

IBM SPSS Modeler 14.2 Solution
Publisher



Hinweis: Lesen Sie vor der Verwendung dieser Informationen und des zugehörigen Produkts die allgemeinen Informationen unter Hinweise auf S. .

Dieses Dokument enthält eigentumsrechtlich geschützte Informationen von SPSS Inc., an IBM Company. Sie werden im Rahmen einer Lizenzvereinbarung bereitgestellt und sind durch Copyright-Gesetze geschützt. Die in dieser Publikation enthaltenen Informationen umfassen keinerlei Produktgewährleistungen und keine der Aussagen in diesem Handbuch darf als solche ausgelegt werden.

Wenn Sie Informationen an IBM bzw. SPSS senden, räumen Sie IBM und SPSS das nicht ausschließliche Recht ein, die Informationen in jeglicher Form zu verwenden bzw. weiterzugeben, die dem Unternehmen geeignet erscheint, ohne dass ihm daraus Verbindlichkeiten Ihnen gegenüber entstehen.

© Copyright IBM Corporation 1994, 2011..

Vorwort

IBM® SPSS® Modeler ist die auf Unternehmensebene einsetzbare Data-Mining-Workbench von IBM Corp.. Mit SPSS Modeler können Unternehmen und Organisationen die Beziehungen zu ihren Kunden bzw. zu den Bürgern durch ein tief greifendes Verständnis der Daten verbessern. Organisationen benutzen die mithilfe von SPSS Modeler gewonnenen Erkenntnisse zur Bindung profitabler Kunden, zur Ermittlung von Cross-Selling-Möglichkeiten, zur Gewinnung neuer Kunden, zur Ermittlung von Betrugsfällen, zur Reduzierung von Risiken und zur Verbesserung der Verfügbarkeit öffentlicher Dienstleistungen.

Die visuelle Benutzeroberfläche von SPSS Modeler erleichtert die Anwendung des spezifischen Geschäftswissens der Benutzer, was zu leistungsstärkeren Vorhersagemodellen führt und die Zeit bis zur Lösungserstellung verkürzt. SPSS Modeler bietet zahlreiche Modellierungsverfahren, beispielsweise Algorithmen für Vorhersage, Klassifizierung, Segmentierung und Assoziationserkennung. Nach der Modellerstellung ermöglicht IBM® SPSS® Modeler Solution Publisher die unternehmensweite Bereitstellung für Entscheidungsträger oder in einer Datenbank.

Über IBM Business Analytics

IBM Business Analytics-Software bietet vollständige, einheitliche und genaue Informationen, auf die Entscheidungsträger vertrauen, um die Unternehmensleistung zu steigern. Ein umfassendes Portfolio von Anwendungen für [Unternehmensinformationen](#), [Vorhersageanalysen](#), [Verwaltung der Finanzleistung und Strategie](#) sowie [Analysen](#) bietet sofort klare und umsetzbare Einblicke in die aktuelle Leistung und ermöglicht die Vorhersage zukünftiger Ergebnisse. In Kombination mit umfassenden Branchenlösungen, bewährten Vorgehensweisen und professionellen Dienstleistungen können Unternehmen jeder Größe optimale Produktivität erreichen, die Entscheidungsfindung zuverlässig automatisieren und bessere Ergebnisse erzielen.

Als Teil dieses Portfolios unterstützt die IBM SPSS Predictive Analytics-Software Unternehmen dabei, zukünftige Ereignisse vorherzusagen und aktiv auf diese Erkenntnisse zu reagieren, um bessere Geschäftsergebnisse zu erzielen. Kunden aus den Bereichen Wirtschaft, Behörden und Bildung aus aller Welt verlassen sich auf die IBM SPSS-Technologie. Sie bringt Ihnen beim Gewinnen, Halten und Ausbauen neuer Kundenbeziehungen einen Wettbewerbsvorteil und verringert gleichzeitig das Betrugs- sowie andere Risiken. Durch Integration der IBM SPSS-Software in den täglichen Betrieb können diese Unternehmen qualifizierte Vorhersagen treffen und dadurch die Entscheidungsfindung so ausrichten und automatisieren, dass Geschäftsziele erreicht werden und ein messbarer Wettbewerbsvorteil entsteht. Wenn Sie weitere Informationen wünschen oder einen Mitarbeiter kontaktieren möchten, ist dies unter <http://www.ibm.com/spss> möglich.

Technischer Support

Kunden mit Wartungsvertrag können den technischen Support in Anspruch nehmen. Kunden können sich an den technischen Support wenden, wenn sie Hilfe bei der Arbeit mit IBM Corp.-Produkten oder bei der Installation in einer der unterstützten Hardware-Umgebungen benötigen. Die Kontaktdaten des Technischen Supports finden Sie auf der IBM Corp.-Website

unter <http://www.ibm.com/support>. Sie müssen bei der Kontaktaufnahme Ihren Namen, Ihre Organisation und Ihre Supportvereinbarung angeben.

Inhalt

1	Informationen zu IBM SPSS Modeler	1
	IBM SPSS Modeler Server	1
	IBM SPSS Modeler-Optionen	2
	IBM SPSS Text Analytics	2
	IBM SPSS Modeler-Dokumentation	3
	Anwendungsbeispiele	4
	Ordner "Demos"	4
2	IBM SPSS Modeler Solution Publisher	6
	So funktioniert IBM SPSS Modeler Solution Publisher	6
	Veröffentlichen von Streams	7
3	Ausführen veröffentlichter Streams	10
	Verwenden der Parameterdatei zur Anpassung der Stream-Ausführung	11
	Einbetten von IBM SPSS Modeler Solution Publisher in Anwendungen	13
	IBM SPSS Modeler Solution Publisher-Beispielanwendung	14
4	Installieren von IBM SPSS Modeler Solution Publisher Runtime	15
	Installation der Runtime	15
	Systemanforderungen	15
	Installation unter Windows	16
	Installation unter UNIX	16
	Starten der Runtime	19
	Ändern des temporären Verzeichnisses	19
	Deinstallation von IBM SPSS Modeler Solution Publisher Runtime	20
	Windows-Deinstallationsprozedur	20
	UNIX-Deinstallationsprozedur	20

5 IBM SPSS Modeler Solution Publisher Runtime Library – API-Referenz 21

Einführung 21
Überblick über API-Prozesse 21
API-Funktionen 22

Anhang

A Hinweise 37

Index 40

Informationen zu IBM SPSS Modeler

IBM® SPSS® Modeler ist ein Set von Data-Mining-Tools, mit dem Sie auf der Grundlage Ihres Geschäftswissens schnell und einfach Vorhersagemodelle erstellen und zur Erleichterung der Entscheidungsfindung in die Betriebsabläufe einbinden können. SPSS Modeler, das auf der Grundlage des den Industrienormen entsprechenden Modells CRISP-DM entwickelt wurde, unterstützt den gesamten Data-Mining-Prozess, von den Daten bis hin zu besseren Geschäftsergebnissen.

SPSS Modeler bietet eine Vielzahl von Modellbildungsmethoden, die aus dem maschinellen Lernen, der künstlichen Intelligenz und der Statistik stammen. Mit den in der Modellierungspalette verfügbaren Methoden können Sie aus Ihren Daten neue Informationen ableiten und Vorhersagemodelle erstellen. Jede Methode besitzt ihre Stärken und eignet sich besonders für bestimmte Problemtypen.

SPSS Modeler kann als Standalone-Produkt oder in Verbindung mit SPSS Modeler Server erworben werden. Außerdem ist eine Reihe von Zusatzoptionen verfügbar, die in den folgenden Abschnitten kurz dargelegt werden. Weitere Informationen finden Sie unter <http://www.ibm.com/software/analytics/spss/products/modeler/>.

IBM SPSS Modeler Server

SPSS Modeler verwendet eine Client/Server-Architektur zur Verteilung von Anforderungen für ressourcenintensive Vorgänge an leistungsstarke Serversoftware, wodurch bei größeren Daten-Sets eine schnellere Leistung erzielt werden kann. Neben den hier aufgeführten Produkten können auch weitere Produkte bzw. Aktualisierungen verfügbar sein. Weitere Informationen finden Sie unter <http://www.ibm.com/software/analytics/spss/products/modeler/>.

SPSS Modeler. SPSS Modeler Clementine Client ist eine im Funktionsumfang vollständige Version des installierten Produkts und kann auf dem Desktop-Computer des Benutzers ausgeführt werden. Es kann im lokalen Modus als Standalone-Produkt oder im verteilten Modus zusammen mit IBM® SPSS® Modeler Server verwendet werden, um im Falle von großen Daten-Sets die Leistung zu verbessern.

SPSS Modeler Server. SPSS Modeler Server wird ständig im verteilten Analysemodus zusammen mit einer oder mehreren IBM® SPSS® Modeler-Installationen ausgeführt, wodurch eine herausragende Leistung bei großen Daten-Sets erzielt werden kann, da speicherintensive Operationen auf dem Server durchgeführt werden können, ohne Daten auf den Client-Computer herunterzuladen. SPSS Modeler Server bietet außerdem Unterstützung für SQL-Optimierung sowie Funktionen zur Modellierung innerhalb der Datenbank, wodurch Leistungsfähigkeit und Automatisierung weiter verbessert werden. Es muss mindestens eine SPSS Modeler-Installation vorhanden sein, um eine Analyse durchzuführen.

IBM SPSS Modeler-Optionen

Die folgenden Komponenten und Funktionen können separat erworben und für die Verwendung mit SPSS Modeler lizenziert werden. Beachten Sie, dass zu einem späteren Zeitpunkt möglicherweise noch weitere Produkte und Updates erhältlich sind. Weitere Informationen finden Sie unter <http://www.ibm.com/software/analytics/spss/products/modeler/>.

- SPSS Modeler Server-Zugriff, der eine bessere Skalierbarkeit und bessere Leistungsfähigkeit bei großen Daten-Sets bietet sowie Unterstützung für SQL-Optimierung und Funktionen zur Modellierung innerhalb der Datenbank.
- SPSS Modeler Solution Publisher, für Scoring in Echtzeit oder automatisiertes Scoring außerhalb der SPSS Modeler-Umgebung. Für weitere Informationen siehe Thema [IBM SPSS Modeler Solution Publisher in Kapitel 2 auf S. 6](#).
- Adapter zum Deployment für IBM SPSS Collaboration and Deployment Services oder die Thin-Client-Anwendung IBM SPSS Modeler Advantage. Für weitere Informationen siehe Thema [Speichern und Bereitstellen von IBM SPSS Collaboration and Deployment Services Repository-Objekten in Kapitel 9 in IBM SPSS Modeler 14.2- Benutzerhandbuch](#).

IBM SPSS Text Analytics

IBM® SPSS® Text Analytics ist ein vollständig integriertes Zusatzprodukt für SPSS Modeler, das hoch entwickelte linguistische Technologien und die Verarbeitung natürlicher Sprache (Natural Language Processing, NLP) benutzt, um eine schnelle Verarbeitung einer großen Vielfalt an unstrukturierten Textdaten zu ermöglichen, um die Schlüsselkonzepte zu extrahieren und zu ordnen und um diese Konzepte in Kategorien zusammenzufassen. Extrahierte Konzepte und Kategorien können mit bestehenden strukturierten Daten, beispielsweise demografischen Informationen, kombiniert und mithilfe der vollständigen Suite der Data-Mining-Tools von IBM® SPSS® Modeler auf die Modellierung angewendet werden, um bessere und fokussiertere Entscheidungen zu ermöglichen.

- Der Text-Mining-Knoten bietet die Modellierung von Konzepten und Kategorien sowie eine interaktive Workbench, in der Sie eine erweiterte Untersuchung von Textlinks und Clustern durchführen, Ihre eigenen Kategorien erstellen und die Vorlagen für linguistische Ressourcen verfeinern können.
- Eine Reihe von Importformaten wird unterstützt, darunter Blogs und andere webbasierte Quellen.
- Benutzerdefinierte Vorlagen, Bibliotheken und Wörterbücher für bestimmte Domänen, wie CRM und Genomforschung, sind ebenfalls eingeschlossen.

Anmerkung: Für den Zugriff auf diese Komponente ist eine separate Lizenz erforderlich. Weitere Informationen finden Sie unter <http://www.ibm.com/software/analytics/spss/products/modeler/>.

IBM SPSS Modeler-Dokumentation

Die vollständige Dokumentation im Online-Hilfe-Format finden Sie im Hilfe-Menü von SPSS Modeler. Dazu gehören die Dokumentation für SPSS Modeler, SPSS Modeler Server und SPSS Modeler Solution Publisher sowie das Anwendungshandbuch und weiteres Material zur Unterstützung.

Die vollständige Dokumentation für die einzelnen Produkte im PDF-Format finden Sie im Ordner *Documentation* auf der jeweiligen Produkt-DVD.

- **IBM SPSS Modeler-Benutzerhandbuch.** Allgemeine Einführung in die Verwendung von SPSS Modeler, in der u. a. die Erstellung von Daten-Streams, der Umgang mit fehlenden Werten, die Erstellung von CLEM-Ausdrücken, die Arbeit mit Projekten und Berichten sowie das Packen von Streams für das Deployment in IBM SPSS Collaboration and Deployment Services, Predictive Applications (Prognoseanwendungen) oder IBM SPSS Modeler Advantage beschrieben werden.
- **Quellen-, Prozess- und Ausgabeknoten in IBM SPSS Modeler.** Beschreibung aller Knoten, die zum Lesen, zum Verarbeiten und zur Ausgabe von Daten in verschiedenen Formaten verwendet werden. Im Grunde sind sie alle Knoten, mit Ausnahme der Modellierungsknoten.
- **IBM SPSS Modeler Modellierungsknoten.** Beschreibungen sämtlicher für die Erstellung von Data Mining-Modellen verwendeter Knoten. IBM® SPSS® Modeler bietet eine Vielzahl von Modellbildungsmethoden, die aus dem maschinellen Lernen, der künstlichen Intelligenz und der Statistik stammen. [Für weitere Informationen siehe Thema Überblick über Modellierungsknoten in Kapitel 3 in IBM SPSS Modeler 14.2-Modellierungsknoten.](#)
- **IBM SPSS Modeler-Algorithmushandbuch.** Beschreibung der mathematischen Grundlagen der in SPSS Modeler verwendeten Modellierungsmethoden.
- **IBM SPSS Modeler-Anwendungshandbuch.** Die Beispiele in diesem Handbuch bieten eine kurze, gezielte Einführung in bestimmte Modellierungsmethoden und -verfahren. Eine Online-Version dieses Handbuchs kann auch über das Hilfe-Menü aufgerufen werden. [Für weitere Informationen siehe Thema Anwendungsbeispiele in IBM SPSS Modeler 14.2-Benutzerhandbuch.](#)
- **Skripterstellung und Automatisierung in IBM SPSS Modeler.** Informationen zur Automatisierung des Systems über Skripterstellung, einschließlich der Eigenschaften, die zur Bearbeitung von Knoten und Streams verwendet werden können.
- **IBM SPSS Modeler Deployment-Handbuch.** Informationen zum Ausführen von SPSS Modeler-Streams und -Szenarien als Schritte bei der Verarbeitung von Jobs im IBM® SPSS® Collaboration and Deployment Services Deployment Manager.
- **IBM SPSS Modeler CLEF-Entwicklerhandbuch.** CLEF bietet die Möglichkeit, Drittanbieterprogramme, wie Datenverarbeitungsroutinen oder Modellierungsalgorithmen, als Knoten in SPSS Modeler zu integrieren.
- **In-Database Mining-Handbuch für IBM SPSS Modeler.** Informationen darüber, wie Sie Ihre Datenbank dazu einsetzen, die Leistung zu verbessern, und wie Sie die Palette der Analysefunktionen über Drittanbieteralgorithmen erweitern.
- **IBM SPSS Modeler Server- und -Leistungshandbuch.** Informationen zur Konfiguration und Verwaltung von IBM® SPSS® Modeler Server.

- **IBM SPSS Modeler Administration Console – Benutzerhandbuch.** Informationen zur Installation und Nutzung der Konsolen-Benutzeroberfläche zur Überwachung und Konfiguration von SPSS Modeler Server. Die Konsole ist als Plugin für die Deployment Manager-Anwendung implementiert.
- **IBM SPSS Modeler Solution Publisher-Handbuch.** SPSS Modeler Solution Publisher ist eine Zusatzkomponente, mit der Unternehmen Streams zur Verwendung außerhalb der SPSS Modeler-Standardumgebung veröffentlichen können.
- **IBM SPSS Modeler-Handbuch zu CRISP-DM.** Schritt-für-Schritt-Anleitung für das Data-Mining mit SPSS Modeler unter Verwendung der CRISP-DM-Methode.

Anwendungsbeispiele

Mit den Data-Mining-Tools in SPSS Modeler kann eine große Bandbreite an geschäfts- und unternehmensbezogenen Problemen gelöst werden; die Anwendungsbeispiele dagegen bieten jeweils eine kurze, gezielte Einführung in spezielle Modellierungsmethoden und -verfahren. Die hier verwendeten Daten-Sets sind viel kleiner als die riesigen Datenbestände, die von einigen Data-Mining-Experten verwaltet werden müssen, die zugrunde liegenden Konzepte und Methoden sollten sich jedoch auch auf reale Anwendungen übertragen lassen.

Sie können auf die Beispiele zugreifen, indem Sie im Menü “Hilfe” in SPSS Modeler auf die Option Anwendungsbeispiele klicken. Die Datendateien und Beispiel-Streams wurden im Ordner *Demos*, einem Unterordner des Produktinstallationsverzeichnis, installiert. [Für weitere Informationen siehe Thema Ordner “Demos” in IBM SPSS Modeler 14.2- Benutzerhandbuch.](#)

Beispiele für die Datenbank-Modellierung. Die Beispiele finden Sie im *IBM SPSS Modeler In-Database Mining-Handbuch*.

Skriptbeispiele. Die Beispiele finden Sie im *IBM SPSS Modeler Handbuch für die Skripterstellung und Automatisierung*.

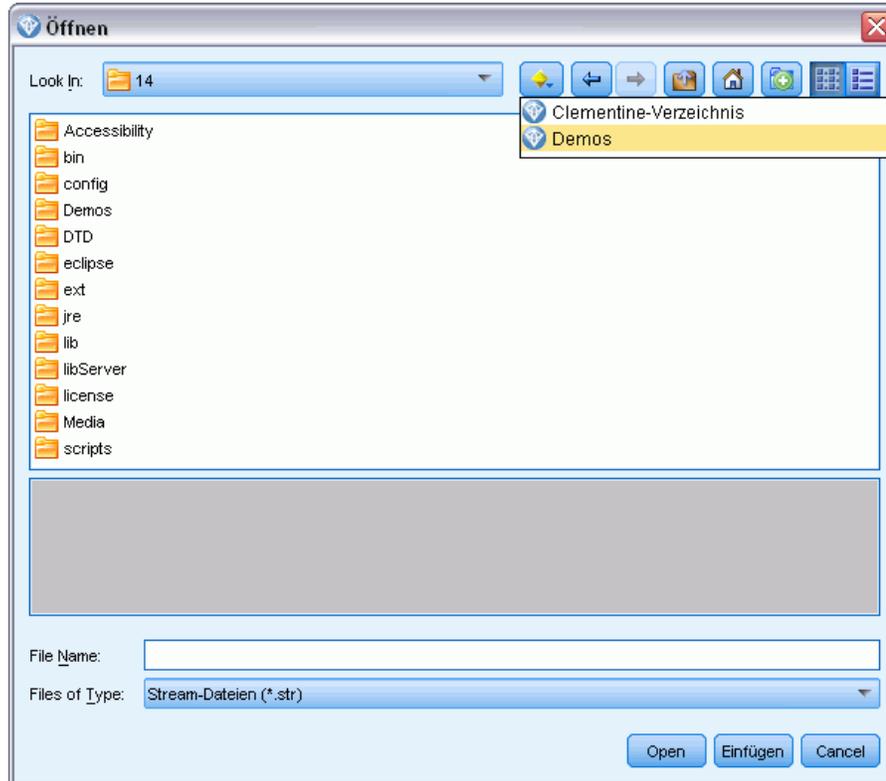
Ordner “Demos”

Die in den Anwendungsbeispielen verwendeten Datendateien und Beispiel-Streams wurden im Ordner *Demos*, einem Unterordner des Produktinstallationsverzeichnis, installiert. Auf diesen Ordner können Sie auch über die Programmgruppe IBM SPSS Modeler 14.2 im

Windows-Startmenü oder durch Klicken auf *Demos* in der Liste der zuletzt angezeigten Verzeichnisse im Dialogfeld "Datei öffnen" zugreifen.

Abbildung 1-1

Auswahl des Ordners "Demos" in der Liste der zuletzt angezeigten Verzeichnisse



IBM SPSS Modeler Solution Publisher

IBM® SPSS® Modeler Solution Publisher ist ein leistungsstarkes Tool, mit dem Sie Data-Mining-Ergebnisse in Ihre Geschäftsabläufe integrieren und auf diese Weise praxisbezogene Probleme lösen können. Mithilfe von SPSS Modeler Solution Publisher können Sie eine “verpackte” Version eines Streams erstellen, der von einer externen Runtime-Engine ausgeführt oder in eine externe Anwendung eingebettet werden kann. Dadurch können Sie Ihre Datenmodellierungs-Streams in einer Produktionsumgebung verwenden, um die alltäglichen Geschäftsprozesse zu unterstützen und den Entscheidungsträgern Ihres Unternehmens die aus dem Data-Mining gewonnenen Erkenntnisse zur Verfügung zu stellen.

Die Verwendung von SPSS Modeler Solution Publisher bietet mehr als nur den Export des Modells (als PMML), denn Sie können damit *vollständige IBM® SPSS® Modeler-Streams* veröffentlichen und verwenden. Sie können also Datenvorbereitungsaufgaben sowie Datensatz- und Feldoperationen, wie das Aggregieren von Daten, das Auswählen von Datensätzen oder das Ableiten neuer Felder, durchführen, bevor Sie Vorhersagen auf der Grundlage eines Modells erstellen. Sie können die Modellergebnisse danach weiter verarbeiten, bevor Sie die Daten speichern – und das alles einfach, indem Sie den veröffentlichten Stream ausführen.

Hinweis: Veröffentlichung wird mithilfe der Registerkarte “Veröffentlichen” des entsprechenden Exportknotens erreicht. [Für weitere Informationen siehe Thema Veröffentlichen von Streams auf S. 7.](#) Für Streams, die in Versionen vor Version 11.0 des Produkts gespeichert wurden, wurde ein gesonderter Publisher-Knoten verwendet. Dieser Knoten wird zur Gewährleistung von Abwärtskompatibilität weiterhin unterstützt, steht jedoch auf der Knotenpalette nicht mehr zur Verfügung.

Lizenzieren von IBM SPSS Modeler Solution Publisher

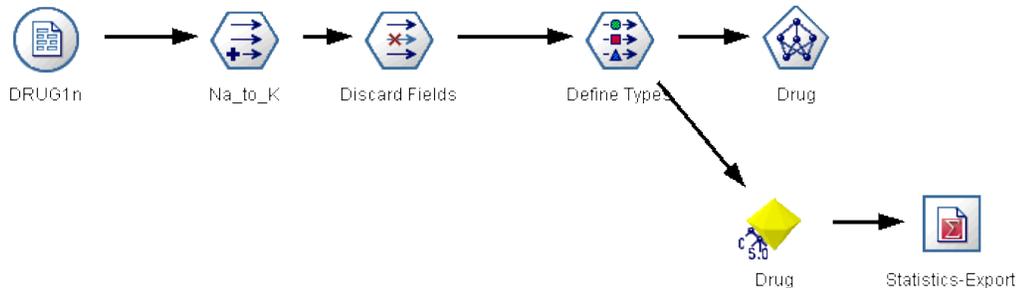
Hinweis: SPSS Modeler Solution Publisher wird als Teil des IBM® SPSS® Collaboration and Deployment Services Scoring Service vertrieben, für den eine gesonderte Lizenz erforderlich ist. <http://www.ibm.com/software/analytics/spss/products/deployment/cds/>

Wenn Sie eine Lizenz erwerben, erhalten Sie ein separates IBM® SPSS® Modeler Solution Publisher Runtime, mit dem Sie die veröffentlichten Streams ausführen können. [Für weitere Informationen siehe Thema Installation der Runtime in Kapitel 4 auf S. 15.](#)

So funktioniert IBM SPSS Modeler Solution Publisher

Die Bereitstellung einer Lösung mithilfe von IBM® SPSS® Modeler Solution Publisher gliedert sich in zwei Phasen: *Veröffentlichen* eines Streams und *Ausführen* eines Streams.

Abbildung 2-1
Veröffentlichen eines Streams



Veröffentlichen. Wenn Sie den Data-Mining-Prozess abgearbeitet haben, erhalten Sie schließlich ein Modell, das eine gute Lösung für Ihr Geschäftsproblem darstellt. Nun können Sie das Modell auf Ihren Geschäftsprozess anwenden. Wenn Sie einen Stream veröffentlichen, wird eine detaillierte Beschreibung des Streams auf den Datenträger geschrieben (als *Image*-Datei und *Parameter*-Datei). [Für weitere Informationen siehe Thema Veröffentlichen von Streams auf S. 7.](#)

Ausführung. Nach der Veröffentlichung des Streams können Sie den im Stream implementierten Prozess neu erstellen, indem Sie den veröffentlichten Stream ausführen. Dies geschieht entweder über die eigenständige IBM® SPSS® Modeler Runtime (*modelerrun.exe*) oder durch Entwicklung einer Anwendung, die SPSS Modeler Runtime Library zur Ausführung des Streams verwendet. Um Streams außerhalb von SPSS Modeler auszuführen (mithilfe der Runtime oder über eine benutzerdefinierte Anwendung), müssen Sie zunächst IBM® SPSS® Modeler Solution Publisher Runtime installieren. [Für weitere Informationen siehe Thema Installation der Runtime in Kapitel 4 auf S. 15.](#)

Veröffentlichen von Streams

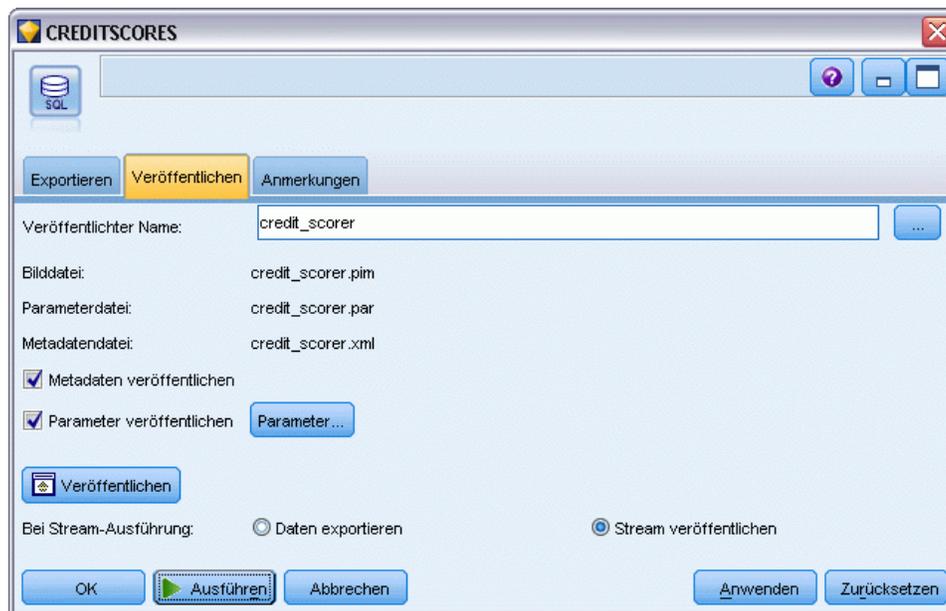
Die Veröffentlichung der Streams erfolgt direkt aus IBM® SPSS® Modeler unter Verwendung eines der standardmäßigen Exportknoten: Datenbank-, Textdatei-, Statistics-Export-, IBM® SPSS® Data Collection-Export-, SAS-Export-, Excel- und XML-Exportknoten. Der Typ des Exportknotens bestimmt das Format der Ergebnisse, die bei jeder Ausführung des veröffentlichten Streams mithilfe von IBM® SPSS® Modeler Solution Publisher Runtime oder einer externen Anwendung geschrieben werden sollen. Beispiel: Wenn die Ergebnisse bei jeder Ausführung des veröffentlichten Streams in eine Datenbank geschrieben werden sollen, müssen Sie einen Datenbankexportknoten verwenden. [Für weitere Informationen siehe Thema Überblick über Exportknoten in Kapitel 7 in IBM SPSS Modeler 14.2- Quellen-, Prozess- und Ausgabeknoten.](#)

So veröffentlichen Sie einen Stream:

- Öffnen bzw. erstellen Sie einen Stream auf normale Art und Weise und fügen Sie am Ende einen Exportknoten an.

- ▶ Geben Sie in der Registerkarte “Veröffentlichen” im Exportknoten einen Stammmamen für die veröffentlichten Dateien an (d. h. den Dateinamen, an den die verschiedenen Erweiterungen *.pim*, *.par* und *.xml* angefügt werden).
- ▶ Klicken Sie auf die Schaltfläche Veröffentlichen, um den Stream zu veröffentlichen, oder wählen Sie Stream veröffentlichen aus, um den Stream bei jeder Ausführung des Knotens automatisch veröffentlichen zu lassen.

Abbildung 2-2
Registerkarte “Veröffentlichen”



Veröffentlichter Name. Geben Sie den Stammmamen für die veröffentlichten Image- und Parameterdateien an.

- Die **Image-Datei** (**.pim*) enthält alle Informationen, die die Runtime benötigt, um den veröffentlichten Stream genau in dem Zustand auszuführen, in dem er sich zum Zeitpunkt des Exports befand. Wenn Sie sich sicher sind, dass Sie keine der Einstellungen für den Stream ändern müssen (beispielsweise die Eingangsdatenquelle oder die Ausgabedatei) können Sie auch nur die Bilddatei verwenden.
- Die **Parameterdatei** (**.par*) enthält konfigurierbare Informationen zu Datenquellen, Ausgabedateien und Ausführungsoptionen. Wenn Sie die Eingabe bzw. Ausgabe des Streams steuern möchten, ohne den Stream erneut veröffentlichen zu müssen, benötigen Sie neben der Bilddatei auch die Parameterdatei.
- Die **Metadatendatei** (**.xml*) beschreibt die Ein- und Ausgaben des Bilds und der zugehörigen Datenmodelle. Sie ist für Anwendungen gedacht, die die Laufzeitbibliothek (Runtime Library) einbetten und denen die Struktur der Ein- und Ausgabedaten bekannt sein muss.

Hinweis: Diese Datei wird nur erstellt, wenn Sie das Kontrollkästchen Metadaten veröffentlichen auswählen.

Parameter veröffentlichen. Falls erforderlich, können Sie Stream-Parameter in die *.par-Datei aufnehmen. Sie können die Werte dieser Stream-Parameter beim Ausführen des Bilds ändern, entweder, indem Sie die Datei *.par bearbeiten oder alternativ über die Laufzeit-API.

Mit diesem Kontrollkästchen aktivieren Sie die Schaltfläche Parameter. Beim Klicken auf die Schaltfläche wird das Dialogfeld “Parameter veröffentlichen” angezeigt.

Abbildung 2-3
Parameterauswahl



Wählen Sie die Parameter aus, die Sie in das veröffentlichte Bild aufnehmen möchten, indem Sie in der Spalte Veröffentlichen das entsprechende Kontrollkästchen aktivieren.

Bei Stream-Ausführung. Gibt an, ob der Stream bei der Ausführung des Knotens automatisch veröffentlicht werden soll.

- **Daten exportieren.** Führt den Exportknoten in der standardmäßigen Art und Weise aus, ohne Veröffentlichung des Streams. (Im Grunde wird der Knoten in SPSS Modeler genau so ausgeführt, wie wenn IBM® SPSS® Modeler Solution Publisher nicht verfügbar wäre.) Bei Auswahl dieser Option, wird der Stream nur veröffentlicht, wenn Sie die Veröffentlichung explizit durch Klicken auf die Schaltfläche Veröffentlichen im Dialogfeld des Exportknotens auslösen. Alternativ können Sie den aktuellen Stream mithilfe des Tools “Veröffentlichen” in der Symbolleiste oder unter Verwendung eines Skripts veröffentlichen.
- **Stream veröffentlichen.** Veröffentlicht den Stream für die Verwendung mithilfe von SPSS Modeler Solution Publisher. Wählen Sie diese Option, wenn der Stream bei jeder Ausführung automatisch veröffentlicht werden soll.

Kommentare

- Wenn Sie vorhaben, den veröffentlichten Stream mit neuen oder aktualisierten Daten auszuführen, dürfen Sie nicht vergessen, dass die Reihenfolge der Felder in der Eingabedatei mit der Reihenfolge der Felder in der im veröffentlichten Stream angegebenen Quellenknoten-Eingabedatei übereinstimmen muss.
- Bei der Veröffentlichung in externe Anwendungen sollten Sie die Filterung überflüssiger Felder bzw. das Umbenennen von Feldern in Erwägung ziehen, um die Eingabeanforderungen zu erfüllen. Beides lässt sich durch Verwendung eines Filterknotens vor dem Exportknoten erreichen. [Für weitere Informationen siehe Thema Filtern bzw. Umbenennen von Feldern in Kapitel 4 in IBM SPSS Modeler 14.2- Quellen-, Prozess- und Ausgabeknoten.](#)

Ausführen veröffentlichter Streams

Das wichtigste Ergebnis bei der Ausführung eines veröffentlichten Streams ist stets die Erzeugung eines Daten-Sets. Die Daten können in einer Datenträgerdatei gespeichert oder in eine Datenbank geschrieben werden. Veröffentlichte Streams können keine Diagramme, Tabellen, generierte Modelle oder Ausgaben, bei denen es sich nicht um Daten handelt, erzeugen.

Über IBM® SPSS® Modeler Solution Publisher veröffentlichte Streams können mithilfe des Programms IBM® SPSS® Modeler Solution Publisher Runtime ausgeführt werden. Das Runtime-Programm, *modelerrun.exe*, wird über die Befehlszeile gestartet. Dabei werden die auszuführende Bilddatei und (optional) die Parameterdatei durch Optionen angegeben. Der Runtime-Befehl lautet wie folgt:

```
modelerrun -nobanner -p <Parameterdatei> -o <Optionen> <Bilddatei>
```

Dabei gilt:

-nobanner	unterdrückt die Starttextmeldung für die Runtime (optional).
<Parameterdatei>	ist der Dateiname der Parameterdatei (optional).
<Optionen>	ist eine kommagetrennte Liste von Option=Wert-Paaren, die Ausführungsoptionen für die Laufzeit angeben. Die Optionen werden weiter unten im Detail angegeben (optional).
<Bilddatei>	ist der Dateiname der auszuführenden Bilddatei (erforderlich).

Folgende Ausführungsoptionen stehen zur Verfügung:

Tabelle 3-1
Gültige Runtime-Optionen

Name	Standardwert	Beschreibung
max_file_size	-1	Maximale Größe der gespeicherten Dateien. Der Wert -1 bedeutet, dass keinerlei Beschränkung vorliegt.
max_sql_string_length	2,048	Maximale Länge für Zeichenketten, die aus einer Datenbank mit SQL importiert werden. Zeichenkettenwerte, die länger sind als dieser Wert, werden ohne Warnmeldung rechts gekürzt. Der zulässige Bereich liegt zwischen 1 und 65.535 Zeichen.
memory_usage	100	Multiplikator für die dynamische Arbeitsspeicherzuordnung. Passen Sie diesen Wert nach oben oder unten an, um den Gesamtwert für die Arbeitsspeicherauslastung des Servers zu regulieren.

Name	Standardwert	Beschreibung
temp_directory	"	Gibt das für temporäre Dateien zu verwendende Verzeichnis an.
request_passwords	J	Regelt, ob Sie bei der Ausführung eines veröffentlichten Streams, für den Datenbankzugriff erforderlich ist, aufgefordert werden, ein Datenbankpasswort einzugeben (Y/N).

Verwenden der Parameterdatei zur Anpassung der Stream-Ausführung

Wenn ein Stream veröffentlicht wird, enthält die Bilddatei Informationen zur Datenquelle und zum Ziel der im ursprünglichen Stream gesammelten Ausgabedaten. Es ist jedoch häufig zweckmäßig, einen Stream mit Daten aus einer anderen Datenquelle auszuführen oder die Ausgabedaten an ein anderes Ziel zu leiten. Dies ist durch Bearbeitung der Parameterdatei des Streams möglich.

Dies ist besonders wichtig, wenn Sie den veröffentlichten Stream auf einem anderen Computer ausführen als dem, auf dem er erstellt wurde. Sie müssen die Speicherorte der Ein- und Ausgabedateien so aktualisieren, dass sie die Struktur des Zielcomputers berücksichtigen. Beachten Sie, dass veröffentlichte Streams keine relativen Pfade unterstützen, sodass es besonders wichtig ist, die in der Parameterdatei angegebenen Datenspeicherorte zu überprüfen. (Wenn Sie beim Veröffentlichen des Streams in den Optionen des Exportknotens relative Pfade angeben, wandelt IBM® SPSS® Modeler diese vor der Erstellung der Bild- und Parameterdatei für den veröffentlichten Stream in absolute Pfade um.)

Außerdem können Sie mithilfe der Parameterdatei Ausführungsoptionen festlegen und die Werte von Stream-Parametern ändern.

Die Parameterwerte werden in der Parameterdatei als <name.attribute>=<value>-Paare angegeben. Bei einigen Parametern enthält der Parametername eine ID-Stelle, mit der Knoten desselben Typs im Stream unterschieden werden. Beispiel: In einem Stream, in dem sich sowohl die Eingabe als auch die Ausgabe auf eine Datenbankverbindung beziehen, könnten die Parameter, die die Eingabeverbindung beschreiben, dbconn0.datasource, dbconn0.user usw. lauten und die der Ausgabeverbindung zugeordneten Parameter könnten die Bezeichnungen dbconn1.datasource, dbconn1.user usw. tragen. Werte werden stets in doppelte Anführungszeichen gesetzt. Die Parameterdatei enthält folgende Parameter, von denen sich alle zur Anpassung der Stream-Ausführung ändern lassen:

Tabelle 3-2
Ausführungsparameter

Parameter	Zweck
dbconnx.datasource	Datenquellename (DSN)
dbconnx.user	Benutzername für Datenbanken mit eingeschränktem Zugriff.
dbconnx.password	Passwort für Datenbanken mit eingeschränktem Zugriff.
dbconnx.epassword	Kodiertes Passwort für Datenbanken mit eingeschränktem Zugriff. Zur Erzeugung eines verschlüsselten Passworts wählen Sie in der SPSS Modeler-Benutzeroberfläche im Menü "Extras" die Option Passwort verschlüsseln. Sie können das kodierte Passwort nach Bedarf kopieren und einfügen.

Parameter	Zweck
dbconnx.password_required	Flag, die angibt, ob zur Eingabe eines Passworts für diese Datenbankverbindung aufgefordert wird oder nicht. (Die möglichen Werte sind "Y" (Ja) oder "N" (Nein).) Dieser Wert ist automatisch auf "N" gesetzt, wenn zum Veröffentlichungszeitpunkt kein Passwort angegeben wurde.
dbconnx.catalog	Der der Datenbankverbindung zugeordnete Katalog.
dbtablex.name	Name der Datenbanktabelle. <i>Anmerkung:</i> Die Reihenfolge der Felder in einer neuen Tabelle muss mit der Reihenfolge der Felder im ursprünglich veröffentlichten Stream übereinstimmen.
dbtablex.schema	Name des Schemas.
dbtablex.catalog	Der der aktuellen Tabelle zugeordnete Katalog.
filex.name	Dateiname für die Datendatei. <i>Anmerkung:</i> Die Reihenfolge der Felder in einer neuen Eingabetabelle muss mit der Reihenfolge der Felder im ursprünglich veröffentlichten Stream übereinstimmen.
filex.path	Speicherort der Datei (Verzeichnisname).
filex.field_names_included	Bei Dateien im Textformat ist dies das Flag, das angibt, ob die Feldnamen als erste Zeile der Datei mit aufgenommen wurden. (Die möglichen Werte sind "Y" (Ja) oder "N" (Nein).)
filex.field_separator	Bei Dateien im Textformat ist dies das zum Trennen der Feldwerte verwendete Zeichen (auch mehrere Zeichen möglich).
filex.decimal_separator	Bei Dateien im Textformat ist dies das als Dezimaltrennzeichen verwendete Zeichen.
options.angle_in_radians	Flag, das angibt, ob Radianten als Maßeinheit in trigonometrischen CLEM-Ausdrücken verwendet werden. (Die möglichen Werte sind "Y" (Ja) oder "N" (Nein).)
options.date_2digit_baseline	Legt das Jahrhundert für Datumsangaben mit zweistelliger Jahreszahl fest.
options.date_baseline	Das Basisjahr (immer 1. Jan.), das von den CLEM-Datumsfunktionen verwendet wird, die mit einem einzelnen Datum arbeiten.
options.time_rollover	Flag, das angibt, ob sich negative Zeitdifferenzen auf die Vergangenheit beziehen.
options.decimal_separator	Das Standarddezimaltrennzeichen in Textdateien.
options.time_format	Zeitformat, das verwendet wird, wenn Zeichenketten von CLEM-Zeitfunktionen als Zeiten interpretiert werden.
options.date_format	Datumsformat, das verwendet wird, wenn Zeichenketten von CLEM-Datumsfunktionen als Datumsangaben interpretiert werden.
options.timestamp_format	Das beim Einlesen von Zeitstempelfeldern als Zeichenketten aus ODBC-Datenquellen verwendete Format.
paramx.value	Wert des Stream-Parameters. <i>Anmerkung:</i> Name, Beschreibung und Speichertyp des Parameters sind als Referenz angegeben, können jedoch nicht geändert werden. Der Wert muss für den Typ gültig sein.

Einbetten von IBM SPSS Modeler Solution Publisher in Anwendungen

Neben der eigenständigen Runtime-Engine bietet IBM® SPSS® Modeler Solution Publisher eine Runtime-Programmiersbibliothek (CLEMRTL), mit der andere Programme die Ausführung des SPSS Modeler Solution Publisher-Streams steuern können. Sie können CLEMRTL-Verfahren in Client-Programmen aufrufen, die in C, C++ geschrieben wurden. Zur Verwendung von CLEMRTL müssen Sie die Header-Datei *clemrtl.h* (verfügbar im Ordner *..\installation\clemrtl\include*) einfügen und die entsprechende Bibliotheksdatei für Ihre Entwicklungsplattform bei der Erstellung mit Ihrer Anwendung verknüpfen. Die Datei "clemrtl.dll" befindet sich im Ordner *..\installation\bin*.

Alle Quelldateien, die Bibliotheksprozeduren referenzieren, müssen den Header *clemrtl.h* enthalten. Diese Header-Datei bietet ANSI C-Prototypen für die Bibliotheks-Prozeduren und definiert nützliche Makros. Es erfordert keine weiteren Header als die für Ihr Programm benötigten. Als Schutz vor Namenskonflikten beginnen alle Namen von Bibliothekstypen und -funktionen mit *clemrtl_* und alle Makronamen ist *CLEMRTL_* vorangestellt.

CLEMRTL erstellt Berichte mit nützlichen Informationen, die wieder an die Anwendung zurückgemeldet werden sollten. CLEMRTL bietet drei Mechanismen für den Umgang mit solchen Nachrichten:

- Meldungen können in ein mit der Funktion *clemrtl_setLogFile()* angegebenes Protokoll geschrieben werden.
- Die Anwendung kann mit der Funktion *clemrtl_getErrorDetail()* Einzelheiten des letzten Fehlers abrufen.
- Mit der Funktion *clemrtl_setReportHandler()* kann die Anwendung ihr eigenes Verfahren für den Umgang mit Berichten bereitstellen.

Das Runtime-System verfügt über einen lokalisierten Meldungskatalog für Berichte. Jeder an die Anwendung weitergegebene Bericht enthält die Zeichenkette der lokalisierten Meldung. Berichte enthalten außerdem den Berichtscode, sodass die Anwendung die Meldung auch anders interpretieren und darstellen kann. Zum Abrufen detaillierter (lokalisierter) Meldungen, muss die Anwendung die entsprechende *messages.cfg*-Datei in ein *config*-Verzeichnis transferieren. Verschiedene Versionen von *messages.cfg* für verschiedene Ländereinstellungen finden Sie im Verzeichnis *<installdir>/config/<locale>*. Dabei ist *<installdir>* das Verzeichnis, in dem SPSS Modeler Solution Publisher installiert wurde, und *<locale>* die gewünschte Ländereinstellung. Wählen Sie die entsprechende Version von *messages.cfg* und fügen Sie sie zu einem *config*-Unterverzeichnis in Ihrem Projektverzeichnis hinzu.

Das Runtime-System benötigt für den korrekten Betrieb keine speziellen Umgebungs- oder Registrierungseinstellungen. Abhängige Bibliotheken müssen mit einer mit der CLEMRTL verknüpften Anwendung verteilt werden. Diese Dateien finden Sie im Verzeichnis *<installdir>/bin*.

Voraussetzungen

Folgende Plattformen und Compiler wurden mit der Runtime-Programmierbibliothek getestet:

Tabelle 3-3
Plattformen und Compiler

Plattform	Vorgeschlagene(r) Compiler
AIX	IBM XL C/C++ Enterprise Edition V8.0 für AIX
HP-UX 11.0	aC++/ANSI C B3910B A.06.05
Solaris 64-Bit	Sun Studio 11 5.8 Patch 121017-06
Linux 32-Bit und 64-Bit	gcc 3.2.3, Intel C++-Compiler 9.1.043
Windows	Visual Studio 2005

Anmerkung: Unter HP-UX muss die Anwendung mit dem aCC-Compiler-Flag “-AA” erstellt werden, um die Verknüpfung mit den CLEMRTL-Bibliotheken zu ermöglichen.

IBM SPSS Modeler Solution Publisher-Beispielanwendung

Eine Beispielanwendung, *dlltest.c*, finden Sie im Ordner `\clemtrl\demo` im Installationsverzeichnis von IBM® SPSS® Modeler Solution Publisher. Verwenden Sie die Beispielanwendung, um sich mit der Einbettung der Runtime-Bibliothek in Ihre eigenen Anwendungen vertraut zu machen.

Die von der Beispielanwendung verwendeten Informationen finden Sie im Ordner `\clemtrl\demo`. Anweisungen dazu entnehmen Sie bitte der Datei *README.txt*. Die Kompilierung von Anwendungen ist je nach Plattform unterschiedlich, weshalb auch Beispiel-Projektdateien enthalten sind.

Installieren von IBM SPSS Modeler Solution Publisher Runtime

Installation der Runtime

Wenn Sie IBM® SPSS® Modeler Solution Publisher als Zusatzkomponente für IBM® SPSS® Modeler erwerben, erhalten Sie ein separates Installationsmedium. Mit diesem Medium installieren Sie IBM® SPSS® Modeler Solution Publisher Runtime. Außerdem können Sie damit veröffentlichte Streams ausführen. Veröffentlichte Streams sind vom Betriebssystem unabhängig. Sie können also jeden veröffentlichten Stream auf jeder von der Runtime unterstützten Plattform ausführen.

Hinweis: Die Funktionalität, mit der Sie Streams aus SPSS Modeler veröffentlichen können, wird mit IBM® SPSS® Modeler installiert und lässt sich durch Erwerb eines gesonderten Autorisierungscode aktivieren. Beachten Sie, dass bei jeder Version von SPSS Modeler die veröffentlichten Dateien erneut exportiert werden müssen. So funktioniert beispielsweise eine Datei, die mit einer früheren Version exportiert wurde, nicht in Version 13 von SPSS Modeler Solution Publisher Runtime.

Systemanforderungen

Folgende Systemanforderungen gelten für die Installation von IBM® SPSS® Modeler Solution Publisher Runtime:

- **Betriebssystem.** Windows Server 2008 R2 für 64-Bit x64-Systeme; Windows Server 2003 Standard Edition R2 für 32-Bit x86- oder 64-Bit x64-Systeme
IBM AIX® 6.1 oder 7.1 für 64-Bit POWER-Systeme; HP-UX 11i V3 für 64-Bit Itanium-Systeme; Oracle Solaris™ 9.x oder 10 für 64-Bit SPARC-Systeme; Red Hat Enterprise Linux 5.x für 32-Bit x86-Systeme; Red Hat Enterprise Linux 5.x oder 6.x für 64-Bit x64- oder IBM System z-Systeme; Red Hat Enterprise Linux Advanced Platform 5.x für 32-Bit x86-Systeme; Red Hat Enterprise Linux Advanced Platform 5.x oder 6.x für 64-Bit x64- oder IBM System z-Systeme; SuSE Linux Enterprise Server 10 oder 11 für 64-Bit x64- oder IBM System z-Systeme
- **Hardware.** Pentium®-Prozessor oder Prozessor der Pentium-Klasse oder höher (für 32-Bit-Windows); x64-Prozessorfamilie (AMD 64 und EM64T) (für 64-Bit-Windows) mit 1 GHz oder schneller
PowerPC-Prozessor, 233 MHz oder schneller und IBM System p für IBM AIX;
Itanium-Prozessor für HP-UX; UltraSPARC II (oder höher) für Solaris; Pentium-Prozessor oder Prozessor der Pentium-Klasse oder höher für 32-Bit-Linux; x64-Prozessorfamilie (AMD 64 und EM64T) oder IBM System z für 64-Bit-Linux
- **Mindestanforderung für freien Festplattenplatz.** 1 Gigabyte (GB) wird empfohlen. Unter HP-UX werden 2 GB empfohlen.

- **RAM.** 4 GB sind empfohlen.
- **Virtuelle Umgebung.** IBM® SPSS® Modeler Server wird von folgenden virtuellen Umgebungen unterstützt.
 - IBM PowerVM Hypervisor für IBM AIX
 - IBM PR/SM oder z/VM für Linux für IBM System z
 - VMWare ESX Server 4.1
 - VMWare vSphere 4.0
- **Cluster-Umgebung.** SPSS Modeler Server wird von folgenden Cluster-Umgebungen unterstützt.
 - IBM HACMP/RSCT für IBM AIX
 - IBM WebSphere App Server 64-Bit
 - Microsoft Cluster Service für Windows
 - Oracle Solaris Cluster
 - Oracle WebLogic App Server 64-Bit
 - Red Hat Cluster Suite für Red Hat Enterprise Linux

Installation unter Windows

Im folgenden Abschnitt wird das Verfahren für die Installation von SPSS Modeler Solution Publisher Runtime unter dem Betriebssystem Windows beschrieben.

- ▶ Legen Sie das IBM SPSS Modeler Solution Publisher-Installationsmedium in das Laufwerk ein.
- ▶ Wählen Sie im Installationsmenü die Option IBM® SPSS® Modeler Solution Publisher Runtime installieren oder IBM® SPSS® Modeler Solution Publisher Runtime für Win64 installieren aus, je nachdem, auf welchem System Sie installieren.
- ▶ Klicken Sie zum Starten im Installationsassistenten auf Weiter.
- ▶ Befolgen Sie die auf dem Bildschirm angezeigten Anweisungen. Klicken Sie zum Fortfahren auf Weiter.
- ▶ Sobald Sie alle Optionen angegeben haben, kann die Installation gestartet werden. Klicken Sie auf Installieren, um mit der Übertragung der Dateien zu beginnen.
- ▶ Wenn alle Dateien installiert wurden, klicken Sie auf Fertig stellen.

Installation unter UNIX

Im folgenden Abschnitt werden die zusätzlichen Systemanforderungen und die Prozedur für die Installation von IBM® SPSS® Modeler Solution Publisher Runtime unter dem Betriebssystem UNIX beschrieben.

Zusätzliche Anforderungen für UNIX

Sie müssen sicherstellen, dass Kernel-Limits im System für den Betrieb von IBM® SPSS® Modeler Server ausreichen. Die “ulimits” für Daten, Arbeitsspeicher und Dateien sind besonders wichtig und sollten in der SPSS Modeler Server-Umgebung auf unbegrenzt eingestellt werden. Gehen Sie dazu wie folgt vor:

- ▶ Fügen Sie *modelersrv.sh* die folgenden Befehle hinzu:

```
ulimit -d unlimited
```

```
ulimit -m unlimited
```

```
ulimit -f unlimited
```

- ▶ Starten Sie SPSS Modeler Server neu.

Solaris.

Für die Installation unter Solaris werden zusätzlich zu den zuvor beschriebenen Voraussetzungen die folgenden versionsspezifischen Patches benötigt. Patches können auf der Website <http://sunsolve.sun.com/pub-cgi/show.pl?target=patchpage> heruntergeladen werden.

- *Solaris 2.9 – Patch-IDs:* 111711-16, 111712-16 und 111722-05 (C++)
- *Solaris 2,10 – Patch-IDs:* 119963 (C++)

Die Bibliothek */usr/lib/iconv/646%UTF-8* muss ebenfalls auf Ihrem System installiert sein. Ist diese derzeit nicht installiert, installieren Sie die Pakete *SUNWiniu8* oder *SUNWiniu8x*, mit denen diese Bibliothek als Teil der Indic iconv-Module für UTF-8 installiert wird.

Linux 32-Bit.

Für einen reibungslosen Betrieb sind außerdem die folgenden Pakete erforderlich. Sie können diese Pakete vom Red Hat Network herunterladen. Möglicherweise müssen Sie sich registrieren, um Zugriff auf das Netzwerk zu erhalten (<https://rhn.redhat.com/network/software/search.pxt>). Verwenden Sie die Suchfunktion, um die einzelnen Paketversionen zu suchen und aufzurufen.

- Das *libstdc++*-Paket (Version: *libstdc++-3.2.3-X.Y.i386*, wobei *X.Y22.1* oder höher ist).
- Das *libgcc*-Paket (Version: *libgcc-3.2.3-X.Y.i386*, wobei *X.Y22.1* oder höher ist).
- Das *compat-libstdc++*-Paket (Version: *compat-libstdc++-33-3.2.3-X.Y.i386*, wobei *X.Y47.3* oder höher ist).

Linux 64-Bit.

Für einen reibungslosen Betrieb sind außerdem die folgenden Pakete erforderlich. Sie können diese Pakete vom Red Hat Network herunterladen. Möglicherweise müssen Sie sich registrieren, um Zugriff auf das Netzwerk zu erhalten (<https://rhn.redhat.com/network/software/search.pxt>). Verwenden Sie die Suchfunktion, um die einzelnen Paketversionen zu suchen und aufzurufen.

- Das *libstdc++*-Paket (Version: *libstdc++-3.2.3-X.Y.x86_64*, wobei *X.Y22.1* oder höher ist).
- Das *libgcc*-Paket (Version: *libgcc-3.2.3-X.Y.x86_64*, wobei *X.Y22.1* oder höher ist).
- Das *compat-libstdc++*-Paket (Version: *compat-libstdc++-33-3.2.3-X.Y.x86_64*, wobei *X.Y47.3* oder höher ist).

AIX.

Neben den zuvor genannten Systemanforderungen erfordert die Installation auf AIX zudem die folgenden, versionspezifischen Patches.

- XL C++ Runtime muss mindestens Level 8.0.0.9 (lspp -l xlc.rte) haben. Die Dateisets, die die Libraries enthalten, heißen IBM C++ Runtime Environment Components for AIX und können unter <http://www.ibm.com/software/awdtools/xlcpp/support/> heruntergeladen werden.
- XL FORTRAN Runtime muss mindestens Level 10.1.0.4 (lspp -l xlf.rte) haben. Die Dateisets, die die Libraries enthalten, heißen XL Fortran Enterprise Edition V10.1 for AIX Runtime Environment und können unter <http://www.ibm.com/software/awdtools/fortran/xlfortran/support/> heruntergeladen werden.

HP-UX.

Für eine Installation unter HP-UX müssen neben den zuvor genannten Systemvoraussetzungen außerdem die C++ Runtime-Bibliotheken installiert sein. Sie können diese als Patch PHSS_26946, wie in der folgenden Tabelle dargestellt, herunterladen. Zugriff auf diese Patches erhalten Sie unter <http://www4.itrc.hp.com/service/patch/mainPage.do>. Eine Registrierung ist erforderlich, jedoch kostenlos.

UNIX-Installationsprozedur

Die folgenden Installationsanweisungen gelten für IBM® SPSS® Modeler Solution Publisher Runtime für UNIX. Sie benötigen Lese- und Schreibberechtigungen für das Zielinstallationsverzeichnis. Melden Sie sich daher mit einem Konto an, das über ausreichende Berechtigungen verfügt.

- ▶ Legen Sie den -Installationsdatenträger in das Laufwerk ein und stellen Sie sicher, dass Sie auf das Laufwerk zugreifen können. Unter Solaris muss der Volume Manager (vold Daemon) laufen.
- ▶ Ändern Sie das Verzeichnis auf den DVD-Mountpoint.
- ▶ Wechseln Sie in das Verzeichnis *modelrun*.
- ▶ Ändern Sie das entsprechende Plattformverzeichnis.
- ▶ Führen Sie das *.bin*-Installationskript aus. Beispiel:


```
./modelersolutionpublisherpia64.bin -i console
```
- ▶ Es werden detaillierte Anweisungen angezeigt. Drücken Sie zum Fortfahren die Eingabetaste.
- ▶ Die Lizenzinformationen werden angezeigt. Lesen Sie die Lizenzvereinbarung, geben Sie 1 ein, um sie zu akzeptieren, und drücken Sie zum Fortfahren die Eingabetaste.
- ▶ Es wird eine Liste der verfügbaren Sprachen angezeigt. Geben Sie die Nummer für die zu installierende Sprachversion ein und drücken Sie zum Fortfahren die Eingabetaste.
- ▶ Sie werden aufgefordert, den Installationsort einzugeben. Um das Standardverzeichnis */usr/IBM/SPSS/ModelerSolutionPublisher<nn>* (wobei *<nn>* die Versionsnummer ist) zu verwenden, drücken Sie die Eingabetaste. Die Runtime wird im angegebenen Verzeichnis installiert.

- ▶ Sie werden aufgefordert, den Installationsort zu bestätigen. Wenn er korrekt ist, geben Sie y ein und drücken Sie die Eingabetaste.
- ▶ Es wird eine Vorinstallationsübersicht angezeigt, damit Sie Ihre bisherigen Eingaben bestätigen können. Drücken Sie zum Fortfahren die Eingabetaste.
- ▶ Es wird eine Meldung angezeigt, die besagt, dass die Installationsroutine nun ausgeführt werden kann. Drücken Sie zum Fortfahren die Eingabetaste.
- ▶ Während die Installationsroutine ausgeführt wird, wird eine Fortschrittsleiste angezeigt. Wenn die Installation abgeschlossen ist, drücken Sie die Eingabetaste, um das Installationsprogramm zu beenden.

Starten der Runtime

Nach der Installation der Runtime können Sie damit Streams ausführen, die mithilfe eines der Exportknoten aus IBM® SPSS® Modeler veröffentlicht wurden. [Für weitere Informationen siehe Thema Veröffentlichen von Streams in Kapitel 2 auf S. 7.](#) Starten Sie die Runtime über die Befehlszeile unter Verwendung von Optionen, die den Namen der veröffentlichten Bilddatei und eine optionale Parameterdatei für die Ausführung angeben.

Zur Ausführung der Runtime geben Sie an der Befehls-Eingabeaufforderung folgende Zeile ein:

```
modelerrun -p <Parameterdatei> -o <Optionen> <Bilddatei>
```

Dabei gilt:

<parameter-file>	ist der Dateiname der veröffentlichten Parameterdatei (optional).
<options>	ist eine kommasetrennte Liste von option=value-Paaren, die Ausführungsoptionen für die Laufzeit angeben.
<image file>	ist der Dateiname der auszuführenden veröffentlichten Bilddatei (*.pim).

Hinweis: Bei Verwendung von Multibyte-Sprachen, wie beispielsweise Chinesisch oder Japanisch, in einer UNIX-Umgebung müssen auch die zu verwendenden Sprachcodes angegeben werden. Das folgende Beispiel zeigt die Sprachcodes für vereinfachtes Chinesisch:

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

dabei gibt -p den Speicherort der pim- und par-Dateien an.

[Für weitere Informationen siehe Thema Ausführen veröffentlichter Streams in Kapitel 3 auf S. 10.](#)

Ändern des temporären Verzeichnisses

Für einige in IBM® SPSS® Modeler Solution Publisher Runtime durchgeführte Operationen müssen u. U. temporäre Dateien erstellt werden. Runtime verwendet standardmäßig das temporäre Systemverzeichnis zum Erstellen von temporären Dateien. Sie können das temporäre Verzeichnis wie folgt ändern:

Hinweis: Sie müssen unbedingt die richtige Syntax verwenden. Achten Sie insbesondere auf Leerzeichen und den Verzeichnispfad.

Angenommen, Sie verwenden SPSS Modeler Solution Publisher Runtime mit folgenden veröffentlichten Dateien: *webtest.pim* und *webtest.par*. Sie möchten den Speicherort für temporäre Verzeichnisse in *C:\published\temp* ändern. So können Sie den Speicherort des von der ausführbaren Datei verwendeten temporären Verzeichnisses ändern:

- ▶ Führen Sie ausgehend von dem Verzeichnis, in dem die Dateien *.pim* und *.par* gespeichert sind, folgenden Befehl aus:

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

Dadurch wird die Runtime angewiesen, "*C:\published\temp*" als temporäres Verzeichnis ausschließlich für diesen spezifischen Vorgang zu verwenden.

Deinstallation von IBM SPSS Modeler Solution Publisher Runtime

Im folgenden Abschnitt werden die Vorgänge für die Deinstallation von IBM® SPSS® Modeler Solution Publisher Runtime beschrieben.

Windows-Deinstallationsprozedur

- ▶ Öffnen Sie in der Windows-Systemsteuerung das Dialogfeld Software.
- ▶ Wählen Sie in der Liste den Eintrag IBM® SPSS® Modeler Solution Publisher <nn> aus.
- ▶ Klicken Sie auf Entfernen.
- ▶ Befolgen Sie die Anweisungen auf dem Bildschirm, um die Runtime zu entfernen.

Hinweis: Sie müssen Sie an Ihrem Computer mit Administratorberechtigungen angemeldet sein, um Programme entfernen zu können.

UNIX-Deinstallationsprozedur

- ▶ Löschen Sie das Installationsverzeichnis mit allen installierten Dateien.

IBM SPSS Modeler Solution Publisher Runtime Library – API-Referenz

Einführung

Mit der CLEMRTL-API können Sie die Ausführung veröffentlichter Streams über Ihre Anwendung steuern. In diesem Abschnitt werden die verfügbaren API-Funktionen beschrieben.

Folgende allgemeine Probleme treten bei der Verwendung der API auf:

- Die API besitzt eine C-Linkage für größtmögliche Kompatibilität, die Bibliothek weist jedoch C++-Abhängigkeiten auf. Dies kann bedeuten, dass sie auf einigen Plattformen nur mit einem C++-Linker verwendet werden kann.
- `clemrtl_initialize_ext()` muss aufgerufen werden, bevor andere Funktionen in der Bibliothek verwendet werden.
- Der Typ `clemrtl_image_handle` wird als Bild-ID verwendet.
- Jede Funktion gibt einen Indikatorcode aus. Die Werte der Statusindikatoren lauten:

CLEMRTL_OK	Erfolg
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar
CLEMRTL_ERROR	Fehlgeschlagen, weitere Informationen über den Fehler über die Funktion <code>clemrtl_getErrorDetail()</code> abrufbar

Überblick über API-Prozesse

Die allgemeine Gliederung einer Anwendung, die die API verwendet, lautet wie folgt:

- ▶ Initialisieren der Bibliothek mithilfe von `clemrtl_initialize_ext()`.
- ▶ Prüfen und Ändern der Ausführungsoptionen mithilfe von `clemrtl_getOption()` und `clemrtl_setOption()`.
- ▶ Öffnen eines Bildes mithilfe von `clemrtl_openImage()` und Abrufen eines Bild-Handle.
- ▶ Prüfen und Ändern der Bildparameter mithilfe von `clemrtl_enumerateParameters()`, `clemrtl_getParameter()` und `clemrtl_setParameter()`.
- ▶ Ausführen des Bildes mithilfe von `clemrtl_execute()`. Wenn dasselbe Bild mehrfach ohne Änderung der Parameter ausgeführt werden soll, verwenden Sie zunächst `clemrtl_prepare()`.
- ▶ Schließen des Bildes mithilfe von `clemrtl_closeImage()`.
- ▶ Zum Abbrechen einer laufenden Ausführung verwenden Sie `clemrtl_interrupt()`.

- ▶ Um Informationen zum letzten Fehler abzurufen, verwenden Sie `clemrtl_getErrorDetail()`. Um Fehlermeldungen und andere Diagnosemeldungen bei ihrem Eintreffen abzurufen, verwenden Sie `clemrtl_setReportHandler()`.

Die Bibliothek ist unter folgenden Bedingungen für eine Verwendung mit mehreren Threads geeignet:

- Globale Funktionen, also Funktionen, die kein Bild-Handle als Argument verwenden, können nicht gleichzeitig aufgerufen werden.
- Bildlokale Funktionen, also diejenigen, die ein Bild-Handle als Argument verwenden, können nicht gleichzeitig aufgerufen werden, wenn sie auf dasselbe Bild-Handle angewendet werden. Ausnahme: `clemrtl_interrupt()`.
- Bildlokale Funktionen können gleichzeitig aufgerufen werden, wenn sie auf verschiedenen Bild-Handles angewendet werden.

Dies bedeutet insbesondere, dass separate Threads Bilder gleichzeitig vorbereiten und ausführen können, vorausgesetzt, Sie verwendet verschiedene Bild-Handles. Verschiedene Bild-Handles können sich auf dasselbe Bild beziehen (`clemrtl_openImage()` erstellt bei jedem Aufruf ein new-Handle, auch für dasselbe Bild), vergessen Sie aber nicht, die Bildparameter bei der Ausführung mehrerer Instanzen desselben Bildes zu ändern, um die Ausgabe nach Bedarf zu ändern. Wenn zwei gleichzeitig ausgeführte Bilder versuchen, in dieselbe Ausgabedatei oder Datenbanktabelle zu schreiben, sind die Ergebnisse nicht vorhersagbar.

API-Funktionen

Im Folgenden finden Sie eine vollständige Liste der der API vorliegenden Funktionen.

initialise

Diese Funktion wurde verworfen und entspricht der Funktion

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

Neue Programme sollten *initialise_ext* aufrufen.

initialise_ext

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

Initialisiert IBM® SPSS® Modeler Solution Publisher Runtime. *Diese Funktion muss vor allen anderen API-Funktionen aufgerufen werden.*

Parameter	Beschreibung
Flags	Steuert bestimmte Aspekte des Initialisierungsprozesses. Der Wert wird als bitweises OR der unten beschriebenen Flags konstruiert.
arg_count	Die Anzahl der zusätzlichen Initialisierungsargumente. Dieser Wert muss kleiner oder Gleich der Länge des Arrays args sein.
args	Zusätzliche Initialisierungsargumente. Argumente werden durch den Typ beschrieben: <pre>typedef struct _clemrtl_init_arg { const char* name; const char* value; } clemrtl_init_arg;</pre>

Das Feld name gibt den Namen einer Konfigurationseigenschaft an und das Feld value ihren Wert. Folgende Konfigurationseigenschaften werden erkannt:

Name	Wert
Installationsverzeichnis	Der vollständige Pfad zum Installationsverzeichnis von IBM® SPSS® Modeler Solution Publisher. Der Installationsordner ist der Ordner, die die Ordner bin und config enthält.

Gibt einen der folgenden Fehlercodes aus:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.

Mit dem Parameter Flags können bestimmte Aspekte des Initialisierungsprozesses gesteuert werden. Der Wert 0 (oder CLEMRTL_INIT_DEFAULTS) gibt das Standardverhalten wie folgt an:

- Festlegen der Standardzeitzone durch Aufrufen von tzset().
- Setzen der Ländereinstellung LC_NUMERIC auf "C".
- Festlegen, dass der globale new-Handler einen Ausnahmefehler ausgibt, wenn der Arbeitsspeicher erschöpft ist.

Dieses Verhalten lässt sich durch Angabe von Kombinationen der folgenden Flags ändern:

Kennzeichen	Beschreibung
CLEMRTL_INIT_NO_TZ	Zeitzone wird nicht festgelegt.
CLEMRTL_INIT_NO_LOCALE	Ländereinstellung wird nicht geändert.
CLEMRTL_INIT_NO_NEW_HANDLER	Es wird kein new-Handler festgelegt.
CLEMRTL_INIT_LOCAL_NEW_HANDLER	Lokalisieren des new-Handlers innerhalb der einzelnen API-Aufrufe.

Für die Bibliothek ist erforderlich, dass die Ländereinstellung LC_NUMERIC auf "C" gesetzt ist, um eine ordnungsgemäße Konvertierung zwischen Zahlen und Zeichenketten durchzuführen (das IBM® SPSS® Modeler-Zahlenformat verwendet die Regeln der Ländereinstellung "C").

Die beinhaltet auch die Konvertierung von aus Textdateien gelesenen bzw. in Textdateien geschriebenen Zahlen.

- Geben Sie `NO_LOCALE` an, wenn Ihre Anwendung von der Ländereinstellung `LC_NUMERIC` abhängt und keine Konvertierung zwischen Zahlen und Zeichenketten erforderlich ist.
- `NO_NEW_HANDLER` und `LOCAL_NEW_HANDLER` schließen sich gegenseitig aus. Für die Bibliothek ist es erforderlich, dass `new` einen Ausnahmefehler ausgibt, wenn der Arbeitsspeicher erschöpft ist.
- Geben Sie `NO_NEW_HANDLER` an, wenn Ihre Anwendung einen eigenen `new`-Handler festlegt oder wenn Sie sicher sind, dass der Compiler/das Runtime-System standardmäßig eine Ausnahme ausgibt.

Geben Sie `LOCAL_NEW_HANDLER` nur an, wenn es für Ihre Anwendung erforderlich ist, dass `new` einen Nullzeiger ausgibt, wenn der Arbeitsspeicher erschöpft ist. *Anmerkung:* Dieser Vorgang ist für eine Umgebung mit mehreren Threads nicht sicher.

getOption

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

Ruft den Wert einer Ausführungsoption ab.

Parameter	Beschreibung
Name	Der Optionsname.
Wert	Ein Puffer für die Aufnahme des Optionswerts.
value_size	Die Größe des Wertepuffers.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Der Vorgang schlägt fehl, wenn der Optionsname nicht zu den oben aufgeführten Ausführungsoptionsnamen gehört – beispielsweise `memory_usage` – oder wenn der Wertepuffer nicht groß genug ist, um den Wert einschließlich eines endenden Nullzeichens aufzunehmen. Optionswerte sind immer Zeichenketten, auch wenn ihre Interpretation numerisch ist. [Für weitere Informationen siehe Thema Ausführen veröffentlichter Streams in Kapitel 3 auf S. 10.](#)

setOption

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

Legt den Wert einer Ausführungsoption fest.

Parameter	Beschreibung
Name	Der Optionsname.
Wert	Der Optionswert.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Der Vorgang schlägt fehl, wenn der Optionsname nicht zu den oben aufgeführten Ausführungsoptionsnamen gehört – beispielsweise `memory_usage`. Optionswerte sind immer Zeichenketten, auch wenn ihre Interpretation numerisch ist.

setLogFile

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

Leitet Protokollmeldungen von allen danach geöffneten Bildern um.

Parameter	Beschreibung
Modalwert	Der Protokollierungsmodus muss einen der folgenden Werte aufweisen: CLEMRTL_NULL_LOG – Meldungen unterdrücken. CLEMRTL_STDERROR_LOG – Meldungen an die Standardfehleraushgabe senden. CLEMRTL_FILE_LOG – Meldungen an die im Parameter <code>log_file</code> genannte Datei senden.
<code>log_file</code>	Zeichenkette, die den Dateinamen enthält.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Der Vorgang schlägt fehl, wenn es sich beim Modus nicht um einen der erkannten Modi handelt oder wenn eine Dateiprotokollierung angefordert wurde und der Dateiname nicht angegeben wurde oder ungültig ist.

openImage

```
int clemrtl_openImage(
    const char* image_file,
```

```
const char* param_file,
clemrtl_image_handle* handle);
```

Öffnet ein veröffentlichtes Stream-Bild.

Parameter	Beschreibung
image_file	Der Name der Bilddatei.
param_file	Der Name der Parameterdatei.
handle	Empfängt das Bild-Handle.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Die Parameterdatei kann NULL sein, wenn keine gesonderte Parameterdatei erforderlich ist. Der Vorgang schlägt fehl, wenn die Bilddatei nicht angegeben wurde, wenn der Dateiname ungültig oder der Inhalt nicht lesbar ist. Wenn der Vorgang erfolgreich ist, kann das zurückgegebene Bild-Handle zur Identifikation der Bildinstanz in nachfolgenden API-Aufrufen verwendet werden.

closelImage

```
int clemrtl_closelImage(
clemrtl_image_handle handle);
```

Schließt ein Bild-Handle und gibt seine Ressourcen frei.

Parameter	Beschreibung
handle	Das Bild-Handle.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Der Vorgang schlägt fehl, wenn ein anderer API-Aufruf für das Bild läuft, insbesondere, wenn das Bild noch immer ausgeführt wird. Wenn der Vorgang erfolgreich ist, wird die Bilddatei geschlossen und das Handle kann nicht in einem nachfolgenden API-Aufruf werden, außer bei einem weiteren Aufruf von `clemrtl_closelImage()`, was jedoch ohne Wirkung bleibt.

enumerateParameters

```
int clemrtl_enumerateParameters(
clemrtl_image_handle handle,
clemrtl_parameter_proc proc,
```

```
void* data);
```

Wendet auf jeden Bildparameternamen und -wert eine Callback-Prozedur an.

Parameter	Beschreibung
handle	Das Bild-Handle.
proc	Die Parameter-Callback-Prozedur.
Daten	Benutzerdefinierte Daten für das Callback.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Die Callback-Prozedur weist folgenden Typ auf:

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

Parameter	Beschreibung
Daten	An clemrtl_enumerateParameters() weitergegebene benutzerdefinierte Daten.
Name	Der Name des Parameters.
Wert	Der Wert des Parameters.

Die Prozedur wird in beliebiger Reihenfolge auf jeden Parameter genau einmal angewendet.

getParameter

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

Ruft den Wert eines Bildparameters ab.

Parameter	Beschreibung
handle	Das Bild-Handle.
Name	Der Name des Parameters.
Wert	Ein Puffer für die Aufnahme des Parameterwerts.
value_size	Größe des Wertepuffers.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Der Vorgang schlägt fehl, wenn der Parameternamen nicht mit dem Namen eines Parameters im Bild übereinstimmt oder wenn der Wertepuffer nicht groß genug ist, um den Wert einschließlich eines endenden Nullzeichens aufzunehmen. Parameternamen verwenden das oben angegebene `name.attribute`-Format – beispielsweise `file0.name` – und Parameterwerte sind immer Zeichenketten, auch wenn ihre Interpretation numerisch ist.

setParameter

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

Legt den Wert eines Bildparameters fest.

Parameter	Beschreibung
handle	Das Bild-Handle.
Name	Der Name des Parameters.
Wert	Der Wert des Parameters.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Der Vorgang schlägt fehl, wenn das Bild-Handle ungültig ist oder wenn der Parameternamen nicht mit dem Namen eines Parameters im Bild übereinstimmt. Parameternamen verwenden das oben angegebene `name.attribute`-Format – beispielsweise `file0.name` – und Parameterwerte sind immer Zeichenketten, auch wenn ihre Interpretation numerisch ist.

getFieldCount

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

Ergibt die Anzahl der Felder in einer Eingabequelle oder einem Ausgabeziel.

Parameter	Beschreibung
handle	Das Bild-Handle.
Schlüssel	Der Name der zu untersuchenden Ein- bzw. Ausgabe laut Parameterdatei. Der Schlüssel (key) kann sich auf eine Datei oder eine Datenbank beziehen.
field_count	Ruft die Anzahl an Feldern ab.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

getFieldTypes

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

Ergibt die Feldtypen für eine Eingabequelle oder ein Ausgabeziel.

Parameter	Beschreibung
handle	Das Bild-Handle.
Schlüssel	Der Name der zu untersuchenden Ein- bzw. Ausgabe laut Parameterdatei. Der Schlüssel (key) kann sich auf eine Datei oder eine Datenbank beziehen.
field_count	Die Anzahl der zu untersuchenden Felder. Dieser Wert muss kleiner oder Gleich der Länge des Arrays field_types sein.
field_types	Ein Array mit mindestens der Länge field_count, das die Feldtypen aufnimmt. Die Typen der ersten field_count-Felder in der Ein- bzw. Ausgabe werden in das Array kopiert. Wenn field_count größer ist als die tatsächliche Anzahl von Feldern bleiben die zusätzlichen Elemente im Array undefiniert. Die Werte für die Feldtypen sind unten in der Tabelle "Datentypen" angegeben.

Datentypen

field_types muss einen der folgenden Werte aufweisen:

Typ	Interpretation	Typischer 'C'-Deklarator
STRING	Mit Null abgeschlossene UTF-8-Zeichenkette.	const char*
INTEGER	Ganze Zahl mit Vorzeichen (32 Bit).	int
LONG	Ganze Zahl mit Vorzeichen (64 Bit).	long long

Typ	Interpretation	Typischer 'C'-Deklarator
REAL	64-Bit-Gleitkommazahl.	double
ZEIT	Ganze Zahl mit Vorzeichen (64 Bit), die die Sekunden seit 0.00 Uhr angibt.	long long
DATE	Ganze Zahl mit Vorzeichen (64 Bit), die die Sekunden seit dem 01.01.1970, 0.00 Uhr angibt.	long long
TIMESTAMP	Ganze Zahl mit Vorzeichen (64 Bit), die die Sekunden seit dem 01.01.1970, 0.00 Uhr angibt.	long long

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

setAlternativeInput

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

Ersetzt eine Dateieingabequelle durch eine alternative Eingabequelle.

Parameter	Beschreibung
handle	Das Bild-Handle.
Schlüssel	Der Name ursprünglichen Eingabequelle laut Parameterdatei. Bei der Eingabequelle muss es sich um eine Datei handeln. Daher nimmt der Schlüssel (key) immer folgendes Format an " <i>fileN</i> " für eine ganze Zahl $N \geq 0$.
field_count	Die Anzahl der Felder in der Eingabe. Der Wert muss genau mit der Anzahl der Felder in der ursprünglichen Dateieingabe übereinstimmen. Anderenfalls schlägt der Aufruf fehl.
field_types	Ein Array von Typen mit mindestens der Länge field_count. Der Wert field_types[i] ist der Typ des i-ten Felds und muss einen der unten in der Tabelle "Datentypen" aufgeführten Werte aufweisen. Die Typen müssen mit den Typen der ursprünglichen Dateieingabe kompatibel sein. Anderenfalls schlägt der Aufruf fehl.

Parameter	Beschreibung
iterator	<p>Eine Funktion, die die alternativen Eingabedaten erstellt. Die Funktion wird wie folgt auf das Argument angewendet:</p> <pre>void** row = iterator(arg);</pre> <p>Die Funktion wird während der Ausführung aufgerufen (innerhalb eines <code>clemrtl_execute</code>-Aufrufs) und wird für jeden Eingabedatensatz einmal aufgerufen. Der Ausgabewert "NULL" zeigt das Ende der Eingabe an. In diesem Fall wird die Funktion nicht erneut aufgerufen und die Ausführung endet schließlich. Anderenfalls ist das Ergebnis ein Datenarray mit mindestens der Länge <code>field_count</code>, wobei <code>row[i]</code> den Wert des <i>i</i>-ten Felds angibt. Ein Wert kann "NULL" sein. Anderenfalls muss er als Zeiger auf ein Datum verweisen, dessen Typ durch die entsprechenden Werte von <code>field_types[i]</code> festgelegt wird. Die Zeiger müssen bis zum nächsten Aufruf des Iterators gültig sein, oder alternativ bis zum Ende der Ausführung, wenn diese vorzeitig endet.</p>
arg	Ein opakes Argument, das bei jedem Aufruf an den Iterator weitergeleitet wird.

Datentypen

`field_types` muss ein Element aus einer begrenzten Liste von Typen sein. [Für weitere Informationen siehe Thema getFieldTypes auf S. 29.](#)

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Die Operation schlägt fehl, wenn die Feldtypen nicht mit denen der ursprünglichen Dateieingabe kompatibel sind, oder wenn die Feldanzahl nicht exakt mit der Anzahl der Felder in der ursprünglichen Eingabedatei übereinstimmt.

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);
```

Ersetzt ein Dateiausgabeziel durch ein alternatives Ausgabeziel.

Parameter	Beschreibung
handle	Das Bild-Handle.
Schlüssel	Der Name des ursprünglichen Ausgabeziels laut Parameterdatei. Beim Ausgabeziel muss es sich um eine Datei handeln. Daher nimmt der Schlüssel (key) immer folgendes Format an " <i>fileN</i> " für eine ganze Zahl $N \geq 0$.
field_count	Die Anzahl der Felder in der Ausgabe. Der Wert muss genau mit der Anzahl der Felder in der ursprünglichen Dateiausgabe übereinstimmen. Anderenfalls schlägt der Aufruf fehl.
field_types	Ein Array von Typen mit mindestens der Länge field_count. Der Wert field_types[i] ist der Typ des i-ten Felds und muss einen der unten in der Tabelle "Datentypen" aufgeführten Werte aufweisen. Die Typen müssen mit den Typen der ursprünglichen Dateiausgabe kompatibel sein. Anderenfalls schlägt der Aufruf fehl.
iterator	<p>Eine Funktion, die die Bildausgabe verwertet. Die Funktion wird wie folgt auf das Argument und eine Datenzeile angewendet:</p> <pre>iterator(arg, row);</pre> <p>Die Funktion wird während der Ausführung aufgerufen (innerhalb eines clemrtl_execute-Aufrufs) und wird für jede Ergebniszeile einmal aufgerufen. Der Wert "NULL" für row (Zeile) zeigt das Ende der Ausgabe an. Danach wird die Funktion nicht mehr aufgerufen. Eine Anwendung sollte sich nicht von diesem letzten Aufruf abhängig sein und nach dem Ende der Ausführung alle externen Ressourcen leeren und schließen. Anderenfalls ist row ein Datenarray mit mindestens der Länge field_count, wobei row[i] den Wert des i-ten Felds angibt. Ein Wert kann "NULL" sein. Anderenfalls muss er als Zeiger auf ein Datum verweisen, dessen Typ durch die entsprechenden Werte von field_types[i] festgelegt wird. Die Funktion muss alle Datenwerte, die sie benötigt, kopieren, da der Arbeitsspeicher nach Rückgabe des Aufrufs eventuell nicht beibehalten wird.</p>
arg	Ein opakes Argument, das bei jedem Aufruf an den Iterator weitergeleitet wird.

Datentypen

field_types muss ein Element aus einer begrenzten Liste von Typen sein. [Für weitere Informationen siehe Thema getFieldTypes auf S. 29.](#)

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_FAIL	Fehlgeschlagen, keine weiteren Details verfügbar.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Die Operation schlägt fehl, wenn die Feldtypen nicht mit denen der ursprünglichen Dateiausgabe kompatibel sind, oder wenn die Feldanzahl nicht exakt mit der Anzahl der Felder in der ursprünglichen Ausgabedatei übereinstimmt.

execute

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

Führt ein Bild aus.

Parameter	Beschreibung
handle	Das Bild-Handle.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Wenn das Bild nicht vorbereitet wurde, wird es zuerst mithilfe der aktuellen Parameterwerte vorbereitet. Der Vorgang schlägt fehl, wenn das Bild-Handle ungültig ist oder wenn während der Vorbereitung oder Ausführung ein Fehler auftritt. Der Aufruf kommt nicht wieder, bis die Ausführung abgeschlossen ist.

prepare

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

Bereitet ein Bild für die Ausführung vor.

Parameter	Beschreibung
handle	Das Bild-Handle.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Der Vorgang schlägt fehl, wenn das Bild-Handle ungültig ist oder wenn während der Vorbereitung ein Fehler auftritt.

Bilder müssen vor der Ausführung vorbereitet werden. Bei der Vorbereitung eines Bildes werden Parameterwerte im Bild fixiert. `clemrtl_execute()` bereitet bei jedem Aufruf automatisch ein Bild vor, wenn es noch nicht bereits vorbereitet wurde. Die Verwendung von `clemrtl_prepare()` zur Vorbereitung eines Bildes ist sinnvoll, wenn das Bild mehrmals mit denselben Parameterwerten ausgeführt werden soll, und kann zu einer erheblichen Leistungssteigerung führen. Nach der Vorbereitung eines Bildes werden nachfolgende Änderungen an Parameterwerten ignoriert;

verwenden Sie wieder `clemrtl_prepare()`, um das Bild mit den neuen Parameterwerten zu aktualisieren.

interrupt

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

Beendet die für ein Bild laufende Ausführung.

Parameter	Beschreibung
handle	Das Bild-Handle.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Der Vorgang bleibt ohne Wirkung, wenn das Bild-Handle ungültig ist oder das Bild nicht ausgeführt wird.

Diese Funktion kann problemlos mit einem weiteren API-Aufruf für dasselbe Image-Handle aufgerufen werden.

getErrorDetail

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

Ruft detaillierte Informationen zu dem letzten Fehler ab, der bei einem Bild auftrat.

Parameter	Beschreibung
handle	Das Bild-Handle.
severity	Ruft den Code für den Schweregrad als einzelnes Zeichen ab: I — Information W — Warnung E — Fehler X — Systemfehler
code	Empfängt die Fehlernummer.
Text	Ein Puffer für die Aufnahme des Meldungstextes.
text_size	Größe des Textpuffers.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Die Ergebnisse sind nicht verlässlich, wenn ein Bild-Handle ungültig ist. Der Meldungstext wird, falls erforderlich gekürzt, damit er in den Textpuffer.

Wenn ein API-Aufruf mit CLEMRTL_ERROR fehlschlägt und der Fehler nicht mit einem bestimmten Bild-Handle in Bezug steht, werden bei Weitergabe von 0 als Bild-Handle Einzelheiten des letzten nicht bildspezifischen Fehlers abgerufen, doch das Ergebnis ist in einer Umgebung mit mehreren Threads nicht verlässlich.

setReportHandler

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

Installiert eine Berichts-Callback-Prozedur für ein Bild.

Parameter	Beschreibung
handle	Das Bild-Handle.
proc	Die Berichts-Callback-Prozedur.
Daten	Benutzerdefinierte Daten für das Callback.

Ergibt einen der folgenden Statuscodes:

Ergebnis	Beschreibung
CLEMRTL_OK	Erfolg.
CLEMRTL_ERROR	Fehlgeschlagen, weitere Details verfügbar.

Die Callback-Prozedur weist folgenden Typ auf:

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

Parameter	Beschreibung
Daten	An clemrtl_setReportHandler() weitergegebene benutzerdefinierte Daten.
severity	Der Code für den Schweregrad als einzelnes Zeichen: I — Information W — Warnung E — Fehler X — Systemfehler

Parameter	Beschreibung
code	Meldungsnummer.
Text	Meldungstext.

Die Prozedur wird bei jedem Eintreffen auf die einzelnen Meldungen angewendet. Wenn die Prozedur auf NULL gesetzt wird, werden alle bestehenden Handler entfernt, die auf dem Bild installiert sind.

Hinweise

This information was developed for products and services offered worldwide.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785, U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing, Legal and Intellectual Property Law, IBM Japan Ltd., 1623-14, Shimotsuruma, Yamato-shi, Kanagawa 242-8502 Japan.

Der folgende Absatz gilt nicht für Großbritannien oder andere Länder, in denen derartige Bestimmungen nicht mit dem dort geltenden Recht vereinbar sind. SPSS INC., AN IBM COMPANY, ÜBERNIMMT FÜR DIE VORLIEGENDE DOKUMENTATION KEINERLEI GEWÄHRLEISTUNG IRGENDWELCHER ART, WEDER AUSDRÜCKLICH NOCH STILLSCHWEIGEND, EINSCHLIESSLICH (JEDOCH NICHT DARAUF BEGRENZT) DER STILLSCHWEIGENDEN GEWÄHRLEISTUNGEN IN BEZUG AUF DIE NICHTVERLETZUNG VON RECHTEN DRITTER, AUF HANDELSÜBLICHKEIT ODER DIE EIGNUNG FÜR EINEN BESTIMMTEN ZWECK. Einige Staaten lassen bei bestimmten Transaktionen keine Ausschlussklauseln ausdrücklicher oder stillschweigender Gewährleistungen zu, sodass diese Erklärung möglicherweise nicht auf Sie zutrifft.

Diese Informationen können technische Ungenauigkeiten oder typografische Fehler enthalten. An den hierin enthaltenen Informationen werden in regelmäßigen Abständen Änderungen vorgenommen, die in spätere Ausgaben der Publikation eingearbeitet werden. SPSS Inc. kann jederzeit ohne Vorankündigung Verbesserungen und/oder Veränderungen an den in dieser Publikation beschriebenen Produkten und/oder Programmen vornehmen.

Alle in diesen Ausführungen enthaltenen Verweise auf Websites, die nicht zu SPSS bzw. IBM gehören, dienen lediglich der Information. Die Nennung bedeutet nicht, dass SPSS bzw. IBM den Inhalt dieser Websites unterstützen. Das Material auf diesen Websites ist kein Bestandteil des Materials für dieses SPSS Inc.-Produkt. Sie verwenden diese Websites auf eigenes Risiko.

Wenn Sie Informationen an IBM bzw. SPSS senden, räumen Sie IBM und SPSS das nicht ausschließliche Recht ein, die Informationen in jeglicher Form zu verwenden bzw. weiterzugeben, die dem Unternehmen geeignet erscheint, ohne dass ihm daraus Verbindlichkeiten Ihnen gegenüber entstehen.

Informationen zu Nicht-SPSS-Produkten stammen von den Herstellern dieser Produkte, ihren veröffentlichten Verlautbarungen oder aus anderen öffentlich verfügbaren Quellen. SPSS hat diese Produkte nicht getestet und kann daher die Richtigkeit der Angaben zu Leistung und Kompatibilität oder anderer Behauptungen in Bezug auf Nicht-SPSS-Produkte nicht bestätigen. Fragen zu den Fähigkeiten von Nicht-SPSS-Produkten sind an die Hersteller dieser Produkte zu richten.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Software Group, Attention: Licensing, 233 S. Wacker Dr., Chicago, IL 60606, USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Diese Informationen enthalten Beispiele für Daten und Berichte, die in alltäglichen Betriebsabläufen verwendet werden. Um sie möglichst umfassend darzulegen, enthalten die Beispiele Namen von Einzelpersonen, Unternehmen, Marken und Produkten. Alle diese Namen sind frei erfunden und jegliche Ähnlichkeit mit Namen und Adressen, die von einem tatsächlichen Handelsunternehmen verwendet werden, ist rein zufällig.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Marken

IBM, das IBM-Logo und ibm.com sind Marken von IBM Corporation, die in vielen Ländern weltweit eingetragen sind. Eine aktuelle Liste der IBM-Marken finden Sie im Internet unter <http://www.ibm.com/legal/copytrade.shtml>.

SPSS ist eine Marke von SPSS Inc., an IBM Company, die in vielen Ländern weltweit eingetragen sind.

Adobe, das Adobe-Logo, PostScript und das PostScript-Logo sind entweder registrierte Marken oder Marken von Adobe Systems Incorporated in den USA und/oder anderen Ländern.

IT Infrastructure Library ist eine eingetragene Marke der Central Computer and Telecommunications Agency, die nun zum Office of Government Commerce gehört.

Intel, das Intel-Logo, Intel Inside, das Intel Inside-Logo, Intel Centrino, das Intel Centrino-Logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium und Pentium sind Marken oder eingetragene Marken der Intel Corporation oder ihrer Tochtergesellschaften in den USA und anderen Ländern.

Linux ist eine eingetragene Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken von Microsoft Corporation in den USA und/oder anderen Ländern.

ITIL ist eine eingetragene Marke und eine eingetragene Gemeinschaftsmarke des Office of Government Commerce und ist beim U.S. Patent and Trademark Office eingetragen.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

Cell Broadband Engine ist eine Marke von Sony Computer Entertainment, Inc. in den USA und/oder anderen Ländern und wird im Rahmen einer Lizenz dieses Unternehmens verwendet.

Java und alle Java-basierten Marken und Logos sind Marken von Sun Microsystems, Inc. in den USA und/oder anderen Ländern.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Weitere Produkt- oder Servicenamen können Marken von IBM, SPSS oder anderen Unternehmen sein.

Index

- Anwendungsbeispiele, 3
- API
 - Alternative Ausgabe, 31
 - Alternative Eingabe, 30
 - Ausführungsoptionen, 24
 - Fehlercodes, 21
 - Feldanzahl, 28
 - Feldtypen, 29
 - Referenz, 21
 - setReportHandler, 35
 - Statuscodes, 21
 - Übersicht, 13, 21
- API-Funktionen
 - closeImage, 21, 26
 - enumerateParameters, 21, 26
 - execute, 21, 33
 - getErrorDetail, 21, 34
 - getFieldCount, 21, 28
 - getFieldTypes, 21, 29
 - getOption, 21, 24
 - getParameter, 21, 27
 - Initialisieren, 22
 - Initialisierung, 21
 - interrupt, 21, 34
 - openImage, 21, 25
 - prepare, 21, 33
 - setAlternativeInput, 21, 30
 - setAlternativeOutput, 21, 31
 - setLogFile, 21, 25
 - setOption, 21, 24
 - setParameter, 21, 28
 - setReportHandler, 21
- Ausführen von Streams
 - Verwendung von IBM SPSS Modeler Solution Publisher, 10–11
- Ausgabeknoten, 6
- Beispiele
 - Anwendungshandbuch, 3
 - Übersicht, 4
- Bereitstellung, 6
- closeImage
 - API-Funktionsreferenz, 26
 - Überblick über API-Prozesse, 21
- Deinstallieren
 - IBM SPSS Modeler Solution Publisher Runtime, 20
 - Dokumentation, 3
- enumerateParameters
 - API-Funktionsreferenz, 26
 - Überblick über API-Prozesse, 21
- execute
 - API-Funktionsreferenz, 33
- Überblick über API-Prozesse, 21
- Fehlerberichte
 - API-Funktionsreferenz, 25, 34–35
 - Fehlercodes, 21
- getErrorDetail
 - API-Funktionsreferenz, 34
 - Überblick über API-Prozesse, 21
- getFieldCount
 - API-Funktionsreferenz, 28
 - Überblick über API-Prozesse, 21
- getFieldTypes
 - API-Funktionsreferenz, 29
 - Überblick über API-Prozesse, 21
- getOption
 - API-Funktionsreferenz, 24
 - Überblick über API-Prozesse, 21
- getParameter
 - API-Funktionsreferenz, 27
 - Überblick über API-Prozesse, 21
- IBM SPSS Modeler, 1
 - Dokumentation, 3
- IBM SPSS Modeler Solution Publisher, 6–7
 - Einbettung in Anwendungen, 13
 - Übersicht, 6
- IBM SPSS Modeler Solution Publisher Runtime
 - Deinstallieren, 20
 - Installieren, 15
 - starten, 19
 - Systemanforderungen, 15–16
 - Temp-Verzeichnis, 19
- IBM SPSS Text Analytics, 2
- initialise
 - API-Funktionsreferenz, 22
 - Flags, 22
- initialise_ext
 - API-Funktionsreferenz, 22
 - Flags, 22
- initialize
 - Überblick über API-Prozesse, 21
- interrupt
 - API-Funktionsreferenz, 34
 - Überblick über API-Prozesse, 21
- Ländereinstellung
 - Festlegen unter Verwendung der API, 22
- Marken, 39
- öffnen
 - Streams, 25

- openImage
 - API-Funktionsreferenz, 25
 - Überblick über API-Prozesse, 21
- .par-Datei, 7
- Parameter
 - API-Bildparameter, 26–28
 - API-Callback-Prozeduren, 26
 - API-Parameterdateien, 25–26
 - Parameter für Stream-Ausführung, 11
- .pim-Datei, 7
- prepare
 - API-Funktionsreferenz, 33
 - Überblick über API-Prozesse, 21
- Protokolldateien, 25

- Rechtliche Hinweise, 37
- Runtime
 - Deinstallieren, 20
 - Optionen, 10
 - starten, 19
 - Temp-Verzeichnis, 19
 - UNIX-Installation, 16
 - Windows-Installation, 16
- Runtime-Programmierbibliothek (CLEMRTL), 13

- Schließen
 - Streams, 26
- setAlternativeInput
 - API-Funktionsreferenz, 30
 - Überblick über API-Prozesse, 21
- setAlternativeOutput
 - API-Funktionsreferenz, 31
 - Überblick über API-Prozesse, 21
- setLogFile
 - API-Funktionsreferenz, 25
 - Überblick über API-Prozesse, 21
- setOption
 - API-Funktionsreferenz, 24
 - Überblick über API-Prozesse, 21
- setParameter
 - API-Funktionsreferenz, 28
 - Überblick über API-Prozesse, 21
- setReportHandler
 - API-Funktionsreferenz, 35
 - Überblick über API-Prozesse, 21
- SPSS Modeler Server, 1
- Statuscodes, 21

- UNIX
 - IBM SPSS Modeler Solution Publisher Runtime installieren, 16

- Veröffentlichen von Streams
 - IBM SPSS Modeler Solution Publisher, 6–7
- Verzeichnis für temporäre Dateien
 - IBM SPSS Modeler Solution Publisher Runtime, 19
- Windows
 - IBM SPSS Modeler Solution Publisher Runtime installieren, 15
- Zeitzone
 - Festlegen unter Verwendung der API, 22