

**IBM SPSS Modeler 18.1.1
Solution Publisher**

IBM

注释

在使用本资料及其支持的产品之前，请阅读第 31 页的『声明』中的信息。

产品信息

本版本适用于 IBM SPSS Modeler V18.1.1 及所有后续发行版和修订版，直到在新版本中另有声明为止。

目录

前言	v
第 1 章 关于 IBM SPSS Modeler	1
IBM SPSS Modeler 产品	1
IBM SPSS Modeler	1
IBM SPSS Modeler Server	1
IBM SPSS Modeler Administration Console	2
IBM SPSS Modeler Batch	2
IBM SPSS Modeler Solution Publisher	2
IBM SPSS Collaboration and Deployment Services 的 IBM SPSS Modeler Server 适配器	2
IBM SPSS Modeler 版本	2
文档	3
SPSS Modeler Professional 文档	3
SPSS Modeler Premium 文档	3
应用程序示例	4
Demos 文件夹	4
许可证跟踪	4
第 2 章 IBM SPSS Modeler Solution Publisher	5
IBM SPSS Modeler Solution Publisher 的工作原理	5
发布流	5
第 3 章 执行已发布的流	7
使用参数文件定制流执行	8
将 IBM SPSS Modeler Solution Publisher 嵌入到应用程序中	9
IBM SPSS Modeler Solution Publisher 样本应用程序	10

第 4 章 安装 IBM SPSS Modeler Solution Publisher Runtime	11
安装 Runtime	11
在 Windows 中安装	11
在 UNIX 中安装	11
安装的故障排除	12
启动 Runtime	13
更改 Temp 目录	13
卸载 IBM SPSS Modeler Solution Publisher Runtime	14
Windows 卸载过程	14
UNIX 卸载过程	14
第 5 章 IBM SPSS 嵌入式预测建模 API 15	
第 6 章 IBM SPSS Modeler Solution Publisher Runtime Library API 参考	17
简介	17
API 流程概述	17
API 函数	18
声明	31
商标	32
产品文档的条款和条件	32
索引	35

前言

IBM® SPSS® Modeler 是 IBM Corp. 企业级数据挖掘工作平台。SPSS Modeler 通过深度的数据分析帮助组织改进与客户和市民的关系。组织通过借助源自 SPSS Modeler 的洞察力可以留住优质客户，识别交叉销售机遇，吸引新客户，检测欺诈，降低风险，促进政府服务交付。

SPSS Modeler 的可视化界面让用户可以应用他们自己的业务专长，这将生成更加强有力的预测模型，缩减实现解决方案所需时间。SPSS Modeler 提供了多种建模技术，例如预测、分类、分割和关联检测算法。模型创建成功后，通过 IBM SPSS Modeler Solution Publisher，在广泛的企业内交付给决策者，或通过数据库交付。

关于 IBM Business Analytics

IBM Business Analytics 软件提供完整、一致和正确的信息，决策人依据此信息来提高业务性能。企业智能、预测分析、财务业绩和战略管理的完整产品组合，和分析应用程序一起提供对当前业绩的清晰、直接和实用的洞察力，以及预测未来结果的能力。结合丰富的行业解决方案，久经证明的实践和专业服务，各种规模的组织都能够实现最高生产力、确信地自动作出决策以及获得更好的结果。

作为此产品服务组合的组成部分，IBM SPSS Predictive Analytics 软件可帮助组织预测未来事件，并在该洞察的基础上提前行动以实现更好的业务结果。全球的商业、政府和学术客户依靠 IBM SPSS 技术作为竞争优势来吸引、挽留和增加客户，同时减少欺诈和降低风险。通过在日常活动中融入 IBM SPSS 软件，组织机构将成为有预见力的企业，即能够指引决策并实现决策的自动化，从而满足企业目标并实现可衡量的竞争优势。有关详细信息或要联系一位代表，请访问 <http://www.ibm.com/spss>。

技术支持

技术支持可供维护客户使用。客户可就 IBM Corp. 产品使用问题或某一受支持硬件环境的安装帮助寻求技术支持。要获取技术支持，请访问 IBM Corp. Web 站点 <http://www.ibm.com/support>。请求帮助时，请准备好标识您自身、组织和支持协议。

第 1 章 关于 IBM SPSS Modeler

IBM SPSS Modeler 是一组数据挖掘工具，通过这些工具可以采用商业技术快速建立预测性模型，并将其应用于商业活动，从而改进决策过程。IBM SPSS Modeler 参照行业标准 CRISP-DM 模型设计而成，可支持从数据到更优商业成果的整个数据挖掘过程。

IBM SPSS Modeler 提供了各种借助机器学习、人工智能和统计学的建模方法。通过建模选项板中的方法，您可以根据数据生成新的信息以及开发预测模型。每种方法各有所长，同时适用于解决特定类型的问题。

SPSS Modeler 可以作为独立产品购买，也可以作为客户机与 SPSS Modeler Server 一起使用。同时提供了大量其他选项，以下各节将对这些选项进行概述。有关更多信息，请参阅<https://www.ibm.com/analytics/us/en/technology/spss/>。

IBM SPSS Modeler 产品

IBM SPSS Modeler 系列产品及关联的软件包括以下各项。

- IBM SPSS Modeler
- IBM SPSS Modeler Server
- IBM SPSS Modeler Administration Console (包含在 IBM SPSS Deployment Manager 中)
- IBM SPSS Modeler Batch
- IBM SPSS Modeler Solution Publisher
- IBM SPSS Collaboration and Deployment Services 的 IBM SPSS Modeler Server 适配器

IBM SPSS Modeler

SPSS Modeler 是具有完整功能的产品，它安装并运行于个人计算机上。您可以在本地方式作为独立产品运行 SPSS Modeler，也可以在分布方式下将其与 IBM SPSS Modeler Server 一起使用来提高大型数据集的性能。

借助 SPSS Modeler，您可以快速直接地构建准确的预测模型，而不进行编程。通过使用唯一可视界面，您可以轻松地查看数据挖掘过程。借助该产品随附的高级分析支持，您可以发现数据中先前隐藏的模式和趋势。您可以构建结果模型并了解影响结果的因素，从而利用业务机会并降低风险。

SPSS Modeler 推出了两个版本：SPSS Modeler Professional 和 SPSS Modeler Premium。请参阅主题第 2 页的『IBM SPSS Modeler 版本』，了解更多信息。

IBM SPSS Modeler Server

SPSS Modeler 使用客户端/服务器体系结构将资源集约型操作的请求分发给功能强大的服务器软件，因而使大数据集的传输速度大大加快。

SPSS Modeler Server 是一个单独授权的产品，在分布分析方式下，该产品在安装了一个或多个 IBM SPSS Modeler 的服务器主机上持续运行。这种运行方式大大提高了 SPSS Modeler Server 对大型数据集的处理速度，因为在服务器上可以运行耗用内存的操作，并且无需将数据下载到客户端计算机上。IBM SPSS Modeler Server 还提供对 SQL 优化和数据库内建模功能的支持，从而在性能和自动化方面带来更多优势。

IBM SPSS Modeler Administration Console

Modeler Administration Console 是一个图形用户界面，用于管理多个 SPSS Modeler Server 配置选项，这些选项还可以通过选项文件进行配置。控制台包含在 IBM SPSS Deployment Manager，可以用于监视和配置 SPSS Modeler Server 安装，并且可供当前 SPSS Modeler Server 客户免费使用。应用程序只能安装在 Windows 计算机上；但是它可以管理安装在任何受支持平台上的服务器。

IBM SPSS Modeler Batch

数据挖掘通常是交互过程，因此，还可以从命令行运行 SPSS Modeler 而不需要图形用户界面。例如，您可能具有长时间运行或重复任务，并且希望在用户不进行干预的情况下执行这些任务。SPSS Modeler Batch 是该产品的一个特殊版本，可提供对 SPSS Modeler 完整分析性能的支持，而无需访问常规的用户界面。要使用 SPSS Modeler Batch，需要 SPSS Modeler Server。

IBM SPSS Modeler Solution Publisher

SPSS Modeler Solution Publisher 是一个支持创建 SPSS Modeler 流的打包版本的工具，该版本的流可以由外部运行时引擎运行或嵌入到外部应用程序中。通过这种方式，您可以发布和部署完整的 SPSS Modeler 流以用于未安装 SPSS Modeler 的环境。SPSS Modeler Solution Publisher 作为 IBM SPSS Collaboration and Deployment Services - Scoring 服务的组成部分分发，需要单独的许可证。通过此许可证，您可以接收 SPSS Modeler Solution Publisher Runtime，它使您能够执行已发布的流。

有关 SPSS Modeler Solution Publisher 的更多信息，请参阅 IBM SPSS Collaboration and Deployment Services 文档。IBM SPSS Collaboration and Deployment Services Knowledge Center 包含名为"IBM SPSS Modeler Solution Publisher"和"IBM SPSS Analytics Toolkit"的部分。

IBM SPSS Collaboration and Deployment Services 的 IBM SPSS Modeler Server 适配器

IBM SPSS Collaboration and Deployment Services 的一些适配器使 SPSS Modeler 和 SPSS Modeler Server 能够与 IBM SPSS Collaboration and Deployment Services 存储库进行交互。通过这种方式，部署到存储库的 SPSS Modeler 流可以由多个用户共享，或者从瘦客户机应用程序 IBM SPSS Modeler Advantage 进行访问。请将适配器安装在托管存储库的系统上。

IBM SPSS Modeler 版本

SPSS Modeler 推出了下列版本。

SPSS Modeler Professional

SPSS Modeler Professional 提供处理大多数类型的结构化数据所需要的所有工具，例如 CRM 系统中跟踪的行为和交互、人口统计信息、采购行为和销售数据。

SPSS Modeler Premium

SPSS Modeler Premium 是一个单独授权的产品，它对 SPSS Modeler Professional 进行了扩展，以便后者能够处理专门的数据和非结构化文本数据。SPSS Modeler Premium 包含 IBM SPSS Modeler Text Analytics：

IBM SPSS Modeler Text Analytics 采用了先进语言技术和自然语言处理 (NLP)，以快速处理大量非结构化文本数据，提取和组织关键概念，以及将这些概念分为各种类别。提取的概念和类别可以和现有结构化数据中

进行组合（例如人口统计学），并且可用于借助 IBM SPSS Modeler 的一整套数据挖掘工具来进行建模，以此实现更好更集中的决策。

IBM SPSS Modeler Subscription

IBM SPSS Modeler Subscription 提供与传统 IBM SPSS Modeler 客户机完全相同的预测性分析功能。通过 Subscription 版本，您可以定期下载产品更新。

文档

文档可以从 SPSS Modeler 中的“帮助”菜单获取。这样会打开可在产品外部公开访问的 Knowledge Center。

作为产品下载的一部分，还会在单独的压缩文件夹中以 PDF 格式提供每个产品的完整文档（包括安装指示信息）。也可以从 Web 下载 PDF 文档，地址为：<http://www.ibm.com/support/docview.wss?uid=swg27046871>。

SPSS Modeler Professional 文档

SPSS Modeler Professional 文档套件（安装指示信息除外）如下。

- **IBM SPSS Modeler 用户指南。**使用 SPSS Modeler 的一般使用介绍，包括如何构建数据流、处理缺失值、构建 CLEM 表达式处理项目和报告以及将用于部署的流打包为 IBM SPSS Collaboration and Deployment Services 或 IBM SPSS Modeler Advantage。
- **IBM SPSS Modeler Source、Process 和 Output 节点。**描述用于以不同格式读取、处理和输出数据的所有节点。实际上这表示所有节点而非建模节点。
- **IBM SPSS Modeler Modeling 节点。**描述所有用于创建数据挖掘模型的节点。IBM SPSS Modeler 提供了各种借助机器学习、人工智能和统计学的建模方法。
- **IBM SPSS Modeler 应用程序指南。**本指南中的示例旨在为具体的建模方法和技术提供具有针对性的简介。还可以从“帮助”菜单获取本指南的在线版本。请参阅主题第 4 页的『应用程序示例』以获取更多信息。
- **IBM SPSS Modeler Python 脚本编制和自动化。**通过编写 Python 脚本实现系统自动化的相关信息，其中包括可以用于处理节点和流的属性的信息。
- **IBM SPSS Modeler 部署指南。**有关在 IBM SPSS Deployment Manager 下以处理作业的步骤形式运行 IBM SPSS Modeler 流的信息。
- **IBM SPSS Modeler CLEF 开发者指南。**CLEF 提供了将第三方程序（例如，数据处理例程或建模算法）作为节点集成到 IBM SPSS Modeler 的功能。
- **IBM SPSS Modeler 数据库内挖掘指南。**有关如何利用数据库的功能通过第三方算法来改进性能并增强分析功能的信息。
- **IBM SPSS Modeler Server 管理和性能指南。**提供有关如何配置和管理 IBM SPSS Modeler Server 的信息。
- **IBM SPSS Deployment Manager 用户指南。**有关使用 Deployment Manager 应用程序中包含的管理控制台用户界面来监视和配置 IBM SPSS Modeler Server 的信息。
- **IBM SPSS Modeler CRISP-DM 指南。**借助 CRISP-DM 方法进行 SPSS Modeler 数据挖掘的分步指南。
- **IBM SPSS Modeler Batch 用户指南。**提供在批处理方式下使用 IBM SPSS Modeler 的完整指导，包括批处理方式执行和命令行自变量的详细信息。本指南仅以 PDF 格式提供。

SPSS Modeler Premium 文档

SPSS Modeler Premium 文档套件（安装指示信息除外）如下。

- **SPSS Modeler Text Analytics 用户指南。**提供有关将文本分析与 SPSS Modeler 配合使用的信息，包括文本挖掘节点、交互式工作台、模板和其他资源。

应用程序示例

SPSS Modeler 中的数据挖掘工具可以帮助解决很多业务和组织问题，应用程序示例将提供有关特定建模方法和技术的简要的针对性说明。此处使用的数据集比某些数据挖掘器管理的大量数据存储小得多，但涉及的概念和方法可扩展到实际应用程序。

要访问示例，请在 SPSS Modeler 中单击“帮助”菜单中的**应用程序示例**。

数据文件和样本流安装在产品安装目录下的 Demos 文件夹中。有关更多信息，请参阅『Demos 文件夹』。

数据库建模示例。请参阅 *IBM SPSS Modeler 数据库内挖掘指南* 中的示例。

脚本编制示例。请参阅 *IBM SPSS Modeler 脚本编写和自动化指南* 中的示例。

Demos 文件夹

与应用程序示例一起使用的数据文件和样本流安装在产品安装目录下的 Demos 文件夹中（例如：C:\Program Files\IBM\SPSS\Modeler\

许可证跟踪

在使用 SPSS Modeler 时，将定期跟踪并记录许可证使用情况。记录的许可证度量为 *AUTHORIZED_USER* 和 *CONCURRENT_USER*，并且记录的度量类型取决于 SPSS Modeler 的许可证类型。

IBM License Metric Tool 可处理生成的日志文件，您可以从中生成许可证使用情况报告。

在记录 SPSS Modeler 客户机日志文件的相同目录中创建许可证日志文件（缺省情况下，%ALLUSERSPROFILE%/IBM/SPSS/Modeler/<version>/log）。

第 2 章 IBM SPSS Modeler Solution Publisher

IBM SPSS Modeler Solution Publisher 是一个强大的工具，用于将数据挖掘结果集成到业务过程中以解决实际问题。使用 IBM SPSS Modeler Solution Publisher，可以创建流的"打包"版本，该版本的流可由外部 Runtime 引擎执行或者嵌入到外部应用程序中。这使您可以在生产环境中部署数据建模流以支持每天的业务过程，以及使用挖掘数据所获得的知识增强组织决策者的能力。

与简单地导出模型（作为 PMML）相比，使用 IBM SPSS Modeler Solution Publisher 提供了更强的性能，因为您能够通过它发布和部署完整的 IBM SPSS Modeler 流。这就意味着，在根据模型创建预测之前您可以执行数据准备，也可以执行记录和字段操作（如聚合数据、选择记录或导出新字段）。然后您可以进一步处理模型结果，然后再保存这些数据 - 所有这些通过执行发布的流即可完成。

注：发布是通过使用相应导出节点的"发布"选项卡实现的。请参阅主题『发布流』以获取更多信息。保存在本产品的 V11.0 之前的发行版中的流使用单独的 Publisher 节点。该节点还支持向后兼容性，但不再节点选项板上可用。

获取 IBM SPSS Modeler Solution Publisher 许可

注：IBM SPSS Modeler Solution Publisher 作为 IBM SPSS Collaboration and Deployment Services 评分服务的一部分分发，需要单独的许可证。<http://www.ibm.com/software/analytics/spss/products/deployment/cds/>

当您购买许可证时，会获取一个单独的 IBM SPSS Modeler Solution Publisher Runtime，通过它可以执行已发布的流。请参阅主题第 11 页的『安装 Runtime』以获取更多信息。

IBM SPSS Modeler Solution Publisher 的工作原理

使用 IBM SPSS Modeler Solution Publisher 部署解决方案涉及两个阶段：发布流和执行流。

发布。当您执行整个数据挖掘过程时，最终将获得一个模型，该模型会为您的业务问题提供很好的解决方案。此时，您将要采用该模型并将其应用到您的业务过程。发布流时，会向磁盘写入该流的详细描述（作为一个图像文件和一个参数文件）。请参阅主题『发布流』以获取更多信息。

执行。发布流之后，可以通过执行已发布的流来重新创建在该流中实现的过程。此步骤是通过使用独立的 IBM SPSS Modeler Runtime (*modelerrun.exe*) 或通过开发一个使用 IBM SPSS Modeler Runtime Library 来执行流的应用程序完成的。要执行 IBM SPSS Modeler 外部的流（使用 Runtime 或自定义应用程序），则必须首先安装 IBM SPSS Modeler Solution Publisher Runtime。请参阅主题第 11 页的『安装 Runtime』以获取更多信息。

发布流

系统使用以下任一标准导出节点直接从 IBM SPSS Modeler 发布流：数据库、平面文件、Statistics 导出、扩展名导出、数据收集导出、SAS 导出、Excel 和 XML 导出节点。导出节点的类型决定每次使用 IBM SPSS Modeler Solution Publisher Runtime 或外部应用程序执行发布的流时要写入的结果格式。例如，如果您想每次运行发布的流时将结果写入数据库，则请使用数据库导出节点。

要发布流

1. 以普通方式打开或构建一个流，并在最后附加一个导出节点。
2. 在该导出节点的"发布"选项卡上，指定发布文件的根名称（即扩展名 .pim、.par 和 .xml 将附加至的文件名）。
3. 单击**发布**以发布该流，或选择**发布流**以便每次执行该节点时都自动发布流。

发布名称。指定发布的映像和参数文件的根名称。

- **图像文件** (*.pim) 提供了 Runtime 执行发布的流时所需的所有信息，这些信息与导出时完全相同。如果您确信不需要更改流的任何设置（如输入数据源或输出数据文件），则可以只部署该图像文件。
- **参数文件** (*.par) 包含有关数据源、输出文件和执行选项的可配置信息。如果您希望能够在不重新发布流的情况下控制流的输入或输出，则同时需要参数文件和图像文件。
- **元数据文件** (*.xml) 介绍图像的输入和输出以及它们的数据模型。它旨在供内嵌 runtime 库并需要了解输入数据和输出数据结构的应用程序使用。

注：仅当选中**发布元数据**选项时，才会产生此文件。

发布参数。如果需要，可以在 *.par 文件中包含流参数。通过编辑 *.par 文件或通过 Runtime API 执行图像时，可以更改这些流参数值。

此选项将启用**参数**按钮。单击该按钮时，将显示"发布参数"对话框。

通过在**发布**列上选中相关的选项，可选择要在发布的图像中包括的参数。

在流执行时。指定执行节点时是否自动发布流。

- **导出数据。**以标准方式执行导出节点，但不发布流。（基本上来说，节点在 IBM SPSS Modeler 中的执行方式与 IBM SPSS Modeler Solution Publisher 不可用时的执行方式相同。）如果您选择此选项，则只有通过单击导出节点对话框上的**发布**进行明确发布时，该流才进行发布。另外，您还可以通过使用工具栏上的发布工具，或通过使用脚本来发布当前流。
- **发布流。**使用 IBM SPSS Modeler Solution Publisher 发布流以用于部署。如果您希望每次执行节点时都自动发布流，则请选择此选项。

注：

- 如果您计划使用新数据或更新后的数据运行发布的流，则要注意的重要一点是，输入文件中字段的顺序必须与发布的流中指定的源节点输入文件中的字段顺序相同。
- 发布到外部应用程序时，请考虑对无关的字段进行过滤，或者对字段进行重命名以符合输入要求。通过在导出节点之前使用过滤节点，可以完成上述两个操作。

第 3 章 执行已发布的流

执行已发布流的主要效果是生成一组数据。数据可以保存到磁盘文件或写到数据库。已发布的流不会生成图形、表格、已生成模型或其他非数据输出。

使用 IBM SPSS Modeler Solution Publisher 发布的流可使用 IBM SPSS Modeler Solution Publisher Runtime 程序执行。*modelerrun.exe* Runtime 程序通过命令行来启动，而且具有指示要执行的映像文件和（可选）参数文件的选项。Runtime 命令如下：

```
modelerrun -nobanner -p <parameter-file> -o <options> <image-file>
```

下表显示了其中的命令条目。

表 1. Runtime 命令条目

命令条目	描述
-nobanner	不显示 Runtime 的启动文本消息（可选）。
<parameter-file>	是参数文件的文件名（可选）。
<options>	是指定 Runtime 执行选项的 option=value 对的列表（以逗号分隔）。下文详细列出选项（可选）。注：在逗号后面不得有任何空格。
<image-file>	是要执行的映像文件的文件名（必需）。

下表显示了可用的执行选项。

表 2. 有效 Runtime 选项。

名称	默认值	描述
max_file_size	-1	保存文件的最大大小。值为 -1 则表示无限制。
max_sql_string_length	2,048	用 SQL 从数据库导入字符串的最大长度。大于此长度的字符串将从右截断，没有警告。有效范围在 1 和 65,535 个字符之间。
memory_usage	100	用于动态内存分配的放大器。可上下调整此值，以调节服务器总内存的使用。
temp_directory	"	指定用于临时文件的目录。
request_passwords	Y	用于在运行要求访问数据库的已发布流时控制是否提示用户输入数据库密码（是/否）。

注：在 **SPSS Modeler Solution Publisher** 中运行包含 CPLEX Optimization 节点的流时，缺省情况下，会使用嵌入的共用版 CPLEX 库。此库具有 1000 个变量和 1000 约束的限制。如果安装 IBM ILOG CPLEX 的完整版本，且希望改为使用完整版本的 CPLEX 引擎以便不具有此类限制，请完成以下适用于您平台的步骤。

- 在 Windows 上，将 OPL 库路径作为 modelerrun.exe 的命令行自变量提供。例如：

```
-o cplex_opl_lib_path="<CPLEX_path>\opl\bin\<Platform_dir>"
```

其中，<CPLEX_path> 是 CPLEX 安装目录（例如，C:\Program Files\IBM\ILOG\CPLEX_Studio127），<Platform_dir> 是特定于平台的目录（例如，x64_win64）。

- 在 Linux 上，编辑 modelerrun 并添加 OPL 库路径。例如：

```
CPLEX_OPL_LIB_PATH=<CPLEX_path>/opl/bin/<Platform_dir>
```

其中，<CPLEX_path> 是 CPLEX 安装目录（例如，/root/Libs_127_FullEdition/Linux_x86_64），<Platform_dir> 是特定于平台的目录（例如，x86-64_linux）。

在 IBM SPSS Modeler Solution Publisher 中运行本地 Apache Spark 流

当您在 IBM SPSS Modeler Solution Publisher 中运行本地 Spark 流时，必须在 modelerrun 命令中指定以下内容（其中，C:/Anaconda2/python.exe 是到 Python 可执行文件的路径）：

```
-o eas_pyspark_python_path=C:/Anaconda2/python.exe
```

使用参数文件定制流执行

发布流时，映像文件含有在原始流中选择的数据源和数据输出目标位置的信息。但是，通常可方便地从不同的数据源对数据运行流或将输出数据重新路由到不同的目标位置。这可通过编辑流的参数文件来完成。

这点在用户不是创建已发布流的计算机上执行已发布流时特别重要。用户需要更新输入和输出文件的位置，以反映目标计算机的结构。请注意，已发布流不支持相对路径，这样确认参数文件中的数据位置就变得特别重要。（如果用户在发布流时在导出节点选项中指定相对路径，则 IBM SPSS Modeler 将在创建已发布流的映像和参数文件之前将相对路径转换为绝对路径。）

您还可以使用参数文件设置执行选项并更改流参数的值。

参数值在参数文件中被指定为 <name.attribute >= <value > 对。有些参数名称含有用来区分数据流中同一类型节点的标识数字。例如，在输入和输出都引用数据库连接的流中，说明输入连接的参数可以是 dbconn0.datasource 、 dbconn0.user 等。而与输出连接关联的参数则可以是 dbconn1.datasource 、 dbconn1.user 等。值总是用双引号引起来。参数文件中包含以下参数（所有参数均可以更改以定制流执行）：

表 3. 执行参数。

参数	目的
dbconnx.datasource	数据源名称 (DSN)。
dbconnx.user	访问权受限数据库的用户名。
dbconnx.password	访问权受限数据库的密码。
dbconnx.epassword	访问权受限数据库的加密密码。要生成加密密码，请选择 IBM SPSS Modeler 用户界面"工具"菜单的加密密码。需要时复制并粘贴加密密码。
dbconnx.password_required	指示是否提示用户输入此数据库连接的密码的标志。（值为"是"或"否"。）发布时如未指定密码则自动设置为"否"。
dbconnx.catalog	与数据库连接关联的目录。
dbtablex.name	基准表格名称。注：新表格中字段的顺序必须与原始发布流中的字段顺序相同。
dbtablex.schema	模式名称。
dbtablex.catalog	与当前表格关联的目录。
filex.name	数据文件的文件名。注：新输入文件中字段的顺序必须与原始发布流中的字段顺序相同。
filex.path	文件位置（目录名称）。
filex.field_names_included	该标志用于指示文本格式文件的字段名称是否作为文件的第一行。（值为"是"或"否"。）
filex.field_separator	用于分隔文本格式文件的字段值的字符。

表 3. 执行参数 (续).

参数	目的
filex.decimal_separator	用于文本格式文件的小数点的字符。
options.angle_in_radians	该标志指示是否使用弧度作为三角法 CLEM 表达式中的度量单位。(值为"是"或"否".)
options.date_2digit_baseline	用于定义按两位年份指定的日期的世纪。
options.date_baseline	使用单个日期的 CLEM 日期函数所用的基线年(始终为 1 月 1 日)。
options.time_rollover	该标志指示负数时间差是否指代过去。
options.decimal_separator	文本文件中的默认小数点字符。
options.time_format	CLEM 时间函数将字符串解释为时间时采用的时间格式。
options.date_format	CLEM 日期函数将字符串解释为日期时采用的日期格式。
options.timestamp_format	以字符串形式从 ODBC 数据源读取时间戳字段时使用的格式。
paramx.value	流参数值。 注: 包含了参数名称、描述以及存储类型以供参考, 但是不能进行更改。 用于该类型的值必须有效。

将 IBM SPSS Modeler Solution Publisher 嵌入到应用程序中

除了提供独立的 Runtime 引擎, IBM SPSS Modeler Solution Publisher 还提供其他程序用来控制 IBM SPSS Modeler Solution Publisher 流执行的 Runtime 编程库 (CLEMRTL)。用户可调用使用 C、和 C++ 编写的客户程序中的 CLEMRTL 程序。要使用 CLEMRTL, 用户需要有头文件 *clemrtl.h* (位于 *..\installation\clemrtl\include* 文件夹), 并务必在构建开发平台时将开发平台的对应库文件链接到用户的应用程序。*clemrtl.dll* 文件可从 *..\installation\bin* 文件夹中获得。

所有引用库程序的源文件都必须包含表头 *clemrtl.h*。此头文件提供库程序的 ANSI C 原型并定义有用的宏。除了程序要求的表头, 该头文件不需要任何其他表头。为了防止名称冲突, 所有以 *clemrtl_* 开头的库类型和函数名称及所有宏名称都添加前缀 *CLEMRTL_*。

CLEMRTL 制作的报告包含有用信息, 应将这些信息以某种方式发送回应用程序。CLEMRTL 为处理这些消息提供三种方法:

- 消息可以写到用 *clemrtl_setLogFile()* 函数指定的日志文件。
- 应用程序可以使用 *clemrtl_getErrorDetail()* 函数抽取最后一个错误的详细信息。
- 应用程序可以使用 *clemrtl_setReportHandler()* 函数提供自己的报告处理程序。

Runtime 系统具有报告的已本地化消息目录。传递到应用程序的所有报告均含有该本地化消息字符串。报告还包含报告代码, 以便应用程序能选择以不同方式解释和表示消息。为了获得详细(本地化的)消息, 应用程序需要在 *config* 目录中含有合适的 *messages.cfg* 文件。对于不同语言环境, 可以在目录 *<installdir>/config/<locale>* 中找到不同版本的 *messages.cfg*, 其中 *<installdir>* 是安装 IBM SPSS Modeler Solution Publisher 的目录, *<locale>* 是所需语言环境。请选择合适版本的 *messages.cfg* 并将它添加到用户项目目录中的 *config* 子目录中。

Runtime 系统不需要特殊环境或注册表设置就能正常运行。依赖的库的分布必须保证有一个应用程序链接到 CLEMRTL。这些文件包括在 *<installdir>/bin* 目录中。

需求

下表显示了已使用 Runtime 编程库测试过的平台和编译器。

表 4. 平台和编译器.

平台	推荐的编译器
AIX	IBM XL C/C++ Enterprise Edition V8.0 for AIX
Linux 64 位	gcc 3.2.3, Intel C++ 编译器 9.1.043
Windows	Visual Studio 2008 R2

IBM SPSS Modeler Solution Publisher 样本应用程序

IBM SPSS Modeler Solution Publisher 安装目录的 `\clemtrl\demo` 文件夹中包含一个示例应用程序 `dlltest.c`。请使用应用程序示例来帮助将 Runtime 库嵌入到应用程序中。

示例应用程序所使用的信息包含在 `\clemtrl\demo` 文件夹中；请参阅 `README.txt` 文件了解说明。不同平台的应用程序编译方法有所不同，因此同时提供了 `makefiles` 示例。

第 4 章 安装 IBM SPSS Modeler Solution Publisher Runtime

安装 Runtime

安装 IBM SPSS Modeler Solution Publisher Runtime 允许您执行已发布的流。已发布流独立于操作系统，因此可以在 Runtime 所支持的任何平台上执行任何已发布流。

此功能（通过它可以 from IBM SPSS Modeler 发布流）随 IBM SPSS Modeler 一起安装。注意，使用 IBM SPSS Modeler 的每个版本时，必须重新导出发布文件。例如，使用较早版本导出的文件不适用于 IBM SPSS Modeler Solution Publisher Runtime 的当前版本。

在 Windows 中安装

以下部分说明在 Windows 操作系统上安装 SPSS Modeler Solution Publisher Runtime 的过程。

1. 启动 IBM SPSS Modeler Solution Publisher 安装。
2. 在安装菜单上，选择安装 **IBM SPSS Modeler Solution Publisher Runtime** 或安装 **IBM SPSS Modeler Solution Publisher Runtime for Win64**，具体取决于要安装至的目标系统。
3. 在安装向导中，单击下一步开始。
4. 按照屏幕上显示的说明进行操作。要继续，请单击 **下一步**。
5. 一旦指定了所有的选项后，就可以开始安装了。单击 **安装** 开始传输文件。
6. 所有文件安装完毕后，单击 **完成**。

在 UNIX 中安装

以下部分说明在 UNIX 操作系统上安装 IBM SPSS Modeler Solution Publisher Runtime 的附加系统要求以及安装过程。

UNIX 的附加要求

您必须确保系统上的内核限制对于 IBM SPSS Modeler Server 的操作足够。数据、内存、文件和进程限制尤其重要，应在 IBM SPSS Modeler Server 环境内设为无限。为此：

1. 添加以下命令到 *modelersrv.sh*：

```
ulimit -d unlimited
ulimit -m unlimited
ulimit -f unlimited
ulimit -u unlimited
```

此外，将堆栈限制设置为系统允许的最大值 (*ulimit -s XXXX*)，例如：

```
ulimit -s 64000
```

2. 重新启动 IBM SPSS Modeler Server。

还需要已安装 *gzip* 文件压缩实用程序和 *GNU cpio* 并且已在 PATH 中，这样安装程序才能解压缩安装文件。此外，在运行 SPSS Modeler Server 的机器上，应该将语言环境设置为 EN_US.UTF-8。

UNIX 安装过程

这些安装说明适用于在 UNIX 上安装的 IBM SPSS Modeler Solution Publisher Runtime。需要具有目标安装目录的读权限和写权限，因此请使用拥有足够权限的帐户登录。

注：您必须以 *root* 用户身份登录进行安装。

1. 从安装包打开 *modelrun* 目录。
2. 切换至相关平台目录。
3. 运行 *.bin* 安装脚本。例如：

```
./modelersolutionpublisherhpie64.bin -i console
```
4. 将显示详细的介绍信息。请按 Enter 键以继续。
5. 将显示许可信息。阅读许可条款，键入 1 接受这些条款，并按 Enter 继续。
6. 这将显示可用语言的列表。输入要安装的语言版本的编号，然后按 Enter 继续。
7. 将提示您输入安装位置。要使用默认目录 */usr/IBM/SPSS/ModelerSolutionPublisher<nn>*（其中 *<nn>* 是版本号），按下 Enter 键。Runtime 将安装在指定目录下。
8. 您将被提示确认安装位置。如果安装位置正确，键入 *y* 并按 Enter。
9. 这将显示安装前摘要，以确认到目前为止输入的信息。请按 Enter 键以继续。
10. 将显示一条消息，提示安装程序已准备好开始运行。请按 Enter 键以继续。
11. 在安装程序运行期间，将出现一个进度条。安装完成时，请按 Enter 键以退出安装程序。

在 UNIX 上配置 ODBC

在 UNIX 系统上，缺省情况下，未配置 DataDirect 驱动程序管理器，因此 IBM SPSS Modeler Solution Publisher Runtime 无法使用 ODBC。要配置 UNIX 以装入 DataDirect 驱动程序管理器，请输入以下命令（其中 *sp_install_dir* 是 Solution Publisher Runtime 的安装目录）：

```
cd sp_install_dir
rm -f libspssodbc.so
ln -s libspssodbc_datadirect.so libspssodbc.so
```

安装的故障排除

安装时发生数字签名无效错误

IBM SPSS Modeler 产品使用 IBM 发放的证书进行数字签名。在某些情况下，您可能在尝试安装 SPSS Modeler 产品时看到以下错误：

```
Error 1330. A file that is required cannot be installed because the cabinet file filename has an invalid digital signature...
```

所有 Windows 用户

如果您尝试在没有因特网连接的机器上安装 SPSS Modeler，并且未安装正确的证书，那么将会看到此消息。请使用以下过程更正此问题。

1. 单击**确定**以确认此消息。
2. 单击**取消**以退出安装程序。
3. 如果用于安装的机器上未连接因特网，请在已连接因特网耳朵机器上执行下一步，并将 *.cer* 文件复制到要用于安装的机器上。

4. 转至 <https://knowledge.symantec.com/support/ssl-certificates-support/index?page=content&id=SO5624>，并遵循指示信息下载 **VeriSign Class 3 Primary Certification Authority - G5** 根证书。将其保存为 `.cer` 文件。
5. 双击此 `.cer` 文件。
6. 在"常规"选项卡中，单击**安装证书**。
7. 遵循证书导入向导中的指示信息，使用缺省选项并在结束时单击**完成**。
8. 重新尝试安装。

启动 Runtime

一旦安装了 Runtime，就可以通过它使用任何导出节点执行已从 IBM SPSS Modeler 中发布的流。请参阅主题第 5 页的『发布流』以获取更多信息。从命令行启动 Runtime，并在命令行中包含表示要执行的已发布映像文件和可选参数文件的名称的选项。

要执行 Runtime，请在命令提示符下键入以下行：

```
modelerrun -p <parameter-file> -o <options> <image-file>
```

下表显示了其中的命令条目。

表 5. Runtime 命令条目

命令条目	描述
<parameter-file>	是已发布参数文件的文件名（可选）。
<options>	是指定 Runtime 执行选项的 option=value 对的列表（以逗号分隔）。注：在逗号后面不得有任何空格。
<image file>	是要执行的已发布图像文件（*.pim）的文件名。

注：在 UNIX 环境中使用多字节语言（例如中文或日文）时，必须指定要使用的语言代码。以下示例显示的是用于简体中文的语言代码：

```
./modelerrun -o locale="zh_CN.GB18030",encoding="GB18030" -p
```

其中 -p 为 pim 和 par 文件的位置。

请参阅第 7 页的第 3 章，『执行已发布的流』，以了解更多信息。

更改 Temp 目录

IBM SPSS Modeler Solution Publisher Runtime 执行的有些操作可能需要创建临时文件。默认情况下，Runtime 在系统临时目录下创建临时文件。可通过以下步骤更改临时目录的位置。

注：必须使用正确的语法，尤其需要注意空格和目录路径，这非常重要。

假设您将 IBM SPSS Modeler Solution Publisher Runtime 与下列已发布文件一起使用 `webtest.pim` 和 `webtest.par`。在这种情况下，要修改临时文件的位置以改为使用目录 `C:\published\temp`。要变更可执行程序使用的临时目录的位置，请执行下列操作：

1. 在保存 `.pim` 和 `.par` 文件的目录下，运行以下命令：

```
modelerrun -o temp_directory="C:\\published\\temp" -p webtest.par webtest.pim
```

此命令可指示 Runtime 使用 `"C:\published\temp"` 作为仅针对此特定进程的临时目录。

卸载 IBM SPSS Modeler Solution Publisher Runtime

下面的部分说明卸载 IBM SPSS Modeler Solution Publisher Runtime 的过程。

Windows 卸载过程

1. 在 Windows 控制面板中，打开添加或删除程序。
2. 从列表中选择 **IBM SPSS Modeler Solution Publisher <nn>**.
3. 单击 **删除**。
4. 按照屏幕上显示的说明删除 Runtime。

注：必须使用管理员特权登录计算机才能删除程序。

UNIX 卸载过程

删除安装目录，包括所有已安装的文件。

第 5 章 IBM SPSS 嵌入式预测建模 API

IBM SPSS 嵌入式预测建模 API 定义一个简单的 Java 接口，使您能够以嵌入式方式集成 IBM SPSS Modeler 设计的工件和您的应用程序。IBM SPSS Modeler Solution Publisher 提供了 Javadoc 格式的完整文档以供使用库的编码人员使用。

在使用嵌入式预测建模 (EPM) API 时请注意以下限制：

- 需要唯一 **Source** 和 **Terminal** 节点标签。EPM API 使用节点标签。尽管标签始终位于 Source 或 Terminal 节点上下文中，但不能模糊不清。
- 不支持使用超节点作为 **Source** 或 **Terminal** 节点。要为所有操作中的超节点提供公共源，不支持将超节点用作 Source 节点或 Terminal 节点
- 输入字段名称限制。字段名称中不能使用某些字符。

请参阅以下信息以获取 API 功能概述。有关更多详细信息，请参阅产品包含的 Javadoc。

预测模型嵌入式评分

用途：由应用程序用于在新数据传入时按需生成预测性分析。应用程序必须处理刷新之后的模型。在这种评分方式下，任何 Source 节点以及评分分支的 Terminal 节点都将替换为应用程序输入和输出。

操作：对于根据应用程序管理的输入数据快速高效地生成预测性分析的重复调用，一次性准备您在 IBM SPSS Modeler 流中设计的评分分支。

预测模型刷新

用途：使用历史数据在流设计中执行模型算法的初始训练。此外，还使用最新的历史记录数据定期“刷新”配置的预测模型算法。

操作：在 IBM SPSS Modeler 流中运行所有预测模型构建器节点，以便使用数据挖掘方法按照配置训练预测模型算法。更新链接到每个模型构建器节点的模型应用器块。流参数和 Source/Terminal 节点属性可以用于实现一定程度的动态。

预测模型评估

用途：由应用程序按照 IBM SPSS Modeler 流中的设计用于传达训练预测模型的准确性、置信度或其他定性评估。该信息通常用于控制刚刚刷新的预测模型到应用程序中的评分的升级过程。“模型刷新”返回新的 PredictiveModel 对象的主要原因是能够比较刚刚刷新的模型与原始模型的评估数据。

操作：在 IBM SPSS Modeler 流中运行所有文档构建器节点（主要是 IBM SPSS Modeler“输出”和“图表”托盘上的非交互式输出节点）以返回产生的文档。流参数和 Source/Terminal 节点属性可以用于实现一定程度的动态。

预测模型流执行

用途：不符合“刷新”和/或“评估”评分方式的所有用例必须使用流执行来处理。

操作：由 IBM SPSS Modeler 流选项控制的执行。或在"所有 Terminal 节点"方式下类似于单击 IBM SPSS Modeler Client 中的绿色大箭头按钮，或如流级别脚本中的编码。流参数和 Source/Terminal 节点属性可以用于实现一定程度的动态。

预测模型评分外部数据

用途：由于整批输入数据的处理方式，通常称为"批处理"流程。

操作：按照设计运行评分分支。来自已配置数据源节点的输入和生成的预测性分析由 Terminal 节点控制为持久存储。流参数和 Source/Terminal 节点属性可以用于实现一定程度的动态。

Source 节点和 Terminal 节点属性的预测模型流参数和规范

用途：一种常见的用法是在流分支中定义表达式中引用的参数。另一个控制点是修改 Source 或 Terminal 节点的特定键属性以变更使用的文件路径等等。

操作：IBM SPSS Modeler 流执行的特定方面可以动态地由应用程序进行修改。

注：参数更改必须在创建 PredictiveScorer 之前进行（请参阅上述的嵌入式评分描述），Source 和 Terminal 节点属性对此对象无任何意义。

第 6 章 IBM SPSS Modeler Solution Publisher Runtime Library API 参考

简介

使用 CLEMRTL API 可以在应用程序中控制已发布流的执行。本节说明了可用的 API 函数。

下面是使用 API 中的一般问题：

- API 含有 C 连接，从而具有最大兼容性，但是库依赖于 C++。在某些平台上，这可能意味着仅能使用具有可识别 C++ 连接器的 API。
- 在使用库中的任何其他函数之前，必须先调用 `clemrtl_initialise_ext()`。
- 类型 `clemrtl_image_handle` 用作图像标识。
- 每个函数都可返回一个状态指示器代码。下表显示了状态指示器的值。

表 6. 状态指示器值

结果	描述
CLEMRTL_OK	成功
CLEMRTL_FAIL	失败，但没有进一步的详细信息
CLEMRTL_ERROR	失败，通过 <code>clemrtl_getErrorDetail()</code> 函数可获取有关该错误的其他信息

API 流程概述

通常，使用 API 的应用程序的大体流程如下：

1. 使用 `clemrtl_initialise_ext()` 初始化库。
2. 使用 `clemrtl_getOption()` 和 `clemrtl_setOption()` 检查和更改执行选项。
3. 使用 `clemrtl_openImage()` 打开图像并接收图像句柄。
4. 使用 `clemrtl_enumerateParameters()`、`clemrtl_getParameter()` 和 `clemrtl_setParameter()` 检查和更改图像参数。
5. 使用 `clemrtl_execute()` 执行图像。如果要在不更改参数的情况下对同一个图像多次执行，则请首先使用 `clemrtl_prepare()`。
6. 使用 `clemrtl_closeImage()` 关闭图像。
7. 要取消正在进行的执行，请使用 `clemrtl_interrupt()`。
8. 要检索有关最后错误的信息，请使用 `clemrtl_getErrorDetail()`。要在错误消息和其他诊断消息到达时接收它们，请使用 `clemrtl_setReportHandler()`。

此库为在下列条件下的多线程使用而设计：

- 不能同时调用全局函数（即，不会将图像句柄作为参数的函数）。
- 将图像本地函数（即，将图像句柄作为参数的函数）应用到同一个图像句柄时不能同时对其进行调用，但 `clemrtl_interrupt()` 函数除外。
- 将图像本地函数应用到不同的图像句柄时可同时对其进行调用。

特别是，如果独立的线程使用的是不同的图像句柄，则这些线程可以同时准备并执行图像。不同的图像句柄可以引用相同的图像（`clemrtl_openImage()` 可在每次被调用时创建一个新句柄，甚至是为相同图像创建的句柄），但要确保在执行相同图像的多个实例时，更改图像参数以按照需要重定向输出。如果两个同时执行的图像都试图写入同一个输出文件或数据库表，则结果将是不可预期的。

API 函数

下面是 API 中函数的完整列表。

初始化

此函数已不再使用，它等效于

```
initialise_ext(flags, 0, 0);
```

新的程序名应为 `initialise_ext`。

initialise_ext

```
int clemrtl_initialise_ext( unsigned flags,
    int arg_count,
    const clemrtl_init_arg* args);
```

初始化 IBM SPSS Modeler Solution Publisher Runtime。必须在调用任何其他 API 函数之前必须先调用此函数。

表 7. `initialise_ext` 参数

参数	描述
flags	对初始化过程的某些方面进行控制。该值可按位构建或按下面说明的标志构建。
arg_count	其他初始化参数的数量。此数字必须小于或等于 args 数组的长度。
args	其他初始化参数。这些参数按类型分为： <pre>typedef struct _clemrtl_init_arg { const char* name; const char* value; } clemrtl_init_arg;</pre>

name 字段为配置属性的名称，value 字段为其值。系统可识别下表中显示的配置属性。

表 8. 配置属性

姓名	值(V)
installation_directory	到 IBM SPSS Modeler Solution Publisher 安装文件夹的完整路径。安装文件夹中包含 bin 和 config 文件夹。

将返回下表显示的错误代码之一。

表 9. 错误代码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_FAIL	初始化失败，不显示进一步的详细信息。

flags 参数可对初始化操作的某些方面进行控制。值等于 0（或 CLEMRTL_INIT_DEFAULTS）可指定以下默认行为：

- 通过调用 `tzset()` 设置默认时域。
- 将 `LC_NUMERIC` 语言环境设置为"C"。
- 将全局新处理程序设置为在内存耗尽时提示发生异常。

通过指定下表中显示的标志的组合可修改此行为。

表 10. *Flags*

Flag	描述
<code>CLEMRTL_INIT_NO_TZ</code>	不设置时区。
<code>CLEMRTL_INIT_NO_LOCALE</code>	不更改语言环境。
<code>CLEMRTL_INIT_NO_NEW_HANDLER</code>	不设置新处理程序。
<code>CLEMRTL_INIT_LOCAL_NEW_HANDLER</code>	在每个 API 调用内定制新处理程序。

此库需要将 `LC_NUMERIC` 语言环境设置为"C"，以便正确地在数字和字符串之间进行转换（IBM SPSS Modeler 数字格式使用"C"环境规则）。这包括对从文本文件中读出和写入文本文件的数字进行转换。

- 如果应用程序依赖于 `LC_NUMERIC` 语言环境并且不需要在数字和字符串之间进行转换，则指定 `NO_LOCALE`。
- `NO_NEW_HANDLER` 和 `LOCAL_NEW_HANDLER` 互斥。此库需要 `new` 在内存耗尽时提示发生异常。
- 如果应用程序设置了自己的新处理程序，或如果确定编译器/Runtime 系统在默认情况下会提示发生异常，则指定 `NO_NEW_HANDLER`。

仅当应用程序依赖于 `new` 在内存耗尽时返回空指针，才指定 `LOCAL_NEW_HANDLER`。注：在多线程环境中使用时，这样设置不安全。

getOption

```
int clemrtl_getOption( const char* name,
                     char* value,
                     int value_size);
```

检索执行选项的值。

表 11. *getOption* 参数

参数	描述
<code>name</code>	选项的名称。
<code>value</code>	接收选项值的缓冲区。
<code>value_size</code>	值缓冲区的大小。

将返回下表显示的状态码之一。

表 12. 状态码

结果	描述
<code>CLEMRTL_OK</code>	成功。
<code>CLEMRTL_FAIL</code>	失败，但没有进一步的详细信息。
<code>CLEMRTL_ERROR</code>	失败，同时显示进一步的详细信息。

如果选项名称不是上面所列出的执行选项名称之一（例如 `memory_usage`）或值缓冲区不够大而无法接收包含终结空字符的值，那么此操作将失败。选项值始终为字符串，即使将它解析为数字。请参阅第 7 页的第 3 章，『执行已发布的流』，以了解更多信息。

setOption

```
int clemrtl_setOption( const char* name,
    const char* value);
```

设置执行选项的值。

表 13. *setOption* 参数

参数	描述
<code>name</code>	选项的名称。
<code>value</code>	选项的值。

将返回下表显示的状态码之一。

表 14. 状态码

结果	描述
<code>CLEMRTL_OK</code>	成功。
<code>CLEMRTL_FAIL</code>	失败，但没有进一步的详细信息。
<code>CLEMRTL_ERROR</code>	失败，同时显示进一步的详细信息。

如果选项名称不是上面所列出的执行选项名称之一（例如 `memory_usage`），那么此操作将失败。选项值始终为字符串，即使将它解析为数字。

setLogFile

```
int clemrtl_setLogFile( int mode,
    const char* log_file);
```

重定向所有此后打开的图像的日志消息。

表 15. *setLogFile* 参数

参数	描述
<code>mode</code>	日志模式必须是下列值之一： <code>CLEMRTL_NULL_LOG</code> — 不显示消息。 <code>CLEMRTL_STDERROR_LOG</code> — 将消息发送到标准错误输出。 <code>CLEMRTL_FILE_LOG</code> — 将消息发送到 <code>log_file</code> 参数指定的文件。
<code>log_file</code>	包含文件名的字符串。

将返回下表显示的状态码之一。

表 16. 状态码

结果	描述
<code>CLEMRTL_OK</code>	成功。
<code>CLEMRTL_FAIL</code>	失败，但没有进一步的详细信息。
<code>CLEMRTL_ERROR</code>	失败，同时显示进一步的详细信息。

如果模式不是被识别的模式之一，或请求用文件进行日志文件时未指定文件名或文件名无效，则此操作将失败。

openImage

```
int clemrtl_openImage( const char* image_file,  
    const char* param_file,  
    clemrtl_image_handle* handle);
```

打开已发布的流图像。

表 17. openImage 参数

参数	描述
image_file	图像文件名。
param_file	参数文件名。
handle	接收图像句柄。

将返回下表显示的状态码之一。

表 18. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_FAIL	失败，但没有进一步的详细信息。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果不需要单独的参数文件，则参数文件可以是空。如果未指定图像文件，或文件名无效，或其内容不可读，则此操作将失败。如果操作成功，则返回的图像句柄可用于识别后来的 API 调用中的图像实例。

closeImage

```
int clemrtl_closeImage( clemrtl_image_handle handle);
```

关闭图像句柄并释放其资源。

表 19. closeImage 参数

参数	描述
handle	图像句柄。

将返回下表显示的状态码之一。

表 20. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_FAIL	失败，但没有进一步的详细信息。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果另一个 API 正在调用图像，特别是如果该图像仍在执行中，则此操作将失败。如果此操作成功，则图像会被关闭，且句柄不能在任何后来的 API 调用中使用，除非再次调用 clemrtl_closeImage() 进行，不过此调用没有效果。

enumerateParameters

```
int clemrtl_enumerateParameters( clemrtl_image_handle handle,  
    clemrtl_parameter_proc proc,  
    void* data);
```

将回调过程应用于图像的每个参数名和参数值中。

表 21. *enumerateParameters* 参数

参数	描述
handle	图像句柄。
proc	参数回调过程。
data	用户指定的回调数据。

将返回下表显示的状态码之一。

表 22. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

回调过程具有下列类型：

```
typedef void (*clemrtl_parameter_proc)( void* data,  
    const char* name,  
    const char* value);
```

表 23. 回调过程参数

参数	描述
data	传递到 <i>clemrtl_enumerateParameters()</i> 的用户指定数据。
name	参数名。
value	参数值。

该过程以任意顺序应用于每个图像参数，并且仅应用一次。

getParameter

```
int clemrtl_getParameter( clemrtl_image_handle handle,  
    const char* name,  
    char* value,  
    int value_size);
```

检索图像参数值。

表 24. *getParameter* 参数

参数	描述
handle	图像句柄。
name	参数名。
value	接收参数值的缓冲区。
value_size	值缓冲区的大小。

将返回下表显示的状态码之一。

表 25. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_FAIL	失败，但没有进一步的详细信息。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果参数名与图像中的任何参数名都不匹配，或如果值缓冲区不够大，无法接收包括终结空字符的参数值，则此操作将失败。参数名称使用上面所述的格式 `name.attribute`（例如 `file0.name`），而参数值始终为字符串（即使将其解释为数字）。

setParameter

```
int clemrtl_setParameter( clemrtl_image_handle handle,  
    const char* name,  
    const char* value);
```

设置图像参数值。

表 26. *setParameter* 参数

参数	描述
handle	图像句柄。
name	参数名。
value	参数值。

将返回下表显示的状态码之一。

表 27. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果图像句柄无效，或如果参数名与图像中的任何参数名都不匹配，则此操作将失败。参数名称使用上面所述的格式 `name.attribute`（例如 `file0.name`），而参数值始终为字符串（即使将其解释为数字）。

getFieldCount

```
int clemrtl_getFieldCount( clemrtl_image_handle handle,  
    const char* key,  
    size_t* field_count);
```

返回输入源或输出目标中的字段数。

表 28. *getFieldCount* 参数

参数	描述
handle	图像句柄。
key	要检查的输入或输出的名称，与参数文件中使用的名称相同。关键字可能为文件或数据库。

表 28. *getFieldCount* 参数 (续)

参数	描述
field_count	接收字段数。

将返回下表显示的状态码之一。

表 29. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_FAIL	失败，但没有进一步的详细信息。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

getFieldTypes

```
int clemrtl_getFieldTypes( clemrtl_image_handle handle,
    const char* key,
    size_t field_count,
    int* field_types);
```

返回输入源或输出目标的字段类型。

表 30. *getFieldTypes* 参数

参数	描述
handle	图像句柄。
key	要检查的输入或输出的名称，与参数文件中使用的名称相同。关键字可能为文件或数据库。
field_count	要检查的字段数。此数字必须小于或等于 <i>field_types</i> 数组的长度。
field_types	长度至少为 <i>field_count</i> 的数组，用于接收字段类型。输入或输出中的第一个 <i>field_count</i> 字段类型将被复制到数组中。如果 <i>field_count</i> 大于实际字段数，则不会定义数组中的多余元素。以下"数据类型"表中列出了字段类型的值。

数据类型

field_types 必须是下表中显示的数据类型之一。

表 31. 数据类型

类型	解释	典型的"C"声明符
STRING	以空字符结尾的 UTF-8 字符串。	const char*
整数	32 位带符号整数。	int
LONG	带符号的 64 位整数。	long long
实数	64 位浮点。	double
时间	64 位带符号整数（午夜后的秒数）。	long long
DATE	64 位带符号整数（从 1970 年 1 月 1 日午夜后的秒数）。	long long
时间戳	64 位带符号整数（从 1970 年 1 月 1 日午夜后的秒数）。	long long

将返回下表显示的状态码之一。

表 32. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_FAIL	失败，但没有进一步的详细信息。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

setAlternativeInput

```
int clemrtl_setAlternativeInput( clemrtl_image_handle handle,
    const char* key,
    size_t field_count,
    int field_types,
    void** (*iterator)(void* arg),
    void* arg);
```

使用其他输入源替换文件输入源。

表 33. setAlternativeInput 参数

参数	描述
handle	图像句柄。
key	用于参考文件的原始输入源名称。输入源必须为一个文件，因此对于某些 $N \geq 0$ 的整数，key 始终采用“fileN”格式。
field_count	输入中的字段数。此值必须与原始文件输入中的字段数完全匹配，否则调用将失败。
field_types	长度至少为 field_count 的类型数组。值 field_types[i] 为第 i 个字段的类型，并且必须为以下“数据类型”表中所列值之一。类型必须与原始文件输入中的类型兼容，否则调用将失败。
iterator	生成其他输入数据的函数。将函数应用到其参数的方法如下： <pre>void** row = iterator(arg);</pre> 在执行（在 clemrtl_execute 的调用中）过程中调用函数，并且为每个输入记录调用一次。返回空值表示输入的结束，在这种情况下不会再调用函数并且执行最后会终止。否则，结果是长度至少为 field_count 的一组数据，其中 row[i] 提供了第 i 个字段的值。值可能为空，否则它必须为指针，并指向其类型由对应 field_types[i] 确定的数据。指针必须有效，直到下次调用迭代器；如果执行过早终止，则直到执行结束。
arg	每次调用时传递到迭代器的不透明参数。

数据类型

field_types 必须为受限类型列表中的一种。请参阅主题第 24 页的『getFieldTypes』以获取更多信息。

将返回下表显示的状态码之一。

表 34. 状态码

结果	描述
CLEMRTL_OK	成功。

表 34. 状态码 (续)

结果	描述
CLEMRTL_FAIL	失败，但没有进一步的详细信息。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果字段类型与原始文件输入的类型不兼容，或字段计数与原始文件输入中的字段数不完全匹配，则操作会失败。

setAlternativeOutput

```
int clemrtl_setAlternativeOutput( clemrtl_image_handle handle,
    const char* key,
    size_t field_count,
    int field_types,
    void (*iterator)(void* arg, void** row),
    void* arg);
```

请使用其他输出目标替换文件输出目标。

表 35. setAlternativeOutput 参数

参数	描述
handle	图像句柄。
key	用于参考文件的原始输出目标名称。输出源必须为一个文件，因此对于某些 $N \geq 0$ 的整数，key 始终采用“fileN”格式。
field_count	输出中的字段数。此值必须与原始文件输出中的字段数完全匹配，否则调用将失败。
field_types	长度至少为 field_count 的类型数组。值 field_types[i] 为第 i 个字段的类型，并且必须为以下“数据类型”表中所列值之一。类型必须与原始文件输出中的类型兼容，否则调用将失败。
iterator	消耗图像输出的函数。函数将应用到其参数和数据行的方式如下： iterator(arg, row); 在执行（在 clemrtl_execute 的调用中）过程中调用函数，并且为由图像生成的每个结果行调用一次。row 值为空表示输出的结束，且输出结束后不再调用函数；执行终止时，应用程序不应依赖这个最后的调用，而是应刷新或关闭所有外部资源等。否则，row 是长度至少为 field_count 的一组数据，其中 row[i] 提供第 i 个结果的值。值可能为空，否则它必须为指针，并指向其类型由对应 field_types[i] 确定的数据。函数必须复制它需要的所有数据值，因为返回调用后可能不会保留内存。
arg	每次调用时传递到迭代器的不透明参数。

数据类型

field_types 必须为受限类型列表中的一种。请参阅主题第 24 页的『getFieldTypes』以获取更多信息。

将返回下表显示的状态码之一。

表 36. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_FAIL	失败，但没有进一步的详细信息。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果字段类型与原始文件输出的类型不兼容，或字段计数与原始文件输出中的字段数不完全匹配，则操作会失败。

执行

```
int clemrtl_execute(clemrtl_image_handle handle);
```

执行图像。

表 37. *execute* 参数

参数	描述
handle	图像句柄。

将返回下表显示的状态码之一。

表 38. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果尚未准备图像，则可首先使用当前的参数值准备。如果图像句柄无效，或如果在准备或执行期间出现错误，则此操作将失败。直到执行完成后才会返回调用。

准备

```
int clemrtl_prepare(clemrtl_image_handle handle);
```

准备要执行的图像。

表 39. *prepare* 参数

参数	描述
handle	图像句柄。

将返回下表显示的状态码之一。

表 40. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果图像句柄无效，或如果在准备期间出现错误，则此操作将失败。

在执行图像之前必须先准备图像。准备图像的操作会将参数值冻结到图像中。 `clemrtl_execute()` 会对每次调用自动准备图像（如果尚未准备图像）；如果要使用同一个参数值对图像多次执行，则使用 `clemrtl_prepare()` 准备图像会非常有用，并且可以显著提高执行效率。一旦已准备好图像，将忽略对参数值的后续更改；再次使用 `clemrtl_prepare()` 可以用新参数值更新该图像。

中断

```
int clemrtl_interrupt(clemrtl_image_handle handle);
```

终止正在进行的对图像的执行。

表 41. *interrupt* 参数

参数	描述
handle	图像句柄。

将返回下表显示的状态码之一。

表 42. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果图像句柄无效，或如果图像不是正在执行，则此操作无效。

即使另一个 API 同时在调用同一个图像句柄，调用此函数也是安全的。

getErrorDetail

```
int clemrtl_getErrorDetail( clemrtl_image_handle handle,
char* severity,
int* code,
char* text,
int text_size);
```

检索有关出现在图像中的最后一个错误的详细信息。

表 43. *getErrorDetail* 参数

参数	描述
handle	图像句柄。
severity	接收用单个字符表示的严重性代码： I — 信息 W — 警告 E — 错误 X — 系统错误
code	接收错误编号。
text	接收消息文本的缓冲区。
text_size	文本缓冲区的大小。

将返回下表显示的状态码之一。

表 44. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

如果图像句柄无效，则结果将不可靠。必要时会截取消息文本以适合文本缓冲区的大小。

如果 API 调用失败，出现 CLEMRTL_ERROR，且错误与特定的图像句柄无关，则传递 0 作为图像句柄这一操作将检索最后的非图像特定错误的详细信息，但此操作在多线程环境中是不可靠的。

setReportHandler

```
int clemrtl_setReportHandler( clemrtl_image_handle handle,
    clemrtl_report_proc proc,
    void* data);
```

为图像安装报告回调过程。

表 45. setReportHandler 参数

参数	描述
handle	图像句柄。
proc	报告回调过程。
data	用户指定的回调数据。

将返回下表显示的状态码之一。

表 46. 状态码

结果	描述
CLEMRTL_OK	成功。
CLEMRTL_ERROR	失败，同时显示进一步的详细信息。

回调过程具有下列类型：

```
typedef void (*clemrtl_report_proc)( void* data,
    char severity,
    int code,
    const char* text);
```

表 47. 回调过程参数

参数	描述
data	传递到 clemrtl_setReportHandler() 的用户指定数据。
severity	用单个字符表示的严重性代码： I — 信息 W — 警告 E — 错误 X — 系统错误
code	消息编号。
text	消息文本。

在每个消息到达时对其应用过程。将过程设置为空可删除图像中安装的所有现有处理程序。

声明

本信息是为在美国提供的产品和服务编写的。IBM 可能以其他语言提供本材料。然而，您可能需要拥有产品或产品版本的该语言副本才能进行访问。

IBM 可能在其他国家或地区不提供本文档中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往：

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

有关双字节 (DBCS) 信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

International Business Machines Corporation"按现状"提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些管辖区域在某些交易中不允许免除明示或暗含的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本出版物中描述的产品进行改进和/或更改，而不另行通知。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：(i) 使其能够在独立创建的程序和其它程序（包括本程序）之间进行信息交换，以及 (ii) 使其能够对已经交换的信息进行相互使用，请与下列地址联系：

IBM Director of Licensing
IBM Corporation

North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或任何同等协议中的条款提供。

所引用的性能数据和客户示例仅作参考用途。实际的性能结果可能会因特定的配置和运营条件而异。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。

关于 IBM 未来方向或意向的声明都可随时更改或收回，而不另行通知，它们仅仅表示了目标和意愿而已。

本信息包含在日常业务操作中使用的数据和报告的示例。为了尽可能完整地说明这些示例，示例中可能会包括个人、公司、品牌和产品的名称。所有这些名字都是虚构的，若与实际个人或业务企业相似，纯属巧合。

商标

IBM、IBM 徽标和 ibm.com 是 International Business Machines Corp. 在全球许多行政管辖地区的商标或注册商标。其他产品和服务名称可能是 IBM 或其他公司的商标。IBM 商标的最新列表可从 Web 上的 "Copyright and trademark information" 处获得，网址为：www.ibm.com/legal/copytrade.shtml。

Adobe、Adobe 徽标、PostScript 和 PostScript 徽标是 Adobe Systems Incorporated 在美国和/或其他国家或地区的注册商标或商标。

Intel、Intel 徽标、Intel Inside、Intel Inside 徽标、Intel Centrino、Intel Centrino 徽标、Celeron、Intel Xeon、Intel SpeedStep、Itanium 和 Pentium 是 Intel Corporation 或其子公司在美国和其他国家或地区的商标或注册商标。

Linux 是 Linus Torvalds 在美国和/或其他国家或地区的注册商标。

Microsoft、Windows、Windows NT 和 Windows 徽标是 Microsoft Corporation 在美国和/或其他国家或地区的商标。

UNIX 是 The Open Group 在美国和其他国家或地区的注册商标。

Java 和所有基于 Java 的商标和徽标是 Oracle 和/或其子公司的商标或注册商标。

产品文档的条款和条件

只有遵守以下条款和条件才会授予使用这些出版物的许可权。

适用性

除用于 IBM Web 站点的任何条款外，还需要遵守这些条款和条件。

个人使用

您可以为了个人使用而非商业性使用复制这些出版物，但前提是保留所有专有权声明。没有 IBM 的明确同意，您不能对这些出版物或者其中的任何部分进行分发、展示或从中派生内容。

商业使用

您仅可在贵公司内部复制、分发和显示这些出版物，但前提是保留所有专有权声明。未经 IBM 的明确许可，您不得制作这些出版物的演绎作品，也不得在贵公司外部复制、分发或显示这些出版物或其部分出版物。

权利

除非本许可权中明确授予，否则不得授予对这些出版物或其中包含的任何信息、数据、软件或其他知识产权的任何许可权、许可证或权利，无论明示的还是暗含的。

只要 IBM 认为这些出版物的使用会损害其利益或者 IBM 判定未正确遵守上述指示信息，IBM 将有权撤销此处授予的许可权。

您不得下载、出口或再出口这些信息，除非完全遵守所有适用的法律和法规（包括所有的美国出口法律和法规）。

IBM 对这些出版物的内容不作任何保证。本出版物以"按现状"的基础提供，不附有任何形式的（无论是明示的，还是暗含的）保证，包括但不限于暗含的有关适销性、非侵权以及适用于某特定用途的保证。

索引

[B]

部署 5, 15

[C]

初始化

标志 18

API 函数参考 18

API 流程概述 17

错误报告

API 函数参考 20, 28, 29

错误代码 17

[D]

打开

流 21

[F]

发布流

IBM SPSS Modeler Solution

Publisher 5

[G]

关闭

流 21

[L]

临时目录

IBM SPSS Modeler Solution

Publisher Runtime 13

[R]

日志文件 20

[S]

时域

使用 API 设置 18

示例

概述 4

应用程序指南 3

输出节点 5, 15

[W]

文档 3

[X]

卸载

IBM SPSS Modeler Solution

Publisher Runtime 14

[Y]

应用程序示例 3

语言环境

使用 API 设置 18

[Z]

执行

API 函数参考 27

API 流程概述 17

执行流

使用 IBM SPSS Modeler Solution

Publisher 7, 8

中断

API 函数参考 28

API 流程概述 17

状态代码 17

准备

API 函数参考 27

API 流程概述 17

A

API

参考 17

错误代码 17

概述 9, 17

其他输出 26

其他输入 25

执行选项 19, 20

状态代码 17

字段计数 23

字段类型 24

setReportHandler 29

API 函数

初始化 17, 18

执行 17, 27

中断 17, 28

准备 17, 27

API 函数 (续)

closeImage 17, 21

enumerateParameters 17, 22

getErrorDetail 17, 28

getFieldCount 17, 23

getFieldTypes 17, 24

getOption 17, 19

getParameter 17, 22

openImage 17, 21

setAlternativeInput 17, 25

setAlternativeOutput 17, 26

setLogFile 17, 20

setOption 17, 20

setParameter 17, 23

setReportHandler 17

C

closeImage

API 函数参考 21

API 流程概述 17

E

enumerateParameters

API 函数参考 22

API 流程概述 17

EPM

API 15

G

getErrorDetail

API 函数参考 28

API 流程概述 17

getFieldCount

API 函数参考 23

API 流程概述 17

getFieldTypes

API 函数参考 24

API 流程概述 17

getOption

API 函数参考 19

API 流程概述 17

getParameter

API 函数参考 22

API 流程概述 17

I

IBM SPSS 可嵌入式预测性分析

API 15

IBM SPSS Modeler 1

文档 3

IBM SPSS Modeler Server 1

IBM SPSS Modeler Solution

Publisher 5, 15

概述 5

嵌入到应用程序中 9

IBM SPSS Modeler Solution Publisher

节点 15

IBM SPSS Modeler Solution Publisher

Runtime

安装 11

启动 13

系统要求 11

卸载 14

temp 目录 13

initialise_ext

标志 18

API 函数参考 18

setAlternativeOutput

API 函数参考 26

API 流程概述 17

setLogFile

API 函数参考 20

API 流程概述 17

setOption

API 函数参考 20

API 流程概述 17

setParameter

API 函数参考 23

API 流程概述 17

setReportHandler

API 函数参考 29

API 流程概述 17

U

UNIX

安装 IBM SPSS Modeler Solution

Publisher Runtime 11

[特别字符]

.par 文件 5

.pim 文件 5

O

openImage

API 函数参考 21

API 流程概述 17

P

parameters

用于流执行的参数 8

API 参数文件 21

API 回调过程 22

API 图像参数 22, 23

Publish 节点 15

R

Runtime

启动 13

卸载 14

选项 7

temp 目录 13

UNIX 安装 11

Windows 安装 11

Runtime 编程库 (CLEMRTL) 9

S

setAlternativeInput

API 函数参考 25

API 流程概述 17



Printed in China