 Solaris 软件组，以及仅用于 SPARC 系统的完整 Solaris 软件组加 OEM 支持。
散列	通过进行一些输入并生成明显比输入短的数字而生成的数字。同一个输出值始终针对同一个输入生成。散列函数可用于表搜索算法、错误检测和篡改检测。如果用于篡改检测，选择散列函数可以使得很难找到生成同一个散列结果的两个输入。MD5 和 SHA-1 是单向散列函数的示例。例如，消息摘要可以接受可变长度的输入（例如磁盘文件），然后将其缩减为较小的值。
散列法	将字符串更改为表示初始字符串的值或密钥的过程。
升级	一种安装过程，将文件和现有文件合并，并尽可能地保留修改。

	升级 Solaris OS 会将 Solaris OS 的新版本与系统的一个或多个磁盘上的现有文件合并。升级将最大限度地保留您对 Solaris OS 的前一版本所做的修改。
升级选项	Solaris 安装程序提供的一种选项。升级过程将新版本的 Solaris 与磁盘上现有的文件合并。升级还尽可能多地保存自上次安装 Solaris 以来的本地修改。
实用程序	一种标准程序，通常在购买计算机时免费装备，用于进行计算机的内务处理。
时区	地球表面 24 个经度分区中的任何一个，每个分区都规定了一个标准时间。
数据集	以下 ZFS 实体的通用名称：克隆、文件系统、快照或卷。
数字证书	一种不可传送、不可伪造的数字文件，由通信双方均已信任的第三方颁发。
私钥	在公钥加密中使用的解密密钥。
探测关键字	一种语法元素，当使用自定义 JumpStart 方法进行安装时，可以提取有关系统的属性信息。探测关键字不需要您按照规则的要求设置匹配条件以及运行配置文件。另请参见规则。
完整 Solaris 软件组	包含完整 Solaris 发行版的软件组。
完整 Solaris 软件组加 OEM 支持	包含完整 Solaris 发行版以及为 OEM 附加的硬件支持的软件组。在基于 SPARC 的服务器上安装 Solaris 软件时建议使用此软件组。
网络安装	一种安装软件的方式，通过网络将软件从带有 CD-ROM 或 DVD-ROM 驱动器的系统安装到没有 CD-ROM 或 DVD-ROM 驱动器的系统。网络安装需要名称服务器和安装服务器。
文档根目录	Web 服务器上的分层结构的根目录，其中包含要向正在访问 Web 服务器的用户显示的文件、映像和数据。
文件服务器	为网络上的系统存储软件和文件的服务器。
无盘客户机	网络上的一种客户机，它的所有磁盘存储均依赖于服务器。
系统配置文件	(system.conf) 一种文本文件，在其中可以指定要在 WAN Boot 安装中使用的 sysidcfg 文件和自定义 JumpStart 文件的位置。
校验和	用于对组进行校验的一组数据项的相加结果。数据项可以是数字，也可以是在计算校验和期间被视为数字的其他字符串。校验和的值可以验证两台设备之间的通信是否成功。
卸载	取消对某个磁盘目录的访问的过程，该磁盘可以连接到计算机上，也可以连接到网络上的远程磁盘。
修补程序分析器	可以手动运行或作为 Solaris 安装程序的一部分运行的脚本。修补程序分析器在您的系统上执行分析，以确定升级到 Solaris 更新时要删除的修补程序（如果有）。
虚拟设备	ZFS 池中的逻辑设备，可以是物理设备、文件或设备集合。
引导	将系统软件装入内存并启动该软件。
引导服务器	一种服务器系统，可以为同一网络子网上的客户机系统提供启动所需的程序和信息。如果安装服务器与要安装 Solaris 软件的系统位于不同的子网上，则必须通过网络安装引导服务器。

引导归档文件

仅适用于 x86： 引导归档文件是用于引导 Solaris OS 的关键文件集合。在挂载根 (/) 文件系统之前的系统启动过程中需要这些文件。系统中将维护以下两个引导归档文件：

- 在系统中用于引导 Solaris OS 的引导归档文件。该引导归档文件有时被称为主引导归档文件。
- 当主引导归档文件被破坏时，用于进行恢复的引导归档文件。该引导归档文件用于在没有挂载根 (/) 文件系统的情况下启动系统。在 GRUB 菜单中，该引导归档文件被称为故障安全。该归档文件的主要作用是重新生成通常用于引导系统的主引导归档文件。

引导环境

强制性文件系统（磁盘片和挂载点）的集合，对 Solaris OS 的运行至关重要。这些磁盘片可以位于同一磁盘上，也可以分布在多个磁盘上。

活动的引导环境是当前引导的引导环境。一次只能引导一个活动的引导环境。非活动引导环境是当前未引导的引导环境，但可以处于等待在下一次重新引导时被激活的状态。

引导装载机

仅适用于 x86： 引导装载机是打开系统后执行的第一个软件程序。此程序将开始执行引导过程。

硬链接

引用磁盘上的文件的目录项。多个此类目录项可以引用同一个物理文件。

语言环境

共享同一种语言、风俗或文化习俗的地理区域、政治区域或社区（美国英语表示为 en_US，英国英语表示为 en_UK）。

域

Internet 命名分层结构的一部分。域表示本地网络上一组共享管理文件的系统。

域名

分配给本地网络上一组共享管理文件的系统的名称。网络信息服务 (NIS) 数据库必须有域名才能正常工作。域名包括一系列用句点分隔的组件名称（例如：tundra.mpk.ca.us）。在一个域名中，越靠右的组件名称所标识的网域范畴越广（通常指远程区域）。

元设备

请参见卷。

证书颁发机构

(CA) 可信任的第三方组织或公司，可以颁发用于创建数字签名和公钥/私钥对的数字证书。CA 可以保证被授予唯一证书的个体的真实身份。

主机名

使网络上的其他系统能够识别某个系统的名称。该名称在特定域内（通常指的是在任何一个组织内）的所有系统中必须是唯一的。主机名可以是字母、数字和减号 (-) 的任意组合，但不能以减号开头或结束。

主系统

用于创建 Solaris Flash 归档文件的系统。系统配置保存在归档文件中。

主引导归档文件

引导归档文件，用于在系统中引导 Solaris OS。该引导归档文件有时被称为主引导归档文件。请参见 [引导归档文件](#)。

状态数据库

一种数据库，用于存储有关 Solaris 卷管理器配置状态的信息。状态数据库是多个复制的数据库副本的集合。每个副本都称为一个 **状态数据库副本**。状态数据库可以跟踪所有已知状态数据库副本的位置和状态。

状态数据库副本

状态数据库的副本。副本可以确保数据库中的数据有效。

子镜像

请参见 [RAID-0 卷](#)。

子网

为了简化路由将单个逻辑网络分为较小物理网络的解决方案。

子网掩码	用于从 Internet 地址中选择子网寻址位的位掩码。掩码长 32 位，它选择 Internet 地址的网络部分和 1 位或多位的本地部分。
自定义 JumpStart	一种安装类型，在此类安装中，基于用户自定义的配置文件，将 Solaris 软件自动安装在系统上。您可以为不同类型的用户和系统创建自定义的配置文件。自定义 JumpStart 安装是您创建的一种 JumpStart 安装。
自定义探测文件	一种必须与 <i>rules</i> 文件位于同一个 JumpStart 目录中的文件，它是一个 Bourne shell 脚本，包含两种函数：探测函数和比较函数。探测函数收集您需要的信息或进行实际的工作，并设置您定义的相应的 <i>SI</i> 环境变量。探测函数成为探测关键字。比较函数调用相应的探测函数并比较探测函数的输出，如果关键字匹配则返回 0，如果关键字不匹配则返回 1。比较函数成为规则关键字。另请参见 <i>rules</i> 文件。
最终用户 Solaris 软件组	一种软件组，包含核心软件组以及向最终用户推荐的软件，包括公用桌面环境 (CDE) 和台式软件。
作业	将由计算机系统完成的用户自定义任务。

索引

数字和符号

#

在 rules 文件中, 33

在配置文件中, 36

(/) 文件系统

JumpStart 设置的值, 140

&& (与) 规则字段, 33

!(感叹号) 规则字段, 33

A

add_install_client 命令, JumpStart 目录访问, 27

AND 规则字段, 33

any

规则关键字, 描述和值, 99, 144

探测关键字, 描述和值, 144

arch 规则关键字, 99, 144

arch 探测关键字, 144

archive_location 关键字, 105-110

auto_install_sample 目录

check 脚本, 51, 73

set_root_pw 结束脚本, 59, 60

将文件复制到 JumpStart 目录, 26, 30, 31

B

setup_install_server 命令的 -b 选项, 93

backup_media 关键字, 110-111

begin.log 文件, 54

boot_device 关键字, 112

bootenv createbe 关键字, 113

bootparams 文件

更新, 165

启用 JumpStart 目录访问, 28

C

-c 选项

pfinstall 命令, 50

add_install_client 命令, 95, 96

check 脚本

custom_probes.ok 文件创建, 72

custom_probes 文件验证, 72, 73

rules.ok 文件创建, 51

rules 文件验证, 51, 52, 73

测试规则, 52, 73

派生的配置文件和, 55

client_arch 关键字, 114

CLIENT MAC ADDR 错误消息, 164

client_root 配置文件关键字, 114

cluster 配置文件关键字

描述和值, 115-116, 116

示例, 37

CPU (处理器)

规则关键字, 99, 144

探测关键字, 144

.cshrc 文件, 59

custom_probes.ok 文件

创建, 72

描述, 72

custom_probes 文件

- 测试 custom_probes, 73
- 命名, 70
- 使用 check 进行验证, 72
- 通过使用 check 进行验证, 73
- 要求, 70

D

- dfstab 文件, 25, 93
- disks 探测关键字, 描述和值, 144
- disksize 规则关键字, 描述和值, 100, 144
- domainname 规则关键字, 100, 144
- domainname 探测关键字, 144
- dontuse 配置文件关键字, 117, 142

E

- eng_profile 示例, 93
- /etc/bootparams 文件
 - 启用 JumpStart 目录访问, 28, 165
- /etc/dfs/dfstab 文件, 25, 93
- /etc/mnttab 文件, 29

F

- fdisk 命令, 65
- fdisk 配置文件关键字
 - 描述和值, 117-119
 - 示例, 37
- filesys 关键字, 120-123, 123-124
- filesys 配置文件关键字
 - 描述和值, 120
 - 示例, 37
- finish.log 文件, 55

G

- geo 关键字, 124
- getfile: RPC 失败: 错误 5: RPC 超时消息, 28

H

- hostaddress 规则关键字, 100, 144
- hostaddress 探测关键字, 144
- hostname 规则关键字
 - 描述和值, 100, 144
 - 示例, 99-103
- hostname 探测关键字, 描述和值, 144

I

- install_config 命令, 28
- install_type 关键字, 125
- install_type 配置文件关键字
 - 测试配置文件, 50-51
 - 示例, 37
 - 要求, 36, 37
- installed 规则关键字, 描述和值, 101, 144
- installed 探测关键字, 描述和值, 144
- IP 地址
 - 规则关键字, 100, 144
 - 探测关键字, 144

J

- JumpStart 目录
 - rules 文件示例, 32
 - 创建
 - 服务器, 25
 - 基于 SPARC 的系统的软盘, 29
 - 基于 x86 的系统的软盘, 29, 31
 - 示例, 93
 - 复制文件
 - 安装文件, 26, 30, 31
 - 使用结束脚本, 56
 - 共享, 25, 93
 - 权限, 25, 29
 - 使用结束脚本添加文件, 56

K

- karch 规则关键字, 101, 144
- karch 探测关键字, 144

L

layout_constraint 关键字, 126-127
 le0: 无载体—收发器电缆问题消息, 160
 locale 关键字, 128

M

marketing_profile 示例, 94
 memsize 规则关键字, 描述和值, 101, 144
 memsize 探测关键字, 描述和值, 144
 metadb 配置文件关键字, 128-129
 mnntab 文件, 29
 model 规则关键字, 描述和值, 102, 144
 model 探测关键字, 描述和值, 144

N

network 规则关键字, 描述和值, 102, 144
 network 探测关键字, 描述和值, 144
 no_master_check 关键字, 129
 noneuclidean 配置文件关键字, 130

O

osname 规则关键字, 102, 144
 osname 探测关键字, 144

P

check 脚本的 -p 选项, 52, 73
 partitioning, 配置文件关键字, 134
 partitioning 关键字, 134
 pfinstall 命令, 47
 probe 规则关键字, 描述和值, 103
 profile 字段中的 = (等号), 54
 profile 字段中的等号 (=), 54
 prtvtoc 命令
 SPARC: 创建磁盘配置文件, 63
 x86: 磁盘配置文件创建, 65

R

check 脚本的 -r 选项, 52, 73
 root (/) 文件系统, 配置文件示例, 20
 root_device 关键字, 140
 rootdisk
 filesystem 的片值, 121
 JumpStart 设置的值, 140
 定义, 140
 RPC 超时消息, 28, 164
 RPC 失败: 错误 5: RPC 超时消息, 28
 rule_keyword 规则字段, 33
 rule_value 规则字段, 33
 rules
 语法, 33
 字段描述, 33
 rules.ok 文件
 创建, 51
 rules.ok 文件, 规则的匹配顺序, 34
 rules.ok 文件
 规则的匹配顺序, 77, 83
 rules.ok 文件
 描述, 51
 rules 文件
 测试规则, 52
 创建, 32
 多行规则, 33
 描述, 32
 命名, 33
 示例, 32
 添加规则, 33
 通过使用 check 进行验证, 52
 自定义 JumpStart 示例, 95
 语法, 33
 注释, 33
 自定义 JumpStart 示例, 94, 95
 rules 文件中的多行, 33
 rules 文件中的反斜杠, 33
 rules 文件中的换行, 33

S

add_install_client 命令的 -s 选项, 96
 set_root_pw 结束脚本, 59, 60

share 命令

共享 JumpStart 目录, 25, 93

shareall 命令, 25, 93

SI_PROFILE 环境变量, 55

Solaris 软件

发行版或版本

installed 规则关键字, 101, 144

installed 探测关键字, 144

osname 规则关键字, 102, 144

osname 探测关键字, 144

组, 115-116

配置文件示例, 37

升级, 116

Solaris 软件的版本

installed 规则关键字, 101, 144

installed 探测关键字, 144

osname 规则关键字, 102, 144

osname 探测关键字, 144

Solaris 软件的发行版

installed 规则关键字, 101, 144

installed 探测关键字, 144

osname 规则关键字, 102, 144

osname 探测关键字, 144

stty 命令, 80, 83

SUNWCall 组, 115-116

SUNWCprog 组, 115-116

SUNWCreq 组, 115-116

SUNWCrnet 组, 115-116

SUNWCuser 组, 115-116

SUNWCXall 组, 115-116

system_type 配置文件关键字

描述和值, 141

示例, 37

T

tip 行连接显示要求, 83

tip 行连接要求, 80

totaldisk 规则关键字, 103, 145

totaldisk 探测关键字, 145

U

UFS, 29

usedisk 配置文件关键字, 描述和值, 142

V

/var/sadm/system/logs/begin.log 文件, 54

/var/sadm/system/logs/finish.log 文件, 55

volcheck 命令, 29, 31

Z

ZFS

概述和规划, 147

关键字, 描述, 151

配置文件关键字

快速参考, 103

配置文件示例, 148

限制, 147

ZFS 的限制, 147

安

安全性, 超级用户口令, 59, 60

备

备选安装程序, 68

变

变量

SI_PROFILE, 55

SYS_MEMSIZE, 49

不

不是 UFS 文件系统消息, 160

测

测试

- 配置文件, 47, 50-51
- 验证 custom_probes 文件
 - 测试 custom_probes, 73
 - 使用 check, 72
- 验证 rules 文件
 - 测试规则, 52
 - 派生的配置文件和, 55
 - 使用 check, 52, 73
 - 自定义 JumpStart 示例, 95
- 验证 rules 文件
 - 使用 check, 51

超

- 超级用户口令, 使用结束脚本进行设置, 59
- 超时 RPC 错误, 164

处

处理器

- 规则关键字, 99, 144
- 探测关键字, 144

创

创建

- custom_probes.ok 文件, 72
- JumpStart 目录, 在服务器上, 25
- RAID-1 卷, 123-124
- rules.ok 文件, 51, 72
- rules 文件, 32
- UFS, 29
- 本地文件系统, 120-123
- 磁盘配置文件, 63
- 配置文件
 - 描述, 35
 - 派生, 54

磁

磁盘配置文件

- 创建
 - 基于 SPARC 的系统, 63
 - 基于 x86 的系统, 65
- 描述, 47, 63

大

大小

- tip 行连接显示尺寸, 80, 83
- 交换空间
 - 配置文件示例, 20
 - 无盘客户机, 114
 - 最大大小, 115
- 内存, 101, 144
- 硬盘
 - 根空间, 114
 - 规则关键字, 100, 103, 144, 145
 - 探测关键字, 144, 145

独

独立系统

- 配置文件示例, 37
- 自定义 JumpStart 安装示例, 18

分

分区

- fdisk 分区, 37, 117-119
- 排除磁盘, 117
- 配置文件关键字, 142
- 示例, 37

服

服务器

- JumpStart 目录创建, 25
- 根空间, 114

复

复制

JumpStart 安装文件, 26, 30, 31

JumpStart 目录文件, 56

感

感叹号 (!) 规则字段, 33

根

根 (/) 文件系统, 非活动的引导环境的软件包要求, 177

根环境, 使用结束脚本进行自定义, 59

更

更改缺省引导设备消息, 165

共

共享 JumpStart 目录, 25, 93

挂

挂载

通过 Solaris 安装, 55

开始脚本警告, 54

远程文件系统, 120

关

关键字

Solaris Flash 归档文件, 自定义 JumpStart, 105-110

探测, 69

归

归档文件

JumpStart 配置文件示例, 40, 41, 42, 43

关键字, 自定义 JumpStart, 105-110

规

规则

rootdisk 匹配规则, 140

测试有效性, 52, 73

多行规则, 33

派生的配置文件, 54, 55

匹配顺序, 34, 77, 83

示例, 34

字段描述, 34

规则关键字, 99

any, 描述和值, 99, 144

arch, 99, 144

disksize, 描述和值, 100, 144

domainname, 100, 144

hostaddress, 100, 144

hostname, 99-103, 144

installed, 描述和值, 101, 144

karch, 101, 144

memsize, 101, 144

model, 102, 144

network, 102, 144

osname, 102, 144

probe, 103

totaldisk, 103, 145

规则文件

通过使用 check 进行验证

派生的配置文件和, 55

规则字段中的 Bourne shell 脚本, 33

核

核心 Solaris 软件组, 115-116

基

基于 GRUB 的引导

- 安装, 83, 85
- 创建配置文件软盘, 31
- 命令参考, 87

交

交换文件系统

- 大小确定, 115
- 内存大小和, 115
- 配置文件示例, 20
- 无盘客户机交换空间, 114

脚

脚本

- 规则字段中的 Bourne shell 脚本, 33
- 结束脚本, 55, 60, 68
- 开始脚本, 53, 55, 68

结

结束规则字段, 描述, 34

结束脚本

- 规则字段, 34
- 设置系统的超级用户口令, 59
- 添加软件包和修补程序, 56
- 自定义根环境, 59

精

精简网络支持软件组, 115-116

警

- 警告: 更改缺省引导设备, 165
- 警告: 时钟快 xxx 天消息, 160

开

开发者 Solaris 软件组, 115-116

- 配置文件示例, 37

开始规则字段, 描述, 33

开始脚本

- 创建派生的配置文件, 54, 55
- 概述, 53
- 规则字段, 33
- 权限, 54
- 站点特定的安装程序, 68

口

口令, 超级用户, 59, 60

令

令牌环卡, 引导错误, 164

路

路径, check 脚本, 52, 73

逻

逻辑 AND 规则字段, 33

名

名称/命名

- custom_probes 文件, 70
- rules 文件, 33
- 派生配置文件的名称, 55
- 系统型号名称, 102, 144
- 主机名, 100, 144

目

目录

JumpStart

- rules 文件示例, 32
- 创建目录, 93
- 复制安装文件, 26, 30, 31
- 复制文件, 56
- 共享目录, 25, 93
- 权限, 25, 29
- 添加文件, 56
- 为系统创建, 29

转到

- JumpStart 目录, 51
- 本地磁盘上 Solaris SPARC 软件的映像, 30
- 本地磁盘上 Solaris 软件的映像, 26
- 本地磁盘上基于 Solaris x86 的软件的映像, 31
- 到 JumpStart 目录, 73

内

内存

- 规则关键字, 101, 144
- 交换空间大小和, 115
- 探测关键字, 144

派

派生的配置文件, 54, 55

配

配置, 创建磁盘配置文件, 63

配置文件

- 测试, 50-51
- 创建, 35
- 规则字段, 34
- 描述, 35
- 命名, 36
- 派生的配置文件, 54, 55
- 匹配系统, 34, 77, 83
- 示例, 37
 - eng_profile, 93

配置文件, 示例 (续)

- marketing_profile, 94
- Solaris Flash, 40, 41, 42, 43
- WAN Boot 安装, 41
- ZFS, 148
- 要求, 33, 36
- 注释, 36
- 配置文件关键, 区分大小写, 103
- 配置文件关键字, 103, 142
 - archive_location, 105-110
 - backup_media, 110-111
 - boot_device, 112
 - bootenv createbe, 113
 - client_arch, 114
 - client_root, 114
 - client_swap, 114
 - cluster
 - 描述和值, 115-116, 116
 - 示例, 37
 - dontuse
 - usedisk 和, 142
 - 描述和值, 117
 - fdisk
 - 描述和值, 117-119
 - 示例, 37
 - filesystems
 - RAID-1 卷, 123-124
 - 本地文件系统, 120-123
 - 描述和值, 120
 - 示例, 37
 - 远程文件系统, 120
 - forced_deployment, 描述和值, 124
 - geo
 - 描述和值, 124
 - install_type
 - 描述和值, 125
 - 示例, 37
 - 适用于 ZFS, 153
 - 要求, 36, 37
 - layout_constraint, 描述和值, 126-127
 - local_customization, 描述和值, 128
 - locale, 描述和值, 128
 - metadb
 - 描述和值, 128-129

配置文件关键字,metadb (续)

- 示例, 37
- no_master_check, 描述和值, 129
- noneuclidean, 130
- partitioning
 - 描述和值, 134
- root_device, 140
- system_type
 - 描述和值, 141
 - 示例, 37
- usedisk, 描述和值, 142
- 创建状态数据库副本 (meatball), 128-129
- 分区
 - examples, 37
 - 排除磁盘, 117
 - 示例, 37
 - 指定磁盘, 142
- 快速参考, 103
- 适用于 ZFS 的 bootenv installbe, 152
- 适用于 ZFS 的 pool, 153
- 适用于 ZFS 的 root_device, 155

匹

匹配

- rootdisk 值, 140
- 规则顺序, 34, 77, 83
- 派生的配置文件, 54

片

片

- 规则关键字, 101, 144
- 配置文件示例, 37
- 探测关键字, 144

平

平台

- 规则关键字, 101, 144
- 匹配系统属性和配置文件, 34, 77, 83
- 探测关键字, 144

平台 (续)

- 无盘客户机, 114
- 系统型号名称, 102, 144

启

- 启动, check 脚本, 51, 52

权

权限

- JumpStart 目录, 25, 29
- 结束脚本, 55
- 开始脚本, 54

缺

缺省

- 分区
 - 排除磁盘, 117
 - 指定磁盘, 142
- 派生配置文件的名称, 55
- 已安装的软件组, 116

日

日志文件

- 结束脚本输出, 55
- 开始脚本输出, 54

软

软件包

- Solaris Live Upgrade
 - 要求, 177
- 管理文件, 53
- 使用自定义 JumpStart 时的要求, 177
- 添加
 - 使用 chroot, 59
 - 使用结束脚本, 56

软件组

- 配置文件示例, 37
- 升级, 116
- 用于配置文件, 115-116

软盘

- JumpStart 目录访问, 27
- x86: JumpStart 目录, 29

删

- 删除, 升级时删除簇, 116

升

升级

- 配置文件关键字, 116, 125, 134
- 升级失败, 170
- 自定义 JumpStart 安装, 75
- 升级失败, 重新引导问题, 170

时

- 时钟快 xxx 天消息, 160

收

- 收发器电缆问题消息, 160

输

输出文件

- 结束脚本日志, 55
- 开始脚本日志, 54

探

探测关键字

- arch, 144
- disks, 144

探测关键字 (续)

- domainname, 144
- hostaddress, 144
- hostname, 144
- installed, 144
- karch, 144
- memsize, 144
- model, 144
- network, 144
- osname, 144
- rootdisk, 144
- totaldisk, 145

添

添加

- 规则到 rules 文件, 33
- 软件组中的软件包, 130
- 升级时添加簇, 116
- 使用结束脚本添加软件包和修补程序, 56

完

- 完整 Solaris 软件组, 115-116
- 完整 Solaris 软件组加 OEM 支持, 115-116

网

- 网络安装, 自定义 JumpStart 安装, 示例, 19
- 网络号, 102, 144

微

微处理器

- 规则关键字, 99, 144
- 探测关键字, 144

未

- 未知的客户机错误消息, 159

文

文件和文件系统

UFS 创建, 29

创建

RAID-1 卷, 123-124

本地文件系统, 120-123

复制

JumpStart 安装文件, 26, 30, 31

使用结束脚本复制 JumpStart 目录文件, 56

挂载远程文件系统, 120

结束脚本输出, 55

开始脚本输出, 54

无

无法从文件/设备引导消息, 160

无盘客户机

交换空间, 114

平台, 114

无载体—收发器电缆问题消息, 160

显

显示

tip 行连接要求, 80, 83

修

修补程序

添加

使用 chroot, 59

使用结束脚本, 56

验

验证

custom_probes 文件

测试, 73

使用 check, 73

rules 文件

测试规则, 52

验证, rules 文件 (续)

派生的配置文件和, 55

使用 check, 51, 52, 73

自定义 JumpStart 示例, 95

要

要求

custom_probes 文件, 70

配置文件, 33, 36

疑

疑难解, 从错误的服务器导, 164

疑难解答

常见安装问题

使用 DHCP 从网络引导, 164

使用 DHCP 从网络引导, 164

一般安装问题

引导系统, 164

引

引导

创建配置文件软盘, 31

使用 GRUB, 命令参考, 87

使用 GRUB 安装, 83, 85

引导: 无法打开 /kernel/unix 消息, 160

硬

硬盘

partitioning

配置文件关键字, 134

rootdisk 值, 140

大小

根空间, 114

规则关键字, 100, 103, 144, 145

探测关键字, 144, 145

分区

排除以使用缺省分区, 117

硬盘,分区 (续)

- 示例, 37
- 指定以使用缺省分区, 142
- 挂载, 120
- 交换空间
 - 配置文件示例, 20, 37
 - 无盘客户机, 114
 - 最大大小, 115

与

- 与 (&&) 规则字段, 33

域

- 规则关键字, 100, 144
- 探测关键字, 144

远

- 远程文件系统, 挂载, 120

站

- 站点特定的安装程序, 68

注

注释

- 在 rules 文件中, 33
- 在配置文件中, 36

转

转到目录

- JumpStart 目录, 51
- 本地磁盘上 Solaris SPARC 软件的映像, 30
- 本地磁盘上 Solaris 软件的映像, 26

转到目录 (续)

- 本地磁盘上基于 Solaris x86 的软件的映像, 31
- 到 JumpStart 目录, 73

准

- 准备安装, 使用自定义 JumpStart, 20, 52

自

- 自定义 JumpStart 安装, 75
- tip 行连接要求, 80, 83
- 概述, 20
- 可选功能, 53
 - 概述, 53
 - 结束脚本, 55, 60
 - 开始脚本, 53, 55
 - 站点特定的安装程序, 68
- 描述, 20
- 配置文件关键字, 103
- 示例, 89, 97
 - check 脚本, 95
 - eng_profile 创建, 93
 - JumpStart 目录, 93
 - marketing_profile 创建, 94
 - RAID-1 卷配置文件, 43, 45
 - rules 文件编辑, 94, 95
 - Solaris Flash 配置文件, 40, 41, 42, 43
 - WAN Boot 安装配置文件, 41
 - 独立系统, 18
 - 非联网的, 18
 - 工程设计系统的设置, 95
 - 联网的, 19
 - 市场营销系统安装, 92
 - 市场营销系统的设置, 96
 - 引导和安装, 97
 - 站点安装, 89, 90
- 引导和安装, 75
- 准备, 20, 52

最

最终用户 Solaris 软件组, 115-116

