EXTRATORES DE PROCESSO DO PPM

ARIS Process Performance Manager 10

Outubro 2026

This document applies to ARIS Process Performance Manager Version 10 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2000-2026 SAG Aris GmbH, Saarbrücken, Germany and/or its subsidiaries and/or its affiliates and/or their licensors.

The name ARIS and all SAG Aris GmbH product names are either trademarks or registered trademarks of SAG Aris GmbH and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

The Software may contain or be distributed with certain commercial third party components (“Third Party Components”), which may be subject to special license terms and conditions (“Third Party Terms”) located at https://aris.com/terms.

Third Party Terms include important licensing and warranty information and disclaimers of third party licensors. For the avoidance of doubt, if Customer uses the Software in accordance with the Documentation, the Third Party Terms do not restrict the rights granted under this Agreement. In the event that a third party manufacturer no longer provides active support for any Third Party Component, Supplier will use commercially reasonable efforts to respond to any issues with the Software related to the Third Party Components.

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with SAG Aris GmbH and/or its subsidiaries and/or its affiliates and/or their licensors.

Índice

1. Geral 6
2. XML 7
2.1. O que é o XML? 7
2.2. Estrutura de um documento XML 7
2.3. PPM e XML 8
3. Visão geral 9
3.1. Extração de dados 9
3.2. Importação de dados 10
4. PPM Process Extractor SAP-2-PPM 12
4.1. Arquitetura 12
4.2. Configuração do sistema R/3 12
4.3. Configuração da tabela R3 (especificação de evento do sistema) 15
4.3.1. Metadados globais 17
4.3.2. Classificar eventos de sistema 19
4.3.3. Adicionar valores de um campo 20
4.3.4. Extrair campos de valor múltiplos 22
4.3.5. Concatenar novos valores de atributo de valores de atributo extraídos 23
4.3.6. Conectar tabelas 26
4.3.6.1. Comparação de relações de chave estrangeira 27
4.3.7. Extração usando condições 28
4.3.7.1. Operadores de condição 29
4.3.7.2. Condições complexas 31
4.3.8. Extrair campos do banco de dados 32
4.3.9. Exemplo: Configuração XML e arquivo de saída 33
4.3.9.1. Configuração da tabela 33
4.3.9.2. Arquivo de saída XML (formato do evento de sistema do PPM) 37
4.3.10. Extração em bloco 39
4.3.10.1. Extração em bloco de grandes volumes de dados 40
4.3.11. Extrair documentos de alteração 42
4.3.11.1. Criar eventos do sistema de origem a partir de alterações individuais 42
4.3.11.2. Extrair eventos do sistema, incluindo alterações 44
4.3.11.3. Extrair o documento da primeira alteração de um campo de tabela 47
4.3.11.4. Extrair o documento da última alteração de um campo de tabela 50
4.3.11.5. Extrair o documento da última alteração de vários campos de tabela 50
4.3.12. Extrair a primeira ou última linha de uma classificação 52
4.3.13. Criar atributos com data invertida 55
4.3.14. Multiplicação de eventos do sistema em tabelas 57
4.3.15. Extrair campos ou tabelas individuais para análise de dados 59
4.3.15.1. Restringir extração de dados 61
4.3.16. Tabelas de extração com colunas de chave/valor 62
4.4. Programa de linha de comando 64
4.4.1. Argumentos do programa de linha de comando 64
4.4.1.1. Argumentos gerais 64
4.4.1.2. Argumentos específicos do sistema de origem 65
4.4.1.3. Argumentos específicos do arquivo de saída 68
4.4.1.4. Extração automática e contínua 69
4.5. Extrair de várias fontes de dados 70
4.6. Comunicação em rede segura SAP 71
4.6.1. Pré-requisitos 71
4.6.2. Configuração XML das fontes de dados SAP 71
5. PPM Process Extractor JDBC-2-PPM 73
5.1. Arquitetura 73
5.2. Configuração do sistema JDBC 73

5.3. Configuração da tabela JDBC 76
5.3.1. Configuração do acesso à tabela 79
5.3.2. Metadados globais 80
5.3.3. Classificar eventos de sistema 81
5.3.4. Concatenar novos valores de atributo de valores de atributo extraídos 82
5.3.5. Extração usando condições 84
5.3.5.1. Operadores de condição 85
5.3.5.2. Condições complexas 90
5.3.6. Extrair campos do banco de dados 91
5.3.6.1. Manipulação de valores NULL 92
5.3.6.2. Extrair campos de valor múltiplos 92
5.3.7. Configuração do acesso à tabela 94
5.3.8. Exemplo: Configuração XML e arquivo de saída 95
5.3.8.1. Configuração da tabela 95
5.3.8.2. Arquivo de saída XML (formato do evento de sistema do PPM) 99
5.3.9. Extração em bloco 102
5.3.10. Extrair a primeira ou última linha de uma classificação 102
5.3.11. Extrair a primeira ou última linha usando um carimbo de data/hora 105
5.3.12. Multiplicação de eventos do sistema em tabelas 107
5.3.13. Extrair campos ou tabelas individuais para análise de dados 109
5.3.13.1. Restringir extração de dados 111
5.3.14. Tabelas de extração com colunas de chave/valor 112
5.4. Programa de linha de comando 113
5.4.1. Argumentos do programa de linha de comando 113
5.4.1.1. Argumentos gerais 113
5.4.1.2. Argumentos específicos do banco de dados de origem 114
5.4.1.3. Argumentos específicos do arquivo de saída 117
5.4.1.4. Extração automática e contínua 117
5.5. Extrair de várias fontes de dados 119
5.6. Extrair dados do tipo de dados BIT 119
5.7. Apêndice 121
5.7.1. Tipos de dados suportados 121
5.7.2. Caracteres especiais e capitalização 122
5.7.2.1. Caso especial: Esquema e nome da tabela 122
6. PPM Process Extractor CSV-2-PPM 125
6.1. CSV 125
6.2. Arquitetura 125
6.3. Leitor de CSV e gerador de eventos do sistema CSV 126
6.3.1. Exemplo 1: Cabeçalho completo 126
6.3.2. Exemplo 2: Cabeçalho com cabeçalho da coluna ausente 127
6.3.3. Exemplo 3: Cabeçalho com vários cabeçalhos da coluna ausentes 128
6.3.4. Exemplo 4: Sem cabeçalho 128
6.4. Configuração CSV 129
6.4.1. Extensão de configuração CSV 132
6.5. Programa de linha de comando 133
6.5.1. Argumentos do programa de linha de comando 133
6.5.2. Argumentos gerais 133
6.5.3. Argumentos específicos do CSV 134
6.5.4. Argumentos específicos do arquivo de saída 135
6.6. Extrair de várias fontes de dados 136
7. Funcionalidades de todos os extratores 137
7.1. Transformação de atributos 137
7.1.1. Operações 139
7.1.1.1. unmask 140
7.1.1.2. trim 140
7.1.1.3. concat 141

7.1.1.4. substring 141
7.1.1.5. if_then_else 142
7.1.1.6. set_with_default 142
7.1.1.7. integer_plus 143
7.1.1.8. integer_minus 143
7.1.1.9. string_equal 143
7.1.1.10. string_in 144
7.1.1.11. boolean_not 144
7.1.1.12. boolean_and 145
7.1.1.13. boolean_or 145
7.1.1.14. exists 146
7.1.1.15. get_null 146
7.1.1.16. getCurrentTimestamp 146
7.1.2. Exemplo de transformação de atributos 147
7.2. Fonte de dados 148
7.2.1. Configuração de vários arquivos de saída 150
7.2.2. Configuração de uma extração de offset em termos de hora ou valor 152
7.3. Arquivo de saída XML (formatos) 153
7.3.1. Formato do evento de sistema PPM 153
7.3.2. Formato PIKIDATA 154
7.3.3. Formato DIMDATA 157
8. Pseudonimização de dados 161
8.1. Sistemas dimensionados 161
8.2. Configuração de atributos pseudonimizados 161
8.3. Dimensões dos dados pseudonimizados 162
8.4. Ferramenta de pseudonimização do PPM 163
8.5. Adicionar novos atributos criptografados 164
9. Suporte e informações legais 165
9.1. Escopo da documentação 165
9.2. Suporte 165
9.3. Legal notices 166

1. Geral

Este manual descreve a função dos extratores do processo PPM SAP-2-PPM, JDBC-2-PPM e CSV-2-PPM.

Os extratores do processo criam arquivos XML a partir de conjuntos de dados de diversos tipos de sistema de origem, que podem ser importados para o ARIS Process Performance Manager (ou PPM)

Observe que este manual não se destina a substituir o treinamento do usuário ou de personalização. É uma fonte de referência contendo informações que suplementam as informações fornecidas na ajuda online.

2. XML

Este capítulo contém informações básicas sobre XML, necessárias para entender os capítulos seguintes.

2.1. O que é o XML?

A abreviatura XML significa e**X**tensible **M**arkup **L**anguage. XML é uma metalinguagem para a descrição de idiomas de exibição, como HTML. As metalinguagens fornecem as regras necessárias para a definição de tipos de documento. Os idiomas de exibição permitem que os documentos sejam exibidos corretamente.

2.2. Estrutura de um documento XML

Um documento XML é sempre composto de dois tipos de caracteres: os dados reais e as chamadas marcas ou marcações. Marcas são instruções XML, que descrevem a divisão do documento em unidades de armazenamento e sua estrutura lógica. A própria estrutura é salva em uma definição de tipo de documento (DTD).

As marcas são sempre escritas em pares e em parênteses angulares. Cada marca inicial sempre tem uma marca final correspondente.

Os atributos XML são usados dentro das marcas. Um atributo só pode ocorrer uma vez dentro de uma marca.

Os documentos XML são compostos por elementos. Um elemento é composto de duas marcas XML e o texto incluído. Elementos em branco são compostos por apenas uma marca e sempre terminam com uma barra (/) antes do parêntese final.

Pode criar documentos XML simples com um editor de texto. No exemplo a seguir, a DTD é especificada entre colchetes no arquivo XML:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE memberlist
[
<!ELEMENT memberlist (no, name, age)>
<!ELEMENT no (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT age (#PCDATA)>
]>
<memberlist>
<no>001</no>
<name>Doe, John</name>
<age>27</age>
</memberlist>
```

Se salvar este documento com o nome de sua escolha e a extensão **.xml**, o Internet Explorer pode exibir o documento em uma forma estruturada.

2.3. PPM e XML

O PPM usa o XML como um formato de dados universal. Toda a configuração do sistema PPM é fornecida por arquivos XML.

Os extratores de processo do PPM extraem dados de diferentes sistemas de origem e os salvam em arquivos XML compatíveis com o PPM.

3. Visão geral

Este capítulo fornece uma visão geral da extração de dados de sistemas aplicativos para uso em ARIS Process Performance Manager. Os dados extraídos podem ser importados para o ARIS Process Performance Manager sem alterações usando a interface de importação XML do PPM e processados usando definições adequadas para fragmentos e mapeamento. A interface de importação XML é descrita em detalhe no manual **Importação de dados do PPM**.

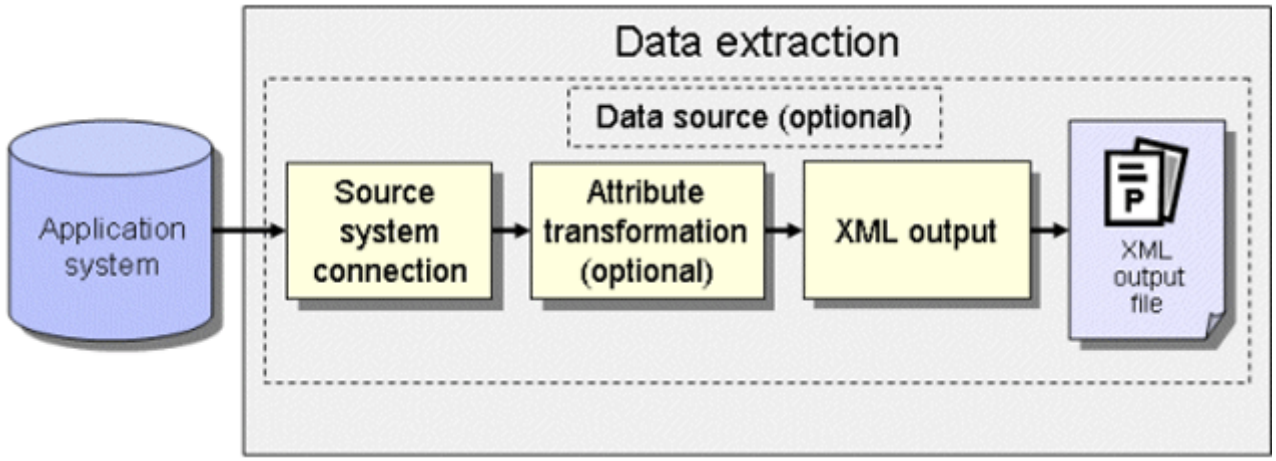
Os extratores de processo PPM CSV-2-PPM, SAP-2-PPM e JDBC-2-PPM são adaptados ao sistema de origem do qual devem ser extraídos dados, através de arquivos de configuração XML.

3.1. Extração de dados

Este capítulo fornece uma visão geral da extração de dados de sistemas aplicativos para uso em ARIS Process Performance Manager. Os dados extraídos podem ser importados para o ARIS Process Performance Manager sem alterações usando a interface de importação XML do PPM e processados usando definições adequadas para fragmentos e mapeamento. A interface de importação XML é descrita em detalhe no manual **Importação de dados do PPM**.

Os extratores de processo PPM CSV-2-PPM, JDBC-2-PPM e SAP-2-PPM são adaptados ao sistema de origem do qual devem ser extraídos dados, através de arquivos de configuração XML.

A seguinte representação geral ilustra o funcionamento básico da extração de dados dos sistemas de origem, a transformação de dados seguinte e a saída em arquivos de saída XML.



Conexão do sistema de origem

A conexão do sistema de origem é criada especificando uma configuração do sistema contendo os dados de acesso necessários para o sistema de origem a ser extraído, por exemplo, número do sistema, tipo de acesso e senha de acesso do usuário do sistema de origem.

Você precisa garantir que o usuário do sistema de origem especificado tenha autorização de acesso apropriada para extrair os campos de dados relevantes.

Transformação de atributos (opcional)

Se necessário, os atributos do sistema de origem podem ser modificados antes de serem importados para o sistema PPM e os tipos de atributos podem ser adicionados e transformados. É necessário criar uma configuração XML adequada para transformação de atributos.

Saída XML

Os dados extraídos são gravados em arquivos de saída, em um formato XML compatível com o PPM.

Fonte de dados (opcional)

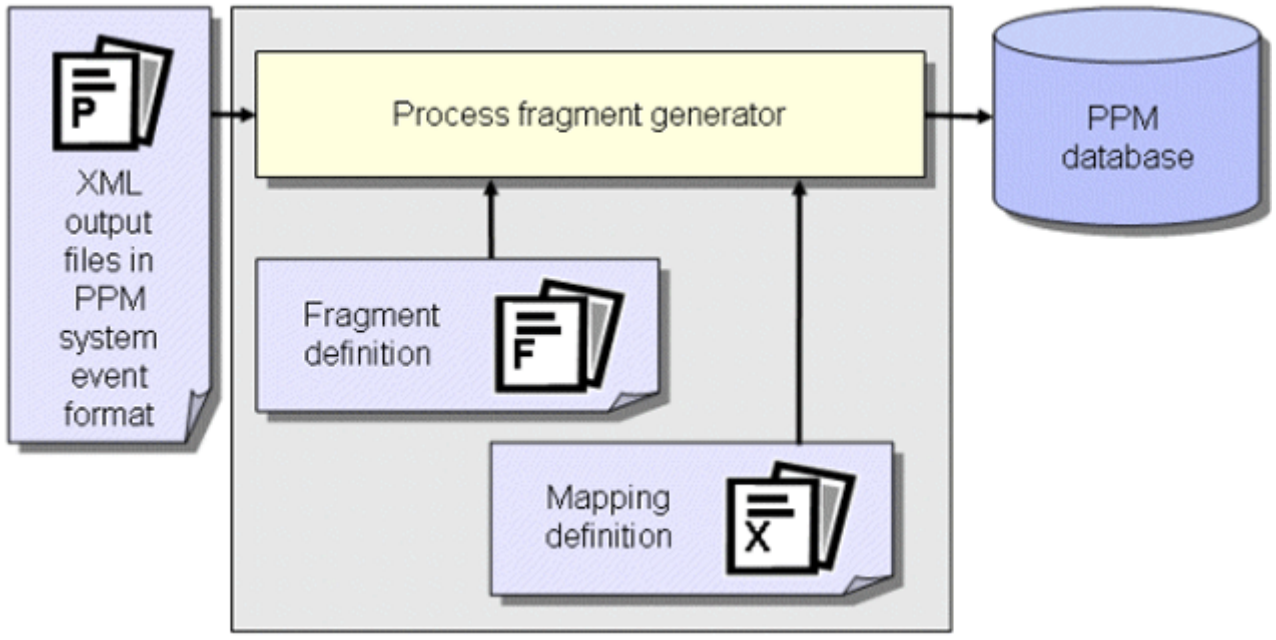
Todos os arquivos XML necessários para a extração, bem como o(s) arquivo(s) de saída, podem ser especificados em um arquivo XML.

3.2. Importação de dados

Além do arquivo de saída XML no formato de evento do sistema PPM, um arquivo no formato de medida independente da instância de processo ou no formato de dados da dimensão também pode ser impresso e importado para um sistema PPM. Apenas a importação dos arquivos de saída XML no formato de evento do sistema PPM é tratada aqui.

Cada evento de sistema no arquivo de saída XML gerado recebe uma definição de fragmento quando importado e é instanciado no banco de dados do PPM. Os atributos do sistema de origem especificados no mapeamento são copiados para os objetos nessa instância de fragmento. As instâncias de fragmento são então salvas no banco de dados do PPM.

Exemplo 1.



Em uma outra etapa de trabalho, as instâncias de fragmento importadas para o banco de dados são compiladas em instâncias de processo usando o comando **runppmimport** do PPM e associadas a tipos de processo. Após o cálculo da medida, as instâncias do processo estão disponíveis para análises detalhadas.

Dica

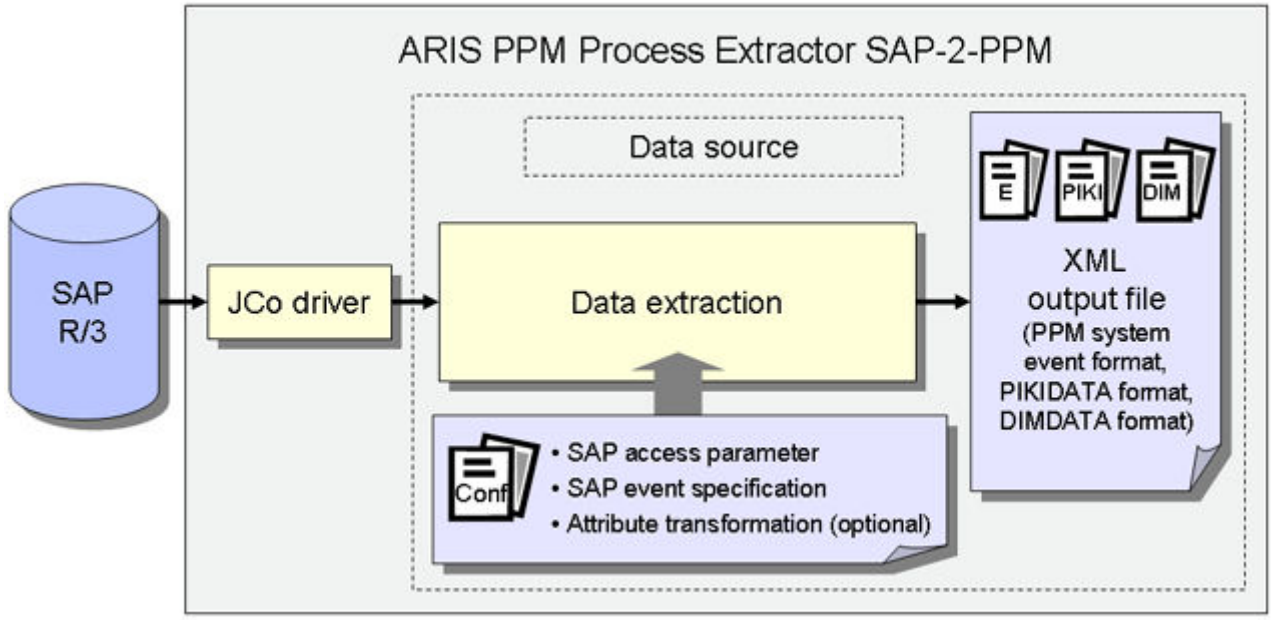
- Para mais informações sobre a configuração da importação XML, consulte o manual **Importação de dados do PPM**.
- A configuração de todos os arquivos relevantes para o **runppmimport** é descrita em detalhe no manual **Personalização do PPM**.

4. PPM Process Extractor SAP-2-PPM

Este capítulo fornece uma visão geral da arquitetura, funcionamento e configuração do PPM Process Extractor SAP-2-PPM.

4.1. Arquitetura

A figura abaixo ilustra a funcionalidade do PPM Process Extractor SAP-2-PPM:



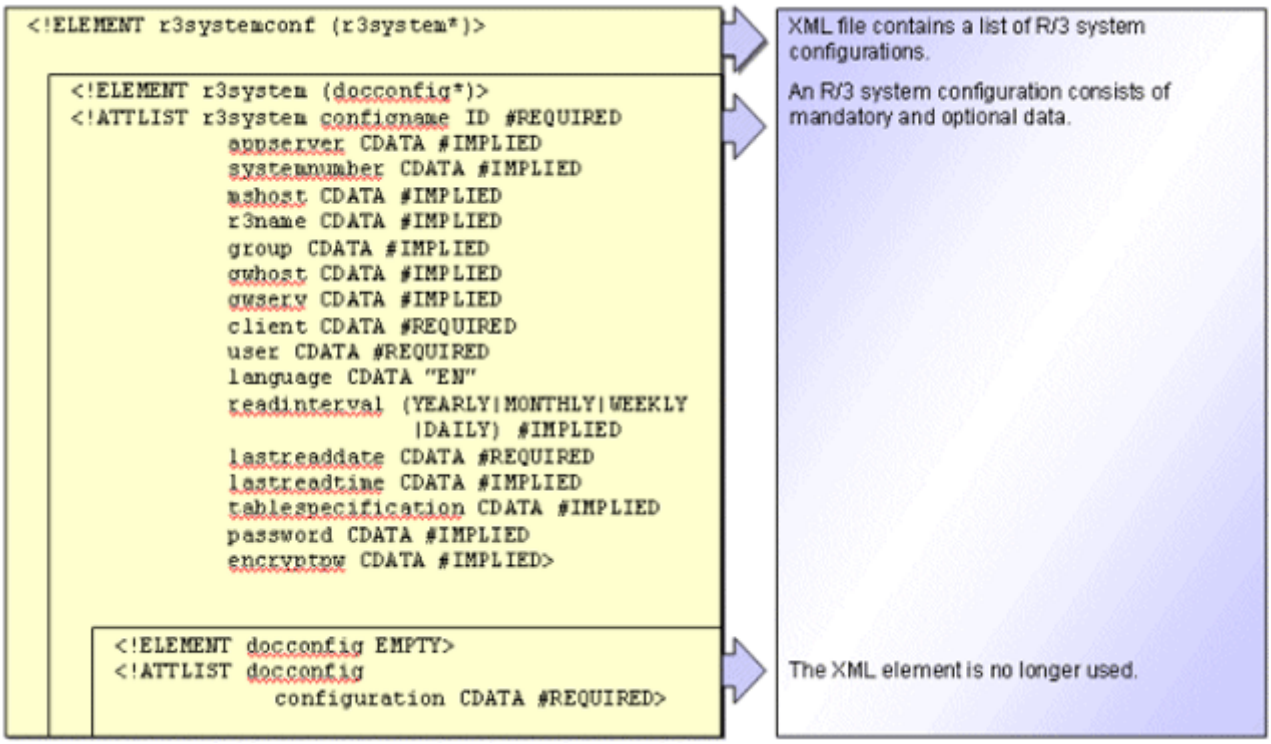
O driver do Java Connector (JCo) é usado para criar uma conexão ao sistema SAP R/3 com os dados de acesso para configuração do sistema R/3. Os dados são então extraídos das tabelas no sistema R/3 com a configuração da tabela R/3 e uma transformação de atributo opcional usando o módulo extração de dados [9] e são gravados em arquivos XML, em um formato de saída compatível com o PPM.

É preciso garantir que o usuário do sistema de origem especificado tenha autorização de acesso apropriada para extrair os campos de dados relevantes e que os requisitos para o sistema de origem R/3 sejam atendidos. Se você tiver alguma dúvida ou preocupação, entre em contato com o administrador do SAP.

4.2. Configuração do sistema R/3

Os dados de acesso ao sistema R/3 são especificados em um arquivo XML. O nome desse arquivo XML é transferido para a ferramenta de linha de comando como um argumento.

O formato do arquivo XML é especificado pela seguinte DTD:



Nome de atributo XML	Descrição	Exemplo
configname	Nome único da configuração	ides_doc_vdap
appserver	Nome do computador ou endereço IP do computador do sistema de origem	172.20.210.21
systemnumber	Número do sistema SAP	00
mshost	Nome do host de mensagens SAP	/H/154.37.28.6/S/4435
r3name	Nome do sistema R/3	A20
group	Nome do grupo de servidores de aplicativos	A20_ALL
gwhost	Nome de computador ou gateway R/3	172.20.210.32
gwserv	Número de serviço do gateway R/3	6789
client	Nome do cliente SAP	100
user	Nome do usuário de sistema SAP	sapuser
language	Idioma dos campos de texto extraídos	DE
readinterval	Já não é usada. Ainda incluído por motivos de compatibilidade.	YEARLY
lastreaddate	Data a partir da qual os dados são extraídos. Não aplicável ao usar uma fonte de dados [148], formato: **hhmmss**	19700101
lastreadtime	Hora a partir da qual os dados são extraídos. Não aplicável ao usar uma fonte de dados [148], formato: **hhmmss**	125708
tablespecification	Já não é usada. Ainda incluído por motivos de compatibilidade.	
password	Senha de acesso para o sistema R/3	mypassword
encryptpw	Senha de acesso criptografada para o sistema R/3	

Na primeira vez que inserir um sistema R/3 na configuração do sistema ou alterar a senha, especifique a senha no atributo XML **password**. Da primeira vez que o sistema R/3 é acessado (por exemplo, teste de

conexão, iniciar extração), a senha é criptografada e salva no atributo XML **encryptpw**. O valor do atributo XML **password** é excluído.

O tipo de acesso a um sistema R/3 é determinado especificando o valor dos atributos XML.

Tipo de sistema R/3	Atributos XML	Exemplo
Servidor de aplicativos	appserver systemnumber	... configname="..." appserver="192.168.102.100" systemnumber="01" mshost="" r3name="" group="" gwhost="" gwserv="" ...
Host de mensagens	mshost r3name group	... configname="test_group_sd" appserver="..." systemnumber="03" mshost="/H/156.243.123.6/H/154.34.37.2/S/4435/H/saproutertest/H/112.4.3.2" r3name="A20" group="A20_ALL" gwhost="" gwserv="" ...

Tipo de sistema R/3	Atributos XML	Exemplo
Servidor gateway	appserver systemnumber gwhost gwserv	... configname="..." appserver="172.20.212.232" systemnumber="01" mshost="" r3name="" group="" gwhost="172.20.212.240" gwserv="6789" ...

```
...
<r3system configname="ides_doc_vbap" appserver="172.20.210.211"
systemnumber="00" client="100" user="sapuser"
language="DE" readinterval="YEARLY"
lastreaddate="19700101" password="testpassword">
<docconfig configuration="SD_VBAP"/>
<docconfig configuration="SD_MSEG"/>
</r3system>
...
```

4.3. Configuração da tabela R3 (especificação de evento do sistema)

A configuração da tabela R/3 especifica quais campos da tabela são extraídos do sistema R/3 e gravados nos eventos de sistema como atributos do sistema de origem. Pode salvar várias configurações de tabela com nomes únicos no arquivo XML.

A configuração da tabela R/3 consiste nos seguintes componentes:

Tabelas globais

As tabelas globais são usadas para extrair informações gravadas para todos os eventos de sistema.

Tabelas de chave estrangeira

O elemento XML **docreftable** contém o nome da tabela de chave estrangeira. Especifica como o intervalo de dados a ser extraído da tabela do evento de sistema é limitado. Os campos de chave primária especificados no elemento XML **pkfield** vinculam a tabela de chave estrangeira à tabela do evento de sistema e a outras tabelas de chave estrangeira.

Tabela do evento de sistema

O elemento XML **doctable** contém o nome da tabela do evento de sistema. Especifica os documentos a serem extraídos para um fluxo de documentos. Cada registro de dados extraído da tabela de eventos do sistema gera um evento do sistema no arquivo de saída (elemento XML **event**).

Tabelas de dados

As informações na tabela de eventos do sistema podem ser complementadas pela extração de campos de dados adicionais de quaisquer outras tabelas de dados (por exemplo, o número do material é extraído da tabela de eventos do sistema e o texto descritivo relacionado com esse número é extraído de uma tabela de dados).

A seguinte estrutura de arquivo XML ilustra a configuração das tabelas das quais os dados devem ser extraídos. O capítulo sobre **configuração do acesso à tabela [94]** descreve que elementos e atributos XML são opcionais.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE r3systemconffields SYSTEM 'xmlextractor_tableconfiguration.dtd'>

<xmlextractor_tableconfiguration>
<configuration name="..." printname="..." classtouse="...">
<globaltable name="..." tablename="..." classtouse="...">
<fieldtoread name="...">
<textref tablename="..." reffieldname="..."
textfieldname="..." langfieldname="..."/>
</fieldtoread>
...
</globaltable>
<docspec>
<docreftable name="..." tablename="..."
classtouse="...">
<condition fieldname="..."
logicaloperator="..." >
<value>...</value>
</condition>
<pkfield name="..." fktablename="..."
fkfieldname="..." logicaloperator="...">
<fkpart readfrom="..." startposition="..."
length="..."/>
<prefix>
<value>...</value>
</prefix>
<postfix>
<value>...</value>
</postfix>
</pkfield>
...
</docreftable>
...
<doctable name="..." tablename="..."
classtouse="..." >
```

```
<condition fieldname="..." logicaloperator="..." >
<value>...</value>
</condition>
<pkfield name="..." fktablename="..."
fkfieldname="..." logicaloperator="...">
<fkpart readfrom="..." startposition="..."
length="..."/>
<prefix>
<value>...</value>
</prefix>
<postfix>
<value>...</value>
</postfix>
</pkfield>
...
<fieldtoread name="...">
<textref tablename="..." reffieldname="..."
textfieldname="..." langfieldname="..."/>
</fieldtoread>
...
</doctable>
</docspec>
<table name="..." tablename="..."
classtouse="...">
<condition fieldname="..." logicaloperator="...">
<value>...</value>
</condition>
<pkfield name="..." fktablename="..."
fkfieldname="..." logicaloperator="...">
<fkpart readfrom="..." startposition="..."
length="..."/>
<prefix>
<value>...</value>
</prefix>
<postfix>
<value>...</value>
</postfix>
</pkfield>
...
<fieldtoread name="...">
<textref tablename="..." reffieldname="..."
textfieldname="..." langfieldname="..."/>
</fieldtoread>
...
</table>
...
</configuration>
...
</xmlextractor_tableconfiguration>
```

4.3.1. Metadados globais

O elemento XML **globaltable** pode ser usado para transferir metadados dos extratores de processo PPM SAP-2-PPM e JDBC-2-PPM para o arquivo de saída como atributos do evento de sistema global.

Os dados típicos para atributos do evento de sistema global incluem os nomes das configurações que foram usadas para extrair os dados. Os próprios dados são determinados a partir dos valores de argumentos especificados na linha de comando. O valor padrão é determinado para argumentos que não são especificados na linha de comando.

A identificação de palavras-chave faz distinção entre maiúsculas e minúsculas.

Os atributos do evento de sistema global aplicam-se a todos os eventos de sistema no arquivo de saída atual.

Palavra-chave	Valor do argumento da linha de comando
SYSTEM_CONFIG_FILE_NAME	-systemconfig <filename> ...
SYSTEM_CONFIG_NAME	-systemconfig ... <configname>
TABLE_CONFIG_FILE_NAME	-tableconfig <filename> ...
TABLE_CONFIG_NAME	-tableconfig ... <configname>
BEGIN_DATE,	-begindate <dd.MM.yyyy>
BEGIN_TIME	-begintime <hh:mm:ss>
END_DATE	-enddate <dd.MM.yyyy>
END_TIME	-endtime <hh:mm:ss>
CPD	-cpd <int>
OUT_FILE_NAME	-outfile <file name>
PIKI_DATA_MAPPING_FILE_NAME	-pikidatamapping <filename>...
PIKI_DATA_MAPPING_NAME	-pikidatamapping ... <pcname>
DIM_DATA_MAPPING_FILE_NAME	-dimdatamapping <filename> ...
DIM_DATA_MAPPING_NAME	-dimdatamapping ... <dimname>

Se nenhum valor puder ser determinado para uma palavra-chave, a própria palavra-chave será gravada no arquivo de saída. Dessa forma pode transferir constantes para o arquivo de saída.

Exemplo 2.

A extração de arquivo abaixo mostra a configuração usada para criar atributos do evento de sistema global.

```
<globaltable name="INFO">
<fieldtoread name="SYSTEM_CONFIG_NAME"/>
</globaltable>
```

O nome de configuração especificado na linha de comando usando -systemconfig **<configname>**, por exemplo, **sapppm** é gravado no arquivo de saída como atributo do evento de sistema global **INFO-SYSTEM_CONFIG_NAME**:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<attribute type="INFO-SYSTEM_CONFIG_NAME ">
sapppm
</attribute>
...
<event>
```

```
...
</event>
...
</eventlist>
```

Ao especificar uma classe Java adequada no atributo XML **classtouse**, os dados também podem ser extraídos de qualquer outra tabela do sistema de origem.

4.3.2. Classificar eventos de sistema

Pode classificar os eventos de sistema na saída XML em ordem alfanumérica crescente com base no conteúdo dos campos definidos como critérios de classificação. É possível especificar vários campos como critérios de classificação. Ao classificar, os campos são priorizados desde o primeiro até ao último especificado, ou seja, os dados são classificados primeiro com base no conteúdo do primeiro campo especificado, depois pelo conteúdo do segundo campo especificado e assim por diante.

> **Dica**
> Pode usar a opção de classificação para criar uma importação XML mais eficiente dos dados extraídos para o sistema PPM. O fato de os eventos de sistema pertencentes a uma instância de processo seguirem imediatamente um ao outro devido à classificação nos arquivos de saída XML significa que eles podem ser importados diretamente para uma EPC, usando a chave de processo durante a importação XML.

Na configuração da tabela, no elemento XML **parameter**, os campos são especificados como critérios de classificação para a tabela do evento de sistema (**doctable**) e as tabelas de chave estrangeira referenciadas (**docreftable**).

Exemplo 3. ***[en-us]***

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="BANF">
<docspec>
<doctable name="EBAN">
<parameter name="ORDER_BY">
<value>BNFPO</value>
<value>BANFN</value>
</parameter>
<condition fieldname="ERDAT"
logicaloperator="creationtimestamp">
<value>yyyyMMdd</value>
</condition>
<pkfield name="BANFN"/>
<pkfield name="BNFPO"/>
<pkfield name="ERDAT"/>
<fieldtoread name="BSART"/>
<fieldtoread name="KONNR"/>
<fieldtoread name="KTPNR"/>
```

```
<fieldtoread name="LIFNR"/>
<fieldtoread name="MFRNR"/>
</doctable>
</docspec>
</configuration>
</xmlextractor_tableconfiguration>
```

Tal como foi especificado na configuração **BANF** (elemento XML **parameter**), os eventos de sistema são gravados no(s) arquivo(s) de saída classificados em ordem crescente pelo número do elemento no número da requisição de compra (campo **BNFPO**) e, como segunda prioridade, pelo número da requisição de compra (campo **BANFN**).

A tabela abaixo mostra todas as opções de configuração:

Elemento/atributo XML	Descrição
parâmetro	Especifica um parâmetro de classificação para uma tabela do evento de sistema (**doctable**) ou uma tabela de chave estrangeira (**docreftable**).
name	Nome do parâmetro. É essencial especificar **ORDER_BY**.
value	Especifica um campo de classificação. O campo tem de ser especificado como um campo de chave primária (**pkfield**).

Pode configurar o parâmetro de classificação em PPM Customizing Toolkit. Para isso, ative a guia **Evento de sistema** no módulo **Extração de dados** do grupo de módulos **Merge do processo**. Mude para modo de edição e selecione **Editar evento de sistema** do menu de contexto da tabela do evento de sistema. Pode especificar as configurações na etapa **Especificar o parâmetro**.

4.3.3. Adicionar valores de um campo

Dependendo do tipo de dados, pode adicionar os valores de um campo como números inteiros ou decimais especificando parâmetros no elemento **table** relevante da configuração da tabela. Uma classe separada deve ser usada para isso.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="SUM">
<docspec>
<docreftable name="...">
...
</docreftable>
<doctable name="...">
...
</doctable>
</docspec>
<table name="VBEP"
classtouse="com.idsscheer.ppm.xmlextractortools.
extractor.sap2ppm.ZNumberOperation_sap2ppm">
<parameter name="FIELDS">
<value>BMENG</value>
```

```
<value>WMENG</value>
</parameter>
<parameter name="OPERATION">
<value>SUM_DECIMAL</value>
</parameter>
<pkfield name="VBELN" fktablename="VBAP"
fkfieldname="VBELN"/>
<pkfield name="POSNR" fktablename="VBAP"
fkfieldname="POSNR"/>
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

Exemplo 4.

A configuração de exemplo especifica que para cada um dos campos de número de ponto flutuante **BMENG** e **WMENG**, o total de vários valores pertencentes a um elemento do documento de vendas (**POSNR**) em um documento de vendas (**VBELN**) é calculado e gravado em um novo atributo do evento de sistema. O nome do novo atributo é composto por **<tablename>-<operator><fieldname>**, por exemplo, **VBEP-SUM_DECIMALWMENG**. A classe **com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.**

ZNumberOperation_sap2ppm é usada para adicionar valores de campo.

Os seguintes números de elemento existem no sistema SAP para o documento de vendas **5696**:

VBELN	POSNR	WMENG	BMENG
5696	10	353,000	103,000
5696	10	215,000	52,000
5696	10	127,000	313,000
5696	20	89,000	79,000
5696	20	456,000	29,000
5696	30	23,000	269,000
5696	30	87,000	159,000
5696	30	124,000	256,000
5696	30	59,000	237,000

Exemplo 5.

Ao extrair usando a configuração de exemplo, as somas decimais são calculadas para os campos **BMENG** e **WMENG** e os seguintes eventos de sistema são gerados para cada número de item:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAP-POSNR">000010</attribute>
<attribute type="VBAP-VBELN">0000005696</attribute>
<attribute type="VBEP-SUM_DECIMALBMENG">468.000</attribute>
<attribute type="VBEP-SUM_DECIMALWMENG">695.000</attribute>
</event>
<event>
```

```
<attribute type="VBAP-POSNR">000020</attribute>
<attribute type="VBAP-VBELN">0000005696</attribute>
<attribute type="VBEP-SUM_DECIMALBMENG">108.000</attribute>
<attribute type="VBEP-SUM_DECIMALWMENG">545.000</attribute>
</event>
<event>
<attribute type="VBAP-POSNR">000030</attribute>
<attribute type="VBAP-VBELN">0000005696</attribute>
<attribute type="VBEP-SUM_DECIMALBMENG">921.000</attribute>
<attribute type="VBEP-SUM_DECIMALWMENG">293.000</attribute>
</event>
</eventlist>
```

4.3.4. Extrair campos de valor múltiplos

A configuração adequada do elemento **table** relevante permite extrair todos os valores diferentes de um campo (**fieldtoread**) e gravá-los em um evento de sistema. Uma classe separada deve ser usada para isso.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="MULTIPLE_VALUES">
<docspec>
<docreftable name="...">
...
</docreftable>
<doctable name="...">
...
</doctable>
</docspec>
<table name="VBEP" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.sap2ppm.
ZTableMultipleValues_sap2ppm">
<pkfield name="VBELN" fktablename="VBAP"
fkfieldname="VBELN"/>
<pkfield name="POSNR" fktablename="VBAP"
fkfieldname="POSNR"/>
<fieldtoread name="WMENG"/>
<fieldtoread name="EDATU"/>
</table>
<table name="VBAK" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.sap2ppm.
ZTableMultipleValues_sap2ppm">
<pkfield name="VBELN" fktablename="VBEP"
fkfieldname="VBELE"/>
<fieldtoread name="VBTYP"/>
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

Exemplo 6.

Na configuração de exemplo, usar a classe **com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.**

ZTableMultipleValues_sap2ppm para todas as tabelas de dados (elementos **table**) especifica que todos os campos a serem extraídos (elementos **fieldtoread**) são extraídos com valores múltiplos se existirem valor múltiplos para um campo no banco de dados do sistema de origem. No arquivo de saída XML, o campo extraído com valores múltiplos aparece como um atributo com o mesmo nome em um evento de sistema:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAP-POSNR">000010</attribute>
<attribute type="VBAP-VBELN">0000004969</attribute>
<attribute type="VBEP-EDATU">19970103</attribute>
<attribute type="VBEP-EDATU">19970107</attribute>
<attribute type="VBEP-WMENG">138.000</attribute>
<attribute type="VBEP-WMENG">317.000</attribute>
<attribute type="VBEP-WMENG">496.000</attribute>
</event>
</eventlist>
```

4.3.5. Concatenar novos valores de atributo de valores de atributo extraídos

A classe **com.idsscheer.ppm.xmlextractortools.extractor.**

sap2ppm.ZTableConcatFKs_sap2ppm pode ser usada ao extrair dados de um sistema SAP para gerar novos atributos do evento do sistema, concatenando os valores dos atributos já extraídos. Os novos eventos de sistema criados podem ser referenciados em outras configurações da tabela de dados (elemento XML **table**).

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="SD">
<docspec>
<doctable name="RBKP">
<condition logicaloperator="eq" fieldname="GJAHR">
<value>2005</value>
</condition>
<pkfield name="BELNR"/>
<pkfield name="GJAHR"/>
</doctable>
</docspec>
<table name="RBKP_NEW" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.
sap2ppm.ZTableConcatFKs_sap2ppm">
<pkfield name="BELNR" fktablename="RBKP"
fkfieldname="BELNR"/>
<pkfield name="GJAHR" fktablename="RBKP"
fkfieldname="GJAHR"/>
<fieldtoread name="AWKEY_GENERATED"/>
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

Se estiver a usar a classe **ZTableConcatFKs_sap2ppm** terá de especificar exatamente um elemento **fieldtoread**. O elemento contém o nome do atributo do evento de sistema a ser criado. No exemplo, é criado um novo atributo do evento de sistema com o nome **RBKP_NEW-AWKEY_GENERATED**. O valor do atributo consiste nos valores dos elementos **pkfield** especificados.

As operações normais **fkpart**, **prefix** e **postfix** podem ser usadas para os elementos **pkfield**.

Um evento de sistema criado com a configuração acima poderia ter a seguinte aparência:

```
...
<event>
<attribute type="RBKP-BELNR">5105601204</attribute>
<attribute type="RBKP-GJAHR">2005</attribute>
<attribute type="RBKP_NEW-AWKEY_GENERATED">51056012042005
</attribute>
</event>
...
```

Caso especial

Na configuração de **tabela**, o elemento especial **pkfield CLIENT** pode ser usado para adicionar automaticamente o número do cliente SAP ao extrair dados. Os atributos XML **name**, **fktablename** e **fkfieldname** têm de ter o valor **CLIENT**.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="SD">
<docspec>
<doctable name="RBKP">
<condition logicaloperator="eq" fieldname="GJAHR">
<value>2005</value>
</condition>
<pkfield name="BELNR"/>
<pkfield name="GJAHR"/>
</doctable>
</docspec>
<table name="RBKP_NEWER" classtouse="com.idsscheer.
ppm.xmlextractortools.extractor.
sap2ppm.ZTableConcatFKs_sap2ppm">
<pkfield name="CLIENT" fktablename="CLIENT"
fkfieldname="CLIENT">
<postfix>
<value> test </value>
</postfix>
</pkfield>
<pkfield name="BELNR" fktablename="RBKP"
fkfieldname="BELNR"/>
<pkfield name="GJAHR" fktablename="RBKP"
fkfieldname="GJAHR"/>
<fieldtoread name="AWKEY_GENERATED"/>
</table>
</configuration>
```

```
</xmlextractor_tableconfiguration>
```

Um evento de sistema que é criado ao extrair dados do cliente 800 poderia ter a seguinte aparência:

```
...
<event>
<attribute type="RBKP-BELNR">5105601204</attribute>
<attribute type="RBKP-GJAHR">2005</attribute>
<attribute type="RBKP_NEWER-AWKEY_GENERATED"
>800 test 51056012042005</attribute>
</event>
...
```

O novo atributo – **RBKP_NEWER-AWKEY_GENERATED** – pode então ser referenciado em outras configurações de **tabela** da mesma forma que os restantes campos da tabela SAP.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="SD">
<docspec>
<doctable name="RBKP">
<condition logicaloperator="eq" fieldname="GJAHR">
<value>2005</value>
</condition>
<pkfield name="BELNR"/>
<pkfield name="GJAHR"/>
</doctable>
</docspec>
<table name="RBKP_NEW" classtouse="com.idsscheer.
ppm.xmlextractortools.extractor.
sap2ppm.ZTableConcatFKs_sap2ppm">
<pkfield name="BELNR" fktablename="RBKP"
fkfieldname="BELNR"/>
<pkfield name="GJAHR" fktablename="RBKP"
fkfieldname="GJAHR"/>
<fieldtoread name="AWKEY_GENERATED"/>
</table>
<table name="BKPF">
<pkfield name="AWKEY" fktablename="RBKP_NEW"
fkfieldname="AWKEY_GENERATED"/>
<fieldtoread name="BELNR"/>
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

Um evento de sistema criado com essa configuração poderia ter a seguinte aparência:

```
...
<event>
<attribute type="RBKP-BELNR">5105601204</attribute>
<attribute type="RBKP-GJAHR">2005</attribute>
<attribute type="RBKP_NEW-AWKEY_GENERATED"
>51056012042005</attribute>
<attribute type="BKPF-BELNR">5100004691</attribute>
</event>
...
```

4.3.6. Conectar tabelas

O cabeçalho, documento e tabelas de dados estão vinculados entre si por relações de chave primária, que são especificadas no elemento XML **pkfield**.

Elemento XML	Atributos XML	Descrição
pkfield	name	Nome da chave primária (nome da coluna da tabela)
	fktablename	Identificador da tabela referenciada por uma relação de chave estrangeira
	fkfieldname	Nome da chave estrangeira (nome da coluna da tabela)
	logicaloperator	Valores válidos: **eq**, **neq**, **gt**, **geq**, **lt**, **leq**. Valor padrão: **eq**
prefix		Cadeia de caracteres inserida antes do valor de chave estrangeira extraído
postfix		Cadeia de caracteres inserida após o valor de chave estrangeira extraído
fkpart	readfrom (opcional)	Direção da subcadeia que forma a chave estrangeira Valores válidos: **left**, **right** Valor padrão: **left**
	startposition	Posição a partir da qual a subcadeia é formada
	length (opcional)	Comprimento da subcadeia Valor padrão: Início ou fim da cadeia de caracteres, dependendo do valor do atributo XML **readfrom**
value		Valor especificado para condições, prefixos e sufixos

Exemplo 7. ***[en-us]***

O extrato de arquivo abaixo mostra a configuração que é usada para extrair os campos do banco de dados **VBELN** e **POSNR** da tabela **VBAP**. As relações de chave estrangeira são estabelecidas pelo conteúdo do campo **TABKEY** da tabela **CDPOS**.

Os primeiros dez caracteres do campo **TABKEY** contêm o valor que é associado ao campo **VBELN** da tabela **VBAP** e os seis caracteres seguintes contêm o valor que é associado ao campo **POSNR** da tabela **VBAP**.

```
<table name="VBAP">
<pkfield name="VBELN" fktablename="CDPOS"
fkfieldname="TABKEY">
<fkpart startposition="0" length="10"/>
</pkfield>
```

```
<pkfield name="POSNR" fktablename="CDPOS"
fkfieldname="TABKEY">
<fkpart startposition="10" length="6"/>
</pkfield>
<fieldtoread name="VBELN"/>
<fieldtoread name="POSNR"/>
</table>
```

4.3.6.1. Comparação de relações de chave estrangeira

Por padrão, é verificada a igualdade das relações de chave estrangeira especificadas no elemento XML **pkfield** (valor padrão: **logicaloperator="eq"**). Isso se aplica às definições correspondentes da tabela do documento (**doctable**), tabela do cabeçalho do documento (**docreftable**) e tabela de dados (**table**).

Pode realizar outras comparações especificando os operadores seguintes:

- **neq** (não igual a)
- **lt** (inferior a)
- **gt** (maior do que)
- **geq** (maior do que ou igual a)
- **leq** (menor do que ou igual a)

As comparações são realizadas lexicograficamente, de acordo com a lista de caracteres Unicode, ou seja, o valor **00999** é inferior ao valor **9**. Os valores de texto alfanuméricos também podem ser comparados.

Exemplo 8.

O período de referência associado para cada valor de dados **EEINV-EINZDAT** deve ser determinado. Este valor representa um intervalo de tempo com uma data de início e de término. Todos os três campos de dados (**EEINV-EINZDAT**, **EANLH-VON**, **EANLH-BIS**) devem ser extraídos. O extrato de arquivo abaixo mostra a configuração correspondente com as comparações de chave estrangeira **menor do que ou igual a** ou **maior do que ou igual a** para extrair valores de campo da tabela **EANLH**:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="ISU">
<docspec>
<doctable name="EEINV">
<pkfield name="EINZBELEG"/>
<pkfield name="VERTRAG"/>
</doctable>
</docspec>
<table name="EEINV">
<pkfield name="EINZBELEG" fktablename="EEINV"
fkfieldname="EINZBELEG"/>
<pkfield name="VERTRAG" fktablename="EEINV"
fkfieldname="VERTRAG"/>
<fieldtoread name="ANLAGE"/>
<fieldtoread name="EINZDAT"/>
```

```
</table>
<table name="EANLH">
<pkfield name="ANLAGE" fktablename="EEINV"
fkfieldname="ANLAGE"/>
<pkfield name="VON" fktablename="EEINV"
fkfieldname="EINZDAT" logicaloperator="leq"/>
<pkfield name="BIS" fktablename="EEINV"
fkfieldname="EINZDAT" logicaloperator="geq"/>
<fieldtoread name="VON"/>
<fieldtoread name="BIS"/>
<fieldtoread name="TARIFTYP"/>
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

Exemplo 9.

Um evento de sistema extraído com essa configuração poderia ter a seguinte aparência:

```
<event>
<attribute type="EEINV-EINZBELEG">000000800002</attribute>
<attribute type="EEINV-VERTRAG">00012532</attribute>
<attribute type="EEINV-ANLAGE">00045678</attribute>
<attribute type="EEINV-EINZDAT">20041015</attribute>
<attribute type="EANLH-VON">20041001</attribute>
<attribute type="EANLH-BIS">20050930</attribute>
<attribute type="EANLH-TARIFTYP">YELLOW-PRIVAT</attribute>
</event>
```

4.3.7. Extração usando condições

A extração de dados das tabelas pode ser limitada pelo uso de condições. Por exemplo, pode usar condições para garantir que, ao extrair dados de uma tabela, apenas serão extraídas as linhas que correspondem a determinado campo de dados. As condições são especificadas diretamente na configuração da especificação da tabela do elemento XML **booleancondition** ou **condition** e aplicam-se apenas à tabela associada.

Uma condição contém o nome do campo de dados, o operador de comparação e um valor concreto. As condições podem ser vinculadas com qualquer grau de complexidade (elemento XML **booleancondition**).

Elemento XML	Atributos XML	Descrição
booleancondition	logicaloperator	Operadores lógicos: **AND**, **OR**, **NOT**
(opcional)		Valor padrão: **AND**
condição	logicaloperator	Operadores de comparação: **eq**, **neq**, **in**, **notin**, **num_gt**, **num_geq**, **num_lt**, **num_leq**, **like**
(opcional)		Operadores para restringirem o intervalo de dados a ser extraído: **creationtimestamp**, **valueconstraint**
		Valor padrão: **eq**
	fieldname	Nome do campo da tabela

Os elementos XML **docbooleancondition** e **doccondition** permitem a extração condicional de uma tabela de dados, dependendo dos campos da tabela já extraídos. São configurados de forma semelhante a

booleancondition e **condition**. O nome da tabela e o nome da coluna que contêm os campos de dados já extraídos são especificados nos atributos XML **tablename** e **fieldname**.

Elemento XML	Atributos XML	Descrição
docbooleancondition	logicaloperator	Operador lógico: **AND**, **OR**, **NOT**
(opcional)		Valor padrão: **AND**
Doccondition	logicaloperator	Operador de comparação: **eq**, **neq**, **in**, **notin**, **exists**, **notexists**
(opcional)		Valor padrão: **eq**
	tablename	Nome da tabela que contém os campos da tabela já extraídos
	fieldname	Nome de coluna

4.3.7.1. Operadores de condição

Operadores comuns

Os seguintes operadores de comparação são suportados pelos elementos XML **condition** e **doccondition**:

Operador	Descrição
eq	O conteúdo do campo é igual ao valor especificado. (Comparação de cadeia de caracteres Java com distinção entre maiúsculas e minúsculas)
neq	O conteúdo do campo não é igual ao valor especificado. (Comparação de cadeia de caracteres Java com distinção entre maiúsculas e minúsculas)
in	O conteúdo do campo é igual a um valor especificado de um conjunto de valores. (Comparação de cadeia de caracteres Java com distinção entre maiúsculas e minúsculas)
notin	O conteúdo do campo não é igual a um valor especificado de um conjunto de valores. (Comparação de cadeia de caracteres Java com distinção entre maiúsculas e minúsculas)

Operadores de condição

Operador	Descrição
num_gt	O conteúdo do campo é maior do que o valor especificado.
num_geq	O conteúdo do campo é maior do que ou igual ao valor especificado.
num_lt	O conteúdo do campo é menor do que o valor especificado.
num_leq	O conteúdo do campo é inferior ou igual ao valor especificado.

Operador	Descrição

like
 	
(apenas para tipos de dados alfanuméricos R/3, ou seja, ACCP, CHAR, CLNT, CUKY, LCHR, NUMC, UNIT, VARC, TIMS ou tipos de dicionário DATS)
	Comparação de valores de campo com uma cadeia de caracteres variável
	
	São permitidos os seguintes espaços reservados:
	
	***** No ou qualquer número de caracteres
	
	? Apenas um caractere
	
	**** Caractere de mascaramento para procurar
	
	espaços reservados ou mascarar caracteres na
	
	forma: **** or ***** or **\?**
	
	Exemplo:
	
	`<condition fieldname="OBJECTID"`
	
	`logicaloperator="like">`
	`<value>*10?0\\20?0*</value>`
	`</condition>`
	
	A pesquisa é realizada para valores como
	
	5551050\20106667 or 1080\204044 but not 34510550\2030*

creationtimestamp	Os carimbos de data/hora (data e hora) são extraídos de campos R/3 com horas. Seus valores formam a base para restringir o intervalo de dados a ser extraído com os parâmetros de linha de comando **-begindate**
	
	(**-begintime**) ou **-enddate** (**-endtime**) [ver capítulo Argumentos específicos do sistema de origem [65]].
	
	Vários campos são separados pela combinação de caracteres **#-#**. Para cada campo, são especificadas as descrições de formato dos campos do sistema de origem nos elementos XML **value**.
	
	Exemplo (para tabela **VBAP**):
	
	`<condition fieldname="ERDAT#-#ERZET"`
	
	`logicaloperator="creationtimestamp">`
	`<value>dd.MM.yyyy</value>`
	`<value>HH:mm:ss</value>`
	`</condition>`

valueconstraint	Especifica um campo R/3 com valores inteiros que são usados para delimitar o intervalo de dados a ser extraído com o parâmetro de linha de comando **-valueconstraint**
	
	[ver capítulo Argumentos específicos do sistema de origem [65]].
	
	O campo a ser extraído tem de ser do tipo de dadosNUMC ou INT4.
	
	Exemplo (para tabela **VBAP**):
	
	`<condition fieldname="POSNR"`
	
	`logicaloperator="valueconstraint"/>`

doccondition operators

Operador	Descrição
exists	O campo existe.
notexists	O campo não existe.

4.3.7.2. Condições complexas

O extrato de arquivo abaixo ilustra as condições aninhadas usando o exemplo da tabela de cabeçalho do documento:

```
...
<docreftable tablename="VBAK" >
<booleancondition logicaloperator="AND">
<condition fieldname="ERDAT#-#ERZET"
logicaloperator="creationtimestamp">
<value>yyyyMMdd</value>
<value>HHmmss</value>
</condition>
<condition fieldname="VBTYP" logicaloperator="in">
<value>C</value>
<value>K</value>
</condition>
<booleancondition logicaloperator="OR">
<booleancondition logicaloperator="AND">
<condition fieldname="VKORG" logicaloperator="eq">
<value>1000</value>
</condition>
<condition fieldname="VKBUR" logicaloperator="eq">
<value>0041</value>
</condition>
</booleancondition >
<booleancondition logicaloperator="AND">
<condition fieldname="VKORG" logicaloperator="eq">
<value>2000</value>
</condition>
<condition fieldname="VKBUR" logicaloperator="neq">
<value>0060</value>
</condition>
</booleancondition >
</booleancondition >
</booleancondition >
...
</docreftable>
...
```

Em termos de lógica proposicional, o extrato de arquivo exibido corresponde à condição seguinte:

ERDAT e ERZET contêm a hora de criação

e

VBTYP é C ou V

e

((VKORG é igual a 1000 **e** VKBUR é igual a 0041) **ou** (VKORG é igual a 2000 **e** VKBUR não é igual a 0060)).

4.3.8. Extrair campos do banco de dados

Os campos do banco de dados na tabela de documento e as tabelas de dados vinculadas, a partir dos quais os valores devem ser extraídos, são especificados no elemento XML **fieldtoread**. Para cada elemento XML **fieldtoread**, uma linha do formulário

```
<attribute type="Type">Value</attribute>
```

é gravada no arquivo de saída XML.

Type é composto pelo nome da tabela, o nome do campo de dados e um nome do campo de texto opcional.

Value é o valor extraído do campo de dados correspondente. Todos os valores são gravados em formato de texto.

Opcionalmente, em vez do valor do campo de dados direto, pode ser extraído o valor da tabela referenciada (elemento XML **textref**). A especificação opcional **langfieldname** extrai o texto específico do idioma do campo de dados.

Informações suplementares são extraídas das tabelas de dados usando relações chave primárias.

Elemento XML	Atributos XML	Descrição
fieldtoread	name	Nome da coluna da tabela que contém o campo de dados a ser extraído
textref	tablename	Nome da tabela da tabela de dados referenciada
	reffieldname	Nome da chave estrangeira
	textfieldname	Nome da coluna da tabela que contém o campo de dados a ser extraído
	langfieldname (opcional)	Nome da coluna da tabela que contém o campo de dados específico do idioma a ser extraído
fkpart	readfrom (opcional)	Direção da subcadeia que forma a chave estrangeira Valores válidos: **left**, **right** Valor padrão: **left**
	startposition	Posição a partir da qual a subcadeia é formada
	length (opcional)	Comprimento da subcadeia Valor padrão: Início ou final da cadeia de caracteres (atributo XML **readfrom**)

Antes que os dados sejam realmente extraídos, é executada uma verificação para saber se as tabelas configuradas e os campos de dados existem e se o usuário do sistema especificado tem autorização de acesso apropriada.

Exemplo 10.

Os valores de campo extraídos são gravados no arquivo de saída XML como atributos de um evento de sistema.

Campos de tabela não localizados:

```
...

  <attribute type="VBAP-WERKS">SB</attribute>
```

```
  ...
```

Exemplo 11.

Campos de tabela localizados:

```
...
<attribute type="VBAP-WERKS-NAME">Saarbrücken</attribute>
...
```

4.3.9. Exemplo: Configuração XML e arquivo de saída

Este capítulo usa um exemplo prático para ilustrar a configuração da tabela para extrair dados de um sistema SAP e mostra um extrato do arquivo de saída XML gerado.

4.3.9.1. Configuração da tabela

O extrato de arquivo abaixo mostra a configuração da tabela **SD_C** para extrair dados de um sistema SAP SD:

```
...
<xmlextractor_tableconfiguration>
<configuration name="SD_C">
<docspec>
<docreftable name="VBAK">
<booleancondition>
<condition fieldname="ERDAT#-#ERZET"
logicaloperator="creationtimestamp">
<value>yyyyMMdd</value>
<value>HHmmss</value>
</condition>
<condition fieldname="VBTYP"
logicaloperator="eq">
<value>C</value>
</condition>
</booleancondition>
<pkfield name="VBELN" />
</docreftable>
<doctable name="Orders" tablename="VBAP">
<pkfield name="VBELN" fktablename="VBAK"
fkfieldname="VBELN"/>
<pkfield name="POSNR" />
<fieldtoread name="ERDAT"/>
<fieldtoread name="ERZET"/>
<fieldtoread name="MATNR">
<textref tablename="MAKT" reffieldname="MATNR"
textfieldname="MAKTX" langfieldname="SPRAS"/>
</fieldtoread>
```

```
<fieldtoread name="KONDM">
<textref tablename="T178T" reffieldname="KONDM"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="SPART">
<textref tablename="TSPAT" reffieldname="SPART"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="WERKS">
<textref tablename="T001W" reffieldname="WERKS"
textfieldname="NAME1"/>
</fieldtoread>
<fieldtoread name="CHARG"/>
<fieldtoread name="PSTYV"/>
<fieldtoread name="ERNAM"/>
<fieldtoread name="NETWR"/>
</doctable>
</docspec>
<table name="VBAK">
<pkfield name="VBELN" fktablename="Orders"
fkfieldname="VBELN"/>
<fieldtoread name="VTWEG">
<textref tablename="TVTWT" reffieldname="VTWEG"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VKORG">
<textref tablename="TVKOT" reffieldname="VKORG"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VDATU"/>
<fieldtoread name="VKBUR">
<textref tablename="TVKBT" reffieldname="VKBUR"
textfieldname="BEZEI" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VKGRP">
<textref tablename="TVGRT" reffieldname="VKGRP"
textfieldname="BEZEI" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VBTYP"/>
<fieldtoread name="AUART"/>
</table>
<table name="MARA">
<pkfield name="MATNR" fktablename="Orders"
fkfieldname="MATNR"/>
<fieldtoread name="MATNR"/>
<fieldtoread name="MTART">
<textref tablename="T134T" reffieldname="MTART"
textfieldname="MTBEZ" langfieldname="SPRAS"/>
</fieldtoread>
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

A configuração da tabela **SD_C** contém as seguintes tabelas:

Tabela de cabeçalho do documento

```
...
<docreftable name="VBAK">
<booleancondition>
<condition fieldname="ERDAT#-#ERZET"
logicaloperator="creationtimestamp">
<value>yyyyMMdd</value>
<value>HHmmss</value>
</condition>
<condition fieldname="VBTYP"
logicaloperator="eq">
<value>C</value>
</condition>
</booleancondition>
<pkfield name="VBELN" />
</docreftable>
...
```

O identificador e o nome da tabela de cabeçalho do documento do sistema R/3 é **VBAK**. Todos os cabeçalhos de documentos de encomenda (**VBTYP=C**) para o período especificado na linha de comando são lidos. Os elementos **value** do operador de condição **creationtimestamp** especificam o formato do carimbo de data/hora especificado.

O nome da coluna da tabela de chave primária é **VBELN**. Para cada valor de campo **VBELN** diferente, os registros de dados são extraídos da tabela do documento vinculado para a qual **VBELN** apresenta o mesmo valor que a tabela de cabeçalho do documento.

Tabela do documento

```
...
<doctable name="Orders" tablename="VBAP">
<pkfield name="VBELN" fktablename="VBAK"
fkfieldname="VBELN"/>
<pkfield name="POSNR" />
<fieldtoread name="ERDAT"/>
<fieldtoread name="ERZET"/>
<fieldtoread name="MATNR">
<textref tablename="MAKT" reffieldname="MATNR"
textfieldname="MAKTX" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="KONDM">
<textref tablename="T178T" reffieldname="KONDM"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="SPART">
<textref tablename="TSPAT" reffieldname="SPART"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="WERKS">
```

```
<textref tablename="T001W" reffieldname="WERKS"
textfieldname="NAME1"/>
</fieldtoread>
<fieldtoread name="CHARG"/>
<fieldtoread name="PSTYV"/>
<fieldtoread name="ERNAM"/>
<fieldtoread name="NETWR"/>
</doctable>
...
```

O identificador **Ordens** é associado à tabela do documento **VBAP**. A relação de chave estrangeira especificada na linha

```
<pkfield name="VBELN" fktablename="VBAK" fkfieldname="VBELN"/>
```

vincula a tabela à tabela de cabeçalho do documento. A menos que especificado de outra forma, as relações de chave estrangeira são criadas com base em valores de campo iguais (ver o capítulo Comparação de relações de chave estrangeira [27]). Neste exemplo, apenas as linhas da tabela para as quais a comparação dos valores de campo **VBAP-VBELN** e **VBAK-VBELN** retornar valores iguais serão extraídas. A chave primária consiste nas colunas **VBELN** e **POSNR**.

Um evento de sistema (elemento XML **event**) é criado no arquivo de saída XML para cada linha extraída da tabela. Para cada elemento XML **fieldtoread**, uma linha do formulário

```
<attribute type="...">...</attribute>
```

é gravada no arquivo de saída.

Os campos de chave primária (**pkfield**) especificados na definição **doctable** são extraídos automaticamente. Não precisa de especificar **fieldtoread** para esses campos.

Para alguns elementos **fieldtoread**, o valor extraído da tabela referenciada (elemento XML **textref**) é gravado em conjunto com o valor do campo de dados extraído. O elemento opcional **langfieldname** extrai o texto específico do idioma do campo de dados.

O tipo de atributo do sistema de origem completo é composto pelo identificador da tabela de cabeçalho do documento (**doctable name**), nome do campo (**fieldtoread name**) e nome do campo de texto referenciado (**textref ... textfieldname**). O tipo de atributo do sistema de origem para o primeiro elemento **fieldtoread** extraído com uma tabela referenciada é semelhante a:

<attribute type="Orders-MATNR-MAKTX">...</attribute>

Tabelas de dados

```
...
<table name="VBAK">
<pkfield name="VBELN" fktablename="Orders"
fkfieldname="VBELN"/>
<fieldtoread name="VTWEG">
<textref tablename="TVTWT" reffieldname="VTWEG"
```

```
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VKORG">
<textref tablename="TVKOT" reffieldname="VKORG"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VDATU"/>
<fieldtoread name="VKBUR">
<textref tablename="TVKBT" reffieldname="VKBUR"
textfieldname="BEZEI" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VKGRP">
<textref tablename="TVGRT" reffieldname="VKGRP"
textfieldname="BEZEI" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VBTYP"/>
<fieldtoread name="AUART"/>
</table>

<table name="MARA">
<pkfield name="MATNR" fktablename="Orders"
fkfieldname="MATNR"/>
<fieldtoread name="MATNR"/>
<fieldtoread name="MTART">
<textref tablename="T134T" reffieldname="MTART"
textfieldname="MTBEZ" langfieldname="SPRAS"/>
</fieldtoread>
</table>
...
```

As relações de chave estrangeira com os campos de chave primária **VBELN** e **MATNR** na tabela **Ordens** são usadas para extrair informações suplementares das tabelas de dados **VBAK** e **MARA**.

4.3.9.2. Arquivo de saída XML (formato do evento de sistema do PPM)

O extrato de arquivo abaixo mostra um intervalo de eventos do sistema a partir do arquivo de saída XML gerado usando o arquivo de configuração especificado no capítulo sobre **Configuração da tabela [33]**:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="Orders-CHARG"></attribute>
<attribute type="Orders-ERDAT">20000214</attribute>
<attribute type="Orders-ERNAM">HDM</attribute>
<attribute type="Orders-ERZET">143616</attribute>
<attribute type="Orders-KONDM"></attribute>
<attribute type="Orders-MATNR">P-100</attribute>
<attribute type="Orders-MATNR-MAKTX">
Thread rod M'8 DIN '8895' without tolerance
</attribute>
<attribute type="Orders-NETWR">28.70</attribute>
```

```
<attribute type="Orders-POSNR">000010</attribute>
<attribute type="Orders-PSTYV">TAN</attribute>
<attribute type="Orders-SPART">01</attribute>
<attribute type="Orders-SPART-VTEXT">Product category 01
</attribute>
<attribute type="Orders-VBELN">0000000001</attribute>
<attribute type="Orders-WERKS">1000</attribute>
<attribute type="Orders-WERKS-NAME1">Plant 1000 (Hamburg)
</attribute>
<attribute type="MARA-MATNR">P-100</attribute>
<attribute type="MARA-MTART">HAWA</attribute>
<attribute type="MARA-MTART-MTBEZ">Trading goods
</attribute>
<attribute type="VBAK-AUART">TA</attribute>
<attribute type="VBAK-VBTYP">C</attribute>
<attribute type="VBAK-VDATU">20000214</attribute>
<attribute type="VBAK-VKBUR"></attribute>
<attribute type="VBAK-VKGRP"></attribute>
<attribute type="VBAK-VKORG">1000</attribute>
<attribute type="VBAK-VKORG-VTEXT">Germany Frankfurt
</attribute>
<attribute type="VBAK-VTWEG">10</attribute>
<attribute type="VBAK-VTWEG-VTEXT">Consumer sales
</attribute>
</event>
<event>
<attribute type="VBAK-VKORG-VTEXT">Germany Frankfurt
</attribute>
<attribute type="VBAK-VTWEG-VTEXT">Consumer sales
</attribute>
<attribute type="Orders-SPART">01</attribute>
<attribute type="Orders-PSTYV">TAD</attribute>
<attribute type="Orders-MATNR">SERVICE</attribute>
<attribute type="Orders-ERDAT">20000214</attribute>
<attribute type="VBAK-VKBUR"></attribute>
<attribute type="VBAK-VTWEG">10</attribute>
<attribute type="Orders-SPART-VTEXT">Product category 01
</attribute>
<attribute type="Orders-ERZET">143616</attribute>
<attribute type="VBAK-VKORG">1000</attribute>
<attribute type="Orders-POSNR">000020</attribute>
<attribute type="MARA-MTART-MTBEZ">Service
</attribute>
<attribute type="Orders-WERKS-NAME1">Plant 1000 (Hamburg)
</attribute>
<attribute type="Orders-CHARG"></attribute>
<attribute type="Orders-WERKS">1000</attribute>
<attribute type="VBAK-VDATU">20000214</attribute>
<attribute type="Orders-MATNR-MAKTX">Repair
</attribute>
<attribute type="Orders-VBELN">0000000001</attribute>
<attribute type="VBAK-AUART">TA</attribute>
<attribute type="VBAK-VKGRP"></attribute>
<attribute type="Orders-NETWR">179.00</attribute>
<attribute type="MARA-MTART">DIEN</attribute>
```

```
<attribute type="VBAK-VBTYP">C</attribute>
<attribute type="Orders-ERNAM">HDM</attribute>
<attribute type="MARA-MATNR">SERVICE</attribute>
<attribute type="Orders-KONDM"></attribute>
</event>
<event>
<attribute type="VBAK-VKGRP-BEZEI">GR. F2 Mr. Mayer
</attribute>
<attribute type="VBAK-VKORG-VTEXT">Germany Frankfurt
</attribute>
<attribute type="VBAK-VTWEG-VTEXT">Consumer sales
</attribute>
<attribute type="Orders-SPART">01</attribute>
<attribute type="Orders-PSTYV">TAN</attribute>
<attribute type="Orders-MATNR">P-100</attribute>
<attribute type="Orders-ERDAT">20000214</attribute>
<attribute type="VBAK-VKBUR">1000</attribute>
<attribute type="VBAK-VTWEG">10</attribute>
<attribute type="Orders-SPART-VTEXT">Product category 01
</attribute>
<attribute type="Orders-ERZET">131257</attribute>
<attribute type="VBAK-VKBUR-BEZEI">Frankfurt office
</attribute>
<attribute type="VBAK-VKORG">1000</attribute>
<attribute type="Orders-POSNR">000010</attribute>
<attribute type="MARA-MTART-MTBEZ">Trading goods</attribute>
<attribute type="Orders-WERKS-NAME1">Plant 1000 (Hamburg)
</attribute>
<attribute type="Orders-CHARG"></attribute>
<attribute type="Orders-WERKS">1000</attribute>
<attribute type="VBAK-VDATU">20000214</attribute>
<attribute type="Orders-MATNR-MAKTX">
Thread rod M'8 DIN '8895' without tolerance
</attribute>
<attribute type="Orders-VBELN">0000000002</attribute>
<attribute type="VBAK-AUART">TA</attribute>
<attribute type="VBAK-VKGRP">101</attribute>
<attribute type="Orders-NETWR">28.70</attribute>
<attribute type="MARA-MTART">HAWA</attribute>
<attribute type="VBAK-VBTYP">C</attribute>
<attribute type="Orders-ERNAM">HDM</attribute>
<attribute type="MARA-MATNR">P-100</attribute>
<attribute type="Orders-KONDM"></attribute>
</event>
...
</eventlist>
```

4.3.10. Extração em bloco

Pode influenciar a relação entre a execução e a eficiência de memória da operação de extração.

O valor do argumento da linha de comando **-cpd** determina o número de eventos do sistema que são mantidos simultaneamente na memória do sistema. O número máximo possível depende do tamanho da memória disponível do sistema.

A memória real utilizada pelo sistema depende dos seguintes fatores:

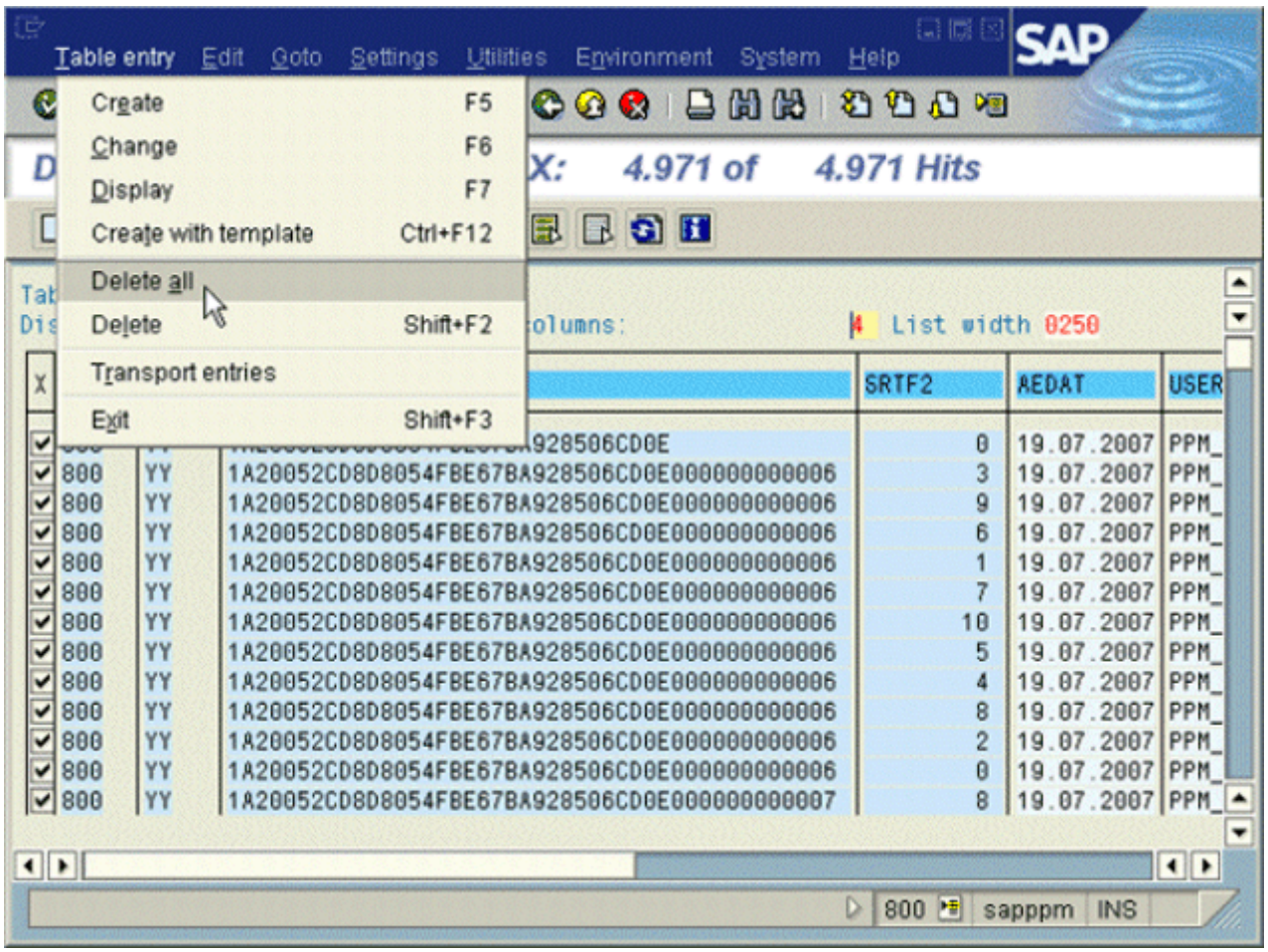
- Número total de eventos do sistema a serem extraídos no período de análise definido
- Número de eventos do sistema mantidos em paralelo no sistema (documentos processados simultaneamente)
- Número de campos de dados a ser extraídos (elementos XML **fieldtoread** e **reftext**).
- Requisitos de memória dos campos de dados extraídos

4.3.10.1. Extração em bloco de grandes volumes de dados

Durante a extração, todos os dados de uma chamada RFC são transferidos de uma vez para o cliente Java usando o SAP JCo. No entanto, grandes volumes de dados podem provocar sobrecarga da memória (OutOfMemoryError). Por exemplo, isso acontece quando as chaves primárias das tabelas do documento e tabelas de cabeçalho do documento estão sendo determinadas. Nesse caso, os dados devem primeiro ser armazenados em buffer no sistema R/3 e, em seguida, transferidos para o cliente Java, um bloco de cada vez. A dimensão do bloco é especificada pelo parâmetro **-cpd**. Existem três modos de bloco diferentes, que diferem em termos de como os dados são armazenados no sistema R/3.

Modo de bloco D

O modo **D** é recomendado para extração em bloco de grandes conjuntos de dados do sistema R/3. Os dados selecionados são armazenados na tabela do banco de dados **/IDS/INDX**. Se a conexão com o Extrator falhar ou se a operação de extração for interrompida manualmente, os dados não serão excluídos automaticamente da tabela do banco de dados. Tem de excluir manualmente os dados armazenados. Para isso, exiba o conteúdo da tabela **/IDS/INDX** usando a transação **SE16**. Em seguida, selecione todas as entradas relevantes e exclua-as usando a opção **Excluir tudo...** no menu principal **Entrada da tabela**.



Exemplo 12.

Quer extrair mais de um milhão de entradas da tabela **CPDOS** (elementos do documento de alteração). Se não usar a extração em bloco, a memória Java será sobrecarregada e a extração será interrompida.

Ao executar a seguinte linha de comando, extrai os dados em blocos (parâmetro do bloco em negrito):

```
runsap2ppm -systemconfig Systemconfig.xml -tableconfig TableconfigCDPOS.xml
-outfile eventsCDPOS -nozip -begindate 19700101 -rfc_blockmode D -cpd 10000
```

Exemplo 13.

Ao especificar o parâmetro **-rfc_blockmode** com o argumento **D**, é iniciada a extração em bloco, que grava todos os dados da tabela do banco de dados **CPDOS** para o arquivo de saída. O parâmetro **-cpd** é usado para especificar a dimensão dos blocos a serem processados. No exemplo, **10000** documentos são processados simultaneamente. O valor padrão é **5000**.

O progresso da extração em bloco é registrado como especificado, por exemplo:

```
I: 09.05.06 11:30:38: [XML] Extracting data from...
...
...
I: 09.05.06 11:32:30: [XML] "CDPOS" table:
10,000 of 1,036,995 rows extracted.
```

```
I: 09.05.06 11:32:30: [XML] Extracting attributes for
10000 source system events...
I: 09.05.06 11:32:39: [XML] "CDPOS" table:
20,000 of 1,036,995 rows extracted.
I: 09.05.06 11:32:39: [XML] Extracting attributes for
10000 source system events...
...
```

Ao especificar **-progress no**, pode suprimir as mensagens de log adicionais para os modos de bloco, que mostram o progresso da extração em bloco.

Modo de bloco G

Os dados selecionados são armazenados em uma tabela interna do grupo de funções. Os dados são mantidos na memória do servidor R/3. Se a conexão com o Extrator falhar ou se a extração for interrompida manualmente, os dados serão excluídos automaticamente do sistema R/3. Estas especificações aplicam-se desde a versão SAP 4.7, ou seja, desde a versão SAP Kernel 6.10.

Modo de bloco M

Os dados selecionados são armazenados na memória ABAP usando **exportar para id de memória**.
Os dados são mantidos na memória do servidor R/3. Se a conexão com o Extrator falhar ou se a extração for interrompida manualmente, os dados serão excluídos automaticamente do sistema R/3. Estas especificações aplicam-se desde a versão SAP 4.7, ou seja, desde a versão SAP Kernel 6.10.

4.3.11. Extrair documentos de alteração

Os documentos de alteração representam um tipo de documento especial do sistema SAP R/3. Estes documentos estão guardados nas tabelas **CDHDR** (cabeçalho de alteração) e **CDPOS** (elemento de alteração). Como essas tabelas são estruturadas da mesma forma que as outras tabelas de documento e cabeçalhos de documento no sistema R/3, a estrutura de configuração pode ser usada para extrair dados das tabelas de documentos de alteração, sem alterações.

A configuração para extração de dados das tabelas de documentos de alteração deve ser projetada de forma que os dados sejam extraídos da tabela do cabeçalho do documento **CDHDR** antes de qualquer outra tabela.

Existem duas formas de extrair documentos de alteração.

4.3.11.1. Criar eventos do sistema de origem a partir de alterações individuais

É criado um evento de sistema para cada documento de alteração.

Exemplo 14.

As alterações para janeiro de 2003 que atendem às seguintes condições devem ser extraídas da tabela do documento de alteração:

- A classe de objeto é **VERKBELEG**.
- A tabela alterada é **VBAK**.
- O campo alterado é **VKGRP**.
- O valor do novo campo é **4711** ou **4712**.

Exemplo 15.

As condições podem ser formuladas no arquivo de configuração XML abaixo:

```
...
<configuration name="OneEventForEveryChangeDoc">
<docspec>
<docreftable name="CDHDR">
<booleancondition logicaloperator="AND">
<condition fieldname="OBJECTCLAS"
logicaloperator="eq">
<value>VERKBELEG</value>
</condition>
<condition fieldname="UDATE#-#UTIME"
logicaloperator="creationtimestamp">
<value>yyyyMMdd</value>
<value>HHmmss</value>
</condition>
</booleancondition>
<pkfield name="OBJECTCLAS"/>
<pkfield name="OBJECTID"/>
<pkfield name="CHANGENR"/>
</docreftable>

<doctable name="CDPOS">
<booleancondition logicaloperator="AND">
<condition fieldname="TABNAME"
logicaloperator="eq">
<value>VBAK</value>
</condition>
<condition fieldname="FNAME"
logicaloperator="eq">
<value>VKGRP</value>
</condition>
<condition fieldname="VALUE_NEW"
logicaloperator="in">
<value>4711</value>
<value>4712</value>
</condition>
</booleancondition>
<pkfield name="OBJECTCLAS" fktablename="CDHDR"
fkfieldname="OBJECTCLAS"/>
<pkfield name="OBJECTID" fktablename="CDHDR"
fkfieldname="OBJECTID"/>
<pkfield name="CHANGENR" fktablename="CDHDR"
fkfieldname="CHANGENR"/>
<pkfield name="TABNAME"/>
<pkfield name="TABKEY"/>
<pkfield name="FNAME"/>
<pkfield name="CHNGIND"/>
```

```
<fieldtoread name="TABNAME"/>
<fieldtoread name="CHNGIND"/>
<fieldtoread name="VALUE_NEW"/>
<fieldtoread name="VALUE_OLD"/>
</doctable>
</docspec>

<table name="CDHDR">
<pkfield name="OBJECTCLAS" fktablename="CDPOS"
fkfieldname="OBJECTCLAS"/>
<pkfield name="OBJECTID" fktablename="CDPOS"
fkfieldname="OBJECTID"/>
<pkfield name="CHANGENR" fktablename="CDPOS"
fkfieldname="CHANGENR"/>
<fieldtoread name="UDATE"/>
<fieldtoread name="UTIME"/>
<fieldtoread name="CHANGENR"/>
<fieldtoread name="USERNAME"/>
</table>
</configuration>
...
```

4.3.11.2. Extrair eventos do sistema, incluindo alterações

As informações sobre documentos de alteração das tabelas SAP **CDPOS** e **CDHDR** são armazenadas em atributos do sistema de origem especiais, de eventos do sistema que usam uma classe específica.

Os documentos de alteração são extraídos usando uma configuração adequada de uma tabela de dados. A classe **com.idsscheer.ppm.xmlextractortools.extractor.**

sap2ppm.ZChangeDocTable_sap2ppm é usada para extrair os documentos de alteração.

Se não houver documentos de alteração para eventos do sistema extraídos, somente os campos da tabela configurada serão transferidos como atributos do sistema de origem.

As seguintes condições devem ser especificadas na configuração da tabela de dados da qual os documentos de alteração devem ser extraídos:

```
<table name="..." classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.sap2ppm.
ZChangeDocTable_sap2ppm">
<booleancondition>
<condition fieldname="OBJECTCLAS" ...>
...
</condition>
<condition fieldname="OBJECTID" ...>
...
</condition>
<condition fieldname="TABNAME" ...>
...
</condition>
<booleancondition logicaloperator="OR">
<condition fieldname="TABKEY" ...>
```

```
...
</condition>
<condition fieldname="TABKEY" ...>
...
</condition>
...
</booleancondition>
</booleancondition>
...
</table>
```

Não especifique outras condições.

A tabela a seguir exibe os atributos do sistema de origem que são criados a partir das informações extraídas do documento de alteração. **CH_DOCS** foi especificado como identificador da tabela de dados na configuração.

Atributo do sistema de origem	Descrição
CH_DOCS-SUM_OF_CHANGES	Número total de alterações no documento extraído (número de operações de salvamento).
CH_DOCS-FIRST_CHANGE_DATE	Data da primeira alteração
CH_DOCS-FIRST_CHANGE_TIME	Hora da primeira alteração
CH_DOCS-LAST_CHANGE_DATE	Data da última alteração
CH_DOCS-LAST_CHANGE_TIME	Hora da última alteração
CH_DOCS-CHANGE_USER_<x>	Nome do processador que criou documentos de alteração. O atributo é criado para cada processador.**<x>** é o número consecutivo do processador.
CH_DOCS-NUM_OF_CHANGES_USER_<x>	Número total de alterações realizadas pelo processador.**<x>** é o número consecutivo do processador.

Exemplo 16.

O extrato de arquivo abaixo ilustra a configuração para extrair as alterações nas tabelas **VBAK** e **VBAP**:

```
<configuration name="SD_CHANGE_DOC_SUM">
<docspec>
<docreftable name="VBAK">
<condition fieldname="ERDAT#-#ERZET"
logicaloperator="creationtimestamp">
<value>yyyyMMdd</value>
<value>HHmmss</value>
</condition>
<pkfield name="VBELN" />
</docreftable>
<doctable name="VBAP">
<pkfield name="VBELN" fktablename="VBAK"
fkfieldname="VBELN"/>
<pkfield name="POSNR" />
<fieldtoread name="VBELN"/>
<fieldtoread name="POSNR"/>
<fieldtoread name="ERDAT"/>
<fieldtoread name="ERZET"/>
<fieldtoread name="ERNAM"/>
</doctable>
```

```
</docspec>
<table name="CH_DOCS" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.
sap2ppm.ZChangeDocTable_sap2ppm">
<booleancondition>
<condition fieldname="OBJECTCLAS">
<value>VERKBELEG</value>
</condition>
<condition fieldname="OBJECTID"
logicaloperator="eq_concatEventAttrValues">
<value>VBAP-VBELN</value>
</condition>
<condition fieldname="TABNAME" logicaloperator="in">
<value>VBAK</value>
<value>VBAP</value>
</condition>
<booleancondition logicaloperator="OR">
<condition fieldname="TABKEY"
logicaloperator="eq_concatEventAttrValues">
<value>CLIENT#-#</value>
<value>VBAP-VBELN#-#</value>
<value>VBAP-POSNR</value>
</condition>
<condition fieldname="TABKEY"
logicaloperator="eq_concatEventAttrValues">
<value>CLIENT#-#</value>
<value>VBAP-VBELN</value>
</condition>
</booleancondition>
</booleancondition>
<pkfield name="VBELN" fktablename="VBAP"
fkfieldname="VBELN"/>
<pkfield name="POSNR" fktablename="VBAP"
fkfieldname="POSNR"/>
</table>
</configuration>
```

Exemplo 17.

O operador **eq_concatEventAttrValues** compara o conteúdo do campo especificado com valores de atributo combinados dos campos especificados. Os valores especificados nos elementos XML **value** são interpretados como identificadores dos atributos do sistema de origem já extraídos.

Os valores de campos individuais (<value>...</value>) são concatenados usando a cadeia de caracteres separadora especificada **#-#** entre os valores de campos individuais. No exemplo, os valores dos campos **OBJECTID** e **TABKEY** contêm o seguinte:

Valor	Descrição	Valor de exemplo
VBAP-VBELN	O valor já extraído do campo **VBELN** na tabela **VBAP** deve corresponder como uma condição para o campo **OBJECTID**.	0000004972
Valor	Descrição	Valor de exemplo
CLIENT	Parâmetro que é substituído pelo número do cliente SAP do qual foi feita a extração.	800
VBAP-VBELN	Valor do campo extraído **VBELN** da tabela **VBAP**.	0000004972

Valor	Descrição	Valor de exemplo
VBAP-POSNR	Valor do campo extraído **POSNRVBELN** da tabela **VBAP**.	000020

Exemplo 18.

Com base nos valores de exemplo da configuração acima, os valores concatenados 800#-#0000004972#-#000020 e 800#-#0000004972 são o resultado para o campo **TABKEY** (chave da linha da tabela alterada).

O extrato do arquivo de saída abaixo ilustra um possível evento de sistema:

```
<event>
<attribute type="CH_DOCS-CHANGE_USER_1">
sapuser
</attribute>
<attribute type="CH_DOCS-CHANGE_USER_2">
ANFEL
</attribute>
<attribute type="CH_DOCS-FIRST_CHANGE_DATE">
20030919
</attribute>
<attribute type="CH_DOCS-FIRST_CHANGE_TIME">
181020
</attribute>
<attribute type="CH_DOCS-LAST_CHANGE_DATE">
20030930
</attribute>
<attribute type="CH_DOCS-NUM_OF_CHANGES_USER_1">
1
</attribute>
<attribute type="CH_DOCS-NUM_OF_CHANGES_USER_2">
3
</attribute>
<attribute type="CH_DOCS-SUM_OF_CHANGES">
4
</attribute>
<attribute type="VBAP-ERDAT">20030919</attribute>
<attribute type="VBAP-ERNAM">ANFEL</attribute>
<attribute type="VBAP-ERZET">174938</attribute>
<attribute type="VBAP-POSNR">000010</attribute>
<attribute type="VBAP-VBELN">0000007500</attribute>
</event>
```

4.3.11.3. Extrair o documento da primeira alteração de um campo de tabela

Use a classe seguinte para extrair informações sobre a primeira alteração de um campo de tabela de um sistema SAP. Especifique a classe com o atributo XML **classtouse** na configuração da tabela do elemento XML **table**:

```
com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZFirstFieldChange_sap2ppm
```

Cálculo

Para alguns documentos do sistema SAP, existe apenas um campo para a data da última alteração. O valor inicial deste campo no primeiro documento de alteração é interpretado como a data de criação do documento. Este valor é extraído, juntamente com a classe a ser usada, das tabelas do documento de alteração **CDHDR** e **CDPOS**.

As condições para usar a classe devem ser as seguintes:

```
<table name="name" classtouse="com.idsscheer.ppm.
xmlextractortools.
extractor.sap2ppm.
ZFirstFieldChange_sap2ppm">
<booleancondition>
<condition fieldname="OBJECTCLAS" ...>
...
</condition>
<condition fieldname="OBJECTID" ...>
...
</condition>
<condition fieldname="TABNAME" ...>
...
</condition>
<condition fieldname="TABKEY" ...>
...
</condition>
<condition fieldname="FNAME" ...>
...
</condition>
</booleancondition>

<pkfield name="OBJECTCLAS"/>
<pkfield name="OBJECTID"/>
<pkfield name="CHANGENR"/>
<pkfield name="TABNAME"/>
<pkfield name="TABKEY"/>
<pkfield name="FNAME"/>
<pkfield name="CHNGIND"/>

<pkfield name="<Field that is used in the tabkey>"
fktablename="<Foreign key table,  from
which the field is extracted>"
fkfieldname="<Foreign key table field>"/>
... Other fields used in the tabkey
<fieldtoread name="CDPOS-VALUE_OLD"/>
... Other fields from the CDPOS table
<fieldtoread name="CDHDR-UDATE"/>
... Other fields from the CDHDR table
</table>
```

A combinação de valores especificados nos campos **OBJECTID** e **TABKEY** é a mesma que para

```
com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZChangeDocTable_sap2ppm
```

(consulte capítulo Extrair eventos do sistema, incluindo alterações [44])

Exemplo 19.

Com essa configuração, irá extrair dados sobre quem modificou pela última vez o campo que contém o estado de aprovação das requisições de compra e quando:

```
<table name="ORIGINAL_ERDAT" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.sap2ppm.
ZFirstFieldChange_sap2ppm">
<booleancondition>
<condition fieldname="OBJECTCLAS" logicaloperator="eq">
<value>EINKBELEG</value>
</condition>
<condition fieldname="OBJECTID"
logicaloperator="eq_concatEventAttrValues">
<value>EKPO-EBELN</value>
</condition>
<condition fieldname="TABNAME" logicaloperator="eq">
<value>EKPO</value>
</condition>
<condition fieldname="TABKEY"
logicaloperator="eq_concatEventAttrValues">
<value>CLIENT#-#EKPO-EBELN#-#EKPO-EBELP</value>
</condition>
<condition fieldname="FNAME" logicaloperator="eq">
<value>AEDAT</value>
</condition>
</booleancondition>

<pkfield name="OBJECTCLAS"/>
<pkfield name="OBJECTID"/>
<pkfield name="CHANGENR"/>
<pkfield name="TABNAME"/>
<pkfield name="TABKEY"/>
<pkfield name="FNAME"/>
<pkfield name="CHNGIND"/>

<pkfield name="EBELN" fktablename="EKPO"
fkfieldname="EBELN"/>
<pkfield name="EBELP" fktablename="EKPO"
fkfieldname="EBELP"/>

<fieldtoread name="CDPOS-VALUE_OLD"/>
</table>
```

Exemplo 20.

O resultado da extração de valor com esta configuração da tabela pode ser o seguinte em um evento de sistema criado:

```
<event>
...
<attribute type="ORIGINAL_ERDAT-VALUE_OLD">
20010402
</attribute>
...
</event>
```

4.3.11.4. Extrair o documento da última alteração de um campo de tabela

Use a classe **com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZLastFieldChange_sap2ppm** para extrair informações sobre a última alteração de um documento de um sistema SAP. A configuração e cálculo da classe são idênticas às usadas para a classe **com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZFirstFieldChange_sap2ppm** (ver o capítulo Extrair o documento da primeira alteração de um campo de tabela [47]), com a diferença de que é usado o último documento de alteração em vez do primeiro para determinar os dados.

4.3.11.5. Extrair o documento da última alteração de vários campos de tabela

As informações acerca dos documentos de alteração são armazenadas nas tabelas do documento de alteração **CDHDR** (cabeçalho do documento de alteração) e **CDPOS** (elementos do documento de alteração). Esta informação pode ser extraída executando a configuração correspondente para a tabela de dados relevante (elemento XML **table**). Use a classe **com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZChangeDocTableNew_sap2ppm** para esse efeito.

Exemplo 21.

Uma configuração com condições que têm de ser especificadas em um elemento **table** poderia ter a seguinte aparência:

```
...
<table name="tablename" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.sap2ppm.
ZChangeDocTableNew_sap2ppm">
<booleancondition>
<condition fieldname="OBJECTCLAS" logicaloperator="...">
...
</condition>
<condition fieldname="OBJECTID" logicaloperator="...">
...
</condition>
<condition fieldname="TABNAME" logicaloperator="...">
...
</condition>
<condition fieldname="TABKEY" logicaloperator="...">
...
</condition>
<condition fieldname="FNAME" logicaloperator="in">
...
```

```
</condition>
</booleancondition>

<pkfield name="OBJECTCLAS"/>
<pkfield name="OBJECTID"/>
<pkfield name="CHANGENR"/>
<pkfield name="TABNAME"/>
<pkfield name="TABKEY"/>
<pkfield name="FNAME"/>
<pkfield name="CHNGIND"/>

<pkfield name="<Field that is used in TABKEY>"
fktablename="<Foreign key table, from
which the field is extracted>"
fkfieldname="<Foreign key table field>"/>
... Other fields used in TABKEY
<fieldtoread name="CDPOS-VALUE_OLD"/>
... Other fields from the CDPOS table
<fieldtoread name="CDHDR-UDATE"/>
... Other fields from the CDHDR table
</table>
...
```

Exemplo 22.

A combinação de valores especificados nos campos **OBJECTID** e **TABKEY** é a mesma que para o classe **com.idsscheer.ppm.xmlextractortools.extractor.**

sap2ppm.ZChangeDocTable_sap2ppm. Abaixo é exibido um exemplo mais concreto.

A configuração de exemplo destina-se a extrair a hora na qual um dos campos dos motivos de bloqueio de faturas logísticas foi alterado:

```
...
<table name="CDPOS" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.sap2ppm.
ZChangeDocTableNew_sap2ppm">
<booleancondition>
<condition fieldname="OBJECTCLAS" logicaloperator="eq">
<value>INCOMINGINVOICE</value>
</condition>
<condition fieldname="OBJECTID"
logicaloperator="eq_concatEventAttrValues">
<value>RSEG-BELNR#-#RSEG-GJAHR</value>
</condition>
<condition fieldname="TABNAME" logicaloperator="eq">
<value>RSEG</value>
</condition>
<condition fieldname="TABKEY"
logicaloperator="eq_concatEventAttrValues">
<value> #-#RSEG-BELNR#-#
RSEG-GJAHR#-#RSEG-BUZEI</value>
</condition>
<condition fieldname="FNAME" logicaloperator="in">
<value>SPGRP</value>
```

```
<value>SPGRM</value>
<value>SPGRT</value>
<value>SPGRG</value>
<value>SPGRV</value>
<value>SPGRQ</value>
<value>SPGRS</value>
<value>SPGRC</value>
</condition>
</booleancondition>
<pkfield name="OBJECTCLAS"/>
<pkfield name="OBJECTID"/>
<pkfield name="CHANGENR"/>
<pkfield name="TABNAME"/>
<pkfield name="TABKEY"/>
<pkfield name="FNAME"/>
<pkfield name="CHNGIND"/>
<pkfield name="BELNR" fktablename="RSEG"
fkfieldname="BELNR"/>
<pkfield name="GJAHR" fktablename="RSEG"
fkfieldname="GJAHR"/>
<pkfield name="BUZEI" fktablename="RSEG"
fkfieldname="BUZEI"/>
</table>
...
```

Exemplo 23.

Um evento de sistema gerado com essa configuração de exemplo poderia ter a seguinte aparência:

```
...
<event>
<attribute type="CDPOS-CHANGE_USER_1">HUETT</attribute>
<attribute type="CDPOS-FIRST_CHANGE_DATE">20010220
</attribute>
<attribute type="CDPOS-FIRST_CHANGE_TIME">154943
</attribute>
<attribute type="CDPOS-LAST_CHANGE_DATE">20010220
</attribute>
<attribute type="CDPOS-LAST_CHANGE_TIME">154943
</attribute>
<attribute type="CDPOS-NUM_OF_CHANGES_USER_1">1
</attribute>
<attribute type="CDPOS-SUM_OF_CHANGES">1</attribute>
...
<event>
...
```

4.3.12. Extrair a primeira ou última linha de uma classificação

A classe **com.idsscheer.ppm.xmlextractortools.**

extractor.sap2ppm.ZSortWithInteger_sap2ppm a ser usada extrai registros de dados de tabelas, classifica-os usando os critérios de classificação especificados e grava o primeiro ou último registro de dados nos

eventos de sistema a serem gerados. A classe tem de ser especificada no atributo **classtouse** do elemento XML **table**.

Exemplo 24.

Os documentos de histórico do estado representam um tipo de documento especial do sistema SAP R/3. Estes documentos são armazenados na tabela **JCDS**. A classe deve determinar a última alteração relacionada com um tipo de estado específico e gravar essa alteração nos eventos do sistema correspondentes. Para isso, todos os números de pedido são extraídos da tabela **AUFK**. Além disso, para cada número de pedido com informações de estado, todas as entradas desse número de pedido com o prefixo **OR** e o estado **I0043** (campo **STAT**) são extraídas da tabela **JCDS**. Estes são classificados por hora de criação (**UDATE**, **UTIME**) e o último registro de dados criado deve ser extraído. Todos os valores de campo pertencentes a este registro de dados são gravados no evento de sistema relevante.

O extrato de arquivo abaixo ilustra a configuração para extrair informações de estado da tabela **JCDS**, nas condições especificadas:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="AUFK_JCDS">
<docspec>
<doctable name="AUFK">
<pkfield name="AUFNR"/>
</doctable>
</docspec>
<table name="JCDS_I0043" tablename="JCDS"
classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.
sap2ppm.ZSortWithInteger_sap2ppm">
<parameter name="SORTCRITERION">
<value>UDATE</value>
<value>UTIME</value>
</parameter>
<parameter name="USE">
<value>MAX</value>
</parameter>
<condition fieldname="STAT"
logicaloperator="eq">
<value>I0043</value>
</condition>
<pkfield name="OBJNR" fktablename="AUFK"
fkfieldname="AUFNR">
<prefix>
<value>OR</value>
</prefix>
</pkfield>
<fieldtoread name="CDTCODE"/>
<fieldtoread name="CHGNR"/>
<fieldtoread name="CHIND"/>
<fieldtoread name="INACT"/>
<fieldtoread name="OBJNR"/>
<fieldtoread name="STAT"/>
<fieldtoread name="TCODE"/>
<fieldtoread name="USNAM"/>
```

```
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

Exemplo 25.

Um evento de sistema extraído com essa configuração poderia ter a seguinte aparência:

```
<event>
<attribute type="AUFK-AUFNR">000000800002</attribute>
<attribute type="JCDS_I0043-CDTCODE"></attribute>
<attribute type="JCDS_I0043-CHGNR">002</attribute>
<attribute type="JCDS_I0043-CHIND">U</attribute>
<attribute type="JCDS_I0043-INACT">X</attribute>
<attribute type="JCDS_I0043-OBJNR">
OR000000800002
</attribute>
<attribute type="JCDS_I0043-STAT">I0043</attribute>
<attribute type="JCDS_I0043-TCODE">KOK2</attribute>
<attribute type="JCDS_I0043-UDATE">19961217</attribute>
<attribute type="JCDS_I0043-USNAM">MUELLERJ</attribute>
<attribute type="JCDS_I0043-UTIME">192122</attribute>
</event>
```

Exemplo 26.

Os critérios de classificação (neste caso **UDATE**, **UTIME**) são automaticamente incluídos na especificação do evento de sistema.

O modelo para criar a configuração exibida é semelhante a isso (os elementos e atributos XML que são opcionais estão destacados na tabela abaixo):

```
<table name="..." tablename="..." classtouse=
"com.idsscheer.ppm.
xmlextractortools.extractor.
sap2ppm.ZSortWithInteger_sap2ppm">
<parameter name="SORTCRITERION">
<value>...</value>
<value>...</value>
...
</parameter>
<parameter name="USE">
<value>...</value>
</parameter>
...
See chapter
R3 table configuration (system event specification) [15]
...
</table>
```

A tabela abaixo reitera as entradas de configuração mais importantes para a definição **table** acima:

Elemento/atributo XML	Valor: Descrição
name	O nome especificado é adicionado como um prefixo aos atributos do sistema de origem extraídos
tablename **(opcional)**	Tabela da qual a informação deve ser extraída. Valor padrão: Valor do atributo XML **name**
classtouse	**com.idsscheer.ppm.xmlextractortools.** **extractor.sap2ppm.ZSortWithInteger_sap2ppm**: Classe Java a ser usada
nome do parâmetro	
SORTCRITERION	Primeiro parâmetro a ser especificado. Tem de ser especificado pelo menos um elemento XML **value** (nome de campo do critério de classificação). Os campos especificados apenas podem incluir valores inteiros. Se vários critérios de classificação forem especificados, eles serão priorizados de cima para baixo, ou seja, os registros de dados serão comparados em primeiro lugar com o primeiro critério (superior) e depois com o segundo critério, se nenhuma classificação completa tiver sido criada, e assim por diante.
USE	Segundo a ser especificado. Tem de ser especificado um único elemento XML **value**. Valores válidos: **MIN** (é selecionado o registro de dados com o menor valor inteiro para o critério de classificação) **MAX** (é selecionado o registro de dados com o maior valor inteiro para o critério de classificação) O primeiro ou o último registro de dados da lista gerada com base nos critérios de classificação especificados é selecionado para ser extraído.

Deve usar essa classe para extrair a primeira ou a última linha de uma classificação com moderação, pois as operações de classificação e seleção resultam em perda de desempenho e memória.

4.3.13. Criar atributos com data invertida

Pode usar a classe **com.idsscheer.ppm.xmlextractortools.extractor. sap2ppm.ZTableInvertDates_sap2ppm** para criar novos atributos de evento do sistema, com datas invertidas, quando extrair dados de um sistema SAP. Os valores de atributos de data já extraídos serão invertidos. Uma data invertida é calculada extraindo o valor da data no formato SAP, por exemplo, 20110529 para 29 de julho de 2011, a partir de 99999999. Os novos eventos de sistema criados podem ser referenciados em outras configurações da tabela de dados (elemento XML **table**).

```
<docspec>
<doctable name="VBAP">
<pkfield name="VBELN" />
<pkfield name="POSNR" />
<pkfield name="ERDAT"/>
<pkfield name="AEDAT"/>
<pkfield name="ABDAT"/>
</doctable>
</docspec>
<table name="VBAP_DATES_INVERTED" tablename="VBAP_DATES_INVERTED"
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZTableInvertDa
tes_sap2ppm">
<pkfield name="ERDAT" fktablename="VBAP" fkfieldname="ERDAT" />
<pkfield name="AEDAT" fktablename="VBAP" fkfieldname="AEDAT" />
</table>
```

Se estiver a usar a classe **ZTableInvertDates_sap2ppm**, não precisa de especificar um elemento **fieldtoread**. Qualquer elemento **fieldtoread** especificado será ignorado. Os novos atributos de evento do sistema consistem no valor do atributo **name** do elemento **table** e no valor do atributo **name** do elemento **pkfield**. No exemplo, são criados dois novos atributos de evento do sistema, **VBAP_DATES_INVERTED-ERDAT** e **VBAP_DATES_INVERTED-AEDAT**.

Pode usar as operações normais **fkpart**, **prefix** e **postfix** para os elementos **pkfield**.

Exemplo 27.

Um evento de sistema criado com a configuração acima poderia ter a seguinte aparência:

```
...
<event>
<attribute type="VBAP-ABDAT">00000000</attribute>
<attribute type="VBAP-AEDAT">19970127</attribute>
<attribute type="VBAP-ERDAT">19970121</attribute>
<attribute type="VBAP-POSNR">000010</attribute>
<attribute type="VBAP-VBELN">0000004974</attribute>
<attribute type="VBAP_DATES_INVERTED-AEDAT">80029872</attribute>
<attribute type="VBAP_DATES_INVERTED-ERDAT">80029878</attribute>
</event>
...
```

Pode usar a classe **ZTableInvertDates_sap2ppm** para converter uma data invertida para uma data normal em formato SAP.

```
<docspec>
<doctable name="VBAP">
<pkfield name="VBELN" />
<pkfield name="POSNR" />
<pkfield name="ERDAT"/>
<pkfield name="AEDAT"/>
<pkfield name="ABDAT"/>
</doctable>
</docspec>
<table name="VBAP_DATES_INVERTED" tablename="VBAP_DATES_INVERTED"
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZTableInvertDa
tes_sap2ppm">
<pkfield name="ERDAT" fktablename="VBAP" fkfieldname="ERDAT" />
<pkfield name="AEDAT" fktablename="VBAP" fkfieldname="AEDAT" />
</table>
<table name="VBAP_DATES_INVERT_AGAIN" tablename="VBAP_DATES_INVERTED"
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZTableInvertDa
tes_sap2ppm">
<pkfield name="ERDAT" fktablename="VBAP_DATES_INVERTED" fkfieldname="ERDAT" />
<pkfield name="AEDAT" fktablename="VBAP_DATES_INVERTED" fkfieldname="AEDAT" />
<pkfield name="ABDAT" fktablename="VBAP_DATES_INVERTED" fkfieldname="ABDAT" />
</table>
```

Exemplo 28.

Retorna o resultado seguinte:

```
...
<event>
```

```
<attribute type="VBAP-ABDAT">00000000</attribute>
<attribute type="VBAP-AEDAT">19970127</attribute>
<attribute type="VBAP-ERDAT">19970121</attribute>
<attribute type="VBAP-POSNR">000010</attribute>
<attribute type="VBAP-VBELN">0000004974</attribute>
<attribute type="VBAP_DATES_INVERTED-AEDAT">80029872</attribute>
<attribute type="VBAP_DATES_INVERTED-ERDAT">80029878</attribute>
<attribute type="VBAP_DATES_INVERT_AGAIN-AEDAT">19970127</attribute>
<attribute type="VBAP_DATES_INVERT_AGAIN-ERDAT">19970121</attribute>
</event>
...
```

4.3.14. Multiplicação de eventos do sistema em tabelas

Ao extrair de um sistema SAP, tem a opção de criar eventos adicionais do sistema com base em uma tabela de eventos do sistema.

No nível do sistema de origem, pode existir um relacionamento 1: n que não pode ser resolvido ao determinar os eventos do sistema. Use a classe **com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZTableMultiplyEvents_sap2ppm** para criar vários eventos de sistema a partir de um evento do sistema e resolver assim a relação 1:n.

```
<configuration name="MultiplyEvents">
<docspec>
<doctable name="VBAK" tablename="VBAK">
<condition fieldname="VBELN" logicaloperator="in">
<value>0000006662</value>
<value>0000006741</value>
</condition>
<pkfield name="VBELN" />
</doctable>
</docspec>
<table name="VBAP"
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZTableMultiply
Events_sap2ppm">
<pkfield name="VBELN" fktablename="VBAK" fkfieldname="VBELN"/>
<fieldtoread name="POSNR"/>
<fieldtoread name="MATNR"/>
</table>
<table name="MAKT">
<condition fieldname="SPRAS" logicaloperator="eq">
<value>de</value>
</condition>
<pkfield name="MATNR" fktablename="VBAP" fkfieldname="MATNR"/>
<fieldtoread name="MAKTX"/>
</table>
</configuration>
```

Exemplo 29.

Este é o arquivo de saída de evento do sistema:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAK-VBELN">0000006662</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">Flatscreen MS 1460 P</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-06</attribute>
<attribute type="VBAP-POSNR">000010</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">Flatscreen MS 1775P</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-10</attribute>
<attribute type="VBAP-POSNR">000020</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">MAG PA/DX 175</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-14</attribute>
<attribute type="VBAP-POSNR">000030</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">Jotachi SN4500</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-18</attribute>
<attribute type="VBAP-POSNR">000040</attribute>
</event>
</eventlist>
```

Exemplo 30.

O exemplo seguinte (excluindo a classe descrita) explica o procedimento durante a extração:

```
<configuration name="MultiplyEvents_Doctable_Only">
<docspec>
<doctable name="VBAK" tablename="VBAK">
<condition fieldname="VBELN" logicaloperator="in">
<value>0000006662</value>
<value>0000006741</value>
</condition>
<pkfield name="VBELN" />
</doctable>
</docspec>
<table name="VBAP" >
<pkfield name="VBELN" fktablename="VBAK" fkfieldname="VBELN"/>
<fieldtoread name="POSNR"/>
<fieldtoread name="MATNR"/>
</table>
<table name="MAKT">
<condition fieldname="SPRAS" logicaloperator="eq">
<value>de</value>
```

```
</condition>
<pkfield name="MATNR" fktablename="VBAP" fkfieldname="MATNR"/>
<fieldtoread name="MAKTX"/>
</table>
</configuration>
```

Uma vez que várias entradas na tabela **VBAP** são associadas a uma entrada na tabela **VBAK**, apenas uma linha aleatória é extraída da tabela **VBAP**.

Este é o arquivo de saída de evento do sistema:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAK-VBELN">0000006662</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">Flatscreen MS 1775P</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-10</attribute>
<attribute type="VBAP-POSNR">000020</attribute>
</event>
</eventlist>
```

4.3.15. Extrair campos ou tabelas individuais para análise de dados

Para permitir a fácil extração de dados para análise de dados, pode extrair todo o conteúdo de uma tabela do sistema de origem para um arquivo no formato de evento, que pode ser importado para um território de análise para análise de dados.

Também pode restringir os dados a serem extraídos [61] definindo condições para a extração de dados.

A tabela a ser extraída não é configurada usando a configuração da tabela, mas o próprio arquivo da fonte de dados. Para isso, o arquivo **datasource.dtd** contém as entradas seguintes:

Elemento/atributo XML	Descrição
analysistype	Atributos XML: Tem de ter o valor **DATA_ANALYTICS**.
realmtable	Elemento XML abrangente para configurar a fonte de dados para análise de dados.
tablename	Atributo XML do elemento **realmtable**: Nome da tabela no sistema de origem
sourcetable	Elemento XML abrangente para configurar a tabela de fonte de dados para análise de dados. Tem de incluir pelo menos um elemento **sourcefield**.
tablename	Atributo XML do elemento **sourcetable**: Nome da tabela no sistema de origem
sourcefield	Contém o nome do campo da tabela do sistema de origem.

O atributo **analysistype** do elemento XML **datasource** tem de ter o valor **DATA_ANALYTICS** se o elemento **<realmtable>** especificar uma tabela de território de análise (valor padrão: PROCESS).

O atributo **tablename** do elemento **<realmtable>** indica o nome da tabela de destino na configuração do território de análise. O nome da tabela não afeta a extração em si, mas é avaliado apenas pela importação do PPM.

Pode encontrar mais informações sobre importação de dados para análise no manual do usuário para análise de dados do PPM.

O elemento XML **<realmtable>** contém o elemento opcional **<sourcetable>** que especifica a tabela a ser extraída. As colunas a serem extraídas desta tabela têm de ser especificadas no elemento **<sourcefield>**. O elemento **<sourcetable>** é opcional. Para o Extrator JDBC ou SAP, devem ser especificadas uma única tabela de origem e pelo menos uma coluna de origem. Caso contrário, uma mensagem de erro será emitida durante a análise do arquivo da fonte de dados.

Uma definição de fonte de dados apenas pode ter, no máximo, uma tabela. É impossível restringir o número de linhas. Todas as linhas são extraídas, incluindo todas as linhas com valores idênticos nas colunas a serem extraídas. Por exemplo, se as colunas **Primeiro nome** e **Sobrenome** tiverem de ser extraídas e a tabela incluir dez entradas **Peter** e **Schmidt**, serão criados dez eventos com valores de atributo idênticos.

O exemplo seguinte explica a configuração:

```
<realmtable tablename="COMPANY_EMPLOYEE">
<sourcetable tablename="EMPLOYEE">
<sourcefield>EMPLOYEE_ID</sourcefield>
<sourcefield>NAME</sourcefield>
</sourcetable>
...
</realmtable>
<dataextraction>
<outputfilename>..\custom\testclient\data\employee.xml</outputfilename>
</dataextraction>
...
<systemconfig>..\custom\testclient\SourceSystemConfig.xml</systemconfig>
```

Em contraste com o comportamento do Extrator convencional JDCB ou SAP, os atributos não recebem mais um nome de tabela como um prefixo no arquivo de saída do evento. Por exemplo, se a tabela **EMPLOYEE** foi extraída, normalmente o extrator gera eventos do tipo **<table name>-<column name>**:

```
<event>
<attribute type="EMPLOYEE-EMPLOYEE_ID">4711</attribute>
<attribute type="EMPLOYEE-NAME">Schmidt</attribute>
</event>
Extracting a table using the <realmtable> element, however, creates only events
without table names:
<event>
<attribute type="EMPLOYEE_ID">4711</attribute>
<attribute type="NAME">Schmidt</attribute>
</event>
```

Valores null no modo ANÁLISE DE DADOS

Se o valor de uma coluna for **null** no momento em que uma tabela do território de análise é extraída, esse valor não é gravado no evento. Se uma linha **EMPLOYEE_ID = 4712** existir sem sobrenome, o extrator cria os eventos seguintes.

```
<event>
<attribute type="EMPLOYEE_ID">4711</attribute>
<attribute type="NAME">Schmidt</attribute>
</event>
<event>
<attribute type="EMPLOYEE_ID">4712</attribute>
</event>
However, if all column values to be extracted are null, the event is written
with empty attributes:
<event>
<attribute type="EMPLOYEE_ID">4711</attribute>
<attribute type="NAME">Schmidt</attribute>
</event>
<event>
<attribute type="EMPLOYEE_ID"></attribute>
<attribute type="NAME"></attribute>
</event>
```

Se existirem várias dessas linhas, elas serão - em contraste com o procedimento de extração comum (analysistype = PROCESS) - transferidas devidamente, de modo que existam tantos elementos <event> no arquivo de eventos, como linhas na tabela de dados do sistema de origem.

4.3.15.1. Restringir extração de dados

Se quiser extrair apenas uma parte específica do conteúdo da tabela, poderá definir as condições de extração de dados. Pode limitar o volume de dados a ser extraído usando um carimbo de data/hora ou valor inteiro (por exemplo, uma sequência).

Para isso, especifique as condições obrigatórias na configuração da fonte de dados, usando o elemento **condition** e o atributo **dataextractiontype**.

```
<datasource name="VBAP" type="JDBC" analysistype="DATA_ANALYTICS"
dataextractiontype="TIME_BASED">
<realmtable tablename="VBAP">
<sourcetable tablename="VBAP2">
<condition logicaloperator="char_creationtimestamp" fieldname="AEDAT">
<value>yyyy-MM-dd</value>
</condition>
<sourcefield>AEDAT</sourcefield>
<sourcefield>ERDAT</sourcefield>
<sourcefield>ERNAM</sourcefield>
<sourcefield>ERZET</sourcefield>
<sourcefield>MATKL</sourcefield>
<sourcefield>MATNR</sourcefield>
<sourcefield>WERKS</sourcefield>
</sourcetable>
</realmtable>
...
</datasource>
```

O atributo **dataextractiontype** pode ter os valores seguintes nas fontes de dados para análise de dados:

- COMPLETE: A extração é executada com um critério de tempo da data **01.01.1990 00:00:00** e um critério de inteiro do valor **0**. Esta é também a configuração padrão se o atributo não existir.
- TIME_BASED: Tem de ser executada uma extração de dados baseada em um carimbo de data/hora.
- VALUE_BASED: Tem de ser executada uma extração de dados baseada em um valor.

Não há verificação se a condição de extração de dados usada corresponde ao tipo de extração de dados configurado (dataextractiontype=). Se a configuração for criada usando o PPM CTK, o PPM CTK garante valores corretos. Se o arquivo for configurado manualmente e uma condição que não corresponde for usada, pode ocorrer um erro durante a extração de dados.

4.3.16. Tabelas de extração com colunas de chave/valor

É possível usar uma classe de extração especial para extrair tabelas que contém células com pares de chave/valor. Um novo atributo de evento é criado para cada chave extraída de um evento.

O exemplo a seguir mostra uma tabela no sistema de origem que contém atributos de valor de chave referentes a um evento.

ID (chave)	TYPE	TEXT	CREATOR	STAT	...
000000532657	F	Crie uma justificativa para o ajuste do benefício do empregado 00001448 devido a qualquer evento.	Criando uma justificativa de ajuste devido a um evento...	COMPLETED	...
...	...	...	...	...	...

Exemplo 31.

Para cada um desses elementos de trabalho (ou seja, para cada linha da tabela "WF_ITEMS"), há atributos arbitrários armazenados na tabela "WF_ITEM_ATTRIBUTES" como chave/valor:

ID (chave)	ATTRIBUTE (chave)	VALUE	...
000000532657	DATEOFVALIDITY	20020101	...
000000532657	EMPLOYEENUMBER	00001448	...
000000532657	EVENT	CREATED	...
000000532657	OBJECTTYPE	FAMILY	...
000000532657	_WF_INITIATOR	USWF-BATCH	...
000000532657	_WF_PRIORITY	5	...
...	...	...	...

Classe de extração

com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZTableKeyValueFieldsToAttributes_sap2ppm

Essa classe de extração cria um novo atributo para o evento do sistema de origem a partir de cada valor de chave encontrado, e o valor se torna o valor do atributo. Para essa classe, é necessário especificar os nomes da coluna de chave e valor como parâmetros separados por "#-#".

Todos os campos de chave e valor devem ser especificados como campos a serem lidos (elemento XML **fieldtoread**). Não são permitidos campos adicionais para serem lidos.

A seguinte configuração se refere à classe de extração JDBC mencionada acima:

```
<xmlextractor_tableconfiguration>
    <configuration name="workflow">
        <docspec>
            <doctable name="WF_ITEMS" tablename="WF_ITEMS">
                <pkfield name="ID" />
                <pkfield name="TYPE" />
                <pkfield name="CREATOR" />
                <pkfield name="TEXT" />
            </doctable>
        </docspec>
        <table name="WF_ITEM_ATTRIBUTES" tablename="WF_ITEM_ATTRIBUTES" classtouse=
        "com.idsscheer.ppm.xmlextractortools.extractor.sap2ppm.ZTableKeyValueFieldsToAttributes_sap2ppm">
            <parameter name="KEYVALUEFIELDS">
                <value>ATTRIBUTE#-#VALUE</value>
            </parameter>
            <pkfield name="ID" fktablename="WF_ITEMS" fkfieldname="ID" />
            <fieldtoread name="ATTRIBUTE" />
            <fieldtoread name="VALUE" />
        </table>
    </configuration>
</xmlextractor_tableconfiguration>
```

Exemplo 32.

O seguinte exemplo de evento foi lido com a configuração acima:

```
<event>

  <attribute type="WF_ITEMS-ID">000000532656</attribute>

  <attribute type="WF_ITEMS-CREATOR">Creating an adjustment reason due to
event...</attribute>

  <attribute type="WF_ITEMS-STAT">COMPLETED</attribute>

  <attribute type="WF_ITEMS-TEXT">Create a benefit adjustment reason for
employee 00001448 due to any event.</attribute>

  <attribute type="WF_ITEMS-TYPE">F</attribute>

  ' <attribute type="WF_ITEM_ATTRIBUTES-DATEOFVALIDITY ">20020101</attribute>

  <attribute type="WF_ITEM_ATTRIBUTES-EMPLOYEENUMBER ">00001448</attribute>

  <attribute type="WF_ITEM_ATTRIBUTES-EVENT ">CREATED</attribute>

  <attribute type="WF_ITEM_ATTRIBUTES-OBJECTTYPE ">FAMILY</attribute>

  <attribute type="WF_ITEM_ATTRIBUTES-_WF_INITIATOR ">USJOSWIGT</attribute>

  <attribute type="WF_ITEM_ATTRIBUTES-_WF_PRIORITY ">5</attribute>

  </event>
```

Caracteres especiais nas chaves

Os valores das colunas de chave são usados no atributo XML **tipo** dos elementos XML **atributo**. Esses valores podem conter caracteres que não podem ser usados nesse atributo XML. Por isso, os valores do

atributo **tipo** do elemento XML **atributo** são verificados em relação à presença de caracteres XML inválidos, e esses caracteres são filtrados, se necessário.

4.4. Programa de linha de comando

Você criou o(s) arquivo(s) de saída XML usando o programa de linha de comando **runsap2ppm.bat**.

Chamar o programa sem parâmetros ou com **-h** ou **-?** gera a ajuda on-line no console, descrevendo todas as opções disponíveis.

4.4.1. Argumentos do programa de linha de comando

4.4.1.1. Argumentos gerais

-version

Exibe o número de versão do extrator de processo PPM usado. Outros argumentos especificados são ignorados.

-information yes|no

Especifica se a informação é para ser exibida (yes) ou não (no) durante a importação. O padrão é **yes**.

-warning yes|no

Especifica se as mensagens de aviso são para ser exibidas (yes) ou não (no) durante a importação. O padrão é **yes**.

-error yes|no

Aqui pode especificar se as mensagens de erro são para ser exibidas (yes) ou não (no) durante a importação. O padrão é **yes**.

-protocolfile <file name>

Especifique o arquivo de log para o qual todas as mensagens são gravadas durante a importação.
Se especificar um arquivo, somente as mensagens de erro críticas que resultam em cancelamento do programa serão exibidas na tela.

-language <ISO code>

Especifique o idioma no qual as informações de log devem ser exibidas.

4.4.1.2. Argumentos específicos do sistema de origem

-datasource <file name> (opcional)

Especifique a fonte de dados que deseja usar para a extração. O arquivo XML contém especificações de todos os arquivos a serem usados para extração e saída XML. Certifique-se de que usa este parâmetro em vez de e não em conjunto com os parâmetros **-datasourcelist**, **-systemconfig**, **-tableconfig**, **-calcconfig**, **-outfile** ou **-nozip**.

-datasourcelist <file name>

Várias fontes de dados podem ser extraídas simultaneamente através do argumento **-datasourcelist**. Isso corresponde a várias extrações consecutivas usando o argumento **-datasource**. São extraídas apenas as fontes de dados SAP.

Ver o capítulo Extrair várias fontes de dados [70].

-systemconfig <filename> <configname>

É aqui que especifica o nome do arquivo XML contendo a(s) configuração(ões) do sistema. O conteúdo do arquivo é específico do sistema de origem e pode conter várias configurações. O nome da configuração a ser usada é especificado no segundo argumento. Se o arquivo XML incluir apenas uma configuração, esta é usada automaticamente. Nesse caso, não é necessário especificar um nome de configuração.

-tableconfig <file name> <config name>

Especifique o nome do arquivo XML contendo a(s) configuração(ões) da tabela. O conteúdo do arquivo é específico do sistema de origem e pode conter várias configurações. O nome da configuração da tabela é especificado no segundo argumento. Se o arquivo XML incluir apenas uma configuração, esta é usada automaticamente. Nesse caso, não é necessário especificar um nome de configuração.

-calcconfig <file name> (opcional)

Especifique o nome do arquivo XML com o qual pode alterar atributos do arquivo de saída XML ou adicionar atributos, incluindo transformações de atributos.

-begindate <dd.mm.yyyy> (opcional)

Especifique a data a partir da qual os dados devem ser extraídos do sistema de origem. Se não usar este parâmetro, é usado o valor do atributo XML **lastreaddate** da fonte de dados ou configuração de sistema especificada. O formato SAP **yyyymmdd** é suportado.

-begintime <hh:mm:ss> (opcional)

Especifique a hora a partir da qual os dados devem ser extraídos do sistema de origem. Se não usar este parâmetro e **-begindate**, é usado o valor do atributo XML **lastreadtime** da fonte de dados ou configuração de sistema especificada. Se não usar o parâmetro, mas usar **-begindate**, será definido o valor padrão **000000**.

-enddate <dd.mm.yyyy> (opcional)

Especifique a data até à qual os dados devem ser extraídos do sistema de origem. Se não usar este parâmetro, será usada a data atual. O formato SAP **yyyymmdd** é suportado.

-endtime <hh:mm:ss> (opcional)

Especifique a hora até à qual os dados devem ser extraídos do sistema de origem. Se não usar o parâmetro, mas usar

-enddate, **235959** será definido como o valor padrão. Se não usar este parâmetro, nem **-enddate**, será usada a hora atual.

Se o PPM Process Extractor SAP-2-PPM for iniciado sem que seja especificado um intervalo de tempo, a hora de início será extraída do arquivo de configuração especificado. A data e hora atuais são usadas como hora de término.

-valueconstraint <parameter1> ... <parameter4> (opcional)

Use este parâmetro para delimitar o volume de dados a ser extraído usando um critério de número inteiro. São permitidos os operadores de comparação "<", "<=", ">" e ">=". Pode comparar o valor do campo correspondente com um ou dois valores (intervalo). Se um último valor extraído (**lastreadvalue**) já tiver sido salvo na configuração do sistema ou fonte de dados, também poderá usar este valor para a comparação (ver o capítulo **Extração automática e contínua [69]**).

```
-valueconstraint 25000020 "<="
```

Exemplo 33.

Apenas são extraídos registros de dados cujo valor inteiro para o campo determinado na especificação do evento seja **menor do que** ou **igual ao** valor **25000020**.

```
-valueconstraint 15000020 ">" 25000020 "<"
```

Exemplo 34.

Intervalo: Somente são extraídos registros de dados cujo valor inteiro para o campo determinado na especificação do evento seja **menor do que** o valor **25000020** e **maior do que** o valor **15000020**.

-save_value_minimum (opcional)

Use este parâmetro para especificar que com **-valueconstraint**, o menor valor da última leitura é salvo como **lastreadvalue** no arquivo de configuração usado (configuração do sistema ou fonte de dados) ao extrair com restrições. Valor padrão: Salvar o valor maior

-cpd <int> (opcional)

Esse parâmetro é usado para especificar o número de eventos do sistema a serem extraídos simultaneamente (documentos processados simultaneamente). O valor especificado extrai gradualmente os campos de chave primária da tabela referenciada usando o elemento XML **doctable** na configuração da tabela e os armazena na memória Java do extrator. A especificação de um valor mais alto resulta em melhor desempenho de extração e uso de memória mais eficiente.

Se não especificar o parâmetro, é usado o valor padrão recomendado de 1000.

-ping (opcional)

Este parâmetro é usado para testar a conexão ao sistema de origem especificado. Não é extraído qualquer dado.

-rfc_blockmode {G|D|M} (opcional)

Este parâmetro permite usar a extração de dados em bloco (ver o capítulo Extração em bloco de grandes volumes de dados [40]). Especifique o modo adequado como argumento. Modos de bloco disponíveis:

G (tabela interna do grupo de funções)

M (memória ABAP)

D (tabela do banco de dados)

-progress {yes|no} (opcional)

Esse parâmetro é usado para especificar se deseja ou não gerar mensagens de log adicionais sobre o progresso da extração de dados em bloco. Valor padrão: **yes**

-rfc_callretrys <number> (opcional)

Use esse parâmetro para especificar o número de vezes que as consultas ao sistema R/3 serão repetidas se ocorrer um erro ou se a conexão falhar. Quando usar os modos de bloco **G** e **M**, o número especificado deve ser **0**. O parâmetro é eficaz globalmente e não apenas para extração em bloco. Valor padrão: **0**

-rfc_delimiter <character> (opcional)

Use esse parâmetro para especificar o caractere a ser usado para separar os valores extraídos. O caractere pode não estar contido nos próprios valores extraídos. Valor padrão: **;**

-removeempty (opcional)

Use este parâmetro para remover os atributos extraídos sem valores antes da transformação de atributos.

-removezerodates (opcional)

Use este parâmetro para remover os valores dos campos da data SAP, ou seja, campos de tabela SAP do tipo **DATS** ou **D** que contenham apenas zeros. Os valores serão removidos dos eventos logo após a extração de dados.

Exemplo 35.

<attribute type="VBAP-ABDAT">00000000</attribute>

4.4.1.3. Argumentos específicos do arquivo de saída

-outfile <file name>

Esse parâmetro é usado para especificar o nome do arquivo de saída XML. A extensão de arquivo é adicionada automaticamente. Por padrão, o arquivo é fornecido como um arquivo ZIP.

-outfileencoding <encoding> (opcional)

Esse parâmetro é usado para especificar a codificação do arquivo de saída XML. Valor padrão: UTF-8

-nozip (opcional)

Use esse parâmetro para especificar que o arquivo de saída deve ser gerado como um arquivo XML em vez de um arquivo ZIP.

-pikidatamapping <file name> <pc name> (opcional)

Use esse parâmetro para especificar o nome do arquivo XML que contém o mapeamento no primeiro argumento. No segundo argumento, especifique o nome da série de medidas na qual as séries de medidas independentes da instância de processo devem ser salvas.

-dimdatamapping <file name> <dim name> (opcional)

Use esse parâmetro para especificar o nome do arquivo XML que contém o mapeamento no primeiro argumento. No segundo argumento, especifique o nome da dimensão para a qual os valores extraídos serão subsequentemente importados.

-sorteventattributes (opcional)

Use esse parâmetro para classificar os atributos do sistema de origem nos eventos do sistema em ordem alfanumérica, usando o tipo de atributo. Esse parâmetro pode ter um impacto negativo na velocidade de extração para grandes volumes de dados.

4.4.1.4. Extração automática e contínua

Pode extrair dados de forma completa e automática eliminando os parâmetros **-begindate**, **-begintime**, **-enddate**, **-endtime** na linha de comando ou usando **-valueconstraint**, sem valor, para o primeiro operador de comparação. Se o intervalo de dados a ser extraído for restrito usando valores inteiros, o menor ou o maior dos últimos valores extraídos será salvo ou atualizado no arquivo de configuração utilizado. Se houver restrições de tempo no intervalo de dados a ser extraído, o horário da última extração de dados será salvo ou atualizado no arquivo de configuração.

Devem ser atendidas as seguintes condições:

- Uma configuração correta para **creationtimestamp** e **valueconstraint**, para os dados a serem extraídos, é especificada na especificação de evento do sistema (ver o capítulo **Operadores de condição [29]**)
- Use sempre a mesma chamada de linha de comando para extração automatizada contínua, por exemplo:

```
runsap2ppm -datasource datasource.xml -valueconstraint ">"
```

- Quando usa o parâmetro **-valueconstraint**, certifique-se de que salva o maior dos últimos valores extraídos para todas as operações de extração (padrão) ou o menor dos últimos valores extraídos, especificando **-save_value_minimum**. O valor de **lastreadvalue** no arquivo de configuração apenas é atualizado se o maior dos últimos valores extraídos for maior do que o valor salvo atualmente quando executa a linha de comando sem o parâmetro **-save_value_minimum**, ou se o menor dos últimos valores extraídos for menor do que o valor salvo atualmente quando usa **-save_value_minimum**.

```
-valueconstraint ">="
```

Exemplo 36.

Apenas são extraídos os registros de dados cujo valor inteiro para restringir a área de dados a ser extraída seja **maior do que** ou **igual ao** valor salvo como o último valor extraído (**lastreadvalue**) na configuração do sistema ou fonte de dados. Se nenhum valor for salvo, o valor padrão **0** é usado (no exemplo, todos os registros de dados com valores de critério inteiro **>=0** seriam extraídos). O maior dos últimos valores extraídos (**300** neste exemplo) é salvo como **lastreadvalue** de acordo com a configuração padrão (ver o parâmetro **-save_value_minimum**).

```
-valueconstraint ">=" 270 "<="
```

Exemplo 37.

São extraídos todos os registros de dados com um valor inteiro de **>=40** e **<=270** para restringir a área de dados a ser extraída. Após a extração, o valor para **lastreadvalue** é atualizado, por exemplo, para **270** se este for realmente o maior dos últimos valores extraídos.

Ao chamar

```
runsap2ppm -datasource datasource.xml -valueconstraint ">"
```

Exemplo 38.

novamente, são extraídos todos os registros de dados cujo valor inteiro para delimitar a área de dados a ser extraída seja maior do que **270** e cujo carimbo de data/hora seja tão antigo ou mais recente que a hora de início da extração de dados (**31.12.1997 15:59:59**).

A hora de início da extração de dados é inserida como **lastreaddate**/**lastreadtime** no arquivo de configuração.

Se eliminar os parâmetros **-begindate**/**-begintime**, extrai sempre os registros de dados cujos carimbos de data/hora são mais recentes (maiores) do que os especificados e extraídos em último lugar usando **lastreaddate**/**lastreadtime**. A data e hora atual são inseridas como lastreaddate/lastreadtime se não usar os parâmetros -enddate e -endtime.

4.5. Extrair de várias fontes de dados

Pode usar uma lista de fontes de dados para extrair simultaneamente de várias fontes de dados. A lista de fontes de dados é especificada em um arquivo de configuração separado.

Cada nova fonte de dados SAP criada no PPM CTK é automaticamente adicionada ao final da lista de fontes de dados do cliente atual. Durante a extração usando a opção de linha de comando **-datasourcelist <file name>** os dados dessas fontes de dados são extraídos consecutivamente como se a extração de dados fosse chamada várias vezes usando a opção **-datasource <file name>**. A sequência da extração da fonte de dados é especificada no arquivo de configuração. As fontes de dados de um tipo diferente de MYSAP são ignoradas durante a extração.

Aplica-se o seguinte à resolução de problemas:

- Se a extração for chamada por meio de uma lista de fontes de dados válida que não contenha nenhuma fonte de dados correspondente, a extração será terminada sem exibir uma mensagem de erro.

- Se a extração for chamada usando uma lista de fontes de dados contendo várias fontes de dados correspondentes e se ocorrer um erro durante a extração de uma dessas fontes de dados, resultando no cancelamento da extração, o procedimento continuará com a fonte de dados seguinte. Isso significa que o cancelamento da extração de uma fonte de dados não leva ao cancelamento de todo o procedimento.
- Se ocorrer um erro em pelo menos uma fonte de dados, o que levaria a um estado de erro de saída durante a extração única dessa fonte de dados, o estado do último erro de saída será retornado pelo processo geral de extração da fonte de dados.

4.6. Comunicação em rede segura SAP

O SAP-2-PPM suporta Comunicação em rede segura SAP (SNC).) O SAP SNC é um protocolo proprietário da SAP para salvaguarda de conexões RFC.

Pode usar o PPM CTK para configurar as definições obrigatórias para importar dados de fontes de dados com SNC. Pode encontrar mais informações na ajuda do PPM CTK.

4.6.1. Pré-requisitos

Para permitir que o SAP-2-PPM leia fontes de dados SAP com, devem ser atendidos os seguintes requisitos.

- Configure o servidor de aplicativos SAP para SNC, incluindo a configuração de usuários SNC adequados.
- Instale software de criptografia certificado pela SAP que seja compatível com GSS API v2. Recomendamos usar o SAP Common Cryptolib, com o qual o PPM já foi testado.
- Configure o ambiente de segurança conforme especificado pelo fabricante do software de criptografia instalado. Por padrão, isso envolve a troca de certificados de segurança com o servidor de aplicativos SAP.
- Use o SAP JCo 3.0.14 ou superior. Recomendamos usar versão mais recente do JCo. Defina as variáveis de ambiente **SNC_LIB** e **SECUDIR** no ambiente de sistema operacional. A biblioteca RFC incorporada no JCo exige que essas variáveis localizem a camada SNC e as credenciais obrigatórias.

Pode encontrar informações detalhadas sobre SAP Common Cryptolib e SAP JCo na documentação SAP correspondente.

4.6.2. Configuração XML das fontes de dados SAP

A definição de configuração do sistema (R3Config.dtd) para uma fonte de dados. SAP com SNC é estendida com os atributos seguintes:

```
<!ATTLIST r3system
…
snc_mode (true|false) "false"
snc_sso (true|false) "true"
snc_qop (1|2|3|8|9) "9"
snc_partnername CDATA #IMPLIED
snc_myname CDATA #IMPLIED
…
>
```

Atributos	Descrição
snc_mode	Ativar/desativar SNC Padrão: Desativado
snc_sso	Logon SNC com nome do usuário e senha (sem início de sessão universal) Padrão: Use SSO
snc_qop	Qualidade do serviço SNC 1. Apenas autenticação 2. Proteção de integridade 3. Proteção de confidencialidade 8. Padrão especificado pelo servidor de aplicativos 9. Configuração de segurança mais elevada (padrão)
snc_partnername	Nome SNC do servidor de aplicativos (por exemplo, p:CN=R3, O=XYZ-INC, C=US)
snc_myname	Nome SNC do cliente RFC (por exemplo, p:CN=Smith, O=XYZ-INC, C=EN)

O atributo **user**, que era obrigatório, é agora opcional. A especificação do nome do usuário e senha não é necessária para a SNC, a menos que tal seja especificamente definido usando o atributo **snc_sso=false**.

A configuração SNC para fontes de dados do extrator PPM-2-SAP e para fontes de dados SAP para análise de dados do PPM é idêntica.

Exemplo 39.

Aqui está um exemplo de uma configuração completa ao usar o SNC:

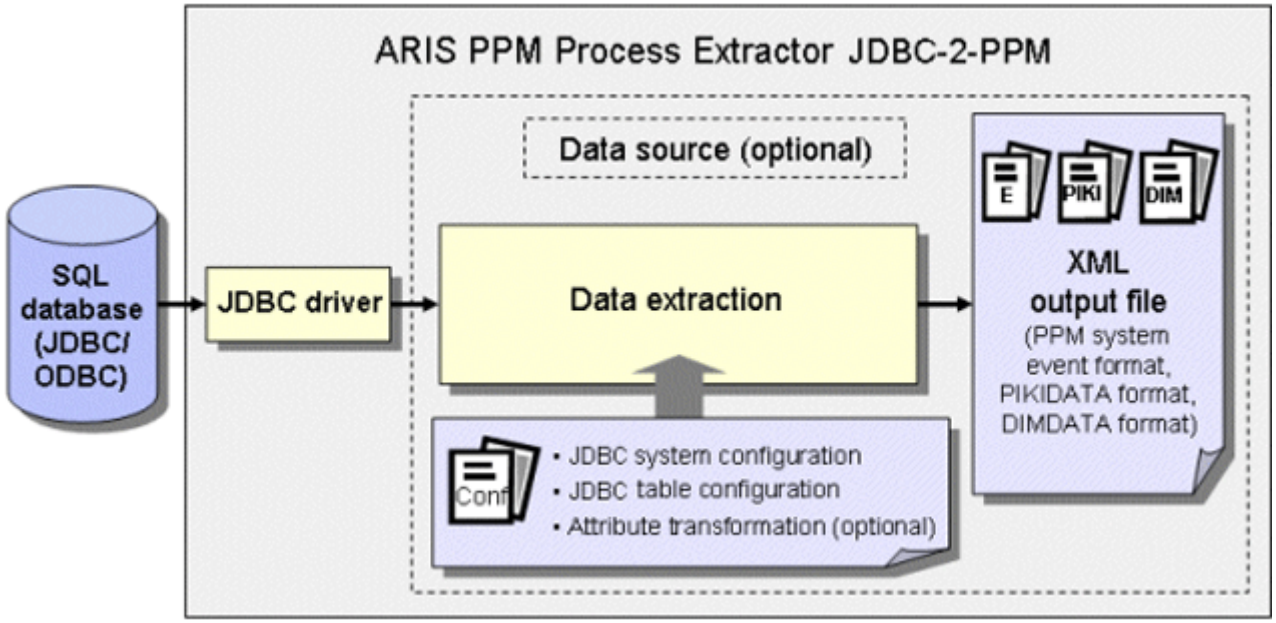
```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE r3systemconf SYSTEM "R3Config.dtd">
<r3systemconf>
<r3system configname="snc_connection"
appserver="xyz.com" systemnumber="00"
client="777" user="" snc_mode="true" snc_sso="true" snc_qop="9"
snc_partnername="p:CN=R3, O=XYZ-INC, C=US"
snc_myname="p:CN=Smith, O=XYZ-INC, C=DE"
language="" lastreaddate="19700101" lastreadtime="000000"
encryptpw="" />
</r3systemconf>
```

5. PPM Process Extractor JDBC-2-PPM

Este capítulo fornece uma visão geral da arquitetura, funcionamento e configuração do PPM Process Extractor JDBC-2-PPM.

5.1. Arquitetura

A figura abaixo ilustra a funcionalidade do PPM Process Extractor JDBC-2-PPM:



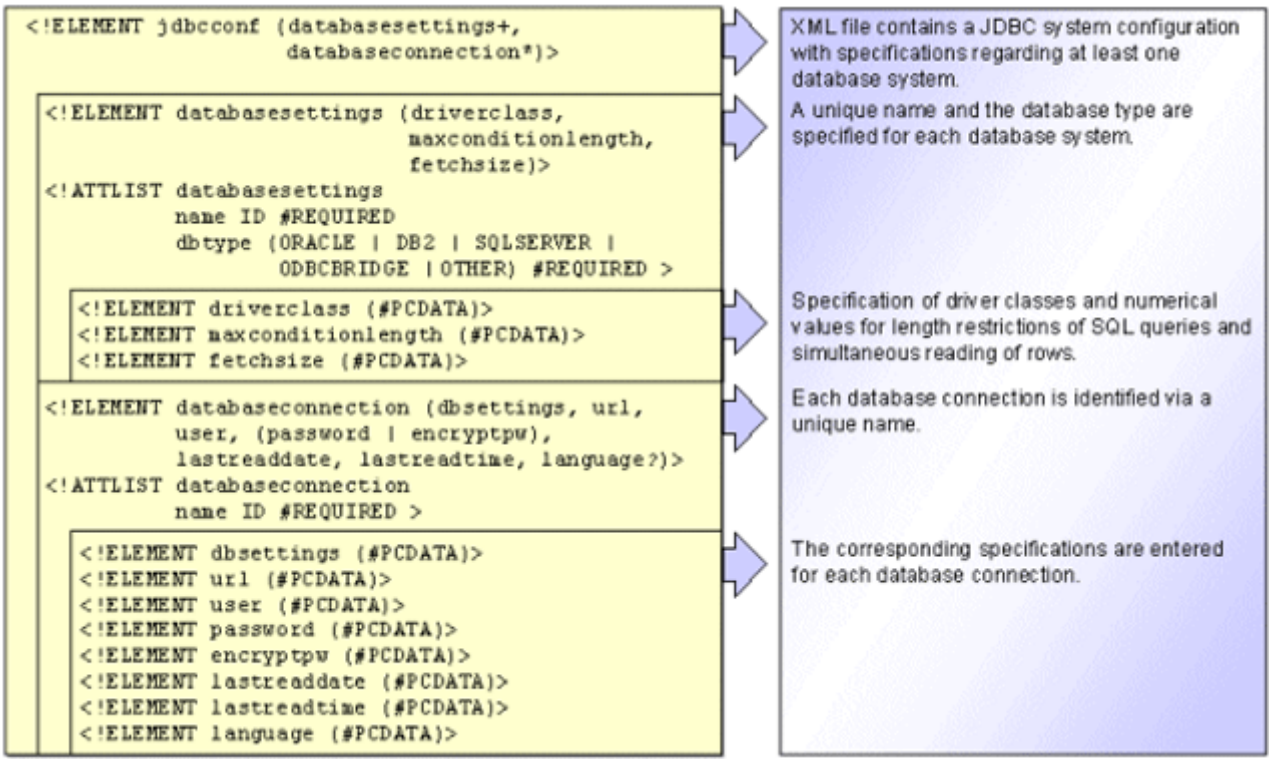
O driver do JDBC é usado para criar uma conexão ao banco de dados de origem SQL, com os dados de acesso para configuração do sistema JDBC. Os dados são então extraídos das tabelas no sistema de banco de dados de origem usando a configuração da tabela JDBC e uma transformação opcional usando o módulo **extração de dados [9]** e são gravados em um arquivo XML, em um formato de saída compatível com o PPM.

Você precisa garantir que o usuário do sistema de origem especificado tenha autorização de acesso apropriada para extrair os campos de dados relevantes.

5.2. Configuração do sistema JDBC

Os dados de acesso ao sistema de banco de dados SQL são especificados em um arquivo XML. O nome desse arquivo XML é transferido para a ferramenta de linha de comando como um argumento.

O formato do arquivo XML é especificado pela seguinte DTD:



Elemento XML ou atributo	Descrição	Exemplo
jdbcconf	Configuração do sistema JDBC especificando os sistemas de banco de dados e as conexões de banco de dados a serem usadas	-
databasesettings	Configurações gerais para o sistema de banco de dados usado	-
name	Nome único das configurações do sistema de banco de dados geral	setting_oracle
dbtype	Tipo de banco de dados usado; valores válidos: **ORACLE** | **DB2** | **SQLSERVER** | **OTHER**	ORACLE
driverclass	Classe de driver a ser usada no sistema de banco de dados	oracle.jdbc.driverOracleDriver
maxconditionlength	O valor especificado limita o comprimento do componente WHERE de uma consulta SQL	1000
fetchsize	Número de registros de dados transferidos simultaneamente do servidor DB para o cliente Java. Para a maioria dos sistemas de banco de dados, o valor padrão é **10**. Valor recomendado: **1000**	1000
databaseconnection	Informações sobre a conexão de banco de dados	-
name	Nome único da conexão de banco de dados	oracle_connection1
dbsettings	Nome único das configurações do sistema de banco de dados geral	setting_oracle
url	URL da conexão de banco de dados	Ver o extrato de arquivo abaixo
user	Nome do usuário do banco de dados	ppmoracle
password	Senha do usuário do banco de dados	ppmoracle
encryptpw	Senha criptografada como uma combinação de valores após avaliação do elemento <password>	62 -62 -56
lastreaddate	Data a partir da qual os dados são extraídos. Não aplicável quando usar um formato fonte de dados [148]: **aaaammdd**	20050131

Elemento XML ou atributo	Descrição	Exemplo
lastreadtime	Hora a partir da qual os dados são extraídos. Não aplicável ao usar uma fonte de dados [148], formato: **hhmmss**	152917
language	Idioma dos campos de texto extraídos	EN

O tipo de banco de dados pode ser um dos valores predefinidos para sistemas de banco de dados **Oracle**, **IBM DB2** ou **MS SQL Server**. O valor **OTHER** deve ser usado se quiser extrair dados de um banco de dados de um tipo diferente dos mencionados acima.

O parâmetro **maxconditionlength** permite especificar um valor que limita o comprimento do componente WHERE de uma consulta SQL ao valor especificado. Isso é necessário para cumprir as restrições de comprimento nos diferentes sistemas de banco de dados. Se a consulta SQL for maior que o comprimento máximo especificado, ela será dividida em várias consultas menores executadas separadamente.

O parâmetro **fetchsize** permite especificar quantas linhas de dados serão transferidas simultaneamente do banco de dados para o extrator JDBC, por cada operação de leitura.

A senha do usuário do banco de dados tem de ser especificada em forma não criptografada e texto simples no elemento XML **password**. Durante a extração a seguir, e após avaliação, a senha é gravada de volta na configuração do sistema JDBC, em formato criptografado, no elemento XML **encryptpw**. A entrada **password** é então excluída. A qualquer momento, a configuração do sistema pode incluir apenas uma das duas entradas.

Para alterar uma senha existente, adicione uma nova entrada **password** ao arquivo de configuração atual e exclua a entrada **encryptpw**. A nova senha é criptografada como parte da extração seguinte.

O parâmetro **idioma** é opcional. Se especificar um valor, este será comparado com o nome de campo da configuração da tabela (**langfieldname**) (ver o capítulo Configuração da tabela JDBC [76]). Por exemplo, se **langfieldname="LANG"** for especificado em uma tabela e **<language>EN</language>** for especificado na configuração do sistema JDBC, apenas serão extraídos os registros de dados para os quais **LANG** tem o valor **EN**.

O extrato de arquivo seguinte mostra uma configuração de exemplo do sistema JDBC:

```
<jdbcconf>
<databasesettings name="setting_sqls" dbtype="SQLSERVER">
<driverclass>com.microsoft.jdbc.sqlserver.SQLServerDriver
</driverclass>
<maxconditionlength>1000</maxconditionlength>
<fetchsize>1000</fetchsize>
</databasesettings>
<databasesettings name="setting_oracle" dbtype="ORACLE">
<driverclass>oracle.jdbc.driver.OracleDriver</driverclass>
<maxconditionlength>1000</maxconditionlength>
<fetchsize>1000</fetchsize>
</databasesettings>
<databaseconnection name="sqls_connection">
<dbsettings>setting_sqls</dbsettings>
<url>jdbc:microsoft:sqlserver:
//PC3:1433;SelectMethod=Cursor;
DatabaseName=ppmdb
</url>
<user>ppmuser</user>
<password>ppmuser</password>
```

```
<lastreaddate>20050228</lastreaddate>
<lastreadtime>000000</lastreadtime>
<language />
</databaseconnection>
<databaseconnection name="oracle_connection1">
<dbsettings>setting_oracle</dbsettings>
<url>jdbc:oracle:thin:@pcppm:1521:orappm_test
</url>
<user>ppmoracle</user>
<password>ppmoracle</password>
<lastreaddate>20051231</lastreaddate>
<lastreadtime>235959</lastreadtime>
</databaseconnection>
<databaseconnection name="oracle_connection2">
<dbsettings>setting_oracle</dbsettings>
<url>jdbc:oracle:thin:@pcppm:1521:orappm_produktiv</url>
<user>ppmoracle2</user>
<encryptpw>48 -62 -76 -60 -108 -57 -92</encryptpw>
<lastreaddate>20050228</lastreaddate>
<lastreadtime>000000</lastreadtime>
</databaseconnection>
```

</jdbcconf>

Para que PPM Process Extractor JDBC-2-PPM possa usar os dados de configuração para conectar ao banco de dados, terá de copiar os drivers JDBC (arquivos JAR e/ou ZIP) para o seguinte diretório da instalação.

<PPM installation directory>\ppm\server\bin\work\data_ppm\drivers

5.3. Configuração da tabela JDBC

A configuração da tabela JDBC (especificação de evento do sistema) especifica quais campos da tabela são extraídos do sistema de banco de dados SQL e gravados nos eventos de sistema como atributos do sistema de origem. Pode salvar várias configurações de tabela com nomes únicos no arquivo XML.

A configuração da tabela JDBC consiste nos seguintes componentes:

Tabelas globais

As tabelas globais são usadas para extrair informações gravadas para todos os eventos de sistema.

Tabelas de chave estrangeira

O elemento XML **docreftable** contém o nome da tabela de chave estrangeira. Especifica como o intervalo de dados a ser extraído da tabela do evento de sistema é limitado. Os campos de chave primária especificados no elemento XML **pkfield** vinculam a tabela de chave estrangeira à tabela do evento de sistema e a outras tabelas de chave estrangeira.

Tabela do evento de sistema

O elemento XML **doctable** contém o nome da tabela do evento de sistema. Especifica os documentos a serem extraídos para um fluxo de documentos. Cada registro de dados extraído da tabela de eventos do sistema gera um evento do sistema no arquivo de saída (elemento XML **event**).

Tabelas de dados

As informações na tabela de eventos do sistema podem ser complementadas pela extração de campos de dados adicionais de quaisquer outras tabelas de dados (por exemplo, o número do material é extraído da tabela de eventos do sistema e o texto descritivo relacionado com esse número é extraído de uma tabela de dados).

Alguns sistemas de banco de dados permitem usar nomes de tabela que não estão em conformidade com o padrão SQL. Extrair essas tabelas com PPM Process Extractor JDBC-2-PPM resulta em uma mensagem de erro. Pode extrair conteúdos criando uma visualização para cada tabela a ser extraída que contenha nomes que não sejam compatíveis com SQL e, em seguida, usando a visualização para extrair com PPM Process Extractor JDBC-2-PPM.

A seguinte estrutura de arquivo XML ilustra a configuração das tabelas das quais os dados devem ser extraídos. Para detalhes sobre quais elementos XML ou atributos são opcionais, consulte a tabela explicativa no capítulo **Configuração do acesso à tabela [79]**.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE r3systemconffields SYSTEM
'jdbc_tableconfiguration.dtd'>
<jdbc_tableconfiguration>
<configuration name="..." printname="..." classtouse="...">
<globaltable name="..." tablename="..." classtouse="...">
<fieldtoread name="..."/>
<textref tablename="..." reffieldname="..."
textfieldname="..." langfieldname="..."/>
</fieldtoread>
...
</globaltable>
<docspec>
<docreftable name="..." tablename="..."
classtouse="...">
<condition fieldname="..." logicaloperator="...">
<value>...</value>
</condition>
<pkfield name="..." fktablename="..."
fkfieldname="...">
<fkpart readfrom="..."
startposition="..." length="..."/>
<prefix>
<value>...</value>
</prefix>
<postfix>
<value>...</value>
</postfix>
</pkfield>
```

```
...
</docreftable>
...
<doctable name="..." tablename="..."
classtouse="...">
<condition fieldname="..." logicaloperator="...">
<value>...</value>
</condition>
<pkfield name="..." fktablename="..."
fkfieldname="...">
<fkpart readfrom="..." startposition="..."
length="..."/>
<prefix>
<value>...</value>
</prefix>
<postfix>
<value>...</value>
</postfix>
</pkfield>
...
<fieldtoread name="...">
<textref tablename="..." reffieldname="..."
textfieldname="..." langfieldname="..."/>
</fieldtoread>
...
</doctable>
</docspec>
<table name="..." tablename="..." classtouse="...">
<condition fieldname="..." logicaloperator="...">
<value>...</value>
</condition>
<pkfield name="..." fktablename="..."
fkfieldname="...">
<fkpart readfrom="..." startposition="..."
length="..."/>
<prefix>
<value>...</value>
</prefix>
<postfix>
<value>...</value>
</postfix>
</pkfield>
...
<fieldtoread name="...">
<textref tablename="..." reffieldname="..."
textfieldname="..." langfieldname="..."/>
</fieldtoread>
...
</table>
...
</configuration>
...
</jdbc_tableconfiguration>
```

5.3.1. Configuração do acesso à tabela

Elemento XML	Atributos XML	Descrição
jdbc_table-configuration		Lista de configurações da tabela
configuração	name	Identificador único da configuração da tabela
	printname (opcional)	Switch. **yes** significa que o nome da configuração é enviado como um prefixo para os nomes dos campos. Valor padrão: **não**
	classtouse (opcional)	Nome da classe Java usada para editar a configuração Valor padrão: Implementação padrão
globaltable (opcional)	name	Identificador da tabela da qual os atributos do sistema global são extraídos
	tablename (opcional)	Nome da tabela no sistema de origem Valor padrão: Especificado no atributo XML **name**
	classtouse (opcional)	Nome da classe Java usada para extrair dados da tabela Valor padrão: Implementação padrão
docspec	-	Agrupamento das especificações de todas as tabelas de chave estrangeira e da tabela de evento do sistema
docreftable	name	Identificador de uma tabela de chave estrangeira
	tablename (opcional)	Nome da tabela no sistema de origem Valor padrão: Especificado no atributo XML **name**
	classtouse (opcional)	Nome da classe Java usada para extrair dados da tabela Valor padrão: Implementação padrão
doctable	name	Identificador da tabela do evento de sistema. Cada linha extraída cria um evento de sistema no arquivo XML.
	tablename (opcional)	Nome da tabela no sistema de origem Valor padrão: Especificado no atributo XML **name**
	classtouse (opcional)	Nome da classe Java usada para extrair dados da tabela Valor padrão: Implementação padrão
tabela (opcional)	name	Identificador da tabela de dados da qual são extraídas informações suplementares
	tablename (opcional)	Nome da tabela no sistema de origem Valor padrão: Especificado no atributo XML **name**
	classtouse (opcional)	Nome da classe Java usada para extrair dados da tabela Valor padrão: Implementação padrão

Cada tabela só pode ser referenciada uma vez na configuração. Para poder extrair uma tabela várias vezes, use identificadores diferentes para fazer referência à mesma tabela (atributo XML **tablename**).

Exemplo 40. *[en-us]*

Quer extrair os campos **VBAP-MATNR_1** e **VBAP-MATNR_2** da tabela **PRODH**. O extrato de arquivo abaixo ilustra a configuração:

```
<table name="PROD_HIER_1" tablename="PRODH"
classtouse="ZTable">
<pkfield name="HNUM" fktablename="VBAP"
fkfieldname="MATNR_1"/>
<fieldtoread name="HVAL"/>
</table>

<table name="PROD_HIER_2" tablename="PRODH"
classtouse="ZTable">
<pkfield name="HNUM" fktablename="VBAP"
fkfieldname="MATNR_2"/>
<fieldtoread name="HVAL"/>
</table>
```

5.3.2. Metadados globais

O elemento XML **globaltable** pode ser usado para transferir metadados dos extratores de processo PPM SAP-2-PPM e JDBC-2-PPM para o arquivo de saída como atributos do evento de sistema global. Os dados típicos para atributos do evento de sistema global incluem os nomes das configurações que foram usadas para extrair os dados. Os próprios dados são determinados a partir dos valores de argumentos especificados na linha de comando. O valor padrão é determinado para argumentos que não são especificados na linha de comando.

A identificação de palavras-chave faz distinção entre maiúsculas e minúsculas.

Os atributos do evento de sistema global aplicam-se a todos os eventos de sistema no arquivo de saída atual.

Palavra-chave	Valor do argumento da linha de comando
SYSTEM_CONFIG_FILE_NAME	-systemconfig <filename> ...
SYSTEM_CONFIG_NAME	-systemconfig ... <configname>
TABLE_CONFIG_FILE_NAME	-tableconfig <filename> ...
TABLE_CONFIG_NAME	-tableconfig ... <configname>
BEGIN_DATE,	-begindate <dd.MM.yyyy>
BEGIN_TIME	-begintime <hh:mm:ss>
END_DATE	-enddate <dd.MM.yyyy>
END_TIME	-endtime <hh:mm:ss>
CPD	-cpd <int>
OUT_FILE_NAME	-outfile <file name>
PIKI_DATA_MAPPING_FILE_NAME	-pikidatamapping <filename>...
PIKI_DATA_MAPPING_NAME	-pikidatamapping ... <pcname>
DIM_DATA_MAPPING_FILE_NAME	-dimdatamapping <filename> ...
DIM_DATA_MAPPING_NAME	-dimdatamapping ... <dimname>

Se nenhum valor puder ser determinado para uma palavra-chave, a própria palavra-chave será gravada no arquivo de saída. Dessa forma pode transferir constantes para o arquivo de saída.

Exemplo 41.

A extração de arquivo abaixo mostra a configuração usada para criar atributos do evento de sistema global.

```
<globaltable name="INFO">
<fieldtoread name="SYSTEM_CONFIG_NAME"/>
</globaltable>
```

O nome de configuração especificado na linha de comando usando -systemconfig **<configname>**, por exemplo, **sapppm** é gravado no arquivo de saída como atributo do evento de sistema global **INFO-SYSTEM_CONFIG_NAME**:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<attribute type="INFO-SYSTEM_CONFIG_NAME ">
sapppm
</attribute>
...
<event>
...
</event>
...
</eventlist>
```

Ao especificar uma classe Java adequada no atributo XML **classtouse**, os dados também podem ser extraídos de qualquer outra tabela do sistema de origem.

5.3.3. Classificar eventos de sistema

Pode classificar os eventos de sistema na saída XML em ordem alfanumérica crescente com base no conteúdo dos campos definidos como critérios de classificação. É possível especificar vários campos como critérios de classificação. Ao classificar, os campos são priorizados desde o primeiro até ao último especificado, ou seja, os dados são classificados primeiro com base no conteúdo do primeiro campo especificado, depois pelo conteúdo do segundo campo especificado e assim por diante.

> **Dica**
> Pode usar a opção de classificação para criar uma importação XML mais eficiente dos dados extraídos para o sistema PPM. O fato de os eventos de sistema pertencentes a uma instância de processo seguirem imediatamente um ao outro devido à classificação nos arquivos de saída XML significa que eles podem ser importados diretamente para uma EPC, usando a chave de processo durante a importação XML.

Na configuração da tabela, no elemento XML **parameter**, os campos são especificados como critérios de classificação para a tabela do evento de sistema (**doctable**) e as tabelas de chave estrangeira referenciadas (**docreftable**).

Exemplo 42. ***[en-us]***

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
```

```
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="BANF">
<docspec>
<doctable name="EBAN">
<parameter name="ORDER_BY">
<value>BNFPO</value>
<value>BANFN</value>
</parameter>
<condition fieldname="ERDAT"
logicaloperator="creationtimestamp">
<value>yyyyMMdd</value>
</condition>
<pkfield name="BANFN"/>
<pkfield name="BNFPO"/>
<pkfield name="ERDAT"/>
<fieldtoread name="BSART"/>
<fieldtoread name="KONNR"/>
<fieldtoread name="KTPNR"/>
<fieldtoread name="LIFNR"/>
<fieldtoread name="MFRNR"/>
</doctable>
</docspec>
</configuration>
</xmlextractor_tableconfiguration>
```

Tal como foi especificado na configuração **BANF** (elemento XML **parameter**), os eventos de sistema são gravados no(s) arquivo(s) de saída classificados em ordem crescente pelo número do elemento no número da requisição de compra (campo **BNFPO**) e, como segunda prioridade, pelo número da requisição de compra (campo **BANFN**).

A tabela abaixo mostra todas as opções de configuração:

Elemento/atributo XML	Descrição
parâmetro	Especifica um parâmetro de classificação para uma tabela do evento de sistema (**doctable**) ou uma tabela de chave estrangeira (**docreftable**).
name	Nome do parâmetro. É essencial especificar **ORDER_BY**.
value	Especifica um campo de classificação. O campo tem de ser especificado como um campo de chave primária (**pkfield**).

Pode configurar o parâmetro de classificação em PPM Customizing Toolkit. Para isso, ative a guia **Evento de sistema** no módulo **Extração de dados** do grupo de módulos **Merge do processo**. Mude para modo de edição e selecione **Editar evento de sistema** do menu de contexto da tabela do evento de sistema. Pode especificar as configurações na etapa **Especificar o parâmetro**.

5.3.4. Concatenar novos valores de atributo de valores de atributo extraídos

A classe **com.idsscheer.ppm.xmlextractortools.extractor.**

jdbc2ppm.ZTableConcatFKs_jdbc2ppm pode ser usada ao extrair dados de um sistema JDBC para gerar novos atributos do evento do sistema, concatenando os valores dos atributos já extraídos. Os novos eventos

de sistema criados podem ser referenciados em outras configurações da tabela de dados (elemento XML **table**).

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="SD">
<docspec>
<doctable name="RBKP">
<condition logicaloperator="eq" fieldname="GJAHR">
<value>2005</value>
</condition>
<pkfield name="BELNR"/>
<pkfield name="GJAHR"/>
</doctable>
</docspec>
<table name="RBKP_NEW" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.
jdbc2ppm.ZTableConcatFKs_jdbc2ppm">
<pkfield name="BELNR" fktablename="RBKP"
fkfieldname="BELNR"/>
<pkfield name="GJAHR" fktablename="RBKP"
fkfieldname="GJAHR"/>
<fieldtoread name="AWKEY_GENERATED"/>
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

Se estiver a usar a classe **ZTableConcatFKs_jdbc2ppm** terá de especificar exatamente um elemento **fieldtoread**. O elemento contém o nome do atributo do evento de sistema a ser criado. No exemplo, é criado um novo atributo do evento de sistema com o nome **RBKP_NEW-AWKEY_GENERATED**. O valor do atributo consiste nos valores dos elementos **pkfield** especificados.

As operações normais **fkpart**, **prefix** e **postfix** podem ser usadas para os elementos **pkfield**.

Um evento de sistema criado com a configuração acima poderia ter a seguinte aparência:

```
...
<event>
<attribute type="RBKP-BELNR">5105601204</attribute>
<attribute type="RBKP-GJAHR">2005</attribute>
<attribute type="RBKP_NEW-AWKEY_GENERATED">51056012042005
</attribute>
</event>
...
```

O novo atributo – **RBKP_NEWER-AWKEY_GENERATED** – pode então ser referenciado em outras configurações de **tabela** da mesma forma que os restantes campos da tabela JDBC.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE xmlextractor_tableconfiguration SYSTEM
'xmlextractor_tableconfiguration.dtd'>
<xmlextractor_tableconfiguration>
<configuration name="SD">
```

```
<docspec>
<doctable name="RBKP">
<condition logicaloperator="eq" fieldname="GJAHR">
<value>2005</value>
</condition>
<pkfield name="BELNR"/>
<pkfield name="GJAHR"/>
</doctable>
</docspec>
<table name="RBKP_NEW" classtouse="com.idsscheer.
ppm.xmlextractortools.extractor.
jdbc2ppm.ZTableConcatFKs_jdbc2ppm">
<pkfield name="BELNR" fktablename="RBKP"
fkfieldname="BELNR"/>
<pkfield name="GJAHR" fktablename="RBKP"
fkfieldname="GJAHR"/>
<fieldtoread name="AWKEY_GENERATED"/>
</table>
<table name="BKPF">
<pkfield name="AWKEY" fktablename="RBKP_NEW"
fkfieldname="AWKEY_GENERATED"/>
<fieldtoread name="BELNR"/>
</table>
</configuration>
</xmlextractor_tableconfiguration>
```

Um evento de sistema criado com essa configuração poderia ter a seguinte aparência:

```
...
<event>
<attribute type="RBKP-BELNR">5105601204</attribute>
<attribute type="RBKP-GJAHR">2005</attribute>
<attribute type="RBKP_NEW-AWKEY_GENERATED"
>51056012042005</attribute>
<attribute type="BKPF-BELNR">5100004691</attribute>
</event>
...
```

5.3.5. Extração usando condições

A extração de dados das tabelas pode ser limitada pelo uso de condições. Por exemplo, pode usar condições para controlar que, ao extrair campos da tabela, somente são extraídos os fluxos de documentos de um tipo de documento em um período específico.

As condições são especificadas diretamente na configuração da tabela do elemento XML **booleancondition** ou **condition** e aplicam-se apenas à tabela associada.

Uma condição contém o nome do campo da tabela, o operador de comparação e um valor concreto. As condições podem ser vinculadas com qualquer grau de complexidade (elemento XML **booleancondition**).

Elemento XML	Atributos XML	Descrição
booleancondition (opcional)	logicaloperator	Operadores lógicos: **AND**, **OR**, **NOT** Valor padrão: **AND**
condição (opcional)	logicaloperator	Operadores de comparação: **eq**, **neq**, **in**, **notin**, **num_gt**, **num_geq**, **num_lt**, **num_leq**, **is_null**, **is_not_null**, **timestamp_eq**, **timestamp_geq**, **timestamp_gt**, **timestamp_leq**, **timestamp_lt**, **time_eq**, **time_geq**, **time_gt**, **time_leq**, **time_lt**, **date_eq**, **date_geq**, **date_gt**, **date_leq**, **date_lt**, **num_eq**, **num_neq**, **num_in**, **num_notin** Operadores para restringirem o intervalo de dados a ser extraído: **char_creationtimestamp**, **date_creationtimestamp**, **valueconstraint** Valor padrão: **eq**
	fieldname	Nome do campo da tabela

Os elementos XML **docbooleancondition** e **doccondition** permitem a extração condicional de uma tabela de dados, dependendo dos campos da tabela já extraídos. São configurados de forma semelhante a **booleancondition** e **condition**. O nome da tabela e o nome da coluna que contêm os campos de dados já extraídos são especificados nos atributos XML **tablename** e **fieldname**.

Elemento XML	Atributos XML	Descrição
docbooleancondition (opcional)	logicaloperator	Operadores lógicos: **AND**, **OR**, **NOT** Valor padrão: **AND**
doccondition (opcional)	logicaloperator	Operadores de comparação: **eq**, **neq**, **in**, **notin**, **exists**, **notexists**, **is_null**, **is_not_null**, Valor padrão: **eq**
	tablename	Nome da tabela que contém os campos da tabela já extraídos
	fieldname	Nome de coluna

5.3.5.1. Operadores de condição

Operadores comuns

Os seguintes operadores de comparação são suportados pelos elementos XML **condition** e **doccondition**:

Operador	Descrição
eq	O conteúdo do campo é igual ao valor especificado.
neq	O conteúdo do campo não é igual ao valor especificado.
in	O conteúdo do campo é igual a um valor especificado de um conjunto de valores.
notin	O conteúdo do campo não é igual a um valor especificado de um conjunto de valores.
is_null	Verifica se o conteúdo do campo é NULL
is_not_null	Verifica se o valor do campo não é igual a NULL

Operadores de condição

Operador	Descrição
num_gt	O conteúdo do campo é maior do que o valor especificado.
num_geq	O conteúdo do campo é maior do que ou igual ao valor especificado.
num_lt	O conteúdo do campo é menor do que o valor especificado.

Operador	Descrição
num_leq	O conteúdo do campo é inferior ou igual ao valor especificado.
num_eq	O conteúdo do campo é igual ao valor especificado.
num_neq	O conteúdo do campo não é igual ao valor especificado.
num_in	O conteúdo do campo é igual a um valor especificado de um conjunto de valores.
num_notin	O conteúdo do campo não é igual a um valor especificado de um conjunto de valores.
timestamp_eq	Verifica se o carimbo de data/hora do campo corresponde ao valor de comparação especificado (**value**). O valor de comparação deve ser especificado no formato **dd.MM.aaaa HH:mm:ss** – isto também se aplica a todos os outros operadores timestamp_*.
timestamp_geq	Verifica se a marca de hora do campo é maior ou igual ao valor de comparação.
timestamp_gt	Verifica se a marca de hora do campo é maior do que o valor de comparação.
timestamp_leq	Verifica se o carimbo de data/hora do campo é inferior ou igual ao valor de comparação.
timestamp_lt	Verifica se o carimbo de data/hora do campo é inferior ao valor de comparação.
time_eq	Verifica a igualdade da hora do campo. O valor de comparação deve ser especificado no formato **HH:mm:ss** – isto também se aplica a todos os outros operadores time_*.
time_geq	Verifica se a hora do campo é maior do que ou igual ao valor de comparação
time_gt	Verifica se a hora do campo é maior do que o valor de comparação
time_leq	Verifica se a hora do campo é menor do que ou igual ao valor de comparação
time_lt	Verifica se a hora do campo é menor do que o valor de comparação
date_eq	Verifica se a data do campo é igual ao valor de comparação. O valor de comparação deve ser especificado no formato **dd.MM.aaaa** – isto também se aplica a todos os outros operadores date_*.
date_geq	Verifica se a data do campo é maior do que ou igual ao valor de comparação
date_gt	Verifica se a data do campo é maior do que o valor de comparação
date_leq	Verifica se a data do campo é menor do que ou igual ao valor de comparação
date_lt	Verifica se a data do campo é menor do que o valor de comparação

like
 	
(apenas com campos do banco de dados da cadeia de caracteres, por exemplo, CHAR, VARCHAR)
	Comparação de valores de campo com uma cadeia de caracteres variável
	
	São permitidos os seguintes espaços reservados:
	
	* No ou qualquer número de caracteres
	
	? Apenas um caractere
	
	\ Caractere de mascaramento para procurar
	
	espaços reservados ou mascarar caracteres na
	
	forma: \\ ou * ou \?
	
	Exemplo:
	
	`<condition fieldname="OBJECTID"`
	
	`logicaloperator="like">`
	
	`<value>*10?0\\20?0*</value>`
	
	`</condition>`
	
	A pesquisa é realizada para valores como
	
	5551050\20106667 or 1080\204044 but not 34510550\2030*

Operador	Descrição

char_
 	
creationtimestamp
	Extrai carimbos de data/hora (data e hora) dos campos do banco de dados de origem do tipo CHAR/ VARCHAR. Seus valores formam a base para restringir o intervalo de dados a ser extraído com os parâmetros de linha de comando **-begindate**
	
	(**-begintime**) ou **-enddate** (**-endtime**) [ver capítulo Argumentos específicos do banco de dados de origem [114]].
	
	Vários campos são separados pela combinação de caracteres **#-#**. Para cada campo, o formato do carimbo de data/hora é especificado no elemento XML **value**. Este formato contém os valores no campo correspondente do banco de dados de origem.
	
	Exemplo:
	
	`<condition fieldname="VC_DATE#-#VC_TIME"`
	`logicaloperator="char_creationtimestamp">`
	`<value>yyyyMMdd</value>`
	`<value>HHmmss</value>`
	`</condition>`
	
	Quando usar o operador **char_creationtimestamp**, esteja ciente dos efeitos possíveis de determinados formatos de hora do campo do banco de dados de origem na classificação e extração de dados. Os campos do banco de dados em um formato de hora são extraídos em ordem alfabética. Por exemplo, se os valores do campo de data no banco de dados forem salvos em formato **ddMMaaaa**, a data **23021999** será extraída quando for extraído o intervalo de tempo **15.01.2000 - 31.12.2000** porque em termos alfabéticos, esta data situa-se entre a data de início e de fim. No entanto, a data **09122000** não está neste intervalo porque é interpretada como anterior à data de início.

date_
 	
creationtimestamp
	Os carimbos de data/hora (data e hora) são extraídos dos campos do banco de dados de origem com horas que são de um tipo de dados de hora dependente do sistema de banco de dados. Seus valores formam a base para restringir o intervalo de dados a ser extraído com os parâmetros de linha de comando **-begindate**
	
	(**-begintime**) ou **-enddate** (**-endtime**) [ver capítulo Argumentos específicos do banco de dados de origem [114]].
	
	Vários campos são separados pela combinação de caracteres **#-#**. Para cada campo a ser extraído, os elementos XML **value** são usados para especificar um dos três tipos de dados de hora correspondentes aos campos do banco de dados de origem. Apenas são permitidos os seguintes tipos de dados nas combinações especificadas: **DATE**/**TIME** ou **TIME**/**DATE** ou **TIMESTAMP** ou **DATE**
	
	Exemplo:
	
	`<condition fieldname="CHG_DATE#-#CHG_TIME"`
	`logicaloperator="date_creationtimestamp">`
	`<value>DATE</value>`
	`<value>TIME</value>`
	`</condition>`

valueconstraint	Especifica um campo do sistema de banco de dados de origem com valores inteiros que são usados para delimitar o intervalo de dados a ser extraído com o parâmetro de linha de comando **-valueconstraint**
	
	[ver o capítulo Argumentos específicos do banco de dados de origem [114]].
	
	O campo a ser extraído tem de ser do tipo de dados inteiro.
	
	Exemplo:
	
	`<condition fieldname="INTEGERFIELD"`
	`logicaloperator="valueconstraint"/>`

Quando usar os operadores **timestamp_***, **time_*** e **date_*** da tabela acima, é preciso saber qual tipo de dados e em qual formato se baseia o valor extraído do sistema de banco de dados correspondente (servidor Oracle, IBM DB2, MS SQL).

As tabelas seguintes fornecem uma visão geral dos diferentes tipos de dados e formatos do banco de dados e mostram alguns exemplos de formatos gerados no arquivo de saída XML.

Tipo de dados Oracle/TIMESTAMP

Formato do valor gravado no banco de dados (exemplo)	Formato de resultado no arquivo de saída XML (exemplo)	Nota
07.05.2005 04:02:36	07.05.2005 04:02:36	-
07.05.2005	07.05.2005 12:25:11	Time = Hora de criação do valor do campo do banco de dados
04:02:36	21.12.2005 04:02:36	Date = Data de criação do valor do campo do banco de dados

Tipo de dados Oracle/DATE

Formato do valor gravado no banco de dados (exemplo)	Formato de resultado no arquivo de saída XML (exemplo)	Nota
07.05.2005 04:02:36	07.05.2005 04:02:36	-
07.05.2005	07.05.2005 12:25:11	Time = Hora de criação do valor do campo do banco de dados
04:02:36	21.12.2005 04:02:36	Date = Data de criação do valor do campo do banco de dados

O tipo de dados Oracle **DATE** salva valores do carimbo de data/hora exclusivamente em formato **dd.MM.aaaa HH:mm:ss**. Para extrair os dados necessários de um banco de dados Oracle, precisa configurar devidamente as condições de extração quando usar operadores **date_***.

Exemplo 43.

Quer extrair todos os registros de dados com a data **16.09.2004**, independentemente da hora. Se especificar a condição seguinte na configuração da tabela:

```
<condition fieldname="TACT_TDATE" logicaloperator="date_eq">
<value>16.09.2004</value>
</condition>
```

Exemplo 44.

apenas serão extraídos registros de dados com o carimbo de data/hora **16.09.2004 00:00:00**. Para garantir que todos os registros da data especificada são extraídos, é necessário reescrever a condição da seguinte maneira:

```
<booleancondition logicaloperator="AND">
<condition fieldname="TACT_TDATE"
logicaloperator="date_geq">
<value>16.09.2004</value>
</condition>
<condition fieldname="TACT_TDATE"
logicaloperator="date_lt">
<value>17.09.2004</value>
```

```
</condition>
</booleancondition>
```

Exemplo 45.

Com a condição especificada, todos os registros de dados com a data **16.09.2004** e qualquer hora serão extraídos.

Tipo de dados IBM DB2/TIMESTAMP

Formato do valor gravado no banco de dados (exemplo)	Formato de resultado no arquivo de saída XML (exemplo)	Nota
07.05.2005 04:02:36	07.05.2005 04:02:36	-

Tipo de dados IBM DB2/DATE

Formato do valor gravado no banco de dados (exemplo)	Formato de resultado no arquivo de saída XML (exemplo)	Nota
07.05.2005	07.05.2005	-

Tipo de dados IBM DB2/TIME

Formato do valor gravado no banco de dados (exemplo)	Formato de resultado no arquivo de saída XML (exemplo)	Nota
04:02:36	04:02:36	-

Tipo de dados Servidor MS SQL/DATETIME

Formato do valor gravado no banco de dados (exemplo)	Formato de resultado no arquivo de saída XML (exemplo)	Nota
07.05.2005 04:02:36	07.05.2005 04:02:36	-
07.05.2005	07.05.2005 00:00:00	A hora é sempre 00:00:00
04:02:36	01.01.1900 04:02:36	A data é sempre 01:01:1900

Tipo de dados Servidor MS SQL/SMALLDATETIME

Formato do valor gravado no banco de dados (exemplo)	Formato de resultado no arquivo de saída XML (exemplo)	Nota
07.05.2005 04:02:36	07.05.2005 04:02:00	Hora apenas correta ao minuto
07.05.2005	07.05.2005 00:00:00	A hora é sempre 00:00:00
04:02:36	01.01.1900 04:02:00	A data é sempre 01.01.1900 e a hora está correta ao minuto

doccondition operators

Operador	Descrição
exists	O campo existe.
notexists	O campo não existe.

5.3.5.2. Condições complexas

O extrato de arquivo abaixo ilustra as condições aninhadas usando o exemplo da tabela de chave estrangeira:

```
...
<docreftable tablename="VBAK" classtouse="ZDocRefTable">
<booleancondition logicaloperator="AND">
<condition fieldname="ERDAT#-#ERZET"
logicaloperator="char_creationtimestamp">
<value>yyyyMMdd</value>
<value>HHmmss</value>
</condition>
<condition fieldname="VBTYP" logicaloperator="in">
<value>C</value>
<value>K</value>
</condition>
<booleancondition logicaloperator="OR">
<booleancondition logicaloperator="AND">
<condition fieldname="VKORG" logicaloperator="eq">
<value>1000</value>
</condition>
<condition fieldname="VKBUR" logicaloperator="eq">
<value>0041</value>
</condition>
</booleancondition >
<booleancondition logicaloperator="AND">
<condition fieldname="VKORG" logicaloperator="eq">
<value>2000</value>
</condition>
<condition fieldname="VKBUR" logicaloperator="neq">
<value>0060</value>
</condition>
</booleancondition >
</booleancondition >
</booleancondition >
...
</docreftable>
...
```

Em termos de lógica proposicional, o extrato de arquivo exibido corresponde à condição seguinte:

ERDAT e ERZET contêm a hora de criação

e

VBTYP é C ou V

e

((VKORG é igual a 1000 **e** VKBUR é igual a 0041) **ou** (VKORG é igual a 2000 **e** VKBUR não é igual a 0060)).

5.3.6. Extrair campos do banco de dados

Os campos do banco de dados na tabela do evento de sistema e as tabelas de dados vinculadas, a partir dos quais os valores devem ser extraídos, são especificados nos elementos XML **fieldtoread**. Para cada elemento XML **fieldtoread**, uma linha do formulário

```
<attribute type="Type">Value</attribute>
```

é gravada no arquivo de saída.

Type é composto pelo nome da tabela, o nome do campo de dados e um nome do campo de texto opcional.

Value é o valor extraído do campo de dados correspondente. Todos os valores são gravados em formato de texto.

Opcionalmente, em vez do valor do campo de dados direto, pode ser extraído o valor da tabela referenciada (elemento XML **textref**). A especificação opcional **langfieldname** extrai o texto específico do idioma do campo de dados.

Informações suplementares são extraídas das tabelas de dados usando relações chave primárias.

Elemento XML	Atributos XML	Descrição
fieldtoread	name	Nome da coluna da tabela que contém o campo de dados a ser extraído
textref	tablename	Nome da tabela da tabela de dados referenciada
	reffieldname	Nome da chave estrangeira
	textfieldname	Nome da coluna da tabela que contém o campo de dados a ser extraído
	langfieldname (opcional)	Nome da coluna da tabela que contém o campo de dados específico do idioma a ser extraído
fkpart	readfrom (opcional)	Direção da subcadeia que forma a chave estrangeira Valores válidos: **left**, **right** Valor padrão: **left**
	startposition	Posição a partir da qual a subcadeia é formada
	length (opcional)	Comprimento da subcadeia Valor padrão: Início ou final da cadeia de caracteres (atributo XML **readfrom**)

Antes que os dados sejam realmente extraídos, é executada uma verificação para saber se as tabelas configuradas e os campos de dados existem e se o usuário do sistema especificado tem autorização de acesso apropriada.

Exemplo 46.

Os valores de campo extraídos são gravados no arquivo de saída como atributos de um evento de sistema.

Campos de tabela não localizados:

```
...

  <attribute type="VBAP-WERKS">SB</attribute>
```

```
  ...
```

Exemplo 47.

Campos de tabela localizados:

```
...

  <attribute type="VBAP-WERKS-NAME">Saarbrücken</attribute>

  ...
```

5.3.6.1. Manipulação de valores NULL

Ao manipular valores null durante a extração, é importante distinguir se é extraído o valor NULL ou uma cadeia de caracteres vazia. Nem todo sistema de banco de dados usado suporta diferenciação entre NULL e cadeias de caracteres vazias:

- O sistema de banco de dados Oracle não diferencia entre nenhum valor (NULL) e uma cadeia de caracteres vazia. Em ambos os casos, nenhum atributo do sistema é criado quando os campos do banco de dados correspondentes são extraídos.
- Os sistemas de banco de dados do servidor IMB DB2 e SQL fazem diferenciação entre NULL e uma cadeia de caracteres vazia. Se um campo do banco de dados retornar NULL como um valor durante a extração, nenhum atributo do evento de sistema será criado. Por outro lado, se o sistema extrair um campo de banco de dados contendo uma cadeia de caracteres vazia, o atributo do evento de sistema correspondente será criado sem valor no arquivo de log XML:

```
...

  <attribute type="AT_EXAMPLE"></attribute>

  ...
```

Se quiser usar valores padrão como **Not specified** em vez de valores null extraídos após a importação para o sistema PPM, pode definir valores padrão (elemento XML **defaultvalue**) no cálculo do atributo correspondente (ver o manual **Personalização do PPM**).

5.3.6.2. Extrair campos de valor múltiplos

A configuração adequada do elemento **table** relevante permite extrair todos os valores diferentes de um campo (**fieldtoread**) e gravá-los em um evento de sistema. Uma classe separada deve ser usada para isso.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE jdbc_tableconfiguration SYSTEM
'jdbc_tableconfiguration.dtd'>
<jdbc_tableconfiguration>
<configuration name="MULTIPLE_VALUES">
```

```
<docspec>
<docreftable name="...">
...
</docreftable>
<doctable name="...">
...
</doctable>
</docspec>
<table name="VBEP" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.jdbc2ppm.
ZTableMultipleValues_jdbc2ppm">
<pkfield name="VBELN" fktablename="VBAP"
fkfieldname="VBELN"/>
<pkfield name="POSNR" fktablename="VBAP"
fkfieldname="POSNR"/>
<fieldtoread name="WMENG"/>
<fieldtoread name="EDATU"/>
</table>
<table name="VBAK" classtouse="com.idsscheer.ppm.
xmlextractortools.extractor.jdbc2ppm.
ZTableMultipleValues_jdbc2ppm">
<pkfield name="VBELN" fktablename="VBEP"
fkfieldname="VBELE"/>
<fieldtoread name="VBTYP"/>
</table>
</configuration>
</jdbc_tableconfiguration>
```

Exemplo 48.

Na configuração de exemplo, usar a classe **com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.**

ZTableMultipleValues_jdbc2ppm para todas as tabelas de dados (elementos **table**) especifica que todos os campos a serem extraídos (elementos **fieldtoread**) são extraídos com valores múltiplos se existirem valor múltiplos para um campo no banco de dados do sistema de origem. No arquivo de saída XML, o campo extraído com valores múltiplos aparece como um atributo com o mesmo nome em um evento de sistema:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAP-POSNR">000010</attribute>
<attribute type="VBAP-VBELN">0000004969</attribute>
<attribute type="VBEP-EDATU">19970103</attribute>
<attribute type="VBEP-EDATU">19970107</attribute>
<attribute type="VBEP-WMENG">138.000</attribute>
<attribute type="VBEP-WMENG">317.000</attribute>
<attribute type="VBEP-WMENG">496.000</attribute>
</event>
</eventlist>
```

5.3.7. Configuração do acesso à tabela

Elemento XML	Atributos XML	Descrição
xmlextractor_table-configuration		Lista de configurações da tabela R/3
configuração	name	Identificador único da configuração da tabela
	printname (opcional)	Switch. **yes** significa que o nome da configuração é enviado como um prefixo para os nomes dos campos. Valor padrão: **não**
	classtouse (opcional)	Nome da classe Java usada para editar a configuração Valor padrão: Implementação padrão
globaltable (opcional)	name	Identificador da tabela da qual os atributos do sistema global são extraídos
	tablename (opcional)	Nome da tabela no sistema de origem Valor padrão: Especificado no atributo XML **name**
	classtouse (opcional)	Nome da classe Java usada para extrair dados da tabela Valor padrão: Implementação padrão
docspec	-	Agrupamento das especificações de todas as tabelas de cabeçalho do documento e da tabela de documento
docreftable	name	Identificador de uma tabela de cabeçalho de documento
	tablename (opcional)	Nome da tabela no sistema de origem Valor padrão: Especificado no atributo XML **name**
	classtouse (opcional)	Nome da classe Java usada para extrair dados da tabela Valor padrão: Implementação padrão
doctable	name	Identificador de uma tabela de documento. Cada linha extraída cria um evento de sistema no arquivo de saída XML.
	tablename (opcional)	Nome da tabela no sistema de origem Valor padrão: Especificado no atributo XML **name**
	classtouse (opcional)	Nome da classe Java usada para extrair dados da tabela Valor padrão: Implementação padrão
tabela (opcional)	name	Identificador da tabela de dados da qual são extraídas informações suplementares
	tablename (opcional)	Nome da tabela no sistema de origem Valor padrão: Especificado no atributo XML **name**
	classtouse (opcional)	Nome da classe Java usada para extrair dados da tabela Valor padrão: Implementação padrão

Cada tabela SAP só pode ser referenciada uma vez na configuração. Para poder extrair uma tabela várias vezes, use identificadores diferentes para fazer referência à mesma tabela (atributo XML **tablename**).

Exemplo 49. *[en-us]*

Quer extrair os campos **VBAP-MATNR_1** e **VBAP-MATNR_2** da tabela **PRODH**. O extrato de arquivo abaixo ilustra a configuração:

```
<table name="PROD_HIER_1" tablename="PRODH" >
<pkfield name="HNUM" fktablename="VBAP"
fkfieldname="MATNR_1"/>
<fieldtoread name="HVAL"/>
</table>

<table name="PROD_HIER_2" tablename="PRODH" >
<pkfield name="HNUM" fktablename="VBAP"
fkfieldname="MATNR_2"/>
<fieldtoread name="HVAL"/>
</table>
```

5.3.8. Exemplo: Configuração XML e arquivo de saída

Este capítulo ilustra a configuração da tabela de um sistema de banco de dados SQL por meio de um exemplo prático e mostra um possível arquivo de saída XML.

5.3.8.1. Configuração da tabela

O arquivo abaixo exibe a configuração **SD_C** para um sistema de banco de dados SQL:

```
...
<configuration name="SD_C">
<docspec>
<docreftable name="VBAK">
<booleancondition>
<condition fieldname="ERDATCHAR#-#ERZETCHAR"
logicaloperator="char_creationtimestamp">
<value>yyyyMMdd</value>
<value>HHmmss</value>
</condition>
<condition fieldname="ERDAT#-#ERZET"
logicaloperator="date_creationtimestamp">
<value>DATE</value>
<value>TIME</value>
</condition>
<condition fieldname="VBTYP" logicaloperator="eq">
<value>C</value>
</condition>
</booleancondition>
<pkfield name="VBELN" />
</docreftable>
<doctable name="Orders" tablename="VBAP">
<pkfield name="VBELN" fktablename="VBAK"
fkfieldname="VBELN"/>
<pkfield name="POSNR" />
<fieldtoread name="VBELN"/>
<fieldtoread name="POSNR"/>
<fieldtoread name="ERDAT"/>
```

```
<fieldtoread name="ERZET"/>
<fieldtoread name="MATNR">
<textref tablename="MAKT" reffieldname="MATNR"
textfieldname="MAKTX" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="KONDM">
<textref tablename="T178T" reffieldname="KONDM"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="SPART">
<textref tablename="TSPAT" reffieldname="SPART"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="WERKS">
<textref tablename="T001W" reffieldname="WERKS"
textfieldname="NAME1"/>
</fieldtoread>
<fieldtoread name="CHARG"/>
<fieldtoread name="PSTYV"/>
<fieldtoread name="ERNAM"/>
<fieldtoread name="NETWR"/>
</doctable>
</docspec>
<table name="VBAK">
<pkfield name="VBELN" fktablename="Orders"
fkfieldname="VBELN"/>
<fieldtoread name="VTWEG">
<textref tablename="TVTWT" reffieldname="VTWEG"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VKORG">
<textref tablename="TVKOT" reffieldname="VKORG"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VDATU"/>
<fieldtoread name="VKBUR">
<textref tablename="TVKBT" reffieldname="VKBUR"
textfieldname="BEZEI" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VKGRP">
<textref tablename="TVGRT" reffieldname="VKGRP"
textfieldname="BEZEI" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VBTYP"/>
<fieldtoread name="AUART"/>
</table>
<table name="MARA">
<pkfield name="MATNR" fktablename="Orders"
fkfieldname="MATNR"/>
<fieldtoread name="MATNR"/>
<fieldtoread name="MTART">
<textref tablename="T134T" reffieldname="MTART"
textfieldname="MTBEZ" langfieldname="SPRAS"/>
</fieldtoread>
```

```
</table>
</configuration>
```

Tabela de chave estrangeira

```
...
<docreftable name="VBAK">
<booleancondition>
<condition fieldname="ERDATCHAR#-#ERZETCHAR"
logicaloperator="char_creationtimestamp">
<value>yyyyMMdd</value>
<value>HHmmss</value>
</condition>
<condition fieldname="ERDAT#-#ERZET"
logicaloperator="date_creationtimestamp">
<value>DATE</value>
<value>TIME</value>
</condition>
<condition fieldname="ERDAT_DATE"
logicaloperator="date_geq">
<value>05.12.2001</value>
</condition>
<condition fieldname="ERZET_TIME"
logicaloperator="time_gt">
<value>22:27:13</value>
</condition>
<condition fieldname="VBTYP"
logicaloperator="eq">
<value>C</value>
</condition>
</booleancondition>
<pkfield name="VBELN" />
</docreftable>
...
```

O identificador e o nome da tabela de chave estrangeira do sistema de banco de dados SQL é **VBAK**. São extraídos todos os documentos de encomenda do tipo **VBTYP=C**. Os elementos **value** do operador de condição **char_creationtimestamp** especificam o formato do carimbo de data/hora a ser gerado a partir dos campos correspondentes do banco de dados. Os elementos **value** do operador de condição **date_creationtimestamp** especificam os tipos de dados dos campos extraídos. Os valores da coluna da tabela **ERDAT** são do tipo **DATE** e os da coluna **ERZET** são do tipo **TIME**.

Apenas as linhas da tabela nas quais os valores dos atributos **ERDAT_DATE** e **ERZET_TIME** são maiores do que/iguais a ou maiores do que os valores especificados com **value** (**05.12.2001** ou **22:27:13**).

O nome da coluna da tabela de chave primária é **VBELN**. Para cada valor de campo **VBELN** diferente, os registros de dados são extraídos da tabela do evento de sistema vinculado para a qual **VBELN** apresenta o mesmo valor que a tabela de chave estrangeira.

Tabela do evento de sistema

```
...
<doctable name="Orders" tablename="VBAP">
<pkfield name="VBELN" fktablename="VBAK"
fkfieldname="VBELN"/>
<pkfield name="POSNR" />
<fieldtoread name="VBELN"/>
<fieldtoread name="POSNR"/>
<fieldtoread name="ERDAT"/>
<fieldtoread name="ERZET"/>
<fieldtoread name="MATNR">
<textref tablename="MAKT" reffieldname="MATNR"
textfieldname="MAKTX" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="KONDM">
<textref tablename="T178T" reffieldname="KONDM"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="SPART">
<textref tablename="TSPAT" reffieldname="SPART"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="WERKS">
<textref tablename="T001W"
reffieldname="WERKS" textfieldname="NAME1"/>
</fieldtoread>
<fieldtoread name="CHARG"/>
<fieldtoread name="PSTYV"/>
<fieldtoread name="ERNAM"/>
<fieldtoread name="NETWR"/>
</doctable>
...
```

A tabela do evento de sistema **VBAP** está associada ao identificador **Orders**. A relação de chave estrangeira especificada na linha

<pkfield name="VBELN" fktablename="VBAK"

fkfieldname="VBELN"/>

vincula a tabela à tabela de chave estrangeira **VBAK**. A chave primária consiste nas colunas **VBELN** e **POSNR**.

Um evento de sistema (elemento XML **event**) é criado no arquivo de saída XML para cada linha extraída da tabela. Para cada elemento XML **fieldtoread**, uma linha do formulário

```
<attribute type="...">...</attribute>
```

é gravada no arquivo de saída. Para alguns elementos **fieldtoread**, o valor extraído da tabela referenciada (elemento XML **textref**) é gravado em conjunto com o valor do campo de dados extraído. O elemento opcional **langfieldname** extrai o texto específico do idioma do campo de dados.

O tipo de atributo do sistema de origem completo é composto pelo identificador da tabela do evento de sistema (**doctable name**), nome do campo (**fieldtoread name**) e nome do campo de texto referenciado (**textref ... textfieldname**). O tipo de atributo do sistema de origem para o primeiro elemento **fieldtoread** extraído com uma tabela referenciada é semelhante a:

<attribute type="Orders-MATNR-MAKTX">...</attribute>

Tabela de dados

```
...
<table name="VBAK">
<pkfield name="VBELN" fktablename="Orders"
fkfieldname="VBELN"/>
<fieldtoread name="VTWEG">
<textref tablename="TVTWT" reffieldname="VTWEG"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VKORG">
<textref tablename="TVKOT" reffieldname="VKORG"
textfieldname="VTEXT" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VDATU"/>
<fieldtoread name="VKBUR">
<textref tablename="TVKBT" reffieldname="VKBUR"
textfieldname="BEZEI" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VKGRP">
<textref tablename="TVGRT" reffieldname="VKGRP"
textfieldname="BEZEI" langfieldname="SPRAS"/>
</fieldtoread>
<fieldtoread name="VBTYP"/>
<fieldtoread name="AUART"/>
</table>
<table name="MARA">
<pkfield name="MATNR" fktablename="Orders"
fkfieldname="MATNR"/>
<fieldtoread name="MATNR"/>
<fieldtoread name="MTART">
<textref tablename="T134T" reffieldname="MTART"
textfieldname="MTBEZ" langfieldname="SPRAS"/>
</fieldtoread>
</table>
...
```

As relações de chave estrangeira com os campos de chave primária **VBELN** e **MATNR** na tabela **Ordens** são usadas para extrair informações suplementares das tabelas de dados **VBAK** e **MARA**.

5.3.8.2. Arquivo de saída XML (formato do evento de sistema do PPM)

O extrato de arquivo abaixo mostra um intervalo de eventos do sistema a partir do arquivo de saída XML gerado usando o arquivo de configuração especificado no capítulo sobre **Configuração da tabela [95]**:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="Orders-CHARG"/>
<attribute type="Orders-ERDAT">20000214</attribute>
<attribute type="Orders-ERNAM">HDM</attribute>
<attribute type="Orders-ERZET">143616</attribute>
<attribute type="Orders-KONDM"/>
<attribute type="Orders-MATNR">P-100</attribute>
<attribute type="Orders-MATNR-MAKTX">
Thread rod M'8 DIN '8895' without tolerance
</attribute>
<attribute type="Orders-NETWR">28.70</attribute>
<attribute type="Orders-POSNR">000010</attribute>
<attribute type="Orders-PSTYV">TAN</attribute>
<attribute type="Orders-SPART">01</attribute>
<attribute type="Orders-SPART-VTEXT">Product category 01
</attribute>
<attribute type="Orders-VBELN">0000000001</attribute>
<attribute type="Orders-WERKS">1000</attribute>
<attribute type="Orders-WERKS-NAME1">

  Plant 1000 (Hamburg)
</attribute>
<attribute type="MARA-MATNR">P-100</attribute>
<attribute type="MARA-MTART">HAWA</attribute>
<attribute type="MARA-MTART-MTBEZ">

  Trading goods

  </attribute>
<attribute type="VBAK-AUART">TA</attribute>
<attribute type="VBAK-VBTYP">C</attribute>
<attribute type="VBAK-VDATU">20000214</attribute>
<attribute type="VBAK-VKBUR"/>
<attribute type="VBAK-VKGRP"/>
<attribute type="VBAK-VKORG">1000</attribute>
<attribute type="VBAK-VKORG-VTEXT">Germany Frankfurt
</attribute>
<attribute type="VBAK-VTWEG">10</attribute>
<attribute type="VBAK-VTWEG-VTEXT">Consumer sales
</attribute>
</event>
<event>
<attribute type="VBAK-VKORG-VTEXT">Germany Frankfurt
</attribute>
<attribute type="VBAK-VTWEG-VTEXT">Consumer sales
</attribute>
<attribute type="Orders-SPART">01</attribute>
<attribute type="Orders-PSTYV">TAD</attribute>
<attribute type="Orders-MATNR">SERVICE</attribute>
<attribute type="Orders-ERDAT">20000214</attribute>
<attribute type="VBAK-VKBUR"/>
```

```
<attribute type="VBAK-VTWEG">10</attribute>
<attribute type="Orders-SPART-VTEXT">Product category 01
</attribute>
<attribute type="Orders-ERZET">143616</attribute>
<attribute type="VBAK-VKORG">1000</attribute>
<attribute type="Orders-POSNR">000020</attribute>
<attribute type="MARA-MTART-MTBEZ">Service
</attribute>
<attribute type="Orders-WERKS-NAME1">

  Plant 1000 (Hamburg)
</attribute>
<attribute type="Orders-CHARG"/>
<attribute type="Orders-WERKS">1000</attribute>
<attribute type="VBAK-VDATU">20000214</attribute>
<attribute type="Orders-MATNR-MAKTX">

  Repair

  </attribute>
<attribute type="Orders-VBELN">0000000001</attribute>
<attribute type="VBAK-AUART">TA</attribute>
<attribute type="VBAK-VKGRP"/>
<attribute type="Orders-NETWR">179.00</attribute>
<attribute type="MARA-MTART">DIEN</attribute>
<attribute type="VBAK-VBTYP">C</attribute>
<attribute type="Orders-ERNAM">HDM</attribute>
<attribute type="MARA-MATNR">SERVICE</attribute>
<attribute type="Orders-KONDM"/>
</event>
<event>
<attribute type="VBAK-VKGRP-BEZEI">GR. F2 Mr. Mayer
</attribute>
<attribute type="VBAK-VKORG-VTEXT">Germany Frankfurt
</attribute>
<attribute type="VBAK-VTWEG-VTEXT">Consumer sales
</attribute>
<attribute type="Orders-SPART">01</attribute>
<attribute type="Orders-PSTYV">TAN</attribute>
<attribute type="Orders-MATNR">P-100</attribute>
<attribute type="Orders-ERDAT">20000214</attribute>
<attribute type="VBAK-VKBUR">1000</attribute>
<attribute type="VBAK-VTWEG">10</attribute>
<attribute type="Orders-SPART-VTEXT">Product category 01
</attribute>
<attribute type="Orders-ERZET">131257</attribute>
<attribute type="VBAK-VKBUR-BEZEI">Frankfurt office
</attribute>
<attribute type="VBAK-VKORG">1000</attribute>
<attribute type="Orders-POSNR">000010</attribute>
<attribute type="MARA-MTART-MTBEZ">

  Trading goods

  </attribute>
```

```
<attribute type="Orders-WERKS-NAME1">

  Plant 1000 (Hamburg)
</attribute>
<attribute type="Orders-CHARG"/>
<attribute type="Orders-WERKS">1000</attribute>
<attribute type="VBAK-VDATU">20000214</attribute>
<attribute type="Orders-MATNR-MAKTX">
Thread rod M'8 DIN '8895' without tolerance
</attribute>
<attribute type="Orders-VBELN">0000000002
</attribute>
<attribute type="VBAK-AUART">TA</attribute>
<attribute type="VBAK-VKGRP">101</attribute>
<attribute type="Orders-NETWR">28.70</attribute>
<attribute type="MARA-MTART">HAWA</attribute>
<attribute type="VBAK-VBTYP">C</attribute>
<attribute type="Orders-ERNAM">HDM</attribute>
<attribute type="MARA-MATNR">P-100</attribute>
<attribute type="Orders-KONDM"/>
</event>
...
</eventlist>
```

5.3.9. Extração em bloco

Pode influenciar a relação entre a execução e a eficiência de memória da operação de extração.

O valor do argumento da linha de comando **-cpd** determina o número de eventos do sistema que são mantidos simultaneamente na memória do sistema. O número máximo possível depende do tamanho da memória disponível do sistema.

A memória real utilizada pelo sistema depende dos seguintes fatores:

- Número total de eventos do sistema a serem extraídos no período de análise definido
- Número de eventos do sistema mantidos em paralelo no sistema (documentos processados simultaneamente)
- Número de campos de dados a ser extraídos (elementos XML **fieldtoread** e **reftext**).
- Requisitos de memória dos campos de dados extraídos

5.3.10. Extrair a primeira ou última linha de uma classificação

A classe Java a ser usada **com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.**

ZSortWithInteger_jdbc2ppm classifica registros em uma tabela de dados por meio do critério de classificação especificado e grava o primeiro ou último registro nos eventos do sistema a serem gerados. A classe tem de ser especificada no atributo **classtouse** do elemento XML **table**.

Exemplo 50.

Com essa classe, pode usar um critério inteiro (quantidade de elementos) para classificar registros de dados referenciados por meio de relações de chave estrangeira, na tabela **ORDER ITEM**. O critério de

classificação é o campo **AMOUNT**. A partir dos dados classificados, o registro de dados com o maior valor de critério de classificação (para **ITEM AMOUNT_MAX**) e o registro de dados com o menor valor de critério de classificação (para **ITEM AMOUNT_MIN**) são gravados nos eventos de sistema gerados, cada um com o número do elemento correspondente. O seguinte extrato de arquivo ilustra a configuração de extração nas condições descritas acima:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE jdbc_tableconfiguration SYSTEM 'jdbc_tableconfiguration.dtd'>
<jdbc_tableconfiguration>
<configuration name="SortWithInteger">
<docspec>
<doctable name="ORDER HEADER">
<pkfield name="NUMBER"/>
</doctable>
</docspec>
<table name="ITEM AMOUNT_MAX" tablename="ORDER ITEM"
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZSortWithInte
ger_jdbc2ppm">
<parameter name="SORTCRITERION">
<value>AMOUNT</value>
</parameter>
<parameter name="USE">
<value>MAX</value>
</parameter>
<pkfield name="NUMBER" fktablename="ORDER HEADER" fkfieldname="NUMBER"/>
<fieldtoread name="ITEM"/>
<fieldtoread name="AMOUNT"/>
</table>
<table name="ITEM AMOUNT_MIN" tablename="ORDER ITEM"
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZSortWithInte
ger_jdbc2ppm">
<parameter name="SORTCRITERION">
<value>AMOUNT</value>
</parameter>
<parameter name="USE">
<value>MIN</value>
</parameter>
<pkfield name="NUMBER" fktablename="ORDER HEADER" fkfieldname="NUMBER"/>
<fieldtoread name="ITEM"/>
<fieldtoread name="AMOUNT"/>
</table>
</configuration>
</jdbc_tableconfiguration>
```

Exemplo 51.

Um evento de sistema extraído com essa configuração poderia ter a seguinte aparência:

```
<event>
<attribute type="ORDER HEADER NUMBER">
4711
</attribute>
<attribute type="ITEM AMOUNT_MAX-ITEM">
20
```

```
</attribute>
<attribute type="ITEM AMOUNT_MAX-AMOUNT">
100000
</attribute>
<attribute type="ITEM AMOUNT_MIN-ITEM">
50
</attribute>
<attribute type="ITEM AMOUNT_MIN-AMOUNT">
470
</attribute>
</event>
```

Exemplo 52.

O critério de classificação (neste caso **AMOUNT**) é automaticamente incluído na especificação do evento de sistema.

O modelo para criar a configuração exibida é semelhante a isso:

```
<table name="..." tablename="..."
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZSortWithInte
ger_jdbc2ppm">
<parameter name="SORTCRITERION">
<value>...</value>
</parameter>
<parameter name="USE">
<value>...</value>
</parameter>
...
See chapter
JDBC table configuration [76]
...
</table>
```

A tabela abaixo reitera as entradas de configuração mais importantes para a definição **table** acima:

Elemento/atributo XML	Valor: Descrição
name	O nome especificado é adicionado como um prefixo aos atributos do sistema de origem extraídos
tablename **(opcional)**	Tabela da qual a informação deve ser extraída. Valor padrão: Valor do atributo XML **name**
classtouse	**com.idsscheer.ppm.xmlextractortools.** **extractor. jdbc2ppm.ZSortWithInteger_jdbc2ppm**: Classe Java a ser usada
nome do parâmetro	
SORTCRITERION	Primeiro parâmetro a ser especificado. Tem de ser especificado um único elemento XML **value** (nome de campo do critério de classificação). O campo especificado apenas pode incluir valores inteiros.

Elemento/atributo XML	Valor: Descrição
USE	Segundo a ser especificado. Tem de ser especificado um único elemento XML **value**. Valores válidos: **MIN** (é selecionado o registro de dados com o menor valor inteiro para o critério de classificação) **MAX** (é selecionado o registro de dados com o maior valor inteiro para o critério de classificação) O primeiro ou o último registro de dados da lista gerada com base no critério de classificação especificado é selecionado para ser extraído.

Deve usar essa classe para extrair a primeira ou a última linha de uma classificação com moderação, pois as operações de classificação e seleção podem resultar em perda de desempenho e memória.

5.3.11. Extrair a primeira ou última linha usando um carimbo de data/hora

Pode usar a classe Java **com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZSortWithTimestamp_jdbc2ppm** para extrair dados pelo carimbo de data/hora (por exemplo, tipo de dados Oracle **TIMESTAMP**) de uma maneira classificada. O primeiro ou último registro extraído é gravado em um evento. A classe é especificada no elemento XML **table** do atributo XML **classtouse**. Os dois parâmetros obrigatórios são especificados nos elementos XML correspondentes:

- Parâmetro **name="SORTCRITERION"** com um elemento XML **value** cujo valor determina o nome de campo da tabela a ser extraída.
- Parâmetro **name="USE"** com um elemento XML **value** que tem de incluir o valor **MAX** (o registro com o valor mais recente é transferido) ou **MIN** (o registro com o valor mais antigo é transferido).

Exemplo 53.

O exemplo seguinte mostra um extrato desse tipo de configuração de tabela.

```
<table ...
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZSortWithTime
stamp_jdbc2ppm">
<parameter name="SORTCRITERION">
<value>...</value>
</parameter>
<parameter name="USE">
<value>...</value>
</parameter>
... conditions ...
... pkfields ...
... fieldstoread ...
</table>
```

Exemplo 54.

O critério de classificação especificado é automaticamente adicionado à saída do evento.

Os campos do último estado da etapa de processo em cada etapa de processo devem ser extraídos da tabela **DBO.WMPROCESSSTEP** (histórico de estado das etapas de processo). Dependendo do valor do campo **AUDITTIMESTAMP** (tipo de dados Oracle TIMESTAMP), a última entrada (**MAX**) tem de ser transferida.

```
<configuration name="AUFK_JCDS_AUFK">
...
<table name="WMPROCESSSTEP_END" tablename="DBO.WMPROCESSSTEP"
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZSortWithTime
stamp_jdbc2ppm">
<parameter name="USE">
<value>MAX</value>
</parameter>
<parameter name="SORTCRITERION">
<value>AUDITTIMESTAMP</value>
</parameter>
<pkfield name="INSTANCEID" fktablename="WMPROCESSSTEP"
fkfieldname="INSTANCEID" />
<pkfield name="INSTANCEITERATION" fktablename="WMPROCESSSTEP"
fkfieldname="INSTANCEITERATION" />
<pkfield name="STEPID" fktablename="WMPROCESSSTEP" fkfieldname="STEPID" />
<pkfield name="STEPITERATION" fktablename="WMPROCESSSTEP"
fkfieldname="STEPITERATION" />
<fieldtoread name="AUDITTIMESTAMP" />
<fieldtoread name="INSERTTIMESTAMP" />
<fieldtoread name="STATUS" />
</table>
...

  </configuration>
```

Exemplo 55.

Um evento extraído com essa configuração poderia ter a seguinte aparência:

```
<event>
...
<attribute type="WMPROCESSSTEP_END-AUDITTIMESTAMP">24.08.2010 09:51:13.477</
attribute>
<attribute type="WMPROCESSSTEP_END-INSERTTIMESTAMP">24.08.2010 09:51:13.550</
attribute>
<attribute type="WMPROCESSSTEP_END-STATUS">2</attribute>
...
</event>
```

Os dados extraídos incluem milissegundos ou nanossegundos

Por padrão, os carimbos de data/hora são extraídos e classificados ao segundo. Se existirem vários registros com valor de carimbo de data/hora idêntico, um deles será selecionado aleatoriamente e usado para determinação do valor.

Se os carimbos de data/hora forem salvos ao milissegundo ou ao nanossegundo no banco de dados, eles poderão ser extraídos e classificados em milissegundos, e respectivamente, em nanossegundos. Para implementar isso, pode especificar o valor **MILLISECOND** ou **NANOSECOND** no atributo **precisionoftime** do elemento XML **databasesettings** do arquivo de configuração JDBC. O valor padrão é **SECOND**.

Com o valor **SECOND**, todos os carimbos de data/hora ou horas são extraídos no formato **MM/dd/aaaa HH:mm:ss** ou **HH:mm:ss** e gravados no arquivo de evento. Os dados são classificados ao segundo, de acordo com o carimbo de data/hora.

Com o valor **MILLISECOND**, os carimbos de data/hora ou horas são extraídos no formato **MM/dd/aaaa HH:mm:ss** ou **HH:mm:ss.SSS** e gravados no arquivo de evento. Os dados são classificados ao milissegundo, de acordo com o carimbo de data/hora.

Com o valor **NANOSECOND**, os carimbos de data/hora ou horas são extraídos no formato **MM/dd/aaaa HH:mm:ss** ou **HH:mm:ss.SSS** e gravados no arquivo de evento. Os dados são classificados ao nanossegundo, de acordo com o carimbo de data/hora.

O valor padrão é **SECOND**, ou seja, as horas são extraídas ao segundo após a atualização de uma configuração do cliente existente.

Exemplo 56.

O seguinte exemplo mostra um extrato do arquivo de configuração XML do JDBC:

```
<jdbcconf>
<databasesettings name="jdbc_oracle" dbtype="ORACLE"
precisionoftime="MILLISECOND">
<driverclass>oracle.jdbc.driver.OracleDriver</driverclass>
<maxconditionlength>2000</maxconditionlength>
<fetchsize>1000</fetchsize>
</databasesettings>
<databaseconnection name="jdbc_oracle10_connection">
<dbsettings>jdbc_oracle</dbsettings>
<url>...</url>
<user>...</user>
...
</databaseconnection>
</jdbcconf>
```

5.3.12. Multiplicação de eventos do sistema em tabelas

Ao extrair de um banco de dados JDBC, tem a opção de criar eventos adicionais do sistema com base em uma tabela de eventos do sistema.

No nível do sistema de origem, pode existir um relacionamento 1: n que não pode ser resolvido ao determinar os eventos do sistema. Use a classe **com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZTableMultiplyEvents_jdbc2ppm** para criar vários eventos de sistema a partir de um evento do sistema e resolver assim a relação 1:n.

```
<configuration name="MultiplyEvents">
<docspec>
<doctable name="VBAK" tablename="VBAK">
<condition fieldname="VBELN" logicaloperator="in">
<value>0000006662</value>
<value>0000006741</value>
</condition>
<pkfield name="VBELN" />
</doctable>
</docspec>
```

```
<table name="VBAP"
classtouse="com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZTableMultipl
yEvents_jdbc2ppm">
<pkfield name="VBELN" fktablename="VBAK" fkfieldname="VBELN"/>
<fieldtoread name="POSNR"/>
<fieldtoread name="MATNR"/>
</table>
<table name="MAKT">
<condition fieldname="SPRAS" logicaloperator="eq">
<value>de</value>
</condition>
<pkfield name="MATNR" fktablename="VBAP" fkfieldname="MATNR"/>
<fieldtoread name="MAKTX"/>
</table>
</configuration>
```

Exemplo 57.

Este é o arquivo de saída de evento do sistema:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAK-VBELN">0000006662</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">Flatscreen MS 1460 P</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-06</attribute>
<attribute type="VBAP-POSNR">000010</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">Flatscreen MS 1775P</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-10</attribute>
<attribute type="VBAP-POSNR">000020</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">MAG PA/DX 175</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-14</attribute>
<attribute type="VBAP-POSNR">000030</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">Jotachi SN4500</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-18</attribute>
<attribute type="VBAP-POSNR">000040</attribute>
</event>
</eventlist>
```

Exemplo 58.

O exemplo seguinte (excluindo a classe descrita) explica o procedimento durante a extração:

```
<configuration name="MultiplyEvents_Doctable_Only">
<docspec>
<doctable name="VBAK" tablename="VBAK">
<condition fieldname="VBELN" logicaloperator="in">
<value>0000006662</value>
<value>0000006741</value>
</condition>

<pkfield name="VBELN" />
</doctable>
</docspec>

<table name="VBAP" >
<pkfield name="VBELN" fktablename="VBAK" fkfieldname="VBELN"/>
<fieldtoread name="POSNR"/>
<fieldtoread name="MATNR"/>
</table>

<table name="MAKT">
<condition fieldname="SPRAS" logicaloperator="eq">
<value>de</value>
</condition>

<pkfield name="MATNR" fktablename="VBAP" fkfieldname="MATNR"/>
<fieldtoread name="MAKTX"/>
</table>
</configuration>
```

Uma vez que várias entradas na tabela **VBAP** são associadas a uma entrada na tabela **VBAK**, apenas uma linha aleatória é extraída da tabela **VBAP**.

Este é o arquivo de saída de evento do sistema:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAK-VBELN">0000006662</attribute>
</event>
<event>
<attribute type="MAKT-MAKTX">Flatscreen MS 1775P</attribute>
<attribute type="VBAK-VBELN">0000006741</attribute>
<attribute type="VBAP-MATNR">M-10</attribute>
<attribute type="VBAP-POSNR">000020</attribute>
</event>
</eventlist>
```

5.3.13. Extrair campos ou tabelas individuais para análise de dados

Para permitir a fácil extração de dados para análise de dados, pode extrair todo o conteúdo de uma tabela do sistema de origem para um arquivo no formato de evento, que pode ser importado para um território de análise para análise de dados.

Também pode restringir os dados a serem extraídos [61] definindo condições para a extração de dados.

A tabela a ser extraída não é configurada usando a configuração da tabela, mas o próprio arquivo da fonte de dados. Para isso, o arquivo **datasource.dtd** contém as entradas seguintes:

Elemento/atributo XML	Descrição
analysistype	Atributos XML: Tem de ter o valor **DATA_ANALYTICS**.
realmtable	Elemento XML abrangente para configurar a fonte de dados para análise de dados.
tablename	Atributo XML do elemento **realmtable**: Nome da tabela no sistema de origem
sourcetable	Elemento XML abrangente para configurar a tabela de fonte de dados para análise de dados. Tem de incluir pelo menos um elemento **sourcefield**.
tablename	Atributo XML do elemento **sourcetable**: Nome da tabela no sistema de origem
sourcefield	Contém o nome do campo da tabela do sistema de origem.

O atributo **analysistype** do elemento XML **datasource** tem de ter o valor **DATA_ANALYTICS** se o elemento **<realmtable>** especificar uma tabela de território de análise (valor padrão: PROCESS).

O atributo **tablename** do elemento **<realmtable>** indica o nome da tabela de destino na configuração do território de análise. O nome da tabela não afeta a extração em si, mas é avaliado apenas pela importação do PPM.

Pode encontrar mais informações sobre importação de dados para análise no manual do usuário para análise de dados do PPM.

O elemento XML **<realmtable>** contém o elemento opcional **<sourcetable>** que especifica a tabela a ser extraída. As colunas a serem extraídas desta tabela têm de ser especificadas no elemento **<sourcefield>**. O elemento **<sourcetable>** é opcional. Para o Extrator JDBC ou SAP, devem ser especificadas uma única tabela de origem e pelo menos uma coluna de origem. Caso contrário, uma mensagem de erro será emitida durante a análise do arquivo da fonte de dados.

Uma definição de fonte de dados apenas pode ter, no máximo, uma tabela. É impossível restringir o número de linhas. Todas as linhas são extraídas, incluindo todas as linhas com valores idênticos nas colunas a serem extraídas. Por exemplo, se as colunas **Primeiro nome** e **Sobrenome** tiverem de ser extraídas e a tabela incluir dez entradas **Peter** e **Schmidt**, serão criados dez eventos com valores de atributo idênticos.

O exemplo seguinte explica a configuração:

```
<realmtable tablename="COMPANY_EMPLOYEE">
<sourcetable tablename="EMPLOYEE">
<sourcefield>EMPLOYEE_ID</sourcefield>
<sourcefield>NAME</sourcefield>
</sourcetable>
...
</realmtable>
<dataextraction>
<outputfilename>..\custom\testclient\data\employee.xml</outputfilename>
</dataextraction>
...
<systemconfig>..\custom\testclient\SourceSystemConfig.xml</systemconfig>
```

Em contraste com o comportamento do Extrator convencional JDCB ou SAP, os atributos não recebem mais um nome de tabela como um prefixo no arquivo de saída do evento. Por exemplo, se a tabela **EMPLOYEE** foi extraída, normalmente o extrator gera eventos do tipo **<table name>-<column name>**:

```
<event>
<attribute type="EMPLOYEE-EMPLOYEE_ID">4711</attribute>
<attribute type="EMPLOYEE-NAME">Schmidt</attribute>
</event>
Extracting a table using the <realmtable> element, however, creates only events
without table names:
<event>
<attribute type="EMPLOYEE_ID">4711</attribute>
<attribute type="NAME">Schmidt</attribute>
</event>
```

Valores null no modo ANÁLISE DE DADOS

Se o valor de uma coluna for **null** no momento em que uma tabela do território de análise é extraída, esse valor não é gravado no evento. Se uma linha **EMPLOYEE_ID = 4712** existir sem sobrenome, o extrator cria os eventos seguintes.

```
<event>
<attribute type="EMPLOYEE_ID">4711</attribute>
<attribute type="NAME">Schmidt</attribute>
</event>
<event>
<attribute type="EMPLOYEE_ID">4712</attribute>
</event>
However, if all column values to be extracted are null, the event is written
with empty attributes:
<event>
<attribute type="EMPLOYEE_ID">4711</attribute>
<attribute type="NAME">Schmidt</attribute>
</event>
<event>
<attribute type="EMPLOYEE_ID"></attribute>
<attribute type="NAME"></attribute>
</event>
```

Se existirem várias dessas linhas, elas serão - em contraste com o procedimento de extração comum (analysistype = PROCESS) - transferidas devidamente, de modo que existam tantos elementos <event> no arquivo de eventos, como linhas na tabela de dados do sistema de origem.

5.3.13.1. Restringir extração de dados

Se quiser extrair apenas uma parte específica do conteúdo da tabela, poderá definir as condições de extração de dados. Pode limitar o volume de dados a ser extraído usando um carimbo de data/hora ou valor inteiro (por exemplo, uma sequência).

Para isso, especifique as condições obrigatórias na configuração da fonte de dados, usando o elemento **condition** e o atributo **dataextractiontype**.

```
<datasource name="VBAP" type="JDBC" analysistype="DATA_ANALYTICS"
dataextractiontype="TIME_BASED">
<realmtable tablename="VBAP">
<sourcetable tablename="VBAP2">
<condition logicaloperator="char_creationtimestamp" fieldname="AEDAT">
<value>yyyy-MM-dd</value>
</condition>
<sourcefield>AEDAT</sourcefield>
<sourcefield>ERDAT</sourcefield>
<sourcefield>ERNAM</sourcefield>
<sourcefield>ERZET</sourcefield>
<sourcefield>MATKL</sourcefield>
<sourcefield>MATNR</sourcefield>
<sourcefield>WERKS</sourcefield>
</sourcetable>
</realmtable>
...
</datasource>
```

O atributo **dataextractiontype** pode ter os valores seguintes nas fontes de dados para análise de dados:

- COMPLETE: A extração é executada com um critério de tempo da data **01.01.1990 00:00:00** e um critério de inteiro do valor **0**. Esta é também a configuração padrão se o atributo não existir.
- TIME_BASED: Tem de ser executada uma extração de dados baseada em um carimbo de data/hora.
- VALUE_BASED: Tem de ser executada uma extração de dados baseada em um valor.

Não há verificação se a condição de extração de dados usada corresponde ao tipo de extração de dados configurado (dataextractiontype=). Se a configuração for criada usando o PPM CTK, o PPM CTK garante valores corretos. Se o arquivo for configurado manualmente e uma condição que não corresponde for usada, pode ocorrer um erro durante a extração de dados.

5.3.14. Tabelas de extração com colunas de chave/valor

É possível usar uma classe de extração especial para extrair tabelas que contém células com pares de chave/valor. Um novo atributo de evento é criado para cada chave extraída de um evento.

Classe de extração

com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZTableKeyValueFieldsToAttributes_jdbc2ppm

O comportamento e a configuração dessa classe de extração são idênticos aos da classe de extração correspondente de um sistema SAP, exceto para o nome de classe diferente. Para ver os detalhes, consulte o capítulo Tabelas de extração com colunas de chave/valor [62].

A seguinte configuração se refere à classe de extração JDBC mencionada acima:

```
<xmlextractor_tableconfiguration>
    <configuration name="workflow">
        <docspec>
            <doctable name="WF_ITEMS" tablename="WF_ITEMS">
                <pkfield name="ID" />
                <pkfield name="TYPE" />
                <pkfield name="CREATOR" />
                <pkfield name="TEXT" />
            </doctable>
        </docspec>
        <table name="WF_ITEM_ATTRIBUTES" tablename="WF_ITEM_ATTRIBUTES" classtouse=
        "com.idsscheer.ppm.xmlextractortools.extractor.jdbc2ppm.ZTableKeyValueFieldsToAttributes_jdbc2ppm">
            <parameter name="KEYVALUEFIELDS">
                <value>ATTRIBUTE#-#VALUE</value>
            </parameter>
            <pkfield name="ID" fktablename="WF_ITEMS" fkfieldname="ID" />
            <fieldtoread name="ATTRIBUTE" />
            <fieldtoread name="VALUE" />
        </table>
    </configuration>
</xmlextractor_tableconfiguration>
```

5.4. Programa de linha de comando

Você criou arquivo(s) de saída XML usando o programa de linha de comando **runjdbc2ppm.bat**.

Chamar o programa sem parâmetros ou com **-h** ou **-?** gera a ajuda on-line no console, descrevendo todas as opções disponíveis.

5.4.1. Argumentos do programa de linha de comando

5.4.1.1. Argumentos gerais

-version

Exibe o número de versão do extrator de processo PPM usado. Outros argumentos especificados são ignorados.

-information yes|no

Especifica se a informação é para ser exibida (yes) ou não (no) durante a importação. O padrão é **yes**.

-warning yes|no

Especifica se as mensagens de aviso são para ser exibidas (yes) ou não (no) durante a importação. O padrão é **yes**.

-error yes|no

Aqui pode especificar se as mensagens de erro são para ser exibidas (yes) ou não (no) durante a importação. O padrão é **yes**.

-protocolfile <file name>

Especifique o arquivo de log para o qual todas as mensagens são gravadas durante a importação.
Se especificar um arquivo, somente as mensagens de erro críticas que resultam em cancelamento do programa serão exibidas na tela.

-language <ISO code>

Especifique o idioma no qual as informações de log devem ser exibidas.

5.4.1.2. Argumentos específicos do banco de dados de origem

-datasource <file name> (opcional)

Especifique a fonte de dados que deseja usar para a extração. O arquivo XML contém especificações de todos os arquivos a serem usados para extração e saída XML. Certifique-se de que usa este parâmetro em vez de e não em conjunto com os parâmetros **-datasourcelist**, **-systemconfig**, **-tableconfig**, **-calcconfig**, **-outfile** ou **-nozip**.

-datasourcelist <file name>

Várias fontes de dados podem ser extraídas simultaneamente através do argumento **-datasourcelist**. Isso corresponde a várias extrações consecutivas usando o argumento **-datasource**. São extraídas apenas as fontes de dados JDBC.

Ver o capítulo Extrair várias fontes de dados [119].

-systemconfig <filename> <configname>

É aqui que especifica o nome do arquivo XML contendo a configuração do sistema. O conteúdo do arquivo é específico do banco de dados de origem e pode conter várias configurações. O nome da configuração a ser usada é especificado no segundo argumento. Se o arquivo XML incluir apenas uma configuração, esta é usada automaticamente. Nesse caso, não é necessário especificar um nome de configuração.

-tableconfig <file name> <config name>

É aqui que especifica o nome do arquivo XML contendo a configuração da tabela. O conteúdo do arquivo é específico do banco de dados de origem. O arquivo deve conter todas as configuração da tabela especificadas na configuração da tabela. O nome da configuração da tabela é especificado no segundo argumento. Se o arquivo XML incluir apenas uma configuração, esta é usada automaticamente. Nesse caso, não é necessário especificar um nome de configuração.

-calcconfig <file name> (opcional)

É aqui que especifica o nome do arquivo XML com o qual pode alterar atributos do arquivo de saída XML ou adicionar atributos, incluindo cálculos de atributos.

-begindate <dd.mm.yyyy> (opcional)

É aqui que especifica a data a partir da qual os dados devem ser extraídos do banco de dados de origem. Se não usar este parâmetro, é usado o valor do atributo XML **lastreaddate** da fonte de dados ou configuração de sistema especificada. O formato **aaaammdd** é suportado.

-begintime <hh:mm:ss> (opcional)

Especifique a hora a partir da qual os dados devem ser extraídos do sistema de origem. Se não usar este parâmetro e **-begindate**, é usado o valor do atributo XML **lastreadtime** da fonte de dados ou configuração de sistema especificada. Se não usar o parâmetro, mas usar **-begindate**, será definido o valor padrão **000000**.

-enddate <dd.mm.yyyy> (opcional)

É aqui que especifica a data até à qual os dados devem ser extraídos do banco de dados de origem. Se não usar este parâmetro, será usada a data atual. O formato **aaaammdd** é suportado.

-endtime <hh:mm:ss> (opcional)

É aqui que especifica a hora até à qual os dados devem ser extraídos do banco de dados de origem. Se não usar o parâmetro, mas usar

-enddate, **235959** será definido como o valor padrão. Se não usar este parâmetro, nem **-enddate**, será usada a hora atual.

Se o PPM Process Extractor JDBC-2-PPM for iniciado sem que seja especificado um intervalo de tempo, a hora de início será extraída do arquivo de configuração especificado. A data e hora atuais são usadas como hora de término.

-valueconstraint <parameter1> ... <parameter4> (opcional)

Use este parâmetro para delimitar o volume de dados a ser extraído usando um critério de número inteiro. São permitidos os operadores de comparação "<", "<=", ">" e ">=". Pode comparar o valor do campo correspondente com um ou dois valores (intervalo). Se um último valor extraído (**lastreadvalue**) já tiver sido salvo na configuração do sistema ou fonte de dados, também poderá usar este valor para a comparação (ver o capítulo Extração automática e contínua [117]).

Exemplo 59. ***[en-us]***

```
-valueconstraint 25000020 "<="
Only data records are extracted whose integer value for the field specified in
the event specification is smaller than or equal to the value 25000020.


  -valueconstraint 15000020 ">" 25000020 "<"
```

Intervalo: Apenas são extraídos registros de dados cujo valor inteiro para o campo especificado na especificação do evento seja **menor do que** o valor **25000020** e **maior do que** o valor **15000020**.

-save_value_minimum (opcional)

Use este parâmetro para especificar que com **-valueconstraint**, o menor valor da última leitura é salvo como **lastreadvalue** no arquivo de configuração usado (configuração do sistema ou fonte de dados) ao extrair com restrições. Valor padrão: Salvar o valor maior

-cpd <int> (opcional)

Esse parâmetro é usado para especificar o número de eventos do sistema a serem extraídos simultaneamente (documentos processados simultaneamente). O valor especificado extrai gradualmente os campos de chave primária da tabela referenciada usando o elemento XML **doctable** na configuração da tabela e os armazena na memória Java do extrator. A especificação de um valor mais alto resulta em melhor desempenho de extração e uso de memória mais eficiente.

Se não especificar o parâmetro, é usado o valor padrão recomendado de 20000.

-ping (opcional)

Este parâmetro é usado para testar a conexão ao banco de dados de origem especificado. Não é extraído qualquer dado.

-removeempty (opcional)

Use este parâmetro para remover os atributos extraídos sem valores antes da transformação de atributos.

5.4.1.3. Argumentos específicos do arquivo de saída

-outfile <file name>

Esse parâmetro é usado para especificar o nome do arquivo de saída XML. A extensão de arquivo é adicionada automaticamente. Por padrão, o arquivo é fornecido como um arquivo ZIP.

-outfileencoding <encoding> (opcional)

Esse parâmetro é usado para especificar a codificação do arquivo de saída XML. Valor padrão: UTF-8

-nozip (opcional)

Use esse parâmetro para especificar que o arquivo de saída deve ser gerado como um arquivo XML em vez de um arquivo ZIP.

-pikidatamapping <file name> <pc name> (opcional)

Use esse parâmetro para especificar o nome do arquivo XML que contém o mapeamento no primeiro argumento. No segundo argumento, especifique o nome da série de medidas na qual as séries de medidas independentes da instância de processo devem ser salvas.

-dimdatamapping <file name> <dim name> (opcional)

Use esse parâmetro para especificar o nome do arquivo XML que contém o mapeamento no primeiro argumento. No segundo argumento, especifique o nome da dimensão para a qual os valores extraídos serão subsequentemente importados.

-sorteventattributes (opcional)

Use esse parâmetro para classificar os atributos do sistema de origem nos eventos do sistema em ordem alfanumérica, usando o tipo de atributo. Esse parâmetro pode ter um impacto negativo na velocidade de extração para grandes volumes de dados.

5.4.1.4. Extração automática e contínua

Pode extrair dados de forma completa e automática eliminando os parâmetros **-begindate**, **-begintime**, **-enddate**, **-endtime** na linha de comando ou usando **-valueconstraint**, sem valor, para o primeiro operador de comparação. Se o intervalo de dados a ser extraído for restrito usando valores inteiros, o menor ou o

maior dos últimos valores extraídos será salvo ou atualizado no arquivo de configuração utilizado. Se houver restrições de tempo no intervalo de dados a ser extraído, o horário da última extração de dados será salvo ou atualizado no arquivo de configuração.

Devem ser atendidas as seguintes condições:

- Uma configuração correta para **char_creationtimestamp** ou **valueconstraint** ou **date_creationtimestamp** para os dados a serem extraídos é especificada na especificação do evento (ver o capítulo Operadores de condição [85]).
- Use sempre a mesma chamada de linha de comando para extração automatizada contínua, por exemplo:

```
runjdbc2ppm -datasource datasource.xml -valueconstraint ">"
```

- Quando usa o parâmetro **-valueconstraint**, certifique-se de que salva o maior dos últimos valores extraídos para todas as operações de extração (padrão) ou o menor dos últimos valores extraídos, especificando

  -save_value_minimum.

  O valor de **lastreadvalue** no arquivo de configuração apenas é atualizado se o maior dos últimos valores extraídos for maior do que o valor salvo atualmente quando executa a linha de comando sem o parâmetro

  -save_value_minimum, ou se o menor dos últimos valores extraídos for menor do que o valor salvo atualmente quando usa

  -save_value_minimum.

```
-valueconstraint ">="
```

Exemplo 60.

Apenas são extraídos os registros de dados cujo valor inteiro para restringir a área de dados a ser extraída seja **maior do que** ou **igual ao** valor salvo como o último valor extraído (**lastreadvalue**) na configuração do sistema ou fonte de dados. Se nenhum valor for salvo, o valor padrão **0** é usado (no exemplo, todos os registros de dados com valores de critério inteiro **>=0** seriam extraídos). O maior dos últimos valores extraídos (**300** neste exemplo) é salvo como **lastreadvalue** de acordo com a configuração padrão (ver o parâmetro **-save_value_minimum**).

```
-valueconstraint ">=" 270 "<="
```

Exemplo 61.

São extraídos todos os registros de dados com um valor inteiro de **>=40** e **<=270** para restringir a área de dados a ser extraída. Após a extração, o valor para **lastreadvalue** é atualizado, por exemplo, para **270** se este for realmente o maior dos últimos valores extraídos.

Ao chamar

```
runjdbc2ppm -datasource datasource.xml -valueconstraint ">"
```

Exemplo 62.

novamente, são extraídos todos os registros de dados cujo valor inteiro para delimitar a área de dados a ser extraída seja maior do que **270** e cujo carimbo de data/hora seja tão antigo ou mais recente que a hora de início da extração de dados (**31.12.1997 15:59:59**).

A hora de início da extração de dados é inserida como **lastreaddate**/**lastreadtime** no arquivo de configuração.

Se eliminar os parâmetros **-begindate**/**-begintime**, extrai sempre os registros de dados cujos carimbos de data/hora são mais recentes (maiores) do que os especificados e extraídos em último lugar usando **lastreaddate**/**lastreadtime**. A data e hora atual são inseridas como lastreaddate/lastreadtime se não usar os parâmetros -enddate e -endtime.

5.5. Extrair de várias fontes de dados

Pode usar uma lista de fontes de dados para extrair simultaneamente de várias fontes de dados. A lista de fontes de dados é especificada em um arquivo de configuração separado.

Cada nova fonte de dados JDBC criada no PPM CTK é automaticamente adicionada ao final da lista de fontes de dados do cliente atual. Durante a extração usando a opção de linha de comando **-datasourcelist <file name>** os dados dessas fontes de dados são extraídos consecutivamente como se a extração de dados fosse chamada várias vezes usando a opção **-datasource <file name>**. A sequência da extração da fonte de dados é especificada no arquivo de configuração. As fontes de dados de um tipo diferente de JDBC são ignoradas durante a extração.

Aplica-se o seguinte à resolução de problemas:

- Se a extração for chamada por meio de uma lista de fontes de dados válida que não contenha nenhuma fonte de dados correspondente, a extração será terminada sem exibir uma mensagem de erro.
- Se a extração for chamada usando uma lista de fontes de dados contendo várias fontes de dados correspondentes e se ocorrer um erro durante a extração de uma dessas fontes de dados, resultando no cancelamento da extração, o procedimento continuará com a fonte de dados seguinte. Isso significa que o cancelamento da extração de uma fonte de dados não leva ao cancelamento de todo o procedimento.
- Se ocorrer um erro em pelo menos uma fonte de dados, o que levaria a um estado de erro de saída durante a extração única dessa fonte de dados, o estado do último erro de saída será retornado pelo processo geral de extração da fonte de dados.

5.6. Extrair dados do tipo de dados BIT

Pode extrair dados do tipo de dados **BIT** a partir de bancos de dados do servidor SQL. Um valor extraído de um campo de dados BIT pode ser mapeado para um valor do tipo de dados BOOLEAN.

Exemplo 63.

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE eventlist SYSTEM "event.dtd">

<eventlist>

<metadata>

<attr_desc type="DATATYPE_BIT-ID">

<ppmdatatype>LONG</ppmdatatype>

</attr_desc>

<attr_desc type="DATATYPE_BIT-VALUE_BIT_2">

<ppmdatatype>BOOLEAN</ppmdatatype>

</attr_desc>

<attr_desc type="DATATYPE_BIT_2-VALUE_BIT_1">

<ppmdatatype>BOOLEAN</ppmdatatype>

</attr_desc>

<attr_desc type="DATATYPE_BIT_2-VALUE_BIT_3">

<ppmdatatype>BOOLEAN</ppmdatatype>

</attr_desc>

<attr_desc type="DATATYPE_BIT_2-VALUE_INT">

<ppmdatatype>LONG</ppmdatatype>

</attr_desc>

</metadata>

<event>

<attribute type="DATATYPE_BIT-ID">1</attribute>

<attribute type="DATATYPE_BIT-VALUE_BIT_2">false</attribute>

<attribute type="DATATYPE_BIT_2-VALUE_BIT_1">false</attribute>

<attribute type="DATATYPE_BIT_2-VALUE_BIT_3">true</attribute>

<attribute type="DATATYPE_BIT_2-VALUE_INT">1</attribute>

</event>

Também pode usar campos de tabelas do tipo de dados **BIT** em condições de restrição da extração de dados. O valor pode ser **true** (respectivamente **1**) ou **false** (respectivamente **0**).

Exemplo 64.

<jdbc_tableconfiguration>

<configuration name="BitConfig">

<docspec>

<doctable name="DATATYPE_BIT">

<condition fieldname="VALUE_BIT_1" logicaloperator="eq">

<value>true</value>

</condition>

<pkfield name="ID"/>

</doctable>

</docspec>

<table name="DATATYPE_BIT">

<pkfield name="ID" fktablename="DATATYPE_BIT" pkfieldname="ID"/>

<fieldtoread name="VALUE_BIT_2"/>

</table>

<table name="DATATYPE_BIT_2" tablename="DATATYPE_BIT">

<pkfield name="VALUE_BIT_2" fktablename="DATATYPE_BIT" fkfieldname="VALUE_BIT_2"/>

<fieldtoread name="VALUE_BIT_1"/>

<fieldtoread name="VALUE_BIT_3"/>

<fieldtoread name="VALUE_INT"/>

</table>

</configuration>

</jdbc_tableconfiguration>

5.7. Apêndice

5.7.1. Tipos de dados suportados

Os sistemas de banco de dados suportados incluem vários tipos de dados que não estão contidos em todos os sistemas de banco de dados ou que possuem aí outros nomes ou semânticas diferentes.

Para evitar um comportamento indesejado, a lista abaixo indica tipos de dados que foram testados com êxito com PPM Process Extractor JDBC-2-PPM.

Sistema de banco de dados	Tipos de dados	Nota
Oracle	CHAR, VARCHAR2, NCHAR, NVARCHAR2, NUMBER, LONG, DATE, TIMESTAMP	Os tipos de dados CLOB, NCLOB, BINARY_FLOAT e BINARY_DOUBLE não são suportados.
IBM DB2	VARCHAR, CHAR, BIGINT, REAL, DOUBLE, FLOAT, DATE, TIME, TIMESTAMP	Os tipos de dados LONG VARCHAR e CLOB não são suportados.
Microsoft SQL Server	CHAR, VARCHAR, NCHAR, NVARCHAR, INT, TINYINT, SMALLINT, BIGINT, REAL, FLOAT, DECIMAL, DATETIME, SMALLDATETIME, BIT	Os tipos de dados LONG VARCHAR e CLOB não são suportados.

5.7.2. Caracteres especiais e capitalização

A partir da versão 9.9, o PPM Process Extractor JDBC-2-PPM suporta o uso de caracteres especiais e capitalização com diferenciação de maiúsculas e minúsculas em nomes de esquemas, tabelas e campos em bancos de dados SQL.

As expressões SQL devem estar entre parênteses para que os sistemas de banco de dados possam processá-las se incluírem caracteres especiais nos nomes de esquemas, tabelas e colunas e ortografia com diferenciação de maiúsculas e minúsculas, por exemplo, como na seguinte instrução SQL para um banco de dados de servidor SQL.

```
SELECT [column with space] FROM [USER.1].[table $%&]
```

A sintaxe pode ser diferente em função do sistema de banco de dados. Por exemplo, em Oracle e DB2, os nomes devem ser definidos em apóstrofos e, no servidor SQL, podem ser usados colchetes.

A configuração do sistema **JdbcConfig.dtd** contém o atributo **non-standard-sql-identifiers (true | false)** para suportar caracteres especiais e ortografia com diferenciação de maiúsculas e minúsculas. Se o atributo **non-standard-sql-identifiers** tiver o valor **true**, os nomes de coluna, tabela e esquema serão definidos em caracteres limitados, adequados para o tipo de banco de dados relevante.

```
<!ATTLIST databasesettings
name ID #REQUIRED
dbtype (ORACLE | DB2 | SQLSERVER | OTHER) #REQUIRED
precisionoftime (SECOND|MILLISECOND) "SECOND"
non-standard-sql-identifiers (true | false) "false"
```

Nos sistemas de banco de dados, os caracteres de limitação não devem fazer parte dos identificadores do banco de dados, como nomes de esquemas, tabelas e colunas.

Compatibilidade com versões anteriores

Para garantir a compatibilidade com versões já existentes, o valor **false** é associado ao atributo **non-standard-sql-identifiers** no PPM CTK sempre que uma configuração é aberta, salva ou migrada. Um valor existente não é substituído automaticamente, mas pode ser alterado manualmente em PPM CTK.

Quando cria uma nova configuração, o PPM CTK define por padrão este valor para **true**.

5.7.2.1. Caso especial: Esquema e nome da tabela

A extração de tabelas localizadas em um esquema do banco de dados diferente daquele do usuário logado apenas é possível no extrator JDBC por meio de um prefixo do nome da tabela. O prefixo é separado por um ponto do nome da tabela, no atributo **tablename** e nos vários elementos da configuração da tabela (jdbc_tableconfiguration.dtd).

Exemplo 65. Exemplo

<doctable name="customer" tablename="USER1.CUSTOMER ">

Como o ponto também pode ocorrer como um caractere especial em nomes de esquemas e tabelas, cada ponto que faz parte do nome do esquema ou da tabela deve ser mascarado com uma barra invertida ****. Para

reconhecer as barras invertidas que fazem parte do nome, elas devem ser mascaradas por uma ocorrência dupla.

Os nomes de esquemas e tabelas podem ser inequivocamente diferenciados nas seguintes configurações de exemplo.

Nome do esquema	Nome da tabela	Sintaxe para o atributo tablename
USER	1.CUSTOMER	USER.1\.CUSTOMER
USER.1.1	ORDER.DE	USER\.1\.1.ORDER\.DE
USER.	ORDER	USER\..ORDER
USER\2	ORDER.DE	USER\\2.ORDER\.DE
USER\.1	ORDER\.44	USER\\\.1.ORDER\\\.44
USER	1.TABLE.1	USER.1.TABLE.1

Se mais de um ponto ocorrer em um esquema e nome de tabela (consulte o exemplo na última linha da tabela) sem ser mascarado pela barra invertida, o primeiro ponto será interpretado como um separador de esquemas e os outros pontos serão vistos como partes do nome da tabela. Isso corresponde ao comportamento da versão anterior à 9.9 e é mantido por motivos de compatibilidade. Ainda é recomendado para mascarar todos os pontos, exceto o separador de esquema com uma barra invertida de prefixo.

Não é permitido mascarar qualquer outro caractere que não seja o ponto ou a barra invertida.

Ao definir chaves estrangeiras, é importante observar que o valor no atributo XML **<fktablename>** refere-se ao atributo XML **<name>** e não a **<tablename>**.

Exemplo 66. Exemplo

```
<jdbc_tableconfiguration>
<configuration name="Example">
<docspec>
<doctable name="USER.1.CUSTOMER" tablename="USER\.1.CUSTOMER">
<condition fieldname="SEQUENCE" logicaloperator="in">
<value>001</value>
<value>099</value>
</condition>
<pkfield name="SEQUENCE"/>
</doctable>
</docspec>
<table name="FK_TABLE">
<pkfield name="SEQUENCE"
fktablename="USER.1.CUSTOMER" fkfieldname="SEQUENCE"/>
<fieldtoread name="FK_COLUMN1"/>
<fieldtoread name="FK_COLUMN2"/>
</table>
</configuration>
</jdbc_tableconfiguration>
```

Se **tablename** estiver ausente, o conteúdo do atributo **name** é usado como nome da tabela para acessar o banco de dados. Os dois valores de atributo para os atributos **name** e **fktablename** são interpretados como cadeias de caracteres puras e, portanto, não são avaliadas em termos de separadores e caracteres de mascaramento.

Por exemplo, se a tabela for especificada apenas com o atributo **name**, da seguinte forma:

<doctable name="USER\.CUSTOMER">.

isso significa que

<doctable name="USER\.CUSTOMER" tablename="USER\.CUSTOMER">.

Assim, o extrator extrairia a tabela **USER.CUSTOMER** do banco de dados.

Se caracteres especiais ocorrerem no nome da tabela, recomendamos especificar sempre o nome da tabela com o atributo **tablename**.

6. PPM Process Extractor CSV-2-PPM

Este capítulo fornece uma visão geral da arquitetura, funcionamento e configuração do PPM Process Extractor CSV-2-PPM.

6.1. CSV

Um arquivo no formato CSV (valores separados por vírgulas) consiste nos registros de dados reais e em um cabeçalho opcional contendo os títulos das colunas. Os cabeçalhos de coluna no cabeçalho e nos campos de dados para os registros de dados são separados pelo mesmo separador, normalmente um ponto e vírgula. Um campo de dados só pode conter o separador se for usado dentro de um valor mascarado. O PPM Process Extractor CSV-2-PPM implementa os cabeçalhos das colunas como tipos de atributos, ou seja, os valores dos registros de dados como valores de atributo.

Exemplo 67. Exemplo

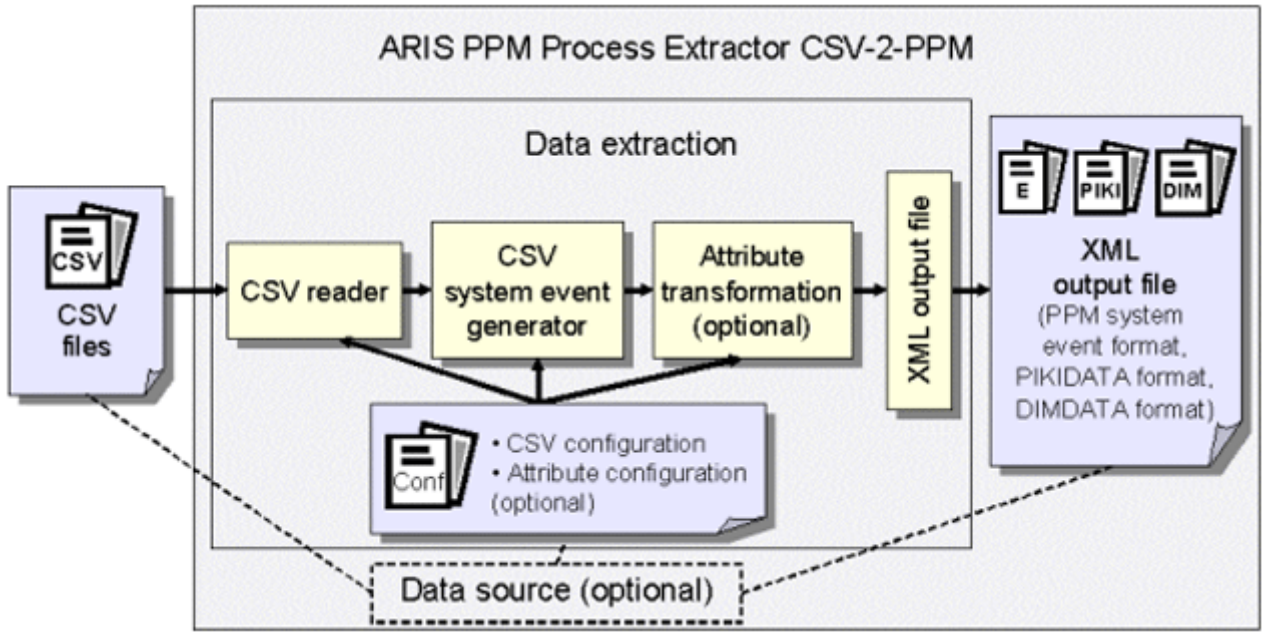
```
ORDER NUMBER,POSITION,RECORDED BY,MATERIAL,QUANTITY
4711,10,"Harry, ""A"" Williams",Mobile 6600,3
4811,23,"Ben, ""B"" Snyder",Mobile 6601,2
4911,15,"George, ""C"" Nyland",Mobile 6602,1
```

O arquivo CSV de exemplo contém um cabeçalho com os cabeçalhos de coluna para cinco colunas de dados e três registros de dados.

A vírgula é usada como separador e os apóstrofos duplos são usados como caractere de mascaramento. O caractere de mascaramento é usado dentro de um valor de campo de dados mascarado, duplicando o caractere de mascaramento usado. Os valores na coluna **RECORDED BY** podem incluir o separador, uma vez que estão mascarados.

6.2. Arquitetura

A figura abaixo ilustra a funcionalidade do PPM Process Extractor CSV-2-PPM:



Os arquivos CSV são lidos com o leitor de CSV, com base na configuração de CSV. O gerador de eventos do sistema CSV e, se aplicável, a transformação do atributo transfere as informações extraídas para o formato do evento de sistema PPM, PIKIDATA ou formato DIMDATA. A saída XML grava os dados no arquivo de saída XML correspondente.

6.3. Leitor de CSV e gerador de eventos do sistema CSV

O leitor de CSV é usado para ler arquivos CSV. Para importar arquivos CSV com um cabeçalho, deve incluir uma especificação adequada no arquivo de configuração CSV, para que a primeira linha do arquivo CSV seja interpretada como um cabeçalho (ver o capítulo **Configuração CSV [129]**).

A capitalização nos títulos das colunas e nos valores dos campos de dados é transferida inalterada dos arquivos CSV.

Os capítulos seguintes descrevem o comportamento do leitor de CSV e do gerador de eventos do sistema CSV usando exemplos simples. Nos exemplos, o arquivo CSV é comparado ao arquivo de saída XML correspondente, no formato de evento do sistema PPM. Um registro de dados CSV e cabeçalhos de coluna são transformados em um evento do sistema no arquivo de saída XML, no formato de evento do sistema PPM.

6.3.1. Exemplo 1: Cabeçalho completo

Entrada no arquivo CSV

```
ORDER NUMBER;POSITION;RECORDED BY;MATERIAL;QUANTITY
4711;10;"Harry, ""A"" Williams";Mobile 6600;3
```

Entrada no arquivo de saída XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="ORDER NUMBER">4711</attribute>
<attribute type="POSITION">10</attribute>
<attribute type="RECORDED BY">Harry, "A" Williams</attribute>
<attribute type="MATERIAL">Mobile 6600</attribute>
<attribute type="QUANTITY">3</attribute>
</event>
...
</eventlist>
```

Se especificar um caractere de mascaramento no arquivo de configuração CSV, todos os valores serão desmascarados ao gerar o arquivo de saída XML.

6.3.2. Exemplo 2: Cabeçalho com cabeçalho da coluna ausente

Se um cabeçalho de coluna estiver ausente dos arquivos CSV, ele será adicionado automaticamente no arquivo de saída XML. Por padrão, o nome de coluna associado é composto pelo prefixo **FIELD_** e pelo número de posição da coluna. A primeira coluna tem a posição 1.

Entrada no arquivo CSV

```
ORDER NUMBER;POSITION;;MATERIAL;QUANTITY
4711;10;"Harry, ""A"" Williams";Mobile 6600;3
```

Entrada no arquivo de saída XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="ORDER NUMBER">4711</attribute>
<attribute type="POSITION">10</attribute>
<attribute type="FIELD_3">Harry, "A" Williams
</attribute>
<attribute type="MATERIAL">Mobile 6600</attribute>
<attribute type="QUANTITY">3</attribute>
</event>
...
</eventlist>
```

O cabeçalho da coluna **FIELD_3** é gerado automaticamente no lugar do cabeçalho da coluna ausente nos arquivos CSV.

6.3.3. Exemplo 3: Cabeçalho com vários cabeçalhos da coluna ausentes

Se vários cabeçalhos de coluna estiverem ausente dos arquivos CSV, eles serão adicionados automaticamente no arquivo de saída XML. Se os cabeçalhos de coluna ausentes estiverem localizados no final do cabeçalho, os separadores não precisam estar presentes entre os cabeçalhos de coluna ausentes.

Entrada no arquivo CSV

```
ORDER NUMBER;POSITION;;;
4711;10;"Harry, ""A"" Williams";Mobile 6600;3
```

ou

```
ORDER NUMBER;POSITION
4711;10;"Harry, ""A"" Williams";Mobile 6600;3
```

Entrada no arquivo de saída XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="ORDER NUMBER">4711</attribute>
<attribute type="POSITION">10</attribute>
<attribute type="FIELD_3">Harry, "A" Williams</attribute>
<attribute type="FIELD_4">Mobile 6600</attribute>
<attribute type="FIELD_5">3</attribute>
</event>
...
</eventlist>
```

Os cabeçalhos das colunas **FIELD_3**, **FIELD_4** e **FIELD_5** são gerados automaticamente no lugar dos cabeçalhos de coluna ausentes dos arquivos CSV. A mesma entrada é gerada no arquivo de saída XML, independentemente de o separador aparecer entre os cabeçalhos de coluna ausentes.

6.3.4. Exemplo 4: Sem cabeçalho

Se os arquivos CSV não incluírem um cabeçalho, um cabeçalho de coluna será gerado automaticamente para cada coluna de dados.

Entrada no arquivo CSV

```
4711;10;"Harry, ""A"" Williams";Mobile 6600;3
```

Entrada no arquivo de saída XML

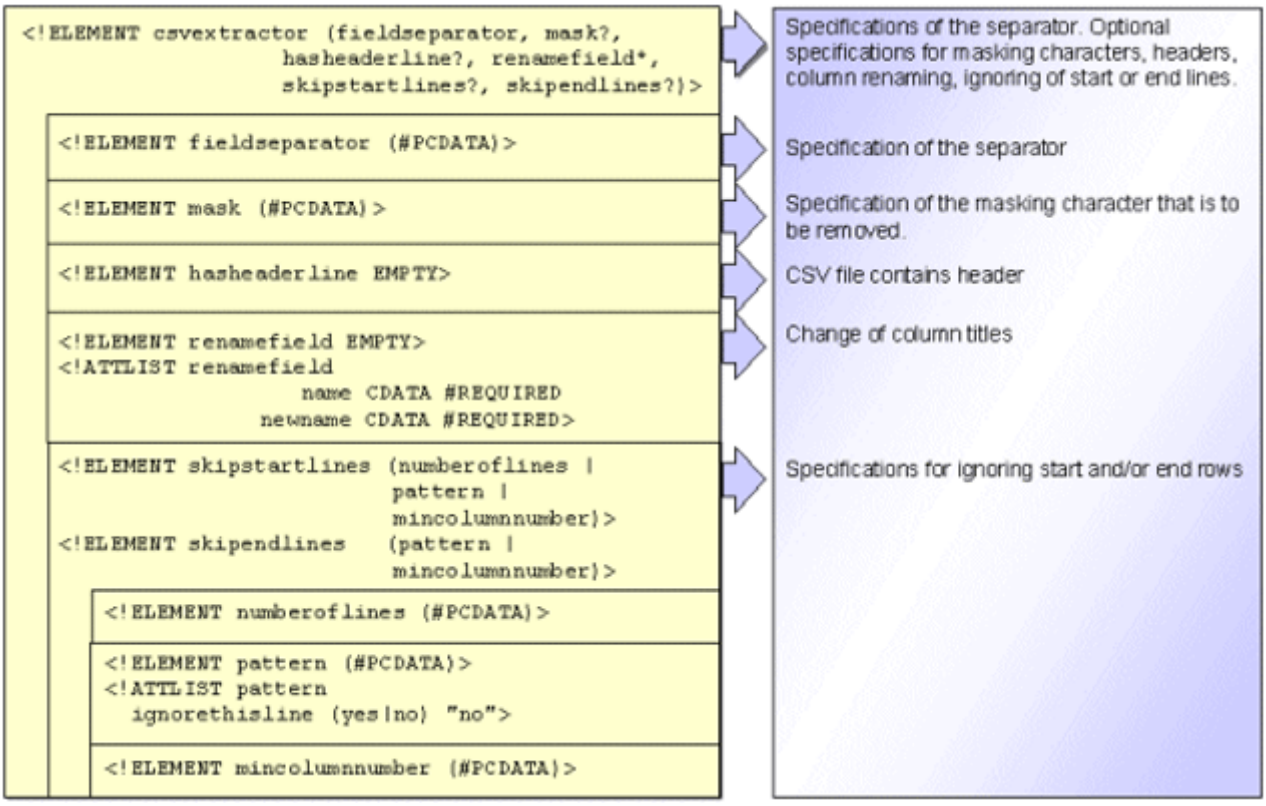
```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="FIELD_1">4711</attribute>
<attribute type="FIELD_2">10</attribute>
<attribute type="FIELD_3">Harry, "A" Williams</attribute>
<attribute type="FIELD_4">Mobile 6600</attribute>
<attribute type="FIELD_5">3</attribute>
</event>
...
</eventlist>
```

6.4. Configuração CSV

A maneira como os dados CSV são processados quando são importados é especificada em um arquivo de configuração XML. O nome desse arquivo é transferido para o programa de linha de comando **runcsv2ppm** como um argumento.

No arquivo de configuração, pode especificar o separador ou renomear cabeçalhos de coluna.

O formato do arquivo XML é especificado pela seguinte DTD:



Elemento ou atributo XML	Descrição	Exemplo
csvextractor	Configuração do processo de importação CSV	Ver abaixo
fieldseparator	Separador	,
Máscara	Caractere de mascaramento (os valores importados não estão mascarados)	"
hasheaderline	Os arquivos CSV incluem o cabeçalho	
renamefield	Renomear uma coluna de dados	Ver abaixo
name	Nome de uma coluna de dados gerada a partir de arquivos CSV	MATERIAL ou FIELD_3
newname	Novo nome da coluna de dados	Designação do elemento
skipstartlines	Linhas iniciais a serem ignoradas	
skipendlines	Linhas finais a serem ignoradas	
numberoflines	Número de linhas iniciais a serem ignoradas. A contagem começa na primeira linha.	10
pattern	Padrão de caracteres para localizar uma linha de dados a partir da qual as linhas devem ser extraídas (**skipstartlines**: por padrão, a linha encontrada também é extraída) ou até à qual as linhas devem ser extraídas (**skipendlines**: a linha encontrada, incluindo todas as linhas seguintes, não é extraída São permitidos os seguintes espaços reservados: ***** No ou qualquer número de caracteres **?** Apenas um caractere **** Caractere de mascaramento para procurar espaços reservados ou mascarar caracteres na forma: **** ou ***** ou **\?** É selecionada a primeira linha do topo que corresponde ao padrão especificado.	Por exemplo, a expressão ***10?*269** pesquisa valores como 555108*269 ou 104*269, mas não 104\\555269.
ignorethisline (apenas com **pattern** em **skipstartlines**)	Na extração, ignore (**yes**) ou não ignore (**no**) a linha de dados que contém o padrão pretendido. Valor padrão: **não**	yes
mincolumn number	Especifica um número mínimo de colunas para definir o início e/ou o final do intervalo de dados a ser extraído. **skipstartlines**: As linhas são extraídas de (inclusivo) a primeira linha encontrada com pelo menos o número especificado de colunas. **skipendlines**: As linhas são extraídas de (exclusivo) a primeira linha encontrada com menos do que o número especificado de colunas.	4

Configuração de exemplo (csvconfig.xml)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE csvextractor SYSTEM 'csvextractor.dtd'>
<csvextractor>
<fieldseparator>;</fieldseparator>
<mask>"</mask>
<hasheaderline/>
<renamefield name="FIELD_3" newname=
"Order item recorded by"/>
<renamefield name="MATERIAL" newname="Item designation"/>
```

```
<skipstartlines>
<pattern ignorethisline="yes">###*</pattern>
</skipstartlines>
<skipendlines>
<pattern>5???;*;*;*;*</pattern>
</skipendlines>
</csvextractor>
```

Arquivo CSV a ser importado (example.csv)

O arquivo consiste em uma linha de comentário, um cabeçalho e cinco registros de dados:

```
### Order data 03/25/2006 ###
ORDER NUMBER;POSITION;;MATERIAL;QUANTITY
4711;10;"Harry, ""A"" Williams";Mobile 6600;3
4811;23;"Ben, ""B"" Snyder";Mobile 6601;2
4911;15;"George, ""C"" Nyland";Mobile 6602;1
5011;6;"George, ""C"" Nyland";Mobile 5405;2
5211;23;"George, ""C"" Nyland";Mobile 5410;1
```

Arquivo de saída no formato do evento de sistema do PPM (event.xml)

O gerador de eventos do sistema CSV usa a configuração CSV de exemplo para gerar o seguinte arquivo de saída XML:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="ORDER NUMBER">4711</attribute>
<attribute type="POSITION">10</attribute>
<attribute type="Order item recorded by"
>Harry, "A" Williams</attribute>
<attribute type="Item designation">Mobile 6600
</attribute>
<attribute type="QUANTITY">3</attribute>
</event>
<event>
<attribute type="ORDER NUMBER">4811</attribute>
<attribute type="POSITION">23</attribute>
<attribute type="Order item recorded by"
>Ben, "B" Snyder</attribute>
<attribute type="Item designation">Mobile 6601
</attribute>
<attribute type="QUANTITY">2</attribute>
</event>
<event>
```

```
<attribute type="ORDER NUMBER">4911</attribute>
<attribute type="POSITION">15</attribute>
<attribute type="Order item recorded by"
>George, "C" Nyland</attribute>
<attribute type="Item designation">Mobile 6602
</attribute>
<attribute type="QUANTITY">1</attribute>
</event>
</eventlist>
```

De acordo com as especificações do arquivo de configuração CSV, a seguinte chamada de linha de comando gera um evento de sistema no arquivo de saída XML para cada registro de dados nos dados CSV:

```
runcsv2ppm -i example.csv -csvconfig csvconfig.xml -outfile event -nozip
-encoding ISO-8859-1
```

Durante a importação e para todos os eventos de sistema, o nome **FIELD_3** associado à terceira coluna de dados é automaticamente substituído por **Order item recorded by** e a coluna de dados **MATERIAL** é renomeada para **Designação do elemento**. Os valores mascarados do tipo de atributo **Order item recorded by** são desmascarados. A linha de comentário é ignorada na extração conforme especificado para **skipstartlines**. As últimas duas linhas da área de dados são ignoradas conforme especificado para **skipendlines** (números de pedido **5011**, **5211**).

6.4.1. Extensão de configuração CSV

Pode especificar o(s) arquivo(s) de entrada CSV e a codificação do(s) arquivo(s) de entrada na configuração do sistema de origem CSV. O elemento XML **csvextractor** pode conter os seguintes elementos XML:

Elemento/atributo XML	Descrição
inputfile	Contém o caminho do(s) arquivo(s) de entrada
encoding	Contém a codificação do(s) arquivo(s) de entrada

```
<csvextractor>
<inputfile>../testclient/data/customers.csv</inputfile>
<encoding>ISO-8859-15</encoding>
<fieldseparator>,</fieldseparator>
<mask>"</mask>
<hasheaderline/>
</csvextractor>
```

O valor para o elemento XML **inputfile** pode ser o caminho para o arquivo CSV. Se tiver especificado o caminho, o parâmetro de linha de comando [134] **-i** será ignorado. O mesmo se aplica ao elemento XML **encoding** e ao parâmetro de linha de comando [134] **-encoding**.

Se tiver especificado o caminho para o arquivo CSV (inputfile) na configuração CSV, os dados do caminho na fonte de dados serão ignorados.

6.5. Programa de linha de comando

O arquivo de saída XML é criado com o programa de linha de comando **runcsv2ppm.bat**.

Chamar o programa sem parâmetros ou com **-h** ou **-?** gera a ajuda on-line no console. Isso descreve todas as opções disponíveis.

6.5.1. Argumentos do programa de linha de comando

6.5.2. Argumentos gerais

-version

Exibe o número de versão do extrator de processo PPM usado. Outros argumentos especificados são ignorados.

-information yes|no

Especifica se a informação é para ser exibida (yes) ou não (no) durante a importação. O padrão é **yes**.

-warning yes|no

Especifica se as mensagens de aviso são para ser exibidas (yes) ou não (no) durante a importação. O padrão é **yes**.

-error yes|no

Aqui pode especificar se as mensagens de erro são para ser exibidas (yes) ou não (no) durante a importação. O padrão é **yes**.

-protocolfile <file name>

Especifique o arquivo de log para o qual todas as mensagens são gravadas durante a importação.
Se especificar um arquivo, somente as mensagens de erro críticas que resultam em cancelamento do programa serão exibidas na tela.

-language <ISO code>

Especifique o idioma no qual as informações de log devem ser exibidas.

6.5.3. Argumentos específicos do CSV

-datasource <file name> (opcional)

Especifique a fonte de dados que deseja usar para a extração. O arquivo XML contém especificações de todos os arquivos a serem usados para extração e saída XML. Certifique-se de que usa este parâmetro em vez de e não em conjunto com os parâmetros **-datasourcelist**, **-csvconfig**, **-i**, **-calcconfig**, **-outfile** ou **-nozip**.

-datasourcelist <file name>

Várias fontes de dados podem ser extraídas simultaneamente através do argumento **-datasourcelist**. Isso corresponde a várias extrações consecutivas usando o argumento **-datasource**. São extraídas apenas as fontes de dados CSV.

Ver o capítulo Extrair várias fontes de dados [136].

-i <csvfilename> [<csvfilename> ...]

É aqui que especifica o(s) arquivo(s) CSV a ser(em) usado(s) para gerar um arquivo de saída XML.

Se o caminho para o arquivo de entrada CSV for especificado no arquivo de configuração CSV, o parâmetro da linha de comando **-i** será ignorado.

-csvconfig <file name>

É aqui que especifica o nome do arquivo de configuração CSV que contém as informações para geração do arquivo de saída XML dos arquivos CSV.

-encoding "<encodingname>"

É aqui que especifica a codificação do(s) arquivo(s) CSV.

Se a codificação for especificada no arquivo de configuração CSV, o parâmetro da linha de comando **-encoding** será ignorado.

Se a codificação não for especificada no arquivo de configuração CSV, o parâmetro da linha de comando será obrigatório.

-calcconfig <file name> (opcional)

É aqui que especifica o nome do arquivo de configuração XML com o qual pode alterar atributos do arquivo de saída XML ou adicionar atributos, incluindo transformações de atributos.

-removeempty (opcional)

Use este parâmetro para remover os atributos extraídos sem valores antes da transformação de atributos.

6.5.4. Argumentos específicos do arquivo de saída

-outfile <file name>

Esse parâmetro é usado para especificar o nome do arquivo de saída XML. A extensão de arquivo é adicionada automaticamente. Por padrão, o arquivo é fornecido como um arquivo ZIP.

-outfileencoding <encoding> (opcional)

Esse parâmetro é usado para especificar a codificação do arquivo de saída XML. Valor padrão: UTF-8

-nozip (opcional)

Use esse parâmetro para especificar que o arquivo de saída deve ser gerado como um arquivo XML em vez de um arquivo ZIP.

-pikidatamapping <file name> <pc name> (opcional)

Use esse parâmetro para especificar o nome do arquivo XML que contém o mapeamento no primeiro argumento. No segundo argumento, especifique o nome da série de medidas na qual as séries de medidas independentes da instância de processo devem ser salvas.

-dimdatamapping <file name> <dim name> (opcional)

Use esse parâmetro para especificar o nome do arquivo XML que contém o mapeamento no primeiro argumento. No segundo argumento, especifique o nome da dimensão para a qual os valores extraídos serão subsequentemente importados.

-sorteventattributes (opcional)

Use esse parâmetro para classificar os atributos do sistema de origem nos eventos do sistema em ordem alfanumérica, usando o tipo de atributo. Esse parâmetro pode ter um impacto negativo na velocidade de extração para grandes volumes de dados.

6.6. Extrair de várias fontes de dados

Pode usar uma lista de fontes de dados para extrair simultaneamente de várias fontes de dados. A lista de fontes de dados é especificada em um arquivo de configuração separado.

Cada nova fonte de dados CSV criada no PPM CTK é automaticamente adicionada ao final da lista de fontes de dados do cliente atual. Durante a extração usando a opção de linha de comando **-datasourcelist <file name>** os dados dessas fontes de dados são extraídos consecutivamente como se a extração de dados fosse chamada várias vezes usando a opção **-datasource <file name>**. A sequência da extração da fonte de dados é especificada no arquivo de configuração. As fontes de dados de um tipo diferente de CSV são ignoradas durante a extração.

Aplica-se o seguinte à resolução de problemas:

- Se a extração for chamada por meio de uma lista de fontes de dados válida que não contenha nenhuma fonte de dados correspondente, a extração será terminada sem exibir uma mensagem de erro.
- Se a extração for chamada usando uma lista de fontes de dados contendo várias fontes de dados correspondentes e se ocorrer um erro durante a extração de uma dessas fontes de dados, resultando no cancelamento da extração, o procedimento continuará com a fonte de dados seguinte. Isso significa que o cancelamento da extração de uma fonte de dados não leva ao cancelamento de todo o procedimento.
- Se ocorrer um erro em pelo menos uma fonte de dados, o que levaria a um estado de erro de saída durante a extração única dessa fonte de dados, o estado do último erro de saída será retornado pelo processo geral de extração da fonte de dados.

7. Funcionalidades de todos os extratores

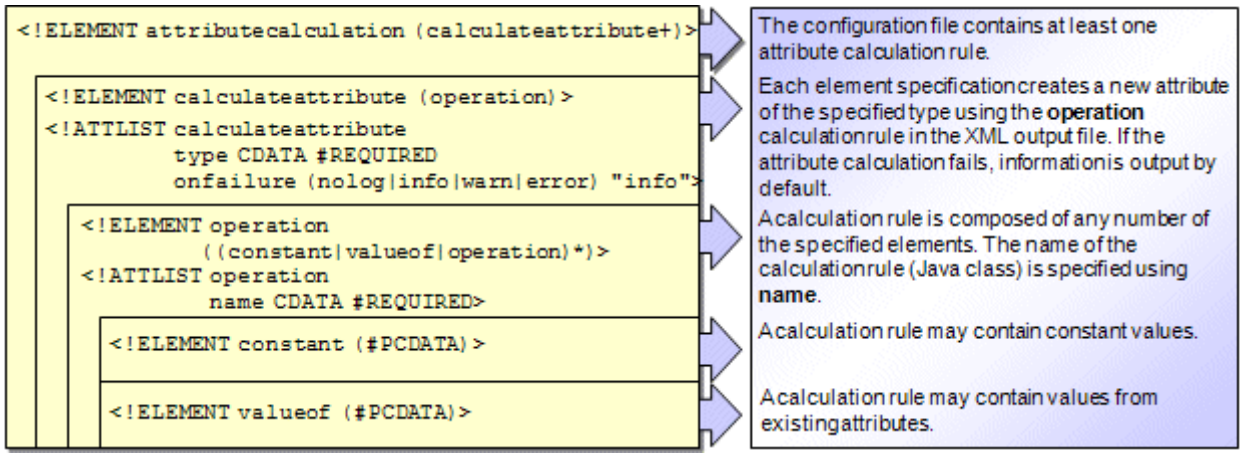
Este capítulo descreve as funcionalidades do programa que todos os extratores de processo têm em comum.

7.1. Transformação de atributos

Para alterar os valores de determinados tipos de atributos nos arquivos de saída ou para criar tipos de atributos adicionais, é necessário especificar isso em um arquivo de configuração XML. O nome desse arquivo de configuração XML é transferido para o programa de linha de comando como um argumento.

Este arquivo de configuração contém as definições para processar tipos de atributos existentes e gerar e transformar novos tipos de atributos ao criar o(s) arquivo(s) de saída XML.

O formato do arquivo XML é especificado pela seguinte DTD:



Elemento XML ou atributo XML	Descrição
attributecalculation	Configuração da transformação de atributos
calculateattribute	Geração de um tipo de atributo de evento do sistema para execução bem-sucedida da regra de transformação (**operation**)
type	Nome do tipo de atributo do evento de sistema gerado
onfailure	Comportamento de cancelamento se a transformação falhar. Valores válidos: **nolog** (sem saída de log) **info** (saída como informações) **warn** (saída como aviso) **error** (saída como mensagem de erro) Valor padrão: **info**
operation	Regra de transformação que é implementada por uma classe Java. Cada operação tem um único valor de retorno na forma de uma cadeia de caracteres. Se a operação falhar, nada é retornado.
name	Nome da operação
constant	Valor constante a ser usado na transformação

Elemento XML ou atributo XML	Descrição
valueof	Valor de um tipo de atributo do sistema de origem existente ou já transformado

Os elementos **calculateattribute** são processados na ordem em que são especificados no arquivo de configuração. O aninhamento de elementos XML **operation** é processado para fora ou de cima para baixo.

Por razões de eficiência, apenas os argumentos que são essenciais para a transformação são avaliados durante uma transformação de atributo. Por exemplo, na operação **if_then_else**, após ser verificada a validade do primeiro argumento, apenas o segundo argumento (then) ou o terceiro argumento (else) é processado. A transformação do atributo é cancelada com uma mensagem de erro assim que um elemento obrigatório não puder retornar um valor.

Se o elemento XML **onfailure** no arquivo de transformação do atributo tiver o valor **error**, uma geração de atributo mal sucedida resulta no evento do sistema correspondente sendo associado ao atributo **EVENT_HAS_ATTRIBUTE_ERRORS=true**, no arquivo de saída XML. Além disso, uma mensagem de erro é gerada no arquivo de log.

Se o elemento XML **onfailure** tiver o valor **warn**, uma geração de atributo mal sucedida resulta no evento do sistema correspondente sendo associado ao atributo **EVENT_HAS_ATTRIBUTE_WARNINGS=true**, no arquivo de saída XML. Além disso, um aviso é gerado no arquivo de log.

Atenção
calculateattribute permite que os valores dos tipos de atributos existentes sejam substituídos após a execução bem sucedida das regras de transformação correspondentes (**operation**). Nenhum aviso é emitido. Se necessário, certifique-se de que os valores dos tipos de atributos existentes não são substituídos acidentalmente, executando uma verificação de existência condicional com uma regra de transformação adequada. Pode usar a operação **exists** (ver o capítulo Operações [139]) para esse propósito.

Se toda a transformação de um atributo for cancelada, o valor do atributo existente permanecerá inalterado. Dependendo do que especificou para a opção **onfailure**, uma mensagem correspondente é emitida com uma referência ao evento de sistema gerado. Nenhum log é emitido para **nolog**.

Exemplo 68.

A extração de dados do sistema de origem do sistema R/3 e o processamento subsequente pelo Gerador de XML criaram a seguinte entrada no arquivo de saída XML **event.xml**:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAP-VBELN">3866</attribute>
<attribute type="VBAP-POSNR">20</attribute>
<attribute type="VBAP-ERNAM">SCHMIDT</attribute>
<attribute type="VBAP-MATNR-MAKTX">100038</attribute>
<attribute type="VBAP-KWMENG">3</attribute>
</event>
</eventlist>
```

Exemplo 69.

Usando as especificações seguintes no arquivo de transformação de atributos **attributetransform.xml**, poderá alterar a visualização dos valores do tipo de atributo **VBAP-ERNAM** para que a concatenação do valor original com uma cadeia de caracteres constante seja usada como o novo valor.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE attributecalculation SYSTEM
"attributetransformation.dtd">
<attributecalculation>
<calculateattribute type="VBAP-ERNAM">
<operation name="concat">
<constant>Walter, "A" </constant>
<valueof>VBAP-ERNAM</valueof>
</operation>
</calculateattribute>
...
</attributecalculation>
```

Exemplo 70.

A chamada de linha de comando seguinte cria o arquivo de saída XML desejado a partir do arquivo de saída XML original usando o arquivo de transformação de atributo mencionado acima.

```
runjdbc2ppm -systemconfig SysConfig.xml ides_doc_vdap -tableconfig
TableConf.xml SD_C -begindate 01.01.2005 -enddate 31.01.2005 -calcconfig
attributetransform.xml -outfile event -outfileencoding ISO-8859-1 -nozip
```

Exemplo 71.

A mesma entrada como a exemplificada acima, então é semelhante a isso:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAP-VBELN">3866</attribute>
<attribute type="VBAP-POSNR">20</attribute>
<attribute type="VBAP-ERNAM">Walter, "A" SCHMIDT
</attribute>
<attribute type="VBAP-MATNR-MAKTX">100038</attribute>
<attribute type="VBAP-KWMENG">3</attribute>
</event>
</eventlist>
```

7.1.1. Operações

Os extratores de processo PPM fornecem as operações descritas nos capítulos seguintes, para transformação ou modificação de tipos de atributos.

Por uma questão de simplicidade, os exemplos de sintaxe mostrados para o elemento XML **operation** usam apenas constantes. No entanto, os elementos **valueof** e **operation** também podem ser especificados em uma operação.

Se um valor não puder ser determinado com uma regra de transformação, a transformação do tipo de atributo apenas será cancelada se o valor não determinado for essencial para a transformação subsequente. Dependendo do que foi especificado para a opção **onfailure**, uma mensagem correspondente pode ser emitida neste caso.

O valor de retorno das operações é geralmente uma cadeia de caracteres alfanumérica.

7.1.1.1. unmask

Description:	**Remove o mascaramento de um valor**
Sintaxe:	Dois parâmetros: 1. Caractere de mascaramento 2. Valor a ser desmascarado

Exemplo:	`<operation name="unmask">`
	`<constant>\</constant>`
	`<constant>\C:\\ppm3\\ppm320\\build3714\\dtd\`
	`</constant>`
	`</operation>`

Valor de retorno:	**C:\ppm3\ppm320\build3714\dtd** Se nenhum valor puder ser determinado para um dos parâmetros, nada é retornado.

O valor a ser desmascarado deve começar e terminar com o caractere de mascaramento especificado.

7.1.1.2. trim

Description:	**Remove os espaços no início e no final de um valor**
Sintaxe:	Um parâmetro: Valor no qual os espaços no início e no final devem ser removidos

Exemplo:	`<operation name="trim">`
	`<constant> PPM </constant>`
	`</operation>`

Valor de retorno:	<...>PPM<...> Se nenhum valor puder ser determinado para um dos parâmetros, nada é retornado.
Nota:	O número de espaços no início e no final do valor do campo não é importante.

7.1.1.3. concat

Description:	Concatenação de qualquer número de valores
Sintaxe:	De um a n parâmetros

Exemplo:	`<operation name="concat">`
	`<constant>Order processing</constant>`
	`<constant>/</constant>`
	`<constant>Cash sales</constant>`
	`</operation>`

Valor de retorno:	**Processamento do pedido/Vendas a dinheiro** Se nenhum valor puder ser determinado para um dos parâmetros, nada é retornado.

7.1.1.4. substring

Description:	Identifica uma subcadeia de um valor
Sintaxe:	Dois a três parâmetros: 1. Valor (cadeia de caracteres) do qual será determinada uma subcadeia 2. Posição inicial: 1 = Primeiro caractere a partir da esquerda -1 = Primeiro caractere a partir da direita 3. Número de caracteres a ser extraído (opcional). Se isso não for especificado, é extraída uma subcadeia executada a partir da posição inicial especificada até ao final do valor.

Exemplo:	`<operation name="substring">`
	`<constant>Order processing/Cash sales`
	`</constant>`
	`<constant>-1</constant>`
	`<constant>10</constant>`
	`</operation>`

Valor de retorno:	**Venda a dinheiro** Se nenhum valor puder ser determinado para um dos parâmetros ou se nenhum número inteiro for especificado para a posição de início e/ou número de caracteres, nada será retornado.

7.1.1.5. if_then_else

Description:	Transformação de atributos condicional
Sintaxe:	Dois a três parâmetros: 1. Valores para verificar a igualdade 2. Valor retornado em caso de igualdade 3. Valor retornado em caso de desigualdade

Exemplo:	Valores para verificar a igualdade usando a cadeia de caracteres **true**:
	`<operation name="if_then_else">`
	`<constant>not true</constant>`
	`<constant>Condition is met</constant>`
	`<constant>Condition is not met`
	`</constant>`
	`</operation>`

Valor de retorno:	**Condition is not met** Se o primeiro valor for igual à cadeia de caracteres **true** (sem diferenciação de maiúsculas e minúsculas), é retornado o segundo valor, caso contrário, (se disponível) é retornado o terceiro valor. Se nenhum valor puder ser determinado para o primeiro parâmetro (condição) ou o terceiro parâmetro, se a condição não for atendida, nada será retornado.

7.1.1.6. set_with_default

Description:	Define um valor para um valor padrão
Sintaxe:	Dois parâmetros: 1. Valor a ser retornado, se existir 2. Valor a ser retornado se nenhum valor puder ser calculado para o primeiro parâmetro.

Exemplo:	`<operation name="set_with_default">`
	`<constant>42</constant>`
	`<constant>1</constant>`
	`</operation>`

Valor de retorno:	**42** Se nenhum valor puder ser determinado para ambos os parâmetros, nada é retornado.

7.1.1.7. integer_plus

Description:	**Adição de parâmetros inteiros**
Sintaxe:	De dois a n parâmetros: Valores a serem adicionados

Exemplo:	`<operation name="integer_plus">`
	`<constant>1</constant>`
	`<constant>4</constant>`
	`<constant>37</constant>`
	`</operation>`

Valor de retorno:	**42** Se nenhum valor puder ser determinado para um dos parâmetros ou se um dos parâmetros tiver um valor não inteiro, nada será retornado.

7.1.1.8. integer_minus

Description:	**Subtração de parâmetros inteiros de outro parâmetro inteiro**
Sintaxe:	De dois a n parâmetros: 1. Valor do qual os outros valores têm de ser subtraídos 2. Valor a ser subtraído ... " n. Valor a ser subtraído

Exemplo:	`<operation name="integer_minus">`
	`<constant>1</constant>`
	`<constant>4</constant>`
	`<constant>37</constant>`
	`</operation>`

Valor de retorno:	**-40** Se nenhum valor puder ser determinado para um dos parâmetros ou se um dos parâmetros tiver um valor não inteiro, nada será retornado.

7.1.1.9. string_equal

Description:	**Compara cadeias de caracteres: Retorna a cadeia de caracteres true se todos os valores forem iguais, caso contrário, a cadeia de caracteres será false.**
Sintaxe:	De dois a n parâmetros: Valores a serem comparados

Description:	**Compara cadeias de caracteres: Retorna a cadeia de caracteres true se todos os valores forem iguais, caso contrário, a cadeia de caracteres será false.**
Exemplo:	<operation name="string_equal"> <constant>42</constant> <constant>42</constant> <constant>37</constant> </operation>
Valor de retorno:	**false** Assim que um valor for encontrado, e que não corresponda ao primeiro valor verificado, **false** é retornado, mesmo que se sigam mais valores. Se nenhum valor puder ser determinado para um dos parâmetros verificados, nada é retornado.

7.1.1.10. string_in

Description:	**Compara uma cadeia de caracteres com uma lista de cadeias de caracteres. Retorna a cadeia de caracteres true se a cadeia de caracteres for idêntica a pelo menos uma das outras cadeias de caracteres, caso contrário, a cadeia de caracteres retornada é false.**
Sintaxe:	De três a n parâmetros: Valores a serem comparados
Exemplo:	<operation name="string_in"> <constant>42</constant> <constant>42</constant> <constant>37</constant> <constant>42</constant> </operation>
Valor de retorno:	**true** Assim que uma cadeia de caracteres for encontrada, que seja idêntica à cadeia de caracteres a ser comparada, **true** é retornado, mesmo que se sigam mais valores. Se nenhum valor puder ser determinado para o primeiro parâmetro, nada é retornado.

7.1.1.11. boolean_not

Description:	**Retorna a cadeia de caracteres false se o parâmetro for igual a true (sem diferenciação de maiúsculas e minúsculas), caso contrário, retorna a cadeia de caracteres true.**
Sintaxe:	Um parâmetro: Valor booleano a ser negado

Description:	**Retorna a cadeia de caracteres false se o parâmetro for igual a true (sem diferenciação de maiúsculas e minúsculas), caso contrário, retorna a cadeia de caracteres true.**

Exemplo:	`<operation name="boolean_not">`
	`..<constant>this is not true</constant>`
	`</operation>`

Valor de retorno:	**true** Se nenhum valor puder ser determinado para o parâmetro, nada é retornado.

7.1.1.12. boolean_and

Description:	**Retorna a cadeia de caracteres true se um dos parâmetros for igual a true (sem diferenciação de maiúsculas e minúsculas), caso contrário, a cadeia de caracteres é false.**
Sintaxe:	De dois a n parâmetros: Valores booleanos a serem vinculados com o Booleano AND

Exemplo:	`<operation name="boolean_and">`
	`..<constant>truE</constant>`
	`..<constant>True</constant>`
	`..<constant>TRUE</constant>`
	`</operation>`

Valor de retorno:	**true** Assim que for encontrado um valor que não seja **true**, **false** é retornado. Qualquer valor subsequente não será verificado. Se nenhum valor puder ser determinado para um dos parâmetros verificados, nada é retornado.

7.1.1.13. boolean_or

Description:	**Retorna a cadeia de caracteres true se um dos parâmetros for igual a true (sem diferenciação de maiúsculas e minúsculas), caso contrário, retorna a cadeia de caracteres false.**
Sintaxe:	De dois a n parâmetros: Valores booleanos a serem vinculados com o Booleano OR

Exemplo:	`<operation name="boolean_or">`
	`<constant>false</constant>`
	`<constant>True</constant>`
	`<constant>this is not true</constant>`
	`</operation>`

Description:	**Retorna a cadeia de caracteres true se um dos parâmetros for igual a true (sem diferenciação de maiúsculas e minúsculas), caso contrário, retorna a cadeia de caracteres false.**
Valor de retorno:	**true** Assim que for encontrado um valor que seja **true**, **true** é retornado. Qualquer valor subsequente não será verificado. Se nenhum valor puder ser determinado para um dos parâmetros verificados, nada é retornado.

7.1.1.14. exists

Description:	**Verificação de existência**
Sintaxe:	De um a n parâmetros: Valores cuja existência deve ser verificada
Exemplo:	<operation name="exists"> <constant>false</constant> <constant>Order volume</constant> <constant>Turnover</constant> </operation>
Valor de retorno:	**true** Retorna **true** se um valor puder ser recuperado para todos os parâmetros, caso contrário, **false**.

7.1.1.15. get_null

Description:	**Não retorna.** **Se a operação externa em uma regra de transformação não retornar nada, o tipo de atributo não será transferido para o arquivo de saída XML.**
Sintaxe:	Sem parâmetros
Exemplo:	<operation name="get_null"/>
Valor de retorno:	Nenhum

7.1.1.16. getCurrentTimestamp

Description:	**Retorna um carimbo de data/hora atual. O formato é especificado no primeiro parâmetro. Inclui todos os formatos data e hora especificados pela classe java.text.SimpleDateFormat (como o operador de condição creationtimestamp), tal como é descrito na referência técnica Importação de dados, na seção Transformações.** **Se especificar apenas um parâmetro, o mesmo carimbo de data/hora - correspondente à primeira vez em que a operação é chamada - será gerado como o valor do atributo em todos os eventos de sistema no(s) arquivo(s) de saída. Se especificar dois parâmetros, a hora da chamada da operação será gravado no evento de sistema como um atributo de carimbo de data/hora. Isso significa que os valores podem diferir dependendo do formato selecionado e da duração da extração. Não há verificação de plausibilidade do segundo parâmetro, apenas precisa existir. Se nenhum valor puder ser determinado para o primeiro parâmetro, nada é retornado.**
Sintaxe:	De um a dois parâmetros

Description:	**Retorna um carimbo de data/hora atual. O formato é especificado no primeiro parâmetro. Inclui todos os formatos data e hora especificados pela classe java.text.SimpleDateFormat (como o operador de condição creationtimestamp), tal como é descrito na referência técnica Importação de dados, na seção Transformações.** **Se especificar apenas um parâmetro, o mesmo carimbo de data/hora - correspondente à primeira vez em que a operação é chamada - será gerado como o valor do atributo em todos os eventos de sistema no(s) arquivo(s) de saída. Se especificar dois parâmetros, a hora da chamada da operação será gravado no evento de sistema como um atributo de carimbo de data/hora. Isso significa que os valores podem diferir dependendo do formato selecionado e da duração da extração. Não há verificação de plausibilidade do segundo parâmetro, apenas precisa existir. Se nenhum valor puder ser determinado para o primeiro parâmetro, nada é retornado.**

Exemplo:	`<operation name="getCurrentTimestamp">`
	`<constant>dd.MM.yyyy HH:mm:ss</constant>`
	`<constant/>`
	`</operation>`

Valor de retorno:	por exemplo, **20.07.2006 15:45:08**, caso contrário, vários valores para diferentes eventos do sistema, dependendo da duração da extração.

7.1.2. Exemplo de transformação de atributos

A importação de dados do sistema de origem gerou a seguinte entrada no arquivo de saída XML:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAP-VBELN">3866</attribute>
<attribute type="VBAP-POSNR">20</attribute>
<attribute type="VBAP-ERNAM">SCHMIDT</attribute>
<attribute type="VBAP-MATNR-MAKTX">100038</attribute>
<attribute type="VBAP-KWMENG">3</attribute>
</event>
</eventlist>
```

Quer vincular o valor do tipo de atributo **VBAP-ERNAM**com uma cadeia de caracteres de constante e gravá-la no novo tipo de atributo **Order item recorded by**. Além disso, quer criar o tipo de atributo **Note on order recording**, que extrai o primeiro nome e sobrenome do tipo de atributo **VBAP-ERNAM** e cria a cadeia de caracteres seguinte:

<First name>< ><Last name>< has recorded the item ><VBAP-POSNR>< of the order ><VBAP-VBELN>< .>

As informações em parênteses angulares representam valores de atributo que são extraídos ou transformados a partir de tipos de atributo ou constantes existentes. Pode especificar as informações necessárias no arquivo de transformação de atributos XML:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE attributecalculation SYSTEM
"attributetransformation.dtd">
<attributecalculation>
<calculateattribute type="Order item recorded by">
<operation name="concat">
<constant>Walter, "A" </constant>
```

```
<valueof>VBAP-ERNAM</valueof>
</operation>
</calculateattribute>
<calculateattribute type="Note on order recording">
<operation name="concat">
<operation name="substring">
<valueof>Order item recorded by</valueof>
<constant>1</constant>
<constant>6</constant>
</operation>
<constant> </constant>
<operation name="substring">
<valueof>Order item recorded by</valueof>
<constant>-1</constant>
<constant>7</constant>
</operation>
<constant> recorded item</constant>
<valueof>VBAP-POSNR</valueof>
<constant> in order </constant>
<valueof>VBAP-VBELN</valueof>
<constant>.</constant>
</operation>
</calculateattribute>
</attributecalculation>
```

Quando voltar a chamar o programa de linha de comando relevante, desta vez com o parâmetro **-calcconfig <XML attribute transformation file>**, será gerada o seguinte arquivo de saída XML:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="VBAP-VBELN">3866</attribute>
<attribute type="VBAP-POSNR">20</attribute>
<attribute type="VBAP-ERNAM">SCHMIDT</attribute>
<attribute type="Order item recorded by">
Walter, "A" SCHMIDT
</attribute>
<attribute type="VBAP-MATNR-MAKTX">100038</attribute>
<attribute type="VBAP-KWMENG">3</attribute>
<attribute type="Note on order recording">
Walter SCHMIDT recorded item 20 in order 3866.
</attribute>
</event>
</eventlist>
```

7.2. Fonte de dados

Todos os arquivos de configuração necessários para extração e o arquivo de saída podem ser especificados em conjunto, em um arquivo de fonte de dados. O arquivo XML é especificado pelo parâmetro **-datasource** no console. Este parâmetro é usado em vez dos parâmetros **-systemconfig**, **-tableconfig**, **-csvconfig**, **-i**, **-calcconfig**, **-outfile** e **-nozip**.

O formato do arquivo XML é especificado pela DTD **datasource.dtd**:

Elemento XML	Atributos XML	Descrição
datasource	name type lastreaddate lastreadtime	Uma fonte de dados tem um nome único, é exatamente de um dos tipos **EVENT, GRAPH, MYSAP, JDBC** ou **CSV** e contém a última hora de extração.
dataextraction (ver o capítulo Configuração de vários arquivos de saída [150])		Especifica o(s) arquivo(s) de saída XML. A extensão de arquivo (**.zip** ou **.xml**) tem de ser especificado explicitamente. É gerado um arquivo XML compactado ou descompactado, dependendo da extensão de arquivo.
systemconfig		Especifica o arquivo de configuração do sistema. O arquivo pode conter apenas uma configuração do acesso ao sistema.
eventspec		Especifica o arquivo de configuração da tabela (especificação do evento). O arquivo pode conter apenas uma configuração da tabela (elemento XML **configuration**).
attribute transformação (opcional)		Especifica o arquivo de transformação de atributos

Se transferir o arquivo de fonte de dados para o programa de linha de comando com o parâmetro **-datasource**, a última hora de extração (**lastreaddate**, **lastreadtime**) é usada e atualizada apenas neste arquivo. A hora, na configuração do sistema usado, não é mais ajustada.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE datasource SYSTEM "datasource.dtd">
<datasource name="BILLING" type="MYSAP"
lastreaddate="19700101" lastreadtime="000000">
<description name="desc_de" language="de">
(Cancelation) invoice,
(Cancelation) credit memo,
Internal credit memo,
Internal allocation,
Debit memo, pro forma invoice
</description>
<description name="desc_en" language="en">
(Cancelation) invoice,
(Cancelation) credit memo,
Internal credit memo,
Internal allocation,
Debit memo, pro forma invoice
</description>
<dataextraction> see chapter
Configuration of multiple output files [150]
</dataextraction>
<systemconfig>C:\Programs\jdbc2ppm\xml\Systemconfig.xml
</systemconfig>
<eventspec>C:\Programs\jdbc2ppm\xml\TableConfiguration.xml
</eventspec>
<attributetransformation>C:\Programs\jdbc2ppm\xml\
attributecalculation.xml
```

```
</attributetransformation>
</datasource>
```

7.2.1. Configuração de vários arquivos de saída

Se volumes de dados muito grandes forem extraídos e gravados em um único arquivo de saída como eventos de sistema, o arquivo poderá tornar-se difícil de tratar. Nesses casos, é possível configurar a fonte de dados usada de forma que os dados extraídos sejam gravados em qualquer número de arquivos de saída XML. Tudo o que precisa fazer é especificar o número máximo de eventos de sistema por arquivo de saída e o nome do arquivo de saída no elemento XML **dataextraction**.

Exemplo 72.

(para uma fonte de dados CSV, mesmo procedimento para os tipos JDBC e SAP)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE datasource SYSTEM "datasource.dtd">
<datasource name="BILLING" type="CSV">
<dataextraction>
<outputfilename>..\custom\<clientname>\data\
BILLING_data_$EXTRACTIONDATE$_$EXTRACTIONTIME$.zip
</outputfilename>
<numberofeventsperxmlfile>
100000
</numberofeventsperxmlfile>
</dataextraction>
...
<systemconfig>...</systemconfig>
<eventspec>...</eventspec>
...
</datasource>
```

Para a fonte de dados **BILLING**, o elemento XML **numberofeventsperxmlfile** especifica que cada arquivo de saída XML deve conter 100000 eventos de sistema, com o último arquivo de saída gerado contendo todos os eventos restantes.

O elemento XML **outputfilename** especifica o caminho e nome dos arquivos de saída. Estas definições geram arquivos XML com nomes no formato **BILLING_data_$EXTRACTIONDATE$_$EXTRACTIONTIME$.zip**, no diretório de saída especificado. A variável do nome **$EXTRACTIONDATE$** contém a data de extração, enquanto **$EXTRACTIONTIME$** contém a hora de extração. Os arquivos de saída gerados são numerados colocando **_<x>** no final do nome, em que **x** é um número consecutivo e o primeiro arquivo de saída gerado não é numerado.

Se 369000 eventos de sistema tiverem sido extraídos em 23 de junho de 2007 às 11:36:45, o uso da configuração de exemplo exibida criará os seguintes arquivos de saída:

- **BILLING_data_20070623_113645.zip**
  (com 100000 eventos de sistema)
- **BILLING_data_20070623_113645_1.zip**
  (com 100000 eventos de sistema)
- **BILLING_data_20070623_113645_2.zip**
  (com 100000 eventos de sistema)

- **BILLING_data_20070623_113645_3.zip**
  (com 69000 eventos de sistema)

A tabela abaixo lista todas as opções de configuração:

Elemento XML	Descrição
numberofeventsperxmlfile	Número fixo de eventos de sistema gravados para um arquivo de saída. O último arquivo de saída criado contém o número restante de eventos. Se este elemento estiver ausente, todos os eventos de sistema serão gravados em um arquivo de saída.
outputfilename	Caminho e padrão de nomeação do arquivo de saída. Os formatos XML e ZIP são suportados. Um arquivo de saída ZIP contém um arquivo de saída XML com o mesmo nome. Os arquivos de saída para uma extração são numerados consecutivamente com **_<x>** no final do nome, apesar de o primeiro arquivo não ser numerado.

As seguintes variáveis são permitidas no nome do arquivo de saída (**outputfilename**) e podem ser usadas em qualquer combinação:

Variável (tipo de fonte de dados)	Descrição
$EXTRACTIONDATE$ (CSV, SAP, JDBC)	Data de extração (formato: aaaaMMdd)
$EXTRACTIONTIME$ (CSV, SAP, JDBC)	Hora de extração (formato: HHmmss)
$BEGINDATE$ (SAP, JDBC)	Data de início do período de extração (formato: yyyyMMdd)
$BEGINTIME$ (SAP, JDBC)	Hora de início do período de extração (formato: HHmmss)
$ENDDATE$ (SAP, JDBC)	Data de término do período de extração (formato: yyyyMMdd)
$ENDTIME$ (SAP, JDBC)	Hora de término do período de extração (formato: HHmmss)
$VALUECONSTRAINT$ (SAP, ver o capítulo Operadores de condição [29] e JDBC, ver o capítulo Operadores de condição [85])	Formato de saída: <Operator>_Value1 ou <Operator>_Value1_<Operator>_Value2 Saída de operadores e valores de comparação de números inteiros usados para restringir o volume de dados extraídos. O nome do arquivo de saída pode ser: **outfile_gt_230_le_300_<x>.xml**, em que **x** é o número consecutivo. Representação do operador: maior do que: **gt** maior ou igual a: **ge** inferior a: **lt** inferior ou igual a: **le**

Pode configurar convenientemente a distribuição dos conjuntos de dados extraídos para vários arquivos de saída em PPM Customizing Toolkit, no **Gerenciamento da fonte de dados** do cliente, por meio de **Configurações adicionais...**, na área **Extração de dados**.

7.2.2. Configuração de uma extração de offset em termos de hora ou valor

As funções descritas neste capítulo são suportadas apenas pelos dois extratores de processo **PPM Process Extractor JDBC-2-PPM** e **PPM Process Extractor SAP-2-PPM** e apenas quando for usada uma configuração da fonte de dados.

Para a extração baseada na hora, pode especificar um valor de hora na configuração da fonte de dados. O valor é subtraído da hora de execução atual como tempo final ao executar o programa de linha de comando correspondente sem parâmetros de hora de término (**-enddate** ou **-endtime**). A extração de dados terminará antes, no valor especificado (em segundos).

Sem especificar um valor de offset da hora, os registros de dados que foram criados mas não gravados explicitamente no banco de dados a ser extraído podem não ser extraídos em nenhuma das duas operações de extração subsequentes, devido ao seu carimbo de data/hora de offset (ver Extração automática e contínua [69]). Isto pode ser evitado especificando um valor de offset.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE datasource SYSTEM "datasource.dtd">

<datasource name="SAP" type="MYSAP" lastreaddate="19700101"
lastreadtime="000000" readoffset="86400">
<description name="default_description" language="en"/>
<description name="default_description" language="de">

  SAP

  </description>
...
</datasource>
```

Exemplo 73.

O atributo XML **readoffset** indica um valor de offset de **86400 segundos**. Quando chamar o programa de linha de comando com

```
runjdbc2ppm -datasource SAP.xml
```

Exemplo 74.

a operação de extração termina um dia (86400 segundos) antes da hora de execução atual. Se a hora de execução do programa for 11.08.08 9:48:05, a operação de extração terminará em 10.08.08, às 9:48:05.

```
...
I: 11.08.08 09:48:06: [XML] Reading data with start date/time 19700101/000000
and end date/time 20080810/094805...
...
```

O valor de offset da última hora de extração é gravado no arquivo de fonte de dados como a hora de início da extração de dados seguinte.

Para extração condicional usando um critério de número inteiro (operador lógico **valueconstraint**, ver Extração usando condições [84]), pode igualmente especificar um valor de offset que é subtraído do valor inicial (**lastreadvalue** do arquivo da fonte de dados) quando executar o programa. Se o resultado dos dados for um valor negativo para o critério inteiro, esse valor será tratado como um valor positivo.

Um valor de offset especificado na configuração da fonte de dados apenas é incluído na extração de dados se a hora de término ou o limite inferior do critério inteiro não for especificado explicitamente.

Dica
Pode configurar facilmente valores de offset em **PPM Customizing Toolkit**, no grupo de módulos **Cliente**, em **Gerenciamento da fonte de dados**. Insira o valor como informações adicionais para a última hora de extração ou o último valor extraído nos componentes da fonte de dados relevante.

Especificar um fuso horário

Se você não especificou a hora de término na linha de comando (argumentos **-enddate** ou **-endtime**) a hora atual do computador local é usada como hora de término, por padrão. Esta hora de término é usada com os operadores **creationtimestamp**, **date_creationtimestamp** ou **char_creationtimestamp** para restringir o volume de dados a ser extraído. Se o campo de dados mencionado nos operadores contiver carimbos de data/hora associados a um fuso horário diferente, poderá especificar esse fuso horário no atributo XML **sourcefieldtimezone** (por exemplo, **Austrália/Camberra**). Na extração de dados subsequente, a hora do computador local é convertida para o fuso horário especificado e, após uma extração bem sucedida, a hora é salva como a última hora de extração (atributos XML **lastreaddate** e **lastreadtime**) na fonte de dados.

Se nenhum fuso horário válido ou o próprio atributo não for especificado no atributo XML **sourcefieldtimezone**, não será realizada uma conversão de fuso horário da hora de término. Com o PPM CTK, o fuso horário pode ser facilmente selecionado usando a interface.

Exemplo 75.

África/Dakar, África/Joanesburgo, Austrália/Sydney, América/Denver, América/Los_Angeles, América/México, Canadá/Central, Europa/Berlim, Europa/Londres, México/Geral, EUA/Central, EUA/Havaí

7.3. Arquivo de saída XML (formatos)

Por padrão, os dados extraídos usando os extratores de processo do PPM são gravados em um arquivo de saída XML no formato de evento do sistema PPM.

Ao especificar um dos parâmetros de linha de comando opcionais **-pikidatamapping** ou **-dimdatamapping**, também pode ter os dados extraídos gravados no(s) arquivo(s) de saída no formato de dados de medida independente da instância de processo ou no formato de dados de dimensão.

Os diferentes formatos de saída são descritos em detalhe no manual

Importação de dados do PPM.

7.3.1. Formato do evento de sistema PPM

Formato padrão do(s) arquivo(s) de saída XML se nenhum dos parâmetros de linha de comando **-pikidatamapping** e **-dimdatamapping** for usado (ver o capítulo **Arquivo de saída XML (formato do evento de sistema do PPM) [37]**).

7.3.2. Formato PIKIDATA

Os dados extraídos devem ser gravados como medidas independentes da instância de processo (PIKI) para um arquivo de saída XML, para a importação do PPM usando o programa de linha de comando **runpikidata.bat**.

O arquivo de configuração XML tem a seguinte estrutura básica:

```
...
<pikidatamapping>
<pikicube name="...">
<pikicolmapping>
<datacol name="..."/>
<keyspec key="..."/>
</pikicolmapping>
...
</pikicube>
...
</pikidatamapping>
```

Elemento XML	Atributos XML	Descrição
pikidatamapping		Elemento raiz da definição de mapeamento. Contém uma lista da série de medidas do PPM para serem configuradas.
pikicube	name	Identificador da série de medidas do PPM (corresponde ao valor do atributo XML **name** para a configuração da série de medidas no PPM)
pikicolmapping		Agrupa elementos de configuração para uma coluna de dados
datacol	name	Nome da coluna de dados na qual os valores PIKI são salvos (corresponde ao nome interno do PIKI na configuração de medidas)
	format (opcional)	Cadeia de caracteres de formato para transformação do valor de dados
keyspec	key	Nome da chave para a associação de valor-chave
	desckey	Nome da chave para a associação de descrição-chave
value		Valor da medida independente da instância de processo

O extrato de arquivo abaixo mostra a série de medidas

PCA Import existente no PPM:

```
...
<pikicube name="PIKICUBE_TURNOVER">
<description language="de" name="PCA-Import"/>
<description language="en" name="PCA Import"/>
<pikidef name="TURNOVER" retrievertype="NUM_KEYINDICATOR"
dimreferring="LOOSE" kigroup="KI_GROUP_COST">
<description language="de" name="Umsatz">
Umsatz
</description>
<description language="en" name="Sales revenues">
Sales revenues
```

```
</description>
<datatype name="COST"/>
</pikidef>
<refdim name="VKORG"/>
<refdim name="DIVISION"/>
<refdim name="TIME" refinement="BY_MONTH"/>
</pikicube>
...
```

Exemplo 76.

Os seguintes pares de valores-chave são o resultado esperado da extração no arquivo de saída XML, no formato de evento do sistema PPM:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="MARA-TURNOVER">225489 EUR</attribute>
<attribute type="MARB-VKORG_ID">1000</attribute>
<attribute type="MARB-VKORG_DESC">Germany Hamburg
</attribute>
<attribute type="DIM_ROUGH_ID">01</attribute>
<attribute type="DIM_ROUGH_DESC">Product category 01
</attribute>
<attribute type="DIM_DETAILED_ID">8112</attribute>
<attribute type="DIM_DETAILED_DESC">8112 description
</attribute>
<attribute type="CAL-TIME">January 2002</attribute>
</event>
<event>
<attribute type="MARA-TURNOVER">135699 EUR</attribute>
<attribute type="MARB-VKORG_ID">4000</attribute>
<attribute type="MARB-VKORG_DESC">Austria Vienna
</attribute>
<attribute type="DIM_ROUGH_ID">01</attribute>
<attribute type="DIM_ROUGH_DESC">Product category 01
</attribute>
<attribute type="DIM_DETAILED_ID">8112</attribute>
<attribute type="DIM_DETAILED_DESC">8112 description
</attribute>
<attribute type="CAL-TIME">January 2002</attribute>
</event>
<event>
<attribute type="MARA-TURNOVER">363521 EUR</attribute>
<attribute type="MARB-VKORG_ID">1000</attribute>
<attribute type="MARB-VKORG_DESC">Germany Hamburg
</attribute>
<attribute type="DIM_ROUGH_ID">07</attribute>
<attribute type="DIM_ROUGH_DESC">High Tech</attribute>
<attribute type="DIM_DETAILED_ID">9128</attribute>
<attribute type="DIM_DETAILED_DESC">9128 description
</attribute>
<attribute type="CAL-TIME">January 2002</attribute>
```

```
</event>
</eventlist>
```

Exemplo 77.

O arquivo de mapeamento abaixo transfere os pares de valores-chave para o formato XML compatível com o PPM para importar valores PIKI:

```
...
<pikidatamapping>
<pikicube name="PIKICUBE_TURNOVER">
<pikicolmapping>
<datacol name="TURNOVER"/>
<keyspec key="MARA-TURNOVER"/>
</pikicolmapping>
<pikicolmapping>
<datacol name="VKORG"/>
<keyspec key="MARB-VKORG_ID"

  desckey="MARB-VKORG_DESC"/>
</pikicolmapping>
<pikicolmapping>
<datacol name="DIVISION"/>
<keyspec key="DIM_ROUGH_ID"

  desckey="DIM_ROUGH_DESC"/>
</pikicolmapping>
<pikicolmapping>
<datacol name="DIVISION"/>
<keyspec key="DIM_DETAILED_ID"
desckey="DIM_DETAILED_DESC"/>
</pikicolmapping>
<pikicolmapping>
<datacol name="TIME" format="MMM yyyy"/>
<keyspec key="CAL-TIME"/>
</pikicolmapping>
</pikicube>
</pikidatamapping>
```

Exemplo 78.

O uso desse arquivo de mapeamento gera o seguinte arquivo de saída XML quando executa o programa:

```
...
<pikidata>
<pikicube name="PIKICUBE_TURNOVER">
<datacols>
<datacol name="TURNOVER"/>
<datacol name="VKORG"/>
<datacol name="DIVISION"/>
<datacol name="DIVISION"/>
<datacol name="TIME" format="MMM yyyy"/>
</datacols>
...
```

```
<datarow>
<value>225489 EUR</value>
<value>1000{Germany Hamburg}</value>
<value>01{Product category 01}</value>
<value>8112{8112 description}</value>
<value>January 2002</value>
</datarow>
<datarow>
<value>135699 EUR</value>
<value>4000{Austria Vienna}</value>
<value>01{Product category 01}</value>
<value>8112{8112 description}</value>
<value>January 2002</value>
</datarow>
<datarow>
<value>363521 EUR</value>
<value>1000{Germany Hamburg}</value>
<value>07{High Tech}</value>
<value>9128{9128 description}</value>
<value>January 2002</value>
</datarow>
...
</pikicube>
</pikidata>
...
```

7.3.3. Formato DIMDATA

Os dados extraídos pelo extrator de processo relevante do PPM devem ser gravados no arquivo de saída XML, no formato de importação especial para dimensões de um, dois ou vários níveis. Os arquivos de saída XML neste formato podem ser importados para o PPM usando o programa de linha de comando **rundimdata.bat** (ver a referência técnica **PPM Data import**).

Para extrair dados corretamente, o extrator de processo relevante requer um arquivo de mapeamento que atribua as chaves (e quaisquer descrições) para os níveis de dimensão individuais aos valores de campo extraídos e corretos. Este arquivo de configuração XML tem a seguinte estrutura:

```
...
<dimdatamapping>
<dimension name="...">
<dimcolmapping>
<datacol name="..."/>
<keyspec key="..."/>
</dimcolmapping>
...
</dimension>
</dimdatamapping>

  ...
```

Elemento XML	Atributos XML	Descrição
dimdatamapping		Elemento raiz da definição de mapeamento. Contém uma lista de dimensões PPM a serem configuradas.
dimension	name	Identificador da dimensão do PPM (corresponde ao valor do atributo XML **name** da configuração da dimensão no sistema PPM)
dimcolmapping		Agrupa elementos de configuração para uma coluna de dados.
datacol	name	Nome da coluna de dados na qual os valores de dimensão são salvos.
	format (opcional)	Cadeia de caracteres de formato para transformação do valor de dados para uso no sistema PPM
keyspec	key	Nome da chave para as associações de valor-chave extraídas
	desckey	Nome da chave para as associações de descrição-chave extraídas
value		Especificação de um valor de dimensão constante

Exemplo 79.

Os seguintes pares de valores-chave são o resultado esperado da extração no arquivo de saída XML, no formato de evento do sistema PPM:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE eventlist SYSTEM "event.dtd">
<eventlist>
<event>
<attribute type="MAT1_ID-MATNR">1000</attribute>
<attribute type="MAT1_DESC-HTEXT">Motor44</attribute>
<attribute type="MAT2_ID-MTART">32</attribute>
<attribute type="MAT2_DESC-MTBEZ">Components</attribute>
</event>
<event>
<attribute type="MAT1_ID-MATNR">1001</attribute>
<attribute type="MAT1_DESC-HTEXT">Wrench77</attribute>
<attribute type="MAT2_ID-MTART">45</attribute>
<attribute type="MAT2_DESC-MTBEZ">Tools</attribute>
</event>
<event>
<attribute type="MAT1_ID-MATNR">1002</attribute>
<attribute type="MAT1_DESC-HTEXT">Pump43</attribute>
<attribute type="MAT2_ID-MTART">33</attribute>
<attribute type="MAT2_DESC-MTBEZ">Trade goods</attribute>
</event>
</eventlist>
```

Exemplo 80.

O arquivo de mapeamento abaixo transfere os pares de valores-chave para o formato XML compatível com o PPM para importar valores de dimensão:

```
...
<dimdatamapping>
<dimension name="MATERIAL">
<dimcolmapping>
```

```
<datacol name="LEVEL1_ID"/>
<keyspec key="MAT1_ID-MATNR"/>
</dimcolmapping>
<dimcolmapping>
<datacol name="LEVEL1_DESC"/>
<keyspec key="MAT1_DESC-HTEXT"/>
</dimcolmapping>
<dimcolmapping>
<datacol name="LEVEL2_ID"/>
<keyspec key="MAT2_ID-MTART"/>
</dimcolmapping>
<dimcolmapping>
<datacol name="LEVEL2_DESC"
<keyspec key="MAT2_DESC-MTBEZ"/>
</dimcolmapping>
</dimension>
</dimdatamapping>
```

Os nomes das colunas de dados para um nível de dimensão PPM são fixos (**LEVEL1_ID**, **LEVEL1_DESC**, **LEVEL2_ID**, **LEVEL2_DESC**, etc.) e não podem ser alterados.

A tabela abaixo ilustra a associação de colunas de dados às chaves e descrições dos níveis de dimensão individuais:

configuração dimdata	Configuração da dimensão
LEVEL1_ID	Identificador do primeiro nível de dimensão
LEVEL1_DESC	Descrição do primeiro nível de dimensão
LEVEL2_ID	Identificador do segundo nível de dimensão
LEVEL2_DESC	Descrição do segundo nível de dimensão
LEVELn_ID	ID do nível de dimensão n-th
LEVELn_DESC	Descrição do nível de dimensão n-th

Exemplo 81.

O arquivo de saída XML gerado no formato DIMDATA tem o seguinte conteúdo:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE dimdata SYSTEM "dimdata.dtd">
<dimdata>
<dim name="MATERIAL">
<datacols>
<datacol name="LEVEL1_ID"/>
<datacol name="LEVEL1_DESC"/>
<datacol name="LEVEL2_ID"/>
<datacol name="LEVEL2_DESC"/>
</datacols>
<datarow>
<value>1001</value>
<value>Motor44</value>
<value>32</value>
<value>Components</value>
</datarow>
```

```
<datarow>
<value>1002</value>
<value>Wrench77</value>
<value>45</value>
<value>Tools</value>
</datarow>
<datarow>
<value>1003</value>
<value>Pump43</value>
<value>33</value>
<value>Trade goods</value>
</datarow>
</dim>
```

8. Pseudonimização de dados

Você pode pseudonimizar os dados extraídos pelos extratores **SAP**, **JDBC** e **CSV**. Os dados extraídos são pseudonimizados antes de serem gravados no arquivo de saída como um evento do sistema de origem.

Você pode pseudonimizar os dados de origem usando o PPM CTK. A opção **Usar pseudonimização durante a extração de dados** fica desativada por padrão, mas você pode ativá-la nas propriedades do cliente no PPM CTK. Se ativar esta opção, o PPM CTK fornece a guia **Pseudonimização** no componente **Extração de dados** do módulo **Mesclagem de processos**. Nesta página, pode especificar para cada atributo do sistema de origem disponível se seu valor deve ser pseudonimizado ou não.

Abra um cliente no PPM CTK -> Propriedades do cliente -> Fontes de dados -> Ativar **Usar pseudonimização durante a extração de dados**.

Para a pseudonimização de dados, o PPM CTK cria uma chave obrigatória e a armazena no arquivo **pseudonymization.key** do diretório **.../custom/<client>/keyfiles**. O arquivo **pseudonymization.key** criado é do tipo binário.

Atenção que deve ativar a pseudonimização antes que quaisquer dados de um atributo do sistema de origem sejam extraídos. Os dados de texto não criptografado já importados não podem ser pseudonimizados.

> **Atenção**
> Recomendamos que crie um backup do arquivo **pseudonymization.key**. O arquivo de chave não pode ser restaurado e não pode usar outra chave de pseudonimização.

8.1. Sistemas dimensionados

Se a pseudonimização for usada para sistemas dimensionados, todos os subservidores deverão usar a mesma chave de pseudonimização, ou seja, deverá garantir que o mesmo arquivo de chave existe em todos os subservidores.

Portanto, se a pseudonimização estiver ativada no subservidor ou no mestre, copie o arquivo chave deste cliente do PPM para todos os outros subservidores e para o mestre desse cenário dimensionado no diretório **.../custom/<client>/keyfiles**. Isso deve ser feito no respectivo cliente, antes da primeira extração de dados de atributos pseudonimizados.

8.2. Configuração de atributos pseudonimizados

O DTD **datasource.dtd** é adaptado para pseudonimização da seguinte forma:

```
<!ELEMENT datasource (description*, realmtable?, dataextraction?,
data, archive?, fragments?, mapping?, systemconfig?, tables?,
eventspec?, attributetransformation?, eventattributetypes?, attributesettings?,
edaeventtype?, pseudonymization?)>
```

```
<!ELEMENT eventattribute (#PCDATA)>

  <!ELEMENT pseudonymization (eventattribute+)>
```

O elemento opcional **pseudonymization** tem de conter pelo menos um subelemento **eventattribute**. Um elemento **eventattribute** contém o nome do atributo do sistema de origem a ser pseudonimizado, expresso como o tipo de atributo que aparece no evento do arquivo de saída.

Exemplo 82.

O exemplo a seguir mostra a pseudonimização de BSEG-ERNAME e LAST_USER.

- Arquivo de saída sem pseudonimização

```
<event>
               <attribute type="BSEG-ERNAME">Test User</attribute>
               <attribute type="CARDNO">308342023837750</attribute>
               <attribute type="CONTACT_REASON">Bonus not received</attribute>
               <attribute type="FIELD_2">Phone</attribute>
               <attribute type="LAST_USER">Second user</attribute>
               <attribute type="SR_CLOSED_DATE">11.06.2003 18:07:52</attribute>
</event>
```

Exemplo 83.

- Configuração da pseudonimização

```
<pseudonymization>
<eventattribute>BSEG-ERNAME</eventattribute>
<eventattribute>LAST_USER</eventattribute>
</pseudonymization>
```

Exemplo 84.

- Arquivo de saída com pseudonimização

```
<event>
               <attribute type="BSEG-ERNAME">WReKoRCNHVdJF6FZ8oLqFGg==</
attribute>
               <attribute type="CARDNO">308342023837750</attribute>
               <attribute type="CONTACT_REASON">Bonus not received</attribute>
               <attribute type="FIELD_2">Phone</attribute>
               <attribute type="LAST_USER">+XMLp1ip51d++gfVWvA+bQ==</attribute>
               <attribute type="SR_CLOSED_DATE">11.06.2003 18:07:52</attribute>
</event>
```

8.3. Dimensões dos dados pseudonimizados

Todos os dados pseudonimizados são do tipo TEXT. Isso significa que os dados devem ter o tipo de dados TEXT no PPM, incluindo dados que eram, por exemplo, do tipo NUMERICAL antes da pseudonimização.

Portanto, deve garantir que os valores sejam atribuídos aos atributos de texto do PPM durante a importação XML e usados apenas para dimensões de texto.

O comprimento do valor pseudonimizado (cipherLength) depende do tamanho do valor antes da pseudonimização (clearLength).

Tem de criar dimensões nas quais os dados pseudonimizados devem ser armazenados com um tamanho de dimensão de 1.000. Dados de texto simples não criptografados não devem exceder 200 caracteres, caso contrário, o algoritmo de pseudonimização pode gerar valores maiores que 1.000. Isso tornaria impossível a reconversão em texto simples.

A importação do PPM registra uma mensagem de erro para cada valor de dimensão demasiado longo.

8.4. Ferramenta de pseudonimização do PPM

A ferramenta de pseudonimização do PPM pseudonimiza um valor de cleartext ou converte um valor pseudonimizado em valor cleartext.

Deve adicionar as credenciais de um usuário do PPM com direitos de administração do sistema nos seguintes exemplos de linha de comando:

- "runppmpseudonymize -in <file> [-encoding <encoding>]..." gera um valor pseudonimizado.
- "runppmpseudonymize -reverse <pseudonymized value> -out <file> [-encoding <encoding>]..." gera um valor cleartext.

```
runppmpseudonymize -user <username> -password <password> [-client <name>]

  {-in <filename>} | {-reverse <pseudonymized value> -out <filename>}

  [-encoding <encoding>]

  [-version]

  [-language <ISO code>][protocoloptions]
```

Parâmetro	Descrição
-in <filename>	Nome do arquivo que contém o valor de texto não criptografado que deve ser pseudonimizado
-reverse <pseudonymized value>	Se esse parâmetro for especificado, esse valor pseudonimizado será convertido em valor de texto não criptografado.
-out <filename>	Nome do arquivo no qual o valor de texto não criptografado deve ser gravado.
-encoding <encoding>	Codificação do arquivo de entrada ou de saída (Padrão: UTF-8)
-language <ISO code>	Language
-version	Número de versão do aplicativo e do esquema do banco de dados

Parâmetro	Descrição
protocoloptions	protocoloptions pode ser composta pelas seguintes instruções: -protocolfile <filename> Log no arquivo <file name> -information {yes|no|default} Log de informações -warning {yes|no|default} Log de avisos -error {yes|no|default} Log de erros

Se o parâmetro **-reverse <pseudonymized value>** for especificado, o valor de texto sem formatação é enviado para o arquivo **-out <filename>** na codificação especificada. Se o arquivo **filename** já existir, ele será substituído.

Se o parâmetro **-in <filename>** for especificado, o valor no arquivo **<filename>** será interpretado como um valor de texto sem formatação e o valor pseudonimizado será enviado diretamente para a saída padrão, ou seja, fora do protocolo. O arquivo é lido na codificação especificada.

O arquivo chave **pseudonymization.key** usado para criptografia/descriptografia é esperado no seguinte diretório: **<PPM installation>\server\bin\work\data_ppm\custom\<client>\keyfiles**. Portanto, o cliente também deve ser especificado na linha de comando usando o parâmetro **-client**. Se o cliente especificado não suportar pseudonimização, será exibida uma mensagem de erro correspondente.

Atenção que o usuário usado para (des)pseudonimização com esta ferramenta deve ter direitos de administração do sistema no PPM.

8.5. Adicionar novos atributos criptografados

No módulo **Importação de dados** do PPM CTK, deve definir o tipo de dados TEXT para o atributo antes do primeiro mapeamento, para que o tipo de dados correto seja atribuído aos atributos do PPM criados automaticamente durante o mapeamento.

Se especificou na configuração da extração de dados para o atributo de evento do sistema de origem "VDARL-RERF", que na verdade possui o tipo de dados DOUBLE no sistema de origem, que deve ser criptografado, deve agora selecionar o tipo de dados TEXT, pois os valores dos atributos do evento do sistema de origem criptografado são textos.

- O comprimento máximo de 200 do valor de texto não criptografado do atributo do sistema de origem não deve ser excedido.
- Se um valor pseudonimizado for usado como um valor de dimensão, tem de garantir que o comprimento da dimensão é definido como 1.000.
- Também para valores inteiros e duplos, o comprimento máximo dos valores não deve ser excedido.

9. Suporte e informações legais

Esta seção fornece algumas informações gerais sobre suporte ao produto e aspectos legais.

9.1. Escopo da documentação

As informações fornecidas descrevem as configurações e os recursos como estavam no momento da publicação. Como a documentação e o software estão sujeitos a ciclos de produção diferentes, a descrição das configurações e dos recursos pode ser diferente das configurações e recursos reais. Informações sobre discrepâncias são fornecidas nas Notas de versão que acompanham o produto. Leia as Notas de versão e considere as informações ao instalar, configurar e usar o produto.

Se você deseja instalar funcionalidades técnicas e/ou de sistema de negócios sem os serviços de consultoria fornecidos pelo SAG ARIS GmbH, precisa ter um amplo conhecimento do sistema a ser instalado, sua finalidade, os sistemas de destino e suas várias dependências. Devido ao número de plataformas e configurações de hardware e software interdependentes, só podemos descrever instalações específicas. Não é possível documentar todas as configurações e dependências.

Ao combinar várias tecnologias, observe as instruções dos fabricantes, especialmente os anúncios referentes a lançamentos em suas páginas da Internet. Não podemos garantir o funcionamento e a instalação adequados de sistemas de terceiros aprovados e não os suportamos. Siga sempre as instruções fornecidas nos manuais de instalação dos fabricantes relevantes. Se encontrar dificuldades, entre em contato com o fabricante relevante.

Se precisar de ajuda para instalar sistemas de terceiros, entre em contato com a sua organização local de vendas do ARIS. Observe que esse tipo de personalização específica do fabricante ou do cliente não é coberto pelo contrato de manutenção de software padrão do ARIS e pode ser realizada somente mediante solicitação e autorização especiais.

9.2. Suporte

Suporte aos Produtos

Fornecemos suporte para produtos ARIS a todos os clientes com um contrato de suporte válido.

Informações sobre disponibilidade e suporte da versão do produto, bem como datas de suporte sustentadas, estão disponíveis em ARIS Community.

Entre em contato com nossos serviços de suporte global na ARIS Community para criar e atualizar incidentes de suporte e postar ideias e solicitações de recursos.

Comunidade

Registre-se na ARIS Community para baixar produtos, atualizações e correções, encontrar informações especializadas e interagir com outros usuários.

Documentos

Para obter informações sobre como usar e ativar produtos e para outras questões de suporte técnico, visite o site de documentação do ARIS.

Se você quiser dar feedback sobre qualquer tópico de documentação, escreva para documentation@aris.com.

9.3. Legal notices

This document applies to ARIS Process Performance Manager Version 10 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2000-2026 SAG Aris GmbH, Saarbrücken, Germany and/or its subsidiaries and/or its affiliates and/or their licensors.

The name ARIS and all SAG Aris GmbH product names are either trademarks or registered trademarks of SAG Aris GmbH and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

The Software may contain or be distributed with certain commercial third party components (“Third Party Components”), which may be subject to special license terms and conditions (“Third Party Terms”) located at https://aris.com/terms.

Third Party Terms include important licensing and warranty information and disclaimers of third party licensors. For the avoidance of doubt, if Customer uses the Software in accordance with the Documentation, the Third Party Terms do not restrict the rights granted under this Agreement. In the event that a third party manufacturer no longer provides active support for any Third Party Component, Supplier will use commercially reasonable efforts to respond to any issues with the Software related to the Third Party Components.

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with SAG Aris GmbH and/or its subsidiaries and/or its affiliates and/or their licensors.