

IBM SPSS Modeler 14.2 Solution
Publisher



Nota: Prima di utilizzare queste informazioni e il relativo prodotto, leggere le informazioni generali disponibili in Note a pag. .

Il presente documento contiene informazioni di proprietà di SPSS Inc, una società IBM. Viene fornito su contratto di licenza ed è protetto dalle leggi sul copyright. Le informazioni contenute nella presente pubblicazione non includono alcuna garanzia di prodotto e qualsiasi dichiarazione inclusa nel presente manuale non deve essere interpretata come tale.

Al momento dell'invio di informazioni a IBM o SPSS, l'utente concede a IBM e SPSS un diritto non esclusivo a utilizzare o distribuire tali informazioni nel modo che ritiene appropriato senza incorrere in alcun obbligo nei confronti dell'utente.

© Copyright IBM Corporation 1994, 2011..

Prefazione

IBM® SPSS® Modeler è l'efficace workbench di data mining aziendale di IBM Corp.. SPSS Modeler consente alle organizzazioni di migliorare le relazioni con i clienti e con il pubblico grazie a un'analisi approfondita dei dati. Le organizzazioni potranno utilizzare le informazioni ottenute tramite SPSS Modeler per mantenere i clienti di valore, cogliere opportunità di vendite incrociate, attrarre nuovi clienti, individuare frodi, diminuire i rischi e migliorare l'offerta di servizi a livello statale.

L'interfaccia visiva di SPSS Modeler favorisce l'applicazione di una competenza aziendale specifica da parte degli utenti, grazie alla quale sarà possibile ottenere modelli di previsione più efficaci e una riduzione nei tempi di sviluppo delle soluzioni. SPSS Modeler offre una vasta gamma di tecniche di creazione di modelli, quali previsione, classificazione, segmentazione e algoritmi per l'individuazione delle associazioni. IBM® SPSS® Modeler Solution Publisher consente quindi di distribuire a livello aziendale i modelli creati in modo che vengano utilizzati dai responsabili dei processi decisionali oppure inseriti in un database.

Informazioni su IBM Business Analytics

Il software IBM Business Analytics fornisce informazioni complete, coerenti e accurate a cui i responsabili delle decisioni possono affidarsi per ottimizzare le prestazioni dell'azienda. Un ampio portafoglio di applicazioni di [business intelligence](#), [analisi predittiva](#), [gestione delle prestazioni e delle strategie finanziarie](#) e [analisi](#) offre una panoramica chiara, istantanea e interattiva delle prestazioni attuali e la possibilità di prevedere i risultati futuri. Utilizzato in combinazione con potenti soluzioni di settore, prassi consolidate e servizi professionali, questo software consente alle aziende di tutte le dimensioni di ottimizzare la produttività, automatizzare le decisioni senza problemi e fornire risultati migliori.

Come parte di questo portafoglio, il software IBM SPSS Predictive Analytics consente alle aziende di prevedere gli eventi futuri e di agire tempestivamente in modo da migliorare i risultati delle attività aziendali. Le aziende, gli enti governativi e le università di tutto il mondo si affidano alla tecnologia IBM SPSS perché rappresenta un vantaggio concorrenziale in termini di attrazione, retention e aumento dei clienti, riducendo al tempo stesso le frodi e limitando i rischi. Incorporando il software IBM SPSS nelle attività quotidiane, le aziende diventano imprese in grado di effettuare previsioni e di gestire e automatizzare le decisioni, per raggiungere gli obiettivi aziendali e vantaggi tangibili sulla concorrenza. Per ulteriori informazioni o per contattare un rappresentante, visitare il sito <http://www.ibm.com/spss>.

Supporto tecnico

Il supporto tecnico è a disposizione dei clienti che dispongono di un contratto di manutenzione. I clienti possono contattare il supporto tecnico per richiedere assistenza per l'utilizzo di IBM Corp. o per l'installazione di uno degli ambienti hardware supportati. Per contattare il supporto tecnico, visitare il sito Web IBM Corp. all'indirizzo <http://www.ibm.com/support>. Ricordare che durante la richiesta di assistenza sarà necessario fornire i dati di identificazione personali, i dati relativi alla propria società e il numero del contratto di manutenzione.

Contenuto

1	Informazioni su IBM SPSS Modeler	1
	IBM SPSS Modeler Server	1
	Opzioni di IBM SPSS Modeler	2
	IBM SPSS Text Analytics	2
	Documentazione di IBM SPSS Modeler	2
	Esempi di applicazioni	4
	Cartella Demos	5
2	IBM SPSS Modeler Solution Publisher	6
	Funzionamento di IBM SPSS Modeler Solution Publisher	6
	Pubblicazione di stream	7
3	Esecuzione di stream pubblicati	10
	Utilizzo del file di parametri per personalizzare l'esecuzione degli stream	11
	Integrazione di IBM SPSS Modeler Solution Publisher con altre applicazioni	12
	Applicazione di esempio IBM SPSS Modeler Solution Publisher	13
4	Installazione di IBM SPSS Modeler Solution Publisher Runtime	15
	Installazione del Runtime	15
	Requisiti di sistema	15
	Installazione in ambiente Windows	16
	Installazione in ambiente UNIX	16
	Avvio del Runtime	19
	Modifica della directory Temp	19
	Disinstallazione di IBM SPSS Modeler Solution Publisher Runtime	20
	Procedura di disinstallazione su computer Windows	20
	Procedura di disinstallazione su computer UNIX	20

**5 Riferimenti all'API della libreria di IBM SPSS Modeler Solution
Publisher Runtime 21**

Introduzione	21
Panoramica del processo API	21
Funzioni API	22

Appendice

A Note	37
---------------	-----------

Indice	40
---------------	-----------

Informazioni su IBM SPSS Modeler

IBM® SPSS® Modeler è un insieme di strumenti di data mining che consente di sviluppare rapidamente modelli predittivi con l'ausilio di competenze aziendali e di eseguirne il deployment nelle operazioni aziendali per migliorare i processi decisionali. Progettato secondo il modello CRISP-DM conforme agli standard di settore, SPSS Modeler supporta l'intero processo di data mining, dai dati a risultati aziendali migliori.

SPSS Modeler offre numerosi metodi di modellazione ricavati dall'apprendimento automatico, dall'intelligenza artificiale e dalla statistica. I metodi disponibili nella palette Modelli consentono di ricavare nuove informazioni dai dati e di sviluppare modelli predittivi. Ogni metodo ha determinati punti di forza e si presta meglio per particolari tipi di problemi.

SPSS Modeler può essere acquistato come prodotto autonomo oppure utilizzato in combinazione con SPSS Modeler Server. È inoltre disponibile una serie di opzioni, come illustrato nelle sezioni seguenti. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/modeler/>.

IBM SPSS Modeler Server

SPSS Modeler utilizza un'architettura client/server per distribuire le richieste di operazioni che utilizzano molte risorse a potenti componenti software server, con un conseguente miglioramento della performance su insiemi di dati di grandi dimensioni. Oltre a quelli elencati di seguito, potrebbero essere disponibili altri prodotti e aggiornamenti. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/modeler/>.

SPSS Modeler. SPSS Modeler è una versione del prodotto completa nelle funzioni, che viene installata ed eseguita sul desktop dell'utente. È possibile eseguirla in modalità locale come prodotto autonomo oppure in modalità distribuita assieme a IBM® SPSS® Modeler Server per ottenere una migliore performance su insiemi di dati di grandi dimensioni.

SPSS Modeler Server. SPSS Modeler Server viene eseguito continuamente in modalità di analisi distribuita assieme a una o più installazioni di IBM® SPSS® Modeler. Una configurazione di questo tipo consente di ottenere performance superiori quando si lavora su insiemi di dati di grandi dimensioni, in quanto le operazioni che richiedono un utilizzo consistente della memoria possono essere eseguite sul server senza necessità di scaricare i dati sul computer client. SPSS Modeler Server supporta inoltre funzionalità di ottimizzazione SQL e modellazione in-database, con ulteriori vantaggi in termini di prestazioni e grado di automazione. Per eseguire un'analisi deve essere disponibile almeno un'installazione SPSS Modeler.

Opzioni di IBM SPSS Modeler

I seguenti componenti e funzionalità possono essere acquistati separatamente e concessi in licenza per l'utilizzo con SPSS Modeler. Tenere presente che potrebbero rendersi disponibili anche prodotti e aggiornamenti aggiuntivi. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/modeler/>.

- Accesso a SPSS Modeler Server per una maggiore scalabilità e prestazioni migliori su insiemi di dati di grandi dimensioni, nonché per il supporto di funzionalità di ottimizzazione SQL e modellazione in-database.
- SPSS Modeler Solution Publisher, per il calcolo automatico o in tempo reale del punteggio al di fuori dall'ambiente di SPSS Modeler. Per ulteriori informazioni, vedere l'argomento [IBM SPSS Modeler Solution Publisher](#) in il capitolo 2 a pag. 6.
- Adattatori per il deployment in IBM SPSS Collaboration and Deployment Services o nell'applicazione thin-client IBM SPSS Modeler Advantage. Per ulteriori informazioni, vedere l'argomento [Archiviazione e deployment di oggetti di IBM SPSS Collaboration and Deployment Services Repository](#) in il capitolo 9 in *Manuale dell'utente di IBM SPSS Modeler 14.2*.

IBM SPSS Text Analytics

IBM® SPSS® Text Analytics è un componente aggiuntivo completamente integrato per SPSS Modeler che utilizza tecnologie linguistiche avanzate e di Natural Language Processing (NLP) per elaborare rapidamente una grande varietà di dati di testo non strutturati, estrarre e organizzare i concetti chiave e raggruppare questi concetti in categorie. È quindi possibile combinare i concetti e le categorie estratti con dati strutturati esistenti, per esempio dati demografici, e applicarli alla modellazione utilizzando la suite completa degli strumenti di data mining di IBM® SPSS® Modeler per prendere decisioni migliori e più mirate.

- Il nodo Text Mining offre funzioni di modellazione di concetti e categorie nonché un workbench interattivo in cui è possibile eseguire esplorazioni avanzate di collegamenti e cluster di testo, creare categorie personalizzate e rifinire i modelli di risorse linguistiche.
- Sono supportati numerosi formati di importazione, inclusi blog e altre sorgenti basate sul Web.
- Sono inclusi inoltre modelli, librerie e dizionari personalizzati per domini specifici, quali CRM e genomica.

Nota: per l'accesso a questo componente, è necessaria una licenza distinta. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/modeler/>.

Documentazione di IBM SPSS Modeler

La documentazione completa nel formato guida in linea è disponibile nel menu Aiuto di SPSS Modeler. Sono incluse la documentazione per SPSS Modeler, SPSS Modeler Server e SPSS Modeler Solution Publisher, nonché la Guida alle applicazioni e altro materiale di supporto.

La documentazione completa in formato PDF dei singoli prodotti è disponibile nella cartella *\Documentation* del DVD di ciascun prodotto.

- **Manuale dell'utente di IBM SPSS Modeler.** Introduzione generale all'utilizzo di SPSS Modeler che illustra come creare stream di dati, gestire valori mancanti, generare espressioni CLEM, utilizzare progetti e report e assemblare stream per il deployment tramite IBM SPSS Collaboration and Deployment Services, le applicazioni predittive o IBM SPSS Modeler Advantage.
- **IBM SPSS Modeler Source, Process, and Output Nodes.** Descrizioni di tutti i nodi utilizzati per leggere, elaborare e generare dati di output in vari formati, ovvero di nodi ad eccezione dei nodi Modelli.
- **IBM SPSS Modeler Nodi Modelli.** Descrizioni di tutti i nodi utilizzati per creare modelli di data mining. IBM® SPSS® Modeler offre numerosi metodi di modellazione ricavati dall'apprendimento automatico, dall'intelligenza artificiale e dalla statistica. [Per ulteriori informazioni, vedere l'argomento Panoramica sui nodi Modelli in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)
- **IBM SPSS Modeler Algorithms Guide.** Descrizione dei fondamenti di matematica per i metodi di modellazione utilizzati in SPSS Modeler.
- **IBM SPSS Modeler Guida alle applicazioni.** Gli esempi inclusi in questa guida forniscono indicazioni mirate e sintetiche su specifici metodi e tecniche di modellazione. Una versione in linea di questa guida è inoltre disponibile dal menu Aiuto. [Per ulteriori informazioni, vedere l'argomento Esempi di applicazioni in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)
- **IBM SPSS Modeler Script e automazione.** Informazioni sulle modalità di automazione del sistema tramite script, incluse le proprietà che è possibile utilizzare per manipolare nodi e stream.
- **IBM SPSS Modeler Deployment Guide.** Informazioni sull'esecuzione di stream e scenari SPSS Modeler come fasi dell'elaborazione di lavori in IBM® SPSS® Collaboration and Deployment Services Deployment Manager.
- **IBM SPSS Modeler Guida per lo sviluppatore CLEF.** CLEF consente di integrare programmi di terze parti (quali routine di elaborazione di dati o algoritmi di modellazione) come nodi in SPSS Modeler.
- **IBM SPSS Modeler Guida alla modellazione in-database.** Informazioni sulle modalità per utilizzare al meglio la potenza del database in uso al fine di ottenere prestazioni migliori ed estendere la gamma di funzionalità analitiche tramite algoritmi di terze parti.
- **IBM SPSS Modeler Server e Guida della performance.** Informazioni su come configurare e amministrare IBM® SPSS® Modeler Server.
- **Manuale dell'utente di IBM SPSS Modeler Administration Console.** Informazioni sull'installazione e l'utilizzo dell'interfaccia utente della console per il monitoraggio e la configurazione di SPSS Modeler Server. La console viene implementata come plug-in dell'applicazione Deployment Manager.
- **IBM SPSS Modeler Solution Publisher Guide.** SPSS Modeler Solution Publisher è un componente aggiuntivo che consente di pubblicare gli stream al di fuori dell'ambiente SPSS Modeler standard.
- **Guida CRISP-DM di IBM SPSS Modeler.** Guida passo a passo al data mining tramite la metodologia CRISP-DM con SPSS Modeler.

Esempi di applicazioni

Mentre gli strumenti per il data mining di SPSS Modeler consentono di risolvere un'ampia gamma di problemi a livello aziendale e organizzativo, gli esempi di applicazioni forniscono indicazioni mirate e sintetiche su specifici metodi e tecniche di modellazione. Gli insiemi di dati utilizzati negli esempi hanno dimensioni molto più limitate rispetto agli enormi archivi di dati gestiti da alcuni data miner, ma i concetti e i metodi coinvolti sono rapportabili alle applicazioni del mondo reale.

È possibile accedere agli esempi facendo clic su Esempi di applicazioni nel menu Aiuto di SPSS Modeler. I file di dati e gli stream di esempio sono installati nella cartella *Demos* nella directory di installazione del prodotto. [Per ulteriori informazioni, vedere l'argomento Cartella Demos in *Manuale dell'utente di IBM SPSS Modeler 14.2*.](#)

Esempi di modellazione in-database. Vedere gli esempi nella *IBM SPSS Modeler Guida alla modellazione in-database*.

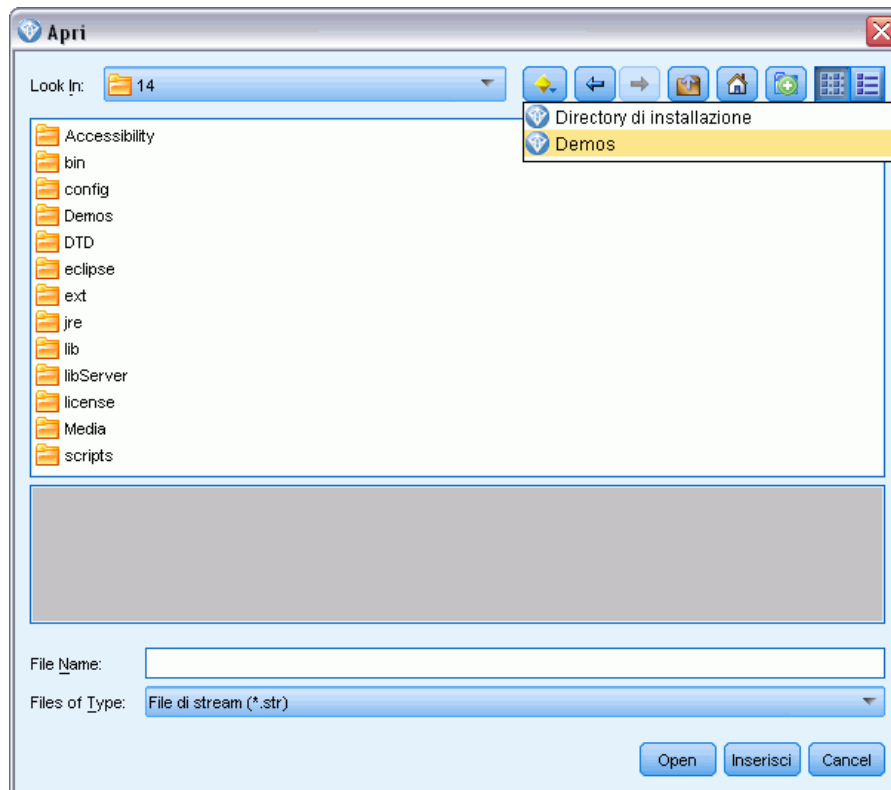
Esempi di script. Vedere gli esempi nella *IBM SPSS Modeler Guida per script e automazione*.

Cartella Demos

I file di dati e gli stream di esempio utilizzati negli esempi di applicazioni sono installati nella cartella *Demos* nella directory di installazione del prodotto. A questa cartella è possibile accedere anche dal gruppo di programmi IBM SPSS Modeler 14.2 nel menu Start di Windows oppure facendo clic su *Demos* nell'elenco delle directory recenti nella finestra di dialogo Apri file.

Figura 1-1

Selezione della cartella *Demos* dall'elenco delle directory utilizzate di recente



IBM SPSS Modeler Solution Publisher

IBM® SPSS® Modeler Solution Publisher è un potente strumento che consente di integrare i risultati di data mining nel processo aziendale per risolvere i problemi del mondo reale. Con SPSS Modeler Solution Publisher è possibile creare una versione “pacchetto” dello stream che può essere eseguita da un motore di runtime esterno o incorporata in un’applicazione esterna. Questo consente di eseguire il deployment degli stream di modellazione dei dati in un ambiente di produzione, per supportare i processi aziendali e fornire ai responsabili dei processi decisionali le conoscenze acquisite con il mining dei dati.

Anziché limitarsi a esportare file di modello (come PMML), SPSS Modeler Solution Publisher permette di pubblicare ed eseguire il deployment di *stream completi di IBM® SPSS® Modeler*. Ciò significa che è possibile effettuare la preparazione dei dati ed eseguire operazioni su record e campi, per esempio aggregare dati, selezionare record o derivare nuovi campi, prima di creare previsioni basate su un modello. Prima di salvare i dati, è quindi possibile elaborare ulteriormente i risultati del modello, eseguendo semplicemente lo stream pubblicato.

Nota: la pubblicazione avviene mediante la scheda Pubblica del nodo di esportazione appropriato. [Per ulteriori informazioni, vedere l'argomento Pubblicazione di stream a pag. 7.](#) Gli stream salvati in versioni del prodotto precedenti alla 11.0 utilizzavano un nodo Publisher separato. Il nodo è ancora supportato per la compatibilità con le versioni precedenti, ma non è più disponibile nella palette dei nodi.

Attivazione della licenza di IBM SPSS Modeler Solution Publisher

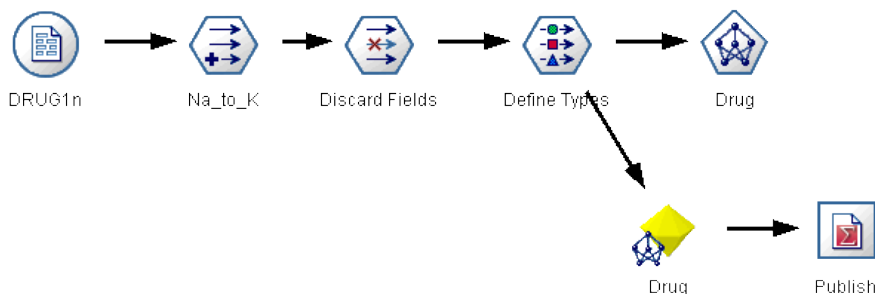
Nota: SPSS Modeler Solution Publisher è distribuito come parte integrante di IBM® SPSS® Collaboration and Deployment Services Scoring Service, per cui è necessaria una licenza separata. <http://www.ibm.com/software/analytics/spss/products/deployment/cds/>

Quando si acquista una licenza, separatamente viene fornito IBM® SPSS® Modeler Solution Publisher Runtime che consente di eseguire stream pubblicati. [Per ulteriori informazioni, vedere l'argomento Installazione del Runtime in il capitolo 4 a pag. 15.](#)

Funzionamento di IBM SPSS Modeler Solution Publisher

Il deployment di una soluzione tramite IBM® SPSS® Modeler Solution Publisher è costituito da due fasi: la *pubblicazione* di uno stream e l'*esecuzione* di uno stream.

Figura 2-1
Pubblicazione di uno stream



Pubblicazione. Il processo di data mining porta infine a un modello che fornisce una buona soluzione del problema aziendale. Sarà possibile utilizzare tale modello ed applicarlo al processo aziendale specifico. Quando si pubblica uno stream, una descrizione dettagliata dello stream viene scritta sul disco (come file *immagine* e file di *parametri*). [Per ulteriori informazioni, vedere l'argomento Pubblicazione di stream a pag. 7.](#)

Esecuzione. Dopo la pubblicazione dello stream, è possibile ricreare il processo implementato nello stream eseguendo lo stream pubblicato. Questa operazione si esegue utilizzando IBM® SPSS® Modeler Runtime autonomo (*modelerrun.exe*) oppure sviluppando un'applicazione che utilizza la libreria di SPSS Modeler Runtime per eseguire lo stream. Per eseguire stream al di fuori di SPSS Modeler (tramite il Runtime o un'applicazione personalizzata), è necessario prima installare IBM® SPSS® Modeler Solution Publisher Runtime. [Per ulteriori informazioni, vedere l'argomento Installazione del Runtime in il capitolo 4 a pag. 15.](#)

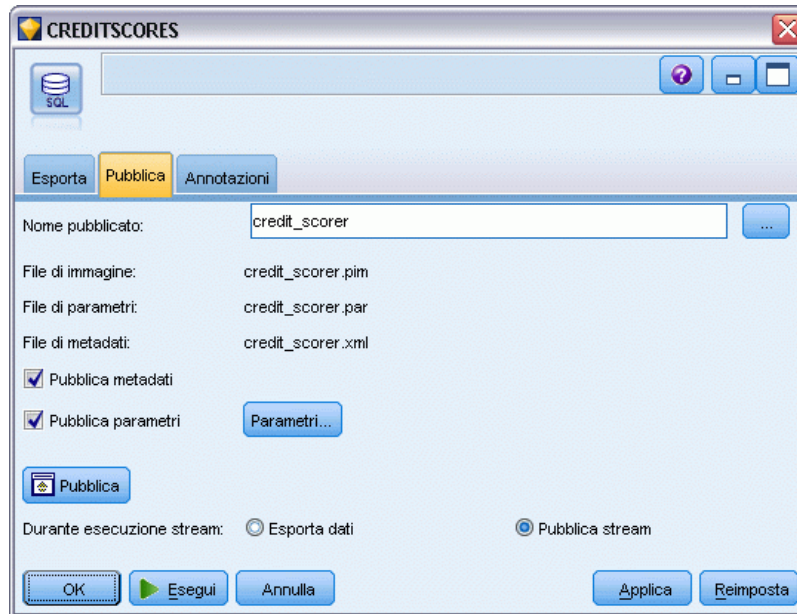
Pubblicazione di stream

La pubblicazione degli stream viene eseguita direttamente da IBM® SPSS® Modeler utilizzando uno qualsiasi dei nodi di esportazione standard: Database, File piatto, Esporta Statistics, Esporta IBM® SPSS® Data Collection, Esporta SAS, Excel ed Esporta XML. Il tipo di nodo di esportazione determina il formato dei risultati da scrivere ogni volta che lo stream pubblicato viene eseguito utilizzando IBM® SPSS® Modeler Solution Publisher Runtime o un'applicazione esterna. Per esempio, se si desidera scrivere i risultati in un database ogni volta che si esegue lo stream pubblicato, utilizzare un nodo Database. [Per ulteriori informazioni, vedere l'argomento Panoramica sui nodi di esportazione in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Per pubblicare uno stream

- ▶ Aprire o creare uno stream come di consueto, quindi collegare un nodo di esportazione alla fine.
- ▶ Nella scheda *Pubblica* del nodo di esportazione, specificare un nome di base per i file pubblicati (ovvero il nome file a cui saranno aggiunte le estensioni *.pim*, *.par* e *.xml*).
- ▶ Fare clic sul pulsante *Pubblica* per pubblicare lo stream oppure selezionare *Pubblica stream* per pubblicare automaticamente lo stream ogni volta che il nodo viene eseguito.

Figura 2-2
Scheda *Pubblica*



Nome pubblicato. Specificare il nome di base per i file immagine e dei parametri pubblicati.

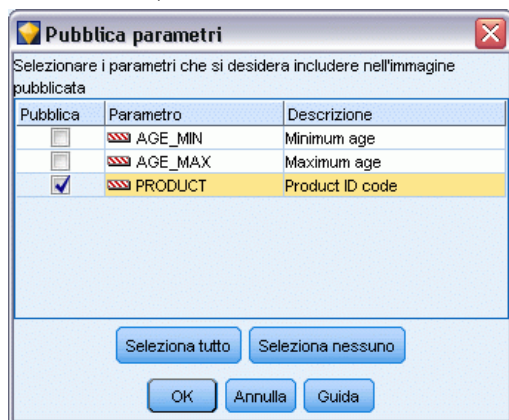
- Il **file immagine** (*.pim) contiene tutte le informazioni necessarie affinché Runtime possa eseguire lo stream pubblicato esattamente come era al momento dell'esportazione. Se non è necessario modificare le impostazioni dello stream quali la sorgenti dati di input o il file dati di output, è possibile eseguire il deployment solo del file immagine.
- Il file *.par è un **file di parametri** contenente informazioni configurabili relative a sorgenti dati, file output e opzioni di esecuzione. Se si desidera controllare l'input o l'output dello stream senza pubblicare nuovamente lo stream, sarà necessario il file di parametri così come il file immagine.
- Il **file di metadati** (*.xml) descrive gli input e gli output dell'immagine e dei rispettivi modelli di dati. Viene utilizzato da applicazioni che incorporano la libreria runtime e che devono conoscere la struttura dei dati di input e output.

Nota: questo file viene generato solo se si seleziona la casella *Pubblica metadati*.

Pubblica parametri. Se necessario, è possibile includere parametri di stream nel file *.par. È possibile modificare questi valori di parametri di stream modificando il file *.par quando si esegue l'immagine, oppure tramite l'API runtime.

Questa casella di controllo attiva il pulsante *Parametri*. Facendo clic sul pulsante viene visualizzata la finestra di dialogo *Pubblica parametri*.

Figura 2-3
Selezione dei parametri



Scegliere i parametri da includere nell'immagine pubblicata selezionando la rispettiva casella di controllo nella colonna Pubblica.

Durante esecuzione stream. Specifica se lo stream viene pubblicato automaticamente quando viene eseguito il nodo.

- **Esporta dati.** Esegue il nodo di esportazione come di consueto, senza pubblicare lo stream (di fatto il nodo viene eseguito in SPSS Modeler come se IBM® SPSS® Modeler Solution Publisher non fosse disponibile). Se si seleziona questa opzione, lo stream non viene pubblicato finché non lo si pubblica esplicitamente facendo clic sul pulsante Pubblica nella finestra di dialogo del nodo di esportazione. In alternativa, è possibile pubblicare lo stream corrente utilizzando lo strumento Pubblica disponibile sulla barra degli strumenti oppure utilizzando uno script.
- **Pubblica stream.** Pubblica lo stream per il deployment utilizzando SPSS Modeler Solution Publisher. Selezionare questa opzione se si desidera pubblicare automaticamente lo stream ogni volta che viene eseguito.

Commenti

- Se si intende eseguire lo stream pubblicato con dati nuovi o aggiornati, è importante tenere presente che l'ordine dei campi nel file di input deve essere uguale all'ordine dei campi nel file di input del nodo di input specificato nello stream pubblicato.
- In caso di pubblicazione in applicazioni esterne, è necessario considerare la ridenominazione dei campi o il filtro di campi estranei per garantire la conformità ai requisiti di input. È possibile eseguire entrambe le operazioni utilizzando un nodo Filtro prima del nodo di esportazione. [Per ulteriori informazioni, vedere l'argomento Applicazione di filtri o ridenominazione di campi in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esecuzione di stream pubblicati

Il principale impatto dell'esecuzione di uno stream pubblicato è sempre quella di generare un insieme di dati. I dati possono essere memorizzati su disco o come database. Gli stream pubblicati non possono generare grafici, tabelle, modelli generati oppure altri tipi di output senza dati.

È possibile eseguire gli stream pubblicati con IBM® SPSS® Modeler Solution Publisher utilizzando il programma IBM® SPSS® Modeler Solution Publisher Runtime. Il programma di runtime, *modelerrun.exe*, viene avviato dalla riga di comando con le opzioni che indicano il file immagine e facoltativamente il file di parametri da eseguire. Il comando di runtime è il seguente:

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

dove

-nobanner	elimina il messaggio di testo di avvio del runtime (facoltativo).
<parameter-file>	è il nome del file di parametri (facoltativo).
<opzioni>	è un elenco separato da virgole di coppie opzione=valore che specificano le opzioni di esecuzione per il runtime. Le opzioni vengono dettagliate di seguito (facoltativo).
<image-file>	è il nome del file immagine da eseguire (obbligatorio).

Sono disponibili le seguenti opzioni di esecuzione:

Tabella 3-1
Opzioni di runtime valide

Name	Valore predefinito	Descrizione
max_file_size	-1	Dimensione massima dei file salvati. Un valore pari a -1 indica nessun limite.
max_sql_string_length	2,048	Lunghezza massima di una stringa importata dal database con SQL. I valori di stringa superiori vengono troncati a destra senza avvertire l'utente. L'intervallo valido è compreso tra 1 e 65.535 caratteri.
memory_usage	100	Moltiplicatore per allocazione di memoria dinamica. Regolare questo valore diminuendolo o aumentandolo per regolare l'utilizzo della memoria totale del server.
temp_directory	"	Specifica la directory da utilizzare per i file temporanei.
request_passwords	Y	Controlla se viene chiesta una password per il database quando viene eseguito uno stream pubblicato che richiede l'accesso al database (Y/N).

Utilizzo del file di parametri per personalizzare l'esecuzione degli stream

Quando uno stream viene pubblicato, il file immagine contiene informazioni sulla sorgente dati e sulla destinazione dei dati di output che sono stati selezionati nello stream originale. Tuttavia, spesso conviene eseguire uno stream rispetto a dati provenienti da una sorgente dati diversa oppure reindirizzare i dati di output su una destinazione diversa. È possibile eseguire questa operazione modificando il file di parametri dello stream.

Si tratta di un'operazione molto importante quando lo stream pubblicato viene eseguito su un computer diverso da quello in cui era stato creato. Sarà necessario aggiornare le posizioni dei file di input e di output per riflettere la struttura del computer di destinazione. Tenere presente che gli stream pubblicati non supportano i relativi percorsi, ed è per questo che è molto importante verificare le posizioni dei dati nel file di parametri. Se si specificano i percorsi relativi nelle opzioni del nodo Esporta quando si pubblica lo stream, IBM® SPSS® Modeler li converte in percorsi assoluti prima di creare i file immagine e di parametri per lo stream pubblicato.

È inoltre possibile utilizzare il file di parametri per impostare le opzioni di esecuzione e modificare i valori dei parametri degli stream.

I valori dei parametri vengono specificati nel file di parametri come coppie <name.attribute>=<value>. In alcuni, il nome del parametro contiene un ID digitale per distinguere i nodi dello stesso tipo nello stream. Per esempio, in uno stream in cui input e output fanno riferimento alla connessione di un database, i parametri che descrivono la connessione di input possono essere dbconn0.datasource, dbconn0.user, e così via e i parametri associati alla connessione di output possono essere dbconn1.datasource, dbconn1.user, ecc. I valori sono sempre racchiusi tra virgolette doppie. Il file di parametri contiene i seguenti parametri, che possono essere modificati per personalizzare l'esecuzione degli stream:

Tabella 3-2
Parametri di esecuzione

Parametro	Scopo
dbconnx.datasource	Nome sorgente dati, Data source name (DSN).
dbconnx.user	Nome utente per database con accesso limitato.
dbconnx.password	Password per database con accesso limitato
dbconnx.epassword	Password codificata per database con accesso limitato. Per generare una password codificata, selezionare Codifica password dal menu Strumenti dell'interfaccia utente di SPSS Modeler. Copiare ed incollare la password codificata se necessario.
dbconnx.password_required	Flag che indica se richiedere una password per questa connessione al database (Il valore è "S" o "N".) Impostato automaticamente su "N" se non viene specificata una password al momento della pubblicazione.
dbconnx.catalog	Catalogo associato alla connessione al database.
dbtablex.name	Nome della tabella di base. <i>Nota:</i> L'ordine dei campi in una nuova tabella deve essere uguale all'ordine dei campi nello stream pubblicato in origine.
dbtablex.schema	Nome dello schema.
dbtablex.catalog	Catalogo associato con la tabella corrente.
filex.name	Nome del file per il file di dati. <i>Nota:</i> l'ordine dei campi in un nuovo file di input deve essere uguale all'ordine dei campi nello stream pubblicato in origine.

Parametro	Scopo
filex.path	Posizione del file (nome della directory).
filex.field_names_included	Per i file in formato testo, il flag che indica se i nomi dei campi sono inclusi come prima riga del file. (Il valore è "S" o "N".)
filex.field_separator	Per i file in formato testo, i caratteri utilizzati per separare i valori di campo.
filex.decimal_separator	Per i file in formato testo, il carattere utilizzato per il punto decimale.
options.angle_in_radians	Flag che indica se i radianti vengono utilizzati come unità di misura nelle espressioni trigonometriche CLEM. (Il valore è "S" o "N".)
options.date_2digit_baseline	Definisce il secolo delle date con l'anno a due cifre.
options.date_baseline	Anno di riferimento (sempre 1 gen) utilizzato dalle funzioni di data CLEM che operano con una singola data.
options.time_rollover	Flag che indica se le differenze di tempo negative si riferiscono al passato.
options.decimal_separator	Il carattere a punto decimale di default nei file di testo.
options.time_format	Formato di ora utilizzato quando le stringhe vengono interpretate come orari dalle funzioni di ora di CLEM.
options.date_format	Formato di ora utilizzato quando le stringhe vengono interpretate come date dalle funzioni di data di CLEM.
options.timestamp_format	Formato utilizzato durante la lettura dei campi Timestamp come stringhe da sorgenti di dati ODBC.
paramx.value	Valore del parametro dello stream. <i>Nota:</i> il nome del parametro, la descrizione e il tipo di archiviazione sono inclusi per riferimento ma non sono modificabili. Il valore deve essere valido per il tipo.

Integrazione di IBM SPSS Modeler Solution Publisher con altre applicazioni

Oltre al motore di runtime autonomo, IBM® SPSS® Modeler Solution Publisher dispone di una libreria di programmazione del runtime (CLEMRTL) che consente ad altri programmi di controllare l'esecuzione dello stream di SPSS Modeler Solution Publisher. È possibile chiamare le procedure CLEMRTL nei programmi client scritti in C e C++. Per utilizzare CLEMRTL, è necessario includere il file di intestazione *clemrtl.h* (disponibile nella cartella *..\installation\clemrtl\include*) e accertarsi di collegare il file della libreria appropriato alla piattaforma di sviluppo nell'applicazione quando viene creato. Il file *clemrtl.dll* è disponibile nella cartella *..\installation\bin*.

Qualsiasi file sorgente che fa riferimento a procedure della libreria deve includere l'intestazione *clemrtl.h*. Questo file di intestazione contiene prototipi ANSI C per le procedure della libreria e consente di definire macro utili. Non richiede l'inclusione di nessun'altra intestazione oltre a quelle richieste dal programma. Per evitare sovrapposizioni di nomi, tutti i tipi di libreria e i nomi di funzioni iniziano con *clemrtl_* e tutti i nomi di macro hanno il prefisso *CLEMRTL_*.

CLEMRTL genera rapporti che contengono informazioni utili che devono essere ritrasmesse in qualche modo all'applicazione. CLEMRTL dispone di tre meccanismi per gestire questo tipo di messaggi:

- I messaggi possono essere scritti in un file di registro specificato dalla funzione `clemrtl_setLogFile()`.
- L'applicazione può recuperare i dettagli relativi all'ultimo errore tramite la funzione `clemrtl_getErrorDetail()`.
- L'applicazione dispone della propria procedura di gestione dei report tramite la funzione `clemrtl_setReportHandler()`.

Il sistema di runtime contiene un catalogo di messaggi localizzati per i report. Qualsiasi report trasmesso all'applicazione include la stringa del messaggio localizzato. I report includono anche il codice del report in modo che l'applicazione possa scegliere se interpretare e presentare il messaggio diversamente. Per ottenere messaggi dettagliati localizzati, l'applicazione deve inviare il file appropriato `messages.cfg` in una directory `config`. È possibile trovare diverse versioni di `messages.cfg` per opzioni internazionali diverse nella directory `<installdir>/config/<locale>`, dove `<installdir>` è la directory in cui è installato SPSS Modeler Solution Publisher e `<locale>` sono le opzioni internazionali desiderate. Selezionare la versione appropriata di `messages.cfg` ed aggiungerla a una sottodirectory `config` nella directory del proprio progetto.

Il sistema di runtime non necessita di un ambiente o di impostazioni di registro speciali per operare correttamente. Le librerie dipendenti devono essere distribuite con un'applicazione collegata a CLEMRTL. Questi file sono inclusi nella directory `<installdir>/bin`.

Requisiti

Di seguito vengono riportate le piattaforme e i compilatori sottoposti a test tramite la libreria di programmazione di runtime.

Tabella 3-3
Piattaforme e compilatori

Piattaforma	Compilatori consigliato
AIX	IBM XL C/C++ Enterprise Edition V8.0 per AIX
HP-UX 11.0	aC++/ANSI C B3910B A.06.05
Solaris a 64 bit	Sun Studio 11 5.8 Patch 121017-06
Linux a 32 bit e 64 bit	gcc 3.2.3, compilatore Intel C++ 9.1.043
Windows	Visual Studio 2005

Nota: In HP-UX, l'applicazione deve essere creata con il flag del compilatore aCC “-AA” per consentire il collegamento alle librerie CLEMRTL.

Applicazione di esempio IBM SPSS Modeler Solution Publisher

Un'applicazione di esempio, `dlltest.c`, è inclusa nella cartella `\clemrtl\demo` della directory di installazione di IBM® SPSS® Modeler Solution Publisher. Utilizzare l'applicazione di esempio per familiarizzare con l'integrazione della libreria di runtime con altre applicazioni.

Le informazioni utilizzate dall'applicazione di esempio sono incluse nella cartella `\clemtrl\demo`; per le istruzioni, vedere il file `README.txt`. La compilazione di un'applicazione varia in base alla piattaforma, quindi vengono inclusi anche makefile di esempio.

Installazione di IBM SPSS Modeler Solution Publisher Runtime

Installazione del Runtime

Quando si acquista IBM® SPSS® Modeler Solution Publisher come componente aggiuntivo di IBM® SPSS® Modeler, si riceve un disco di installazione separato. Utilizzare questo disco per installare IBM® SPSS® Modeler Solution Publisher Runtime, che consente di eseguire stream pubblicati. Gli stream pubblicati non dipendono dal sistema operativo in uso. È pertanto possibile eseguire qualsiasi stream pubblicato su qualsiasi piattaforma supportata dal Runtime.

Nota: la funzionalità che consente di pubblicare stream da SPSS Modeler viene installata con IBM® SPSS® Modeler ed è possibile attivarla acquistando un codice di autorizzazione separato. Si noti che i file pubblicati devono essere riesportati con ogni versione di SPSS Modeler. Per esempio, un file esportato con una versione precedente non funzionerà con la versione 13 di SPSS Modeler Solution Publisher Runtime.

Requisiti di sistema

Per l'installazione di IBM® SPSS® Modeler Solution Publisher Runtime sono necessari i seguenti requisiti:

- **Sistema operativo.** Windows Server 2008 R2 per sistemi x64 a 64 bit; Windows Server 2003 Standard Edition R2 per sistemi x86 a 32 bit o x64 a 64 bit
IBM AIX® 6.1 o 7.1 per sistemi POWER a 64 bit; HP-UX 11i V3 per sistemi Itanium a 64 bit; Oracle Solaris™ 9.x o 10 per sistemi SPARC a 64 bit; Red Hat Enterprise Linux 5.x per sistemi x86 a 32 bit; Red Hat Enterprise Linux 5.x o 6.x per sistemi x64 a 64 bit o sistemi IBM System z; Red Hat Enterprise Linux Advanced Platform 5.x per sistemi x86 a 32 bit; Red Hat Enterprise Linux Advanced Platform 5.x o 6.x per sistemi x64 a 64 bit o sistemi IBM System z; SuSE Linux Enterprise Server 10 o 11 per sistemi x64 a 64 bit o sistemi IBM System z
- **Hardware.** Processore Pentium® o di classe Pentium o superiore (per Windows a 32 bit); famiglia di processori x64 (AMD 64 ed EM64T) (per Windows a 64 bit) da 1 GHz o superiori
Processore PowerPC da 233 MHz o superiore e IBM System p per IBM AIX; processore Itanium per HP-UX; UltraSPARC II (o superiore) per Solaris; processore Pentium o di classe Pentium o superiore per Linux a 32 bit; famiglia di processori x64 (AMD 64 ed EM64T) o IBM System z per Linux a 64 bit
- **Spazio libero su disco.** Si consiglia 1 gigabyte (GB). 2 GB consigliati se il sistema operativo in uso è HP-UX.
- **RAM.** Si consiglia 4 GB.
- **Ambiente virtuale.** Gli ambienti virtuali seguenti supportano IBM® SPSS® Modeler Server.
 - IBM PowerVM Hypervisor per IBM AIX
 - IBM PR/SM o z/VM per Linux per IBM System z

- VMWare ESX Server 4.1
- VMWare vSphere 4.0
- **Ambiente cluster.** Gli ambienti cluster seguenti supportano SPSS Modeler Server.
 - IBM HACMP/RSCT per IBM AIX
 - IBM WebSphere App Server a 64 bit
 - Microsoft Cluster Service per Windows
 - Oracle Solaris Cluster
 - Oracle WebLogic App Server a 64 bit
 - Red Hat Cluster Suite per Red Hat Enterprise Linux

Installazione in ambiente Windows

Nella seguente sezione è descritta la procedura per installare SPSS Modeler Solution Publisher Runtime sul sistema operativo Windows.

- ▶ Inserire il disco di installazione di IBM SPSS Modeler Solution Publisher nella relativa unità.
- ▶ Nel menu di installazione, scegliere Installa IBM® SPSS® Modeler Solution Publisher Runtime o Installa IBM® SPSS® Modeler Solution Publisher Runtime for Win64, a seconda del sistema su cui viene eseguita l'installazione.
- ▶ Nell'installazione guidata fare clic su Avanti per iniziare.
- ▶ Seguire le istruzioni visualizzate sullo schermo. Per continuare, fare clic su Avanti.
- ▶ Dopo aver specificato tutte le opzioni, è possibile procedere all'installazione. Fare clic su Installa per iniziare a trasferire i file.
- ▶ Dopo che tutti i file sono stati installati, fare clic su Fine.

Installazione in ambiente UNIX

Nella seguente sezione sono descritti i requisiti di sistema aggiuntivi e la procedura per installare IBM® SPSS® Modeler Solution Publisher Runtime sul sistema operativo UNIX.

Requisiti aggiuntivi per UNIX

È necessario verificare che i limiti del kernel sul sistema siano sufficienti a consentire il funzionamento di IBM® SPSS® Modeler Server. I parametri ulimit relativi a dati, memoria e file sono particolarmente importanti e dovrebbero essere impostati su un valore illimitato nell'ambiente SPSS Modeler Server. A questo scopo, eseguire le operazioni seguenti:

- ▶ Aggiungere i seguenti comandi a *modelersrv.sh*:

```
ulimit -d unlimited
```

```
ulimit -m unlimited
```

```
ulimit -f unlimited
```

- Riavviare SPSS Modeler Server.

Solaris.

Oltre ai requisiti di sistema citati in precedenza, l'installazione su Solaris richiede anche le seguenti patch di versione specifiche. È possibile scaricare le patch dalla pagina <http://sunsolve.sun.com/pub-cgi/show.pl?target=patchpage>.

- *ID patch Solaris 2,9*: 111711-16, 111712-16 e 111722-05 (C++)
- *ID patch Solaris 2,10*: 119963 (C++)

Inoltre è necessario installare nel sistema la libreria `/usr/lib/iconv/646%UTF-8`. Se non è ancora stata installata, procedere all'installazione dei pacchetti `SUNWiniu8` o `SUNWiniu8x` per installare la libreria come parte dei moduli Indic iconv per UTF-8.

Linux a 32-bit.

Per una corretta esecuzione sono necessari i seguenti pacchetti aggiuntivi. È possibile scaricare questi pacchetti da Red Hat Network. Per accedere alla rete potrebbe essere necessario registrarsi (<https://rhn.redhat.com/network/software/search.pxt>). Utilizzare la funzionalità di ricerca e visualizzare ciascuna versione dei pacchetti specifica.

- Il pacchetto `libstdc++` (versione: `libstdc++-3.2.3-X.Y.i386` dove `X.Y` indica `22.1` o versione successiva).
- Il pacchetto `libgcc` (versione: `libgcc-3.2.3-X.Y.i386` dove `X.Y` indica `22.1` o versione successiva).
- Il pacchetto `compat-libstdc++` (versione: `compat-libstdc++-33-3.2.3-X.Y.i386` dove `X.Y` è `47.3` o superiore).

Linux a 64-bit.

Per una corretta esecuzione sono necessari i seguenti pacchetti aggiuntivi. È possibile scaricare questi pacchetti da Red Hat Network. Per accedere alla rete potrebbe essere necessario registrarsi (<https://rhn.redhat.com/network/software/search.pxt>). Utilizzare la funzionalità di ricerca e visualizzare ciascuna versione dei pacchetti specifica.

- Il pacchetto `libstdc++` (versione: `libstdc++-3.2.3-X.Y.x86_64` dove `X.Y` indica `22.1` o versione successiva).
- Il pacchetto `libgcc` (versione: `libgcc-3.2.3-X.Y.x86_64` dove `X.Y` indica `22.1` o versione successiva).
- Il pacchetto `compat-libstdc++` (versione: `compat-libstdc++-33-3.2.3-X.Y.x86_64` dove `X.Y` indica `47.3` o versione successiva).

AIX.

Oltre ai requisiti di sistema citati in precedenza, per l'installazione su AIX è necessario quanto segue:

- Il livello di runtime di XL C++ deve essere almeno 8.0.0.9 (lspp -l xlc.rte). Gli insiemi di file contenenti le librerie sono indicati come IBM C++ Runtime Environment Components for AIX e possono essere scaricati da <http://www.ibm.com/software/awdtools/xlcpp/support/>.
- Il livello di runtime di XL FORTRAN deve essere almeno 10.1.0.4 (lspp -l xlf.rte). Gli insiemi di file contenenti le librerie sono indicati come XL Fortran Enterprise Edition V10.1 for AIX Runtime Environment e possono essere scaricati da <http://www.ibm.com/software/awdtools/fortran/xlfortran/support/>.

HP-UX.

Oltre ai requisiti di sistema citati in precedenza, l'installazione su HP-UX richiede inoltre che siano installate le librerie C++ Runtime. È possibile scaricarle come patch PHSS_26946, come mostrato nella tabella seguente. Per accedere alle patch, visitare la pagina <http://www4.itrc.hp.com/service/patch/mainPage.do>. La registrazione è obbligatoria ed è gratuita.

Procedura di installazione su computer UNIX

Le seguenti istruzioni di installazione si riferiscono alla versione IBM® SPSS® Modeler Solution Publisher Runtime per UNIX. È necessario disporre di autorizzazioni in lettura e scrittura per la directory di installazione di destinazione. Accedere pertanto con un account che disponga di autorizzazioni sufficienti.

- ▶ Inserire il disco di installazione nel lettore, assicurandosi di poter accedere a questa unità. Su Solaris deve essere in esecuzione il gestore dei volumi (ovvero il daemon vold).
- ▶ Cambiare le directory nel punto di avvio del DVD.
- ▶ Cambiare le directory con la directory *modelrun*.
- ▶ Passare alla directory della piattaforma pertinente.
- ▶ Eseguire lo script di installazione *.bin*. Per esempio:


```
./modelersolutionpublisherhpia64.bin -i console
```
- ▶ Vengono visualizzati i dettagli di introduzione. Premere Invio per continuare.
- ▶ Vengono visualizzate le informazioni sulla licenza. Leggere la licenza, digitare 1 per accettarla e premere Invio per continuare.
- ▶ Viene visualizzato un elenco delle lingue disponibili. Immettere il numero della versione di lingua da installare e premere Invio per continuare.
- ▶ Viene richiesto di immettere il percorso di installazione. Per utilizzare la directory di default, */usr/IBM/SPSS/ModelerSolutionPublisher<nn>* (dove *<nn>* è il numero di versione), premere Invio. Il Runtime verrà installato nella directory specificata.
- ▶ Viene richiesto di confermare il percorso di installazione. Se il percorso è corretto, digitare *y* e premere Invio.

- ▶ Viene visualizzato un riepilogo delle informazioni specificate. Premere Invio per continuare.
- ▶ Un messaggio indica che la routine di installazione è pronta per essere eseguita. Premere Invio per continuare.
- ▶ Una barra indica lo stato di avanzamento dell'esecuzione della routine di installazione. Al termine dell'installazione, premere Invio per uscire dal programma di installazione.

Avvio del Runtime

Dopo aver installato il Runtime, è possibile utilizzarlo per eseguire stream che sono stati pubblicati da IBM® SPSS® Modeler utilizzando uno dei nodi di esportazione. [Per ulteriori informazioni, vedere l'argomento Pubblicazione di stream in il capitolo 2 a pag. 7.](#) Avviare il Runtime dalla riga di comando, con opzioni indicanti il nome del file di immagine pubblicato e un file di parametri facoltativo da eseguire.

Per eseguire il Runtime, al prompt dei comandi digitare la riga seguente:

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

dove

<parameter-file>	è il nome di file del file di parametri pubblicato (facoltativo).
<options>	è un elenco separato da virgole di coppie option=value che specificano le opzioni di esecuzione per il runtime.
<image file>	è il nome di file del file di immagine pubblicato (*.pim) da eseguire.

Nota: quando si utilizzano lingue multibyte come il cinese o il giapponese in ambiente UNIX, è necessario specificare i codici di lingua da utilizzare. L'esempio seguente mostra i codici di lingua per il cinese semplificato:

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

in cui -p è la posizione dei file PIM e PAR.

[Per ulteriori informazioni, vedere l'argomento Esecuzione di stream pubblicati in il capitolo 3 a pag. 10.](#)

Modifica della directory Temp

Alcune operazioni eseguite da IBM® SPSS® Modeler Solution Publisher Runtime potrebbero richiedere la creazione di file temporanei. Per default, per la creazione di file temporanei in Runtime viene utilizzata la directory temporanea di sistema. È possibile modificare il percorso della directory temporanea, attenendosi alla procedura seguente.

Nota: è molto importante utilizzare la sintassi corretta. Prestare particolare attenzione agli spazi e al percorso della directory.

Si supponga di utilizzare SPSS Modeler Solution Publisher Runtime con i seguenti file pubblicati, *webtest.pim* e *webtest.par*, e di voler modificare la posizione dei file temporanei per utilizzare la directory *C:\published\temp*. Per modificare la posizione della directory temp utilizzata dall'eseguibile:

- ▶ Dalla directory in cui sono salvati i file *.pim* e *.par*, eseguire il seguente comando:

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

Runtime utilizzerà la directory "*C:\published\temp*" come directory temporanea solo per quel processo specifico.

Disinstallazione di IBM SPSS Modeler Solution Publisher Runtime

Nella sezione seguente sono descritte le procedure per la disinstallazione di IBM® SPSS® Modeler Solution Publisher Runtime.

Procedura di disinstallazione su computer Windows

- ▶ Nel Pannello di controllo di Windows, selezionare Installazione applicazioni.
- ▶ Dall'elenco, selezionare IBM® SPSS® Modeler Solution Publisher <nn>.
- ▶ Fare clic su Rimuovi.
- ▶ Per rimuovere il Runtime, seguire le istruzioni visualizzate sullo schermo.

Nota: per rimuovere programmi, è necessario accedere al computer con i privilegi di amministratore.

Procedura di disinstallazione su computer UNIX

- ▶ Rimuovere la directory di installazione, inclusi tutti i file installati.

Riferimenti all'API della libreria di IBM SPSS Modeler Solution Publisher Runtime

Introduzione

L'API CLEMRTL consente di controllare l'esecuzione di stream pubblicati dall'applicazione. In questa sezione sono descritte le funzioni API disponibili.

Di seguito sono riportati argomenti di interesse generale che riguardano l'utilizzo dell'API:

- L'API viene avviata al collegamento C per la massima compatibilità, ma la libreria ha dipendenze C++. In alcune piattaforme questo significa che può essere utilizzata solo con un linker che riconosce C++.
- Prima di utilizzare qualsiasi altra funzione della libreria, è necessario chiamare `clemrtl_initialise_ext()`.
- Il tipo `clemrtl_image_handle` è utilizzato come un identificatore di immagine.
- Ogni funzione restituisce un codice di indicatore di stato. Di seguito sono elencati i valori dell'indicatore di stato:

CLEMRTL_OK	Esito positivo
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con informazioni aggiuntive sull'errore disponibili mediante la funzione <code>clemrtl_getErrorDetail()</code>

Panoramica del processo API

Di seguito è riportata la struttura generale di un'applicazione che utilizza l'API:

- ▶ Inizializzare la libreria con `clemrtl_initialise_ext()`.
- ▶ Controllare e modificare le opzioni di esecuzione con `clemrtl_getOption()` e `clemrtl_setOption()`.
- ▶ Aprire un'immagine con `clemrtl_openImage()` e ricevere un handle di immagine.
- ▶ Controllare e modificare i parametri dell'immagine con `clemrtl_enumerateParameters()`, `clemrtl_getParameter()` e `clemrtl_setParameter()`.
- ▶ Eseguire l'immagine con `clemrtl_execute()`. Se è necessario eseguire ripetutamente la stessa immagine senza modificarne i parametri, utilizzare prima `clemrtl_prepare()`.

- ▶ Chiudere l'immagine con `clemrtl_closeImage()`.
- ▶ Per annullare un'esecuzione in corso, utilizzare `clemrtl_interrupt()`.
- ▶ Per recuperare informazioni sull'ultimo errore, utilizzare `clemrtl_getErrorDetail()`. Per ricevere messaggi di errore e altri messaggi diagnostici al loro arrivo, utilizzare `clemrtl_setReportHandler()`.

La libreria è stata progettata per l'utilizzo a thread multipli alle seguenti condizioni:

- Non è possibile chiamare contemporaneamente funzioni globali, ovvero quelle che non assumono un handle di immagine come argomento.
- Non è possibile chiamare contemporaneamente funzioni locali di immagine, ovvero quelle che assumono un handle di immagine come argomento, quando sono applicate allo stesso handle di immagine, con l'eccezione di `clemrtl_interrupt()`.
- È possibile chiamare contemporaneamente funzioni locali di immagine quando sono applicate a handle di immagine diversi.

In particolare, questo significa che thread separati possono preparare ed eseguire immagini contemporaneamente, a condizione che utilizzino handle di immagine diversi. Handle di immagine diversi possono fare riferimento alla stessa immagine (`clemrtl_openImage()` crea un nuovo handle ogni volta che viene chiamato, anche per la stessa immagine), tuttavia è necessario ricordarsi di modificare i parametri dell'immagine quando si eseguono istanze multiple della stessa immagine per reindirizzare l'output in base alle specifiche esigenze. Se due immagini eseguite contemporaneamente tentano di scrivere entrambe nello stesso file o tabella di database di output, i risultati saranno imprevedibili.

Funzioni API

Di seguito è riportato l'elenco completo delle funzioni esposte nell'API.

initialise

Questa funzione è obsoleta ed è equivalente a

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

I nuovi programmi devono chiamare *initialise_ext*.

initialise_ext

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

Inizializza IBM® SPSS® Modeler Solution Publisher Runtime. È necessario chiamare questa funzione prima di qualsiasi altra funzione API.

Parametro	Descrizione
flag	Controlla alcuni aspetti del processo di inizializzazione. Il valore è costruito come un OR esclusivo bit per bit dei flag descritti di seguito.
arg_count	Numero di argomenti di inizializzazione aggiuntivi. Deve essere minore o uguale alla lunghezza della matrice args.
args	Argomenti di inizializzazione aggiuntivi. Gli argomenti sono descritti per tipo: typedef struct _clemrtl_init_arg { const char* name; const char* value; } clemrtl_init_arg;

Il campo name è il nome di una proprietà di configurazione e il campo value è il valore corrispondente. Sono riconosciute le seguenti proprietà di configurazione:

Nome	Valore
directory_installazione	Percorso completo alla cartella di installazione di IBM® SPSS® Modeler Solution Publisher. La cartella di installazione è quella contenente le cartelle bin e config.

Restituisce i seguenti codici di errore:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Inizializzazione non riuscita, senza ulteriori dettagli disponibili.

Il parametro flag fornisce il controllo di determinati aspetti del processo di inizializzazione. Il valore 0 (o CLEMRTL_INIT_DEFAULTS) specifica il comportamento di default, come indicato di seguito:

- Impostare il fuso orario di default chiamando tzset().
- Impostare le impostazioni internazionali LC_NUMERIC su "C".
- Impostare il nuovo handler globale per generare un'eccezione quando la memoria è esaurita.

È possibile modificare questo comportamento specificando combinazioni dei seguenti flag:

Flag	Descrizione
CLEMRTL_INIT_NO_TZ	Non impostare il fuso orario.
CLEMRTL_INIT_NO_LOCALE	Non modificare le impostazioni internazionali.
CLEMRTL_INIT_NO_NEW_HANDLER	Non impostare il nuovo handler.
CLEMRTL_INIT_LOCAL_NEW_HANDLER	Localizzare il nuovo handler in ogni chiamata API.

Per convertire correttamente tra numeri e stringhe, è necessario impostare le impostazioni internazionali LC_NUMERIC su "C" (il formato di numero di IBM® SPSS® Modeler utilizza le regole delle impostazioni internazionali "C"). È inclusa la conversione di numeri letti da e scritti in file di testo.

- Specificare `NO_LOCALE` se l'applicazione dipende dalle impostazioni internazionali `LC_NUMERIC` e non richiede conversione tra numeri e stringhe.
- `NO_NEW_HANDLER` e `LOCAL_NEW_HANDLER` si escludono a vicenda. È necessario che `new` generi un'eccezione quando la memoria è esaurita.
- Specificare `NO_NEW_HANDLER` se l'applicazione imposta il proprio nuovo handler oppure se si è certi che il compilatore/sistema di runtime generi un'eccezione per default.

Specificare `LOCAL_NEW_HANDLER` se l'applicazione necessita che `new` restituisca un puntatore nullo quando la memoria è esaurita. *Nota:* questo utilizzo non è sicuro in un ambiente a thread multipli.

getOption

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

Recupera il valore di un'opzione di esecuzione.

Parametro	Descrizione
nome	Nome dell'opzione.
valore	Buffer che riceve il valore dell'opzione.
dimensione_valore	Dimensione del buffer del valore.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se il nome dell'opzione non è uno dei nomi di opzione di esecuzione elencati in precedenza, per esempio `utilizzo_memoria`, oppure se il buffer dei valori non è sufficientemente grande da ricevere il valore, incluso un carattere di terminazione nullo. I valori delle opzioni sono sempre stringhe, anche quando l'interpretazione è numerica. [Per ulteriori informazioni, vedere l'argomento Esecuzione di stream pubblicati in il capitolo 3 a pag. 10.](#)

setOption

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

Imposta il valore di un'opzione di esecuzione.

Parametro	Descrizione
nome	Nome dell'opzione.
valore	Valore dell'opzione.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se il nome dell'opzione non è uno dei nomi di opzione di esecuzione elencati in precedenza, per esempio utilizzo_memoria. I valori delle opzioni sono sempre stringhe, anche quando l'interpretazione è numerica.

setLogFile

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

Reindirizza i messaggi di registro da tutte le immagini aperte successivamente.

Parametro	Descrizione
Modalità	La modalità di registrazione deve essere uno dei valori riportati di seguito: CLEMRTL_NULL_LOG — sopprime messaggi. CLEMRTL_STDERROR_LOG — invia messaggi all'output degli errori standard. CLEMRTL_FILE_LOG — invia messaggi al file denominato dal parametro file_registro.
file_registro	Stringa che contiene il nome file.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se la modalità non è una delle modalità riconosciute oppure se è necessaria la registrazione in un file e il nome file non è stato specificato oppure non è valido.

openImage

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

Apri l'immagine di uno stream pubblicato.

Parametro	Descrizione
file_immagine	Il nome file dell'immagine.
file_parametri	Il nome file dei parametri.
handle	Riceve l'handle di immagine.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

Il file dei parametri può essere NULL se non è richiesto un file dei parametri separato. L'operazione non riesce se non si specifica il file di immagine, se il nome file non è valido oppure se il contenuto non è leggibile. Se l'operazione riesce, l'handle di immagine restituito può essere utilizzato per identificare l'istanza dell'immagine nelle successive chiamate API.

closeImage

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

Chiude un handle di immagine e ne libera le risorse.

Parametro	Descrizione
handle	L'handle di immagine.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se è in corso un'altra chiamata API sull'immagine, in particolare se l'immagine è ancora in esecuzione. Se l'operazione riesce, l'immagine viene chiusa e non è possibile utilizzare l'handle in nessun'altra chiamata API successiva, eccetto che in un'altra chiamata di `clemrtl_closeImage()`, che non produce effetti.

enumerateParameters

```
int clemrtl_enumerateParameters(
    clemrtl_image_handle handle,
```



```
clemrtl_parameter_proc proc,
void* data);
```

Applica una procedura di richiamata a ogni nome e valore di parametro di immagine.

Parametro	Descrizione
handle	L'handle di immagine.
proc	La procedura di richiamata dei parametri.
dati	Dati definiti dall'utente per la richiamata.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

Di seguito è riportato il tipo della procedura di richiamata:

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

Parametro	Descrizione
dati	Dati definiti dall'utente passati a clemrtl_enumerateParameters().
nome	Nome del parametro.
valore	Valore del parametro.

La procedura viene applicata esattamente una sola volta a ogni parametro di immagine in ordine arbitrario.

getParameter

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

Recupera il valore di un parametro di immagine.

Parametro	Descrizione
handle	L'handle di immagine.
nome	Nome del parametro.
valore	Buffer che riceve il valore del parametro.
dimensione_valore	Dimensione del buffer del valore.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se il nome di parametro non corrisponde al nome di qualsiasi parametro dell'immagine oppure se il buffer del valore non è sufficientemente grande da ricevere il valore, incluso un carattere di terminazione nullo. I nomi di parametro utilizzano il formato `nome.attributo` descritto in precedenza, per esempio `file0.nome` e i valori di parametro sono sempre stringhe, anche se l'interpretazione è numerica.

setParameter

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

Imposta il valore di un parametro di immagine.

Parametro	Descrizione
handle	L'handle di immagine.
nome	Nome del parametro.
valore	Valore del parametro.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se l'handle di immagine non è valido oppure se il nome di parametro non corrisponde al nome di qualsiasi parametro dell'immagine. I nomi di parametro utilizzano il formato `nome.attributo` descritto in precedenza, per esempio `file0.nome` e i valori di parametro sono sempre stringhe, anche se l'interpretazione è numerica.

getFieldCount

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

Restituisce il numero di campi in una sorgente di input o destinazione di output.

Parametro	Descrizione
handle	L'handle di immagine.
chiave	Nome dell'input o dell'output da esaminare, come utilizzato nel file di parametri. La chiave può fare riferimento a un file o a un database.
field_count	Riceve il numero di campi.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

getFieldTypes

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

Restituisce i tipi di campo per una sorgente di input o destinazione di output.

Parametro	Descrizione
handle	L'handle di immagine.
chiave	Nome dell'input o dell'output da esaminare, come utilizzato nel file di parametri. La chiave può fare riferimento a un file o a un database.
field_count	Numero di campi da esaminare. Deve essere minore o uguale alla lunghezza della matrice field_types.
field_types	Matrice di lunghezza almeno pari a field_count che riceve i tipi di campo. Nella matrice vengono copiati i tipi dei primi campi field_count dell'input o dell'output. Se field_count è maggiore del numero effettivo di campi, gli elementi aggiuntivi della matrice vengono lasciati non definiti. I valori dei tipi di campo vengono elencati nella tabella dei tipi di dati riportata di seguito.

Tipi di dati

I field_types devono essere uno dei seguenti tipi:

Tipo	Interpretazione	Dichiaratore 'C' tipico
STRING	Stringa di caratteri con terminazione NULL UTF-8.	const char*
INTEGER	Numero intero firmato a 32 bit.	int
LONG	Numero intero firmato a 64 bit.	long long

Tipo	Interpretazione	Dichiaratore 'C' tipico
REAL	Numero a virgola mobile a 64 bit.	double
ORA	Numero intero firmato a 64 bit (secondi da mezzanotte).	long long
DATE	Numero intero firmato a 64 bit (secondi dalla mezzanotte del 01/01/1970).	long long
TIMESTAMP	Numero intero firmato a 64 bit (secondi dalla mezzanotte del 01/01/1970).	long long

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

setAlternativeInput

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

Sostituisce la sorgente di input di un file con una sorgente di input alternativa.

Parametro	Descrizione
handle	L'handle di immagine.
chiave	Nome della sorgente di input originale, come utilizzata nel file di parametri. La sorgente di input deve essere un file cosicché la chiave avrà sempre la forma "fileN" per alcuni numeri interi $N \geq 0$.
field_count	Numero di campi nell'input. Il valore deve corrispondere esattamente al numero di campi nell'input del file originale. In caso contrario la chiamata non riuscirà.
field_types	Matrice di tipi di lunghezza pari almeno a field_count. Il valore field_types[i] è il tipo dell'i-esimo campo e deve essere uno dei valori elencati nella tabella dei tipi di dati riportata di seguito. I tipi devono essere compatibili con quelli dell'input del file originale. In caso contrario, la chiamata non riuscirà.

Parametro	Descrizione
iterator	<p>Funzione che produce dati di input alternativi. Si applica ai rispettivi argomenti come indicato di seguito:</p> <pre>void** row = iterator(arg);</pre> <p>La funzione viene chiamata durante l'esecuzione (all'interno di una chiamata a <code>clemrtl_execute</code>) ed è chiamata una volta per ogni record di input. Un valore restituito NULL indica la fine dell'input. In questo caso, la funzione non viene più chiamata e l'esecuzione verrà terminata. In caso contrario, il risultato è una matrice di dati di lunghezza almeno pari a <code>field_count</code>, in cui <code>row[i]</code> fornisce il valore dell'i-esimo campo. Un valore può essere NULL. In caso contrario, deve essere un puntatore a un dato il cui tipo è determinato dal corrispondente <code>field_types[i]</code>. I puntatori devono rimanere validi fino alla chiamata successiva dell'iteratore o fino alla fine dell'esecuzione se l'esecuzione termina prematuramente.</p>
arg	Argomento opaco che viene passato all'iteratore a ogni chiamata.

Tipi di dati

I `field_types` devono essere uno di questi tipi. [Per ulteriori informazioni, vedere l'argomento `getFieldTypes` a pag. 29.](#)

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se i tipi di campo non sono compatibili con quelli dell'input del file originale oppure se il conteggio dei campi non corrisponde esattamente al numero di campi nell'input del file originale.

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);
```

Sostituisce la destinazione output di un file con una destinazione output alternativa.

Parametro	Descrizione
handle	L'handle di immagine.
chiave	Nome della destinazione di output originale, come utilizzata nel file di parametri. La destinazione di output deve essere un file cosicché la chiave avrà sempre la forma " <i>fileN</i> " per alcuni numeri interi $N \geq 0$.
field_count	Numero di campi nell'output. Il valore deve corrispondere esattamente al numero di campi nell'output del file originale. In caso contrario la chiamata non riuscirà.
field_types	Matrice di tipi di lunghezza pari almeno a field_count. Il valore field_types[i] è il tipo dell'i-esimo campo e deve essere uno dei valori elencati nella tabella dei tipi di dati riportata di seguito. I tipi devono essere compatibili con quelli dell'output del file originale. In caso contrario, la chiamata non riuscirà.
iterator	<p>Funzione che consuma l'output dell'immagine. Si applica al rispettivo argomento e a una riga di dati come indicato di seguito:</p> <pre>iterator(arg, row);</pre> <p>La funzione viene chiamata durante l'esecuzione (all'interno di una chiamata a clemrtl_execute) ed è chiamata una volta per ogni riga di risultati prodotta dall'immagine. Un valore row di NULL indica la fine dell'output, dopo il quale la funzione non viene più chiamata. Un'applicazione non dovrebbe basarsi su questa chiamata finale e dovrebbe chiudere eventuali risorse esterne quando l'esecuzione viene terminata. In caso contrario, row è una matrice di dati di lunghezza almeno pari a field_count, in cui row[i] fornisce il valore dell'i-esimo risultato. Un valore può essere NULL. In caso contrario, deve essere un puntatore a un dato il cui tipo è determinato dal corrispondente field_types[i]. La funzione deve copiare gli eventuali valori di dati di cui ha bisogno perché la memoria potrebbe non venire conservata dopo che la chiamata viene restituita.</p>
arg	Argomento opaco che viene passato all'iteratore a ogni chiamata.

Tipi di dati

I field_types devono essere uno di questi tipi. [Per ulteriori informazioni, vedere l'argomento getFieldTypes a pag. 29.](#)

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_FAIL	Operazione non riuscita, senza ulteriori dettagli disponibili.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se i tipi di campo non sono compatibili con quelli dell'output del file originale oppure il conteggio dei campi non corrisponde esattamente al numero di campi nell'output del file originale.

execute

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

Esegue un'immagine.

Parametro	Descrizione
handle	L'handle di immagine.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

Se l'immagine non è stata preparata, viene preparata inizialmente utilizzando i valori di parametro correnti. L'operazione non riesce se l'handle di immagine non è valido oppure se si verifica un errore durante la preparazione o l'esecuzione. La chiamata non restituisce finché l'esecuzione non è completa.

prepare

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

Prepara un'immagine per l'esecuzione.

Parametro	Descrizione
handle	L'handle di immagine.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non riesce se l'handle di immagine non è valido oppure se si verifica un errore durante la preparazione.

È necessario preparare un'immagine prima della sua esecuzione. Durante la preparazione, i valori di parametro dell'immagine vengono congelati. Se non è stata già preparata, `clemrtl_execute()` prepara un'immagine automaticamente in ogni chiamata. L'utilizzo di `clemrtl_prepare()` per preparare un'immagine è utile se l'immagine deve essere eseguita più volte con gli stessi valori di parametro e può migliorare sensibilmente le performance. Dopo che un'immagine è stata preparata, le successive modifiche apportate ai valori di parametro

vengono ignorate. Utilizzare nuovamente `clemrtl_prepare()` per aggiornare l'immagine con i nuovi valori di parametro.

interrupt

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

Termina l'esecuzione in corso di un'immagine.

Parametro	Descrizione
handle	L'handle di immagine.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

L'operazione non ha effetto se l'handle di immagine non è valido oppure se l'immagine non è in esecuzione.

Questa funzione è sicura per chiamare contemporaneamente con un'altra chiamata API sullo stesso handle di immagine.

getErrorDetail

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

Restituisce informazioni dettagliate sull'ultimo errore che si è verificato in un'immagine.

Parametro	Descrizione
handle	L'handle di immagine.
severità	Riceve il codice di severità come carattere singolo: I—information (informazione) W—warning (avviso) E—error (errore) X—system error (errore di sistema)
code	Riceve il numero di errore.
testo	Buffer che riceve il testo del messaggio.
dimensione_testo	Dimensione del buffer del testo.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

I risultati saranno inaffidabili se l'handle di immagine non è valido. Per adattarlo al buffer del testo, il testo del messaggio viene troncato all'occorrenza.

Se una chiamata API non riesce con CLEMRTL_ERROR e l'errore non è correlato a un handle di immagine specifico, passando 0 come handle di immagine verranno recuperati i dettagli dell'ultimo errore non specifico di un'immagine, ma il risultato non è affidabile in ambienti a thread multipli.

setReportHandler

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

Installa una procedura di richiamata di report per un'immagine.

Parametro	Descrizione
handle	L'handle di immagine.
proc	La procedura di richiamata del report.
dati	Dati definiti dall'utente per la richiamata.

Restituisce uno dei seguenti codici di stato:

Risultato	Descrizione
CLEMRTL_OK	Operazione riuscita.
CLEMRTL_ERROR	Operazione non riuscita, con ulteriori dettagli disponibili.

Di seguito è riportato il tipo della procedura di richiamata:

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

Parametro	Descrizione
dati	Dati definiti dall'utente passati a clemrtl_setReportHandler().
severità	Il codice di severità come carattere singolo: I—information (informazione) W—warning (avviso) E—error (errore) X—system error (errore di sistema)

Parametro	Descrizione
code	Il numero del messaggio.
testo	Il testo del messaggio.

La procedura è applicata a ogni messaggio al suo arrivo. Impostando la procedura su NULL, qualsiasi handler esistente installato nell'immagine viene rimosso.

Note

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.

Il seguente paragrafo non si applica per il Regno Unito o altri paesi in cui le presenti disposizioni non sono conformi alle leggi locali: SPSS INC., UNA SOCIETÀ IBM, FORNISCE IL PRESENTE DOCUMENTO “COSÌ COM'È” SENZA GARANZIA DI ALCUN TIPO, ESPRESSA O IMPLICITA, COMPRESE, TRA LE ALTRE, LE GARANZIE IMPLICITE DI NON VIOLAZIONE DEI DIRITTI ALTRUI, COMMERCIALIZZABILITÀ O IDONEITÀ PER UNO SCOPO SPECIFICO. Alcuni stati non consentono limitazioni di garanzie espresse o implicite in determinate transazioni, pertanto quanto sopra potrebbe non essere applicabile.

Le presenti informazioni possono includere imprecisioni tecniche o errori tipografici. Le modifiche periodiche apportate alle informazioni contenute in questa pubblicazione verranno inserite nelle nuove edizioni della pubblicazione. SPSS Inc. può apportare miglioramenti e/o modifiche al/ai prodotto/i e/o al/ai programma/i descritti nella presente pubblicazione in qualsiasi momento senza preavviso.

Eventuali riferimenti contenuti nelle presenti informazioni a siti Web non SPSS e non IBM vengono forniti per comodità dell'utente e non implicano in alcun modo l'approvazione di tali siti Web. I materiali presenti in tali siti Web non sono parte dei materiali di questo prodotto SPSS Inc. e l'utilizzo di tali siti Web è a rischio esclusivo dell'utente.

Al momento dell'invio di informazioni a IBM o SPSS, l'utente concede a IBM e SPSS un diritto non esclusivo a utilizzare o distribuire tali informazioni nel modo che ritiene appropriato senza incorrere in alcun obbligo nei confronti dell'utente.

Le informazioni relative ai prodotti non SPSS provengono dai fornitori di tali prodotti, dagli annunci pubblicati o da altre fonti pubbliche disponibili. SPSS non ha testato tali prodotti e non può confermare l'accuratezza delle dichiarazioni relative alle prestazioni, alla compatibilità o ad altre caratteristiche relative ai prodotti non SPSS. Le domande relative alle funzionalità dei prodotti non SPSS devono essere indirizzate ai fornitori di tali prodotti.

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.

Le presenti informazioni includono esempi di dati e report utilizzati in operazioni aziendali quotidiane. Per fornire una descrizione il più possibile esaustiva, gli esempi includono nomi di persone, società, marchi e prodotti. Tutti questi nomi sono fittizi e ogni somiglianza a nomi e indirizzi utilizzati da aziende reali è puramente casuale.

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

Marchi

IBM, il logo IBM e [ibm.com](http://www.ibm.com) sono marchi di IBM Corporation, registrati in numerose giurisdizioni nel mondo. Un elenco aggiornato dei marchi IBM è disponibile sul Web all'indirizzo <http://www.ibm.com/legal/copytrade.shtml>.

SPSS è un marchio di SPSS Inc., una società IBM, registrati in numerose giurisdizioni nel mondo.

Adobe, il logo Adobe, PostScript e il logo PostScript sono marchi o marchi registrati di Adobe Systems Incorporated negli Stati Uniti e/o in altri paesi.

IT Infrastructure Library è un marchio registrato della Central Computer and Telecommunications Agency ora parte dell'Office of Government Commerce.

Intel, il logo Intel, Intel Inside, il logo Intel Inside, Intel Centrino, il logo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium e Pentium sono marchi o marchi registrati di Intel Corporation o delle sue consociate negli Stati Uniti e in altri paesi.

Linux è un marchio registrato di Linus Torvalds negli Stati Uniti e/o negli altri paesi.

Microsoft, Windows, Windows NT e il logo Windows sono marchi di Microsoft Corporation negli Stati Uniti e/o negli altri paesi.

ITIL è un marchio registrato e un marchio comunitario registrato dell'Office of Government Commerce ed è registrato presso l'Ufficio Brevetti degli Stati Uniti.

UNIX è un marchio registrato di The Open Group negli Stati Uniti e in altri paesi.

Cell Broadband Engine è un marchio di Sony Computer Entertainment, Inc. negli Stati Uniti e/o in altri paesi ed è utilizzato su licenza.

Java e tutti i marchi e i logo basati su Java sono marchi di Sun Microsystems, Inc. negli Stati Uniti e/o negli altri paesi.

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.

I nomi di altri prodotti o servizi possono essere marchi di IBM, SPSS o altre società.

- apertura
 - stream, 25
- API
 - cenni generali, 12, 21
 - codici di errore, 21
 - codici di stato, 21
 - conteggio campi, 28
 - input alternativo, 30
 - opzioni di esecuzione, 24
 - output alternativo, 31
 - riferimenti, 21
 - setReportHandler, 35
 - tipi di campo, 29
- chiusura
 - stream, 26
- closeImage
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 26
- codici di errore, 21
- codici di stato, 21
- deployment, 6
- directory temporanea
 - IBM SPSS Modeler Solution Publisher Runtime, 19
- disinstallazione
 - IBM SPSS Modeler Solution Publisher Runtime, 20
- documentazione, 2
- enumerateParameters
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 26
- esecuzione di stream
 - utilizzo di IBM SPSS Modeler Solution Publisher, 10–11
- esempi
 - cenni generali, 4
 - Guida alle applicazioni, 2
- esempi di applicazioni, 2
- execute
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 33
- file *.par*, 7
- file *.pim*, 7
- file di registro, 25
- funzioni API
 - closeImage, 21, 26
 - enumerateParameters, 21, 26
 - execute, 21, 33
 - getErrorDetail, 21, 34
 - getFieldCount, 21, 28
 - getFieldTypes, 21, 29
 - getOption, 21, 24
 - getParameter, 21, 27
 - inizializzazione, 21–22
 - 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
- fuso orario
 - impostazione tramite l'API, 22
- getErrorDetail
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 34
- getFieldCount
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 28
- getFieldTypes
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 29
- getOption
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 24
- getParameter
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 27
- IBM SPSS Modeler, 1
 - documentazione, 2
- IBM SPSS Modeler Solution Publisher, 6–7
 - cenni generali, 6
 - integrazione con altre applicazioni, 12
- IBM SPSS Modeler Solution Publisher Runtime
 - avvio, 19
 - directory temp, 19
 - disinstallazione, 20
 - Installazione, 15
 - requisiti di sistema, 15–16
- IBM SPSS Text Analytics, 2
- initialise
 - flag, 22
 - riferimenti alle funzioni API, 22
- initialise_ext
 - flag, 22
 - riferimenti alle funzioni API, 22
- initialize
 - panoramica del processo API, 21
- interrupt
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 34
- locale
 - impostazione tramite l'API, 22

-
- marchi, 38
 - nodi output, 6
 - note legali, 37
 - openImage
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 25
 - parameters
 - parametri per esecuzione di stream, 11
 - parametri
 - file di parametri API, 25–26
 - parametri di immagini API, 26–28
 - procedure di richiamata API, 26
 - prepare
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 33
 - pubblicazione di stream
 - IBM SPSS Modeler Solution Publisher, 6–7
 - Runtime
 - avvio, 19
 - directory temp, 19
 - disinstallazione, 20
 - installazione su computer UNIX, 16
 - installazione su computer Windows, 16
 - opzioni, 10
 - Runtime programming library (CLEMRTL), 12
 - segnalazione di errori
 - riferimenti alle funzioni API, 25, 34–35
 - setAlternativeInput
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 30
 - setAlternativeOutput
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 31
 - setLogFile
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 25
 - setOption
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 24
 - setParameter
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 28
 - setReportHandler
 - panoramica del processo API, 21
 - riferimenti alle funzioni API, 35
 - SPSS Modeler Server, 1
 - UNIX
 - installazione IBM SPSS Modeler Solution Publisher
 - Runtime, 16
 - Windows
 - installazione IBM SPSS Modeler Solution Publisher
 - Runtime, 15