



Pentaho Data Integration version 2.3.0

A new batch of features!

Key differences with Kettle 2.2.2

List of changes on 20-06-'06

Compiled by Matt Casters, [mcasters \(at\) pentaho.org](mailto:mcasters@pentaho.org)

Send additional changes you found to this address.

Index

1. Changes summary.....	4
1.1. Preface.....	4
1.2. Overview.....	4
2. General changes.....	5
2.1. New splash screen.....	5
2.2. New login screen.....	5
2.3. Databases.....	6
2.4. Data grids.....	6
2.5. Platforms, Look & feel.....	7
2.6. Localizations.....	8
2.7. Localization management: Translator.....	9
2.8. Repository import and export.....	10
3. Spoon.....	11
3.1. Extra options.....	11
3.2. Search meta-data.....	11
3.3. Set environment variable.....	12
3.4. Execution log history.....	12
3.5. Replay.....	12
3.6. Generate mapping against target step.....	13
3.7. Safe mode.....	14
4. Chef.....	15
4.1. New logging window.....	15
4.2. Delete job entry copies.....	15
5. Steps.....	16
5.1. Text file input.....	16
5.2. Get System Info.....	17
5.3. Excel Input.....	18
5.4. XML Input.....	19
5.5. Table output.....	20
5.6. Insert / Update.....	21
5.7. Delete (new step).....	22
5.8. Sort rows.....	23
5.9. Value Mapper (New step).....	24
5.10. Dimension Lookup.....	25
5.11. Combination Lookup.....	26
5.12. Get rows from result.....	27
5.13. Set Variable (New step).....	28
5.14. Get Variable (New step).....	28
5.15. Get file names (New step).....	29
5.16. Get files from result (New step).....	30
5.17. Set files in result (New step).....	31
5.18. Blocking step (New step).....	32
5.19. Call Database Procedure.....	32
5.20. Injector (new step).....	33
6. Job Entries.....	34
6.1. Start.....	34

6.2. Transformation.....	34
6.3. Job.....	35
6.4. Shell.....	36
6.5. Mail.....	37
7. Source code improvements.....	38
7.1. A few extra lines of code.....	38
7.2. Trackers.....	38
7.3. Unit testing.....	38
7.4. Committers.....	38
7.5. Other contributors.....	39

1. Changes summary

1.1. Preface

Over the last 5 months, a lot of things have happened to the Pentaho Data Integration project, a.k.a. Kettle. Numerous developers joined the project and there were bug fixes provided by people in various regions of the world. In fact there have been so many changes to the code-base that it was a challenge to write this very summary. We believe one of the reasons for the many changes is the popularity that “Kettle” gained in this period. This has led to a long list of challenges that the software received and met in the course of the 2.3.0 development.

1.2. Overview

These are the most notable changes that have been made:

- Overall performance enhancements
- New error handling code (with replay) for reading text files and excel documents
- New steps: Delete, Value mapper, Set Variable, Get Variable, Get File names, Get files from result, Set files in result and the Blocking step
- Search meta-data functionality in Spoon: look for a value or parameter
- All sorts of parameterizing functionality in transformations and jobs
- The possibility to export and import of repositories, even in batch for backup
- Localization of the code into English, Chinese, French, German and Dutch
- Many enhancements to the existing steps and job entries
- Improved logging capabilities
- Easier fixed target table mapping
- Improved look & feel for all platforms
- ...

2. General changes

2.1. New splash screen

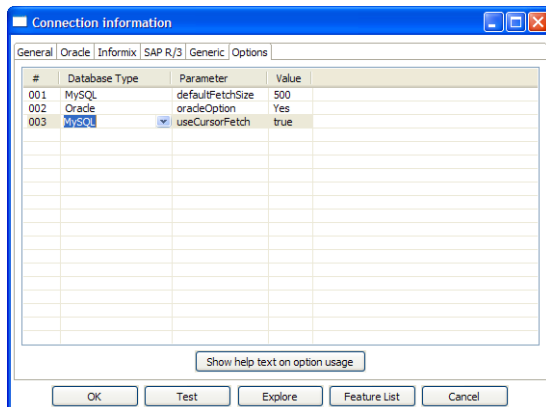


2.2. New login screen

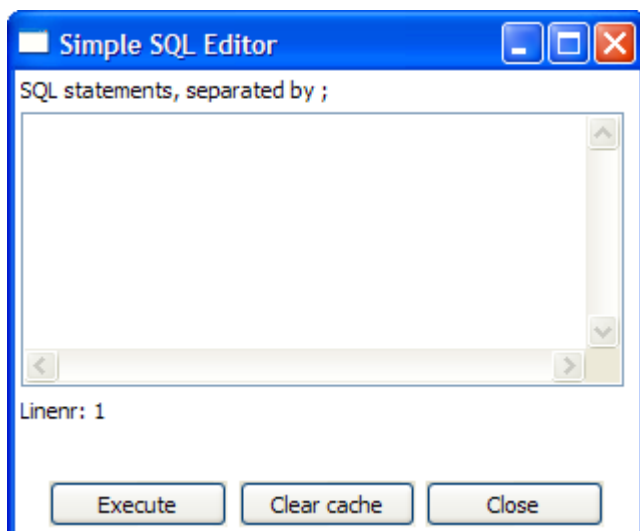


2.3. Databases

- Added support for ExtenDB (<http://www.extendb.com/>, patch by Mason Sharp <msharp (at) extendb.com>)
- Extra database specific options can be specified. These options are added to the connection settings.(through the URL or by setting Properties at connect time)



- Better support for handling reserved keyword quoting, as well as quoting of tables and fields with spaces, slashes, dashes, plus, etc. in it.
- Repository support on most (if not all) supported database platforms.
- Better handling of schema and catalog names in the repository explorer.
- We also added a “clear cache” button to the SQL Editor because the database cache often is no longer valid when working with 3rd party tools to create or alter tables. Because this is where the problem becomes visible, this is where we added the button:



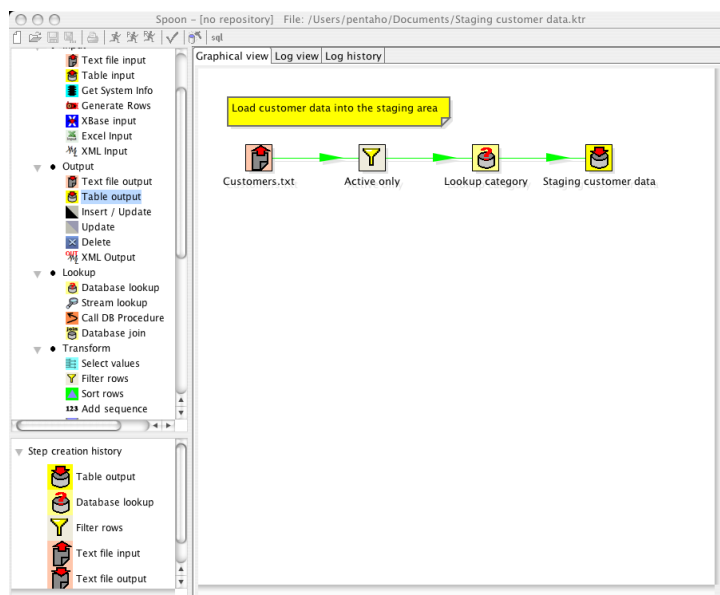
2.4. Data grids

- Serious performance upgrades for large grids
- Selected lines are now bright blue, no longer gray.

- Clicking below the last line automatically adds a new line
- Going beyond the last line with the cursor automatically adds a new line.

2.5. Platforms, Look & feel

- Both Linux and OS-X are now brought alongside of the Windows platform in terms of functionality and looks.
- All OS-X and Linux specific bugs are gone.
- Thanks to the new SWT 3.2 libraries, the look and feel of Windows is taken over automatically, giving a nicer user-experience on that platform. These are to be seen in the screen shots in this document.
- We now also have support for AMD64/Linux (x86_64)
- There is also support for the new Intel Macs using Universal Binaries.



- Preliminary FreeBSD/x86 support.

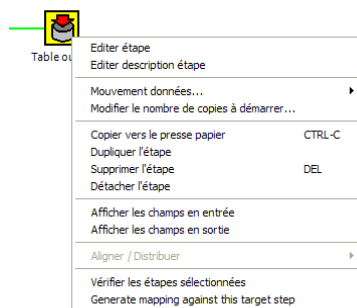
2.6. Localizations

Because of popular demand, we localized version 2.3.0 to allow us to translate the externalized strings. So far there are 4 languages we have available or partly available:

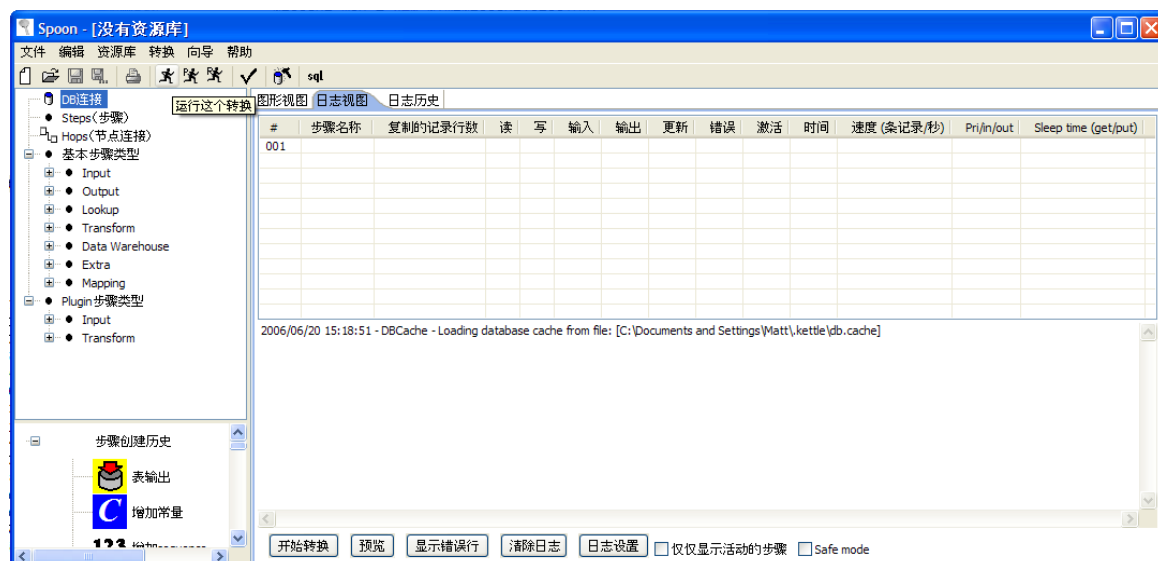
- English (100%)
- Simplified Chinese (>90%)
- French (>75%)
- German (>40%)
- Dutch (10%)

The next languages that will become available are most likely Spanish and Portuguese.

An example of a pop-up menu in French:

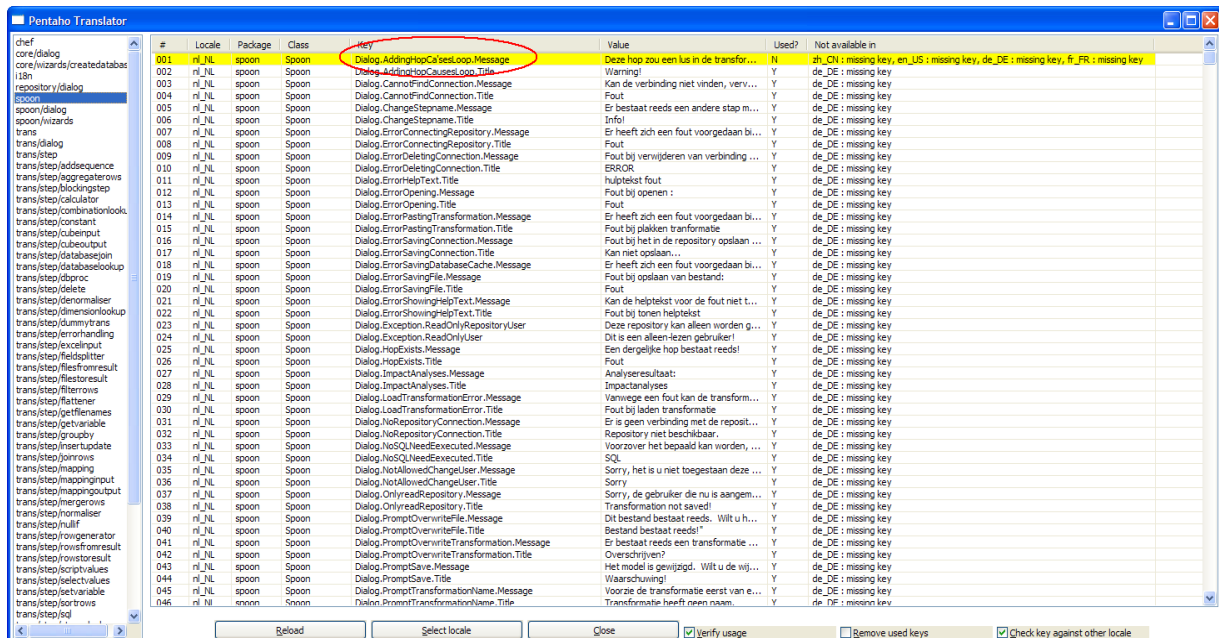


Spoon running in Simplified Chinese:



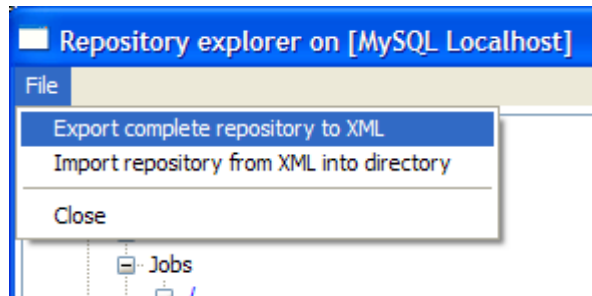
2.7. Localization management: Translator

To keep an overview of the translation efforts, we have created a new program called Pentaho Translator that allows you to browse the keys in the different languages and source packages. It makes it easier to detect typos, unused keys, etc.:



2.8. Repository import and export

Although basic support for import and export was already available in version 2.2.2, import and export was seriously improved and made available in a menu in the repository explorer:



The export functionality puts the complete contents of the repository into a single XML document. The export function is also available in Pan with the new “exprep” option.

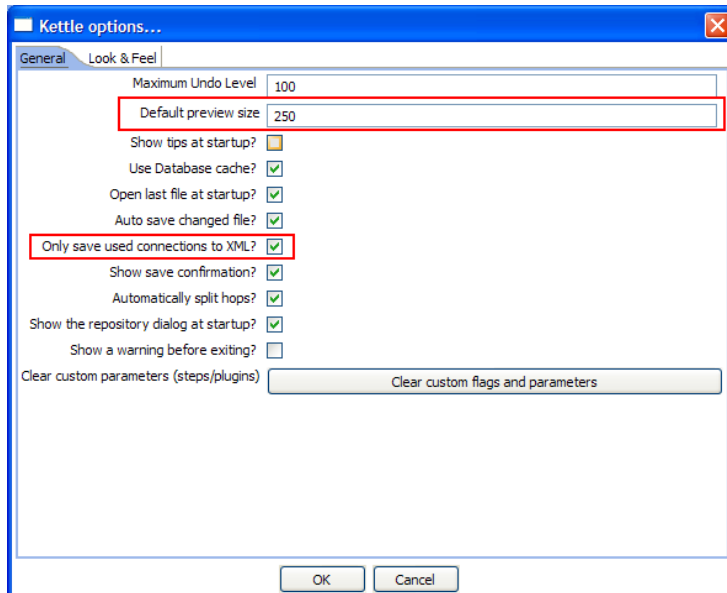
Import allows you to import the repository into a certain directory of an existing repository.

3. Spoon

For a list of the changes that were done in the steps, please see the corresponding chapter below.

3.1. Extra options

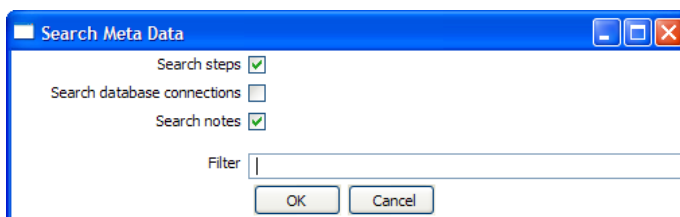
These are the new options that were added:



As you can see, it is now possible to set the default number of row to preview. Some people like to see more than others.

The other option will only save the connections that are used in a transformation in XML. This is handy if you have large numbers of connections defined.

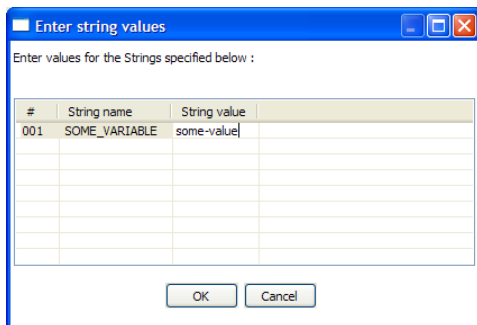
3.2. Search meta-data



This option will search in the transformation in any possible field, connector or note. A detailed result is shown.

3.3. Set environment variable

The possibility to set an environment variable was added because it becomes easier that way to test transformations that use dynamically set variables. Normally these variables are set by a different transformation in a job. However, during development and testing, you might want to set these manually.



This screen is also presented when you run a transformation that use undefined variables. That way you can define them right before execution time.

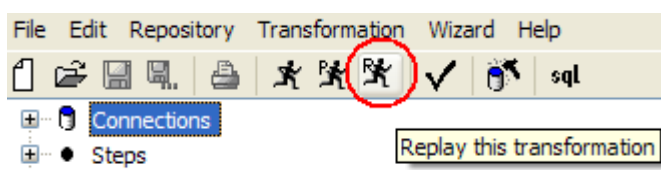
3.4. Execution log history

If you store the logging of the transformation in a database table (using the transformation settings, logging tab) you can see the overview of the runs in the log history tab in Spoon:

#	Batch ID	Status	Read	Written	Updated	Input	Output	Errors	Start date	End date	Log date	Dependency date	Replay date
001	2	end	0	452	0	452	0	0	1900/01/01 00:00:00.000	2006/06/06 15:52:37.000	2006/06/06 15:52:37.000	2006/06/06 15:52:37.000	2006/06/06 15:52:37.000
002	1	stop	0	308999	0	309001	0	0	1900/01/01 00:00:00.000	2006/06/06 15:51:41.000	2006/06/06 15:52:32.000	2006/06/06 15:51:41.000	2006/06/06 15:51:41.000

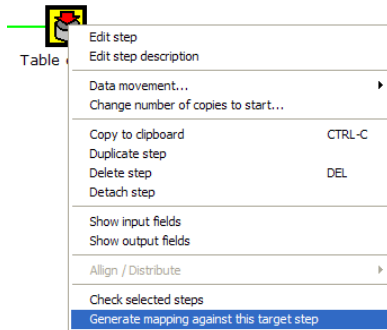
3.5. Replay

It is now possible to re-run a transformation that failed. Replay functionality is implemented for Text File Input and Excel input. It allows you to send files that had errors back to the source and have the data corrected. ONLY the lines that failed before are then processed during the replay if a .line file is present. It uses the date in the filename of the .line file to match the entered replay date.

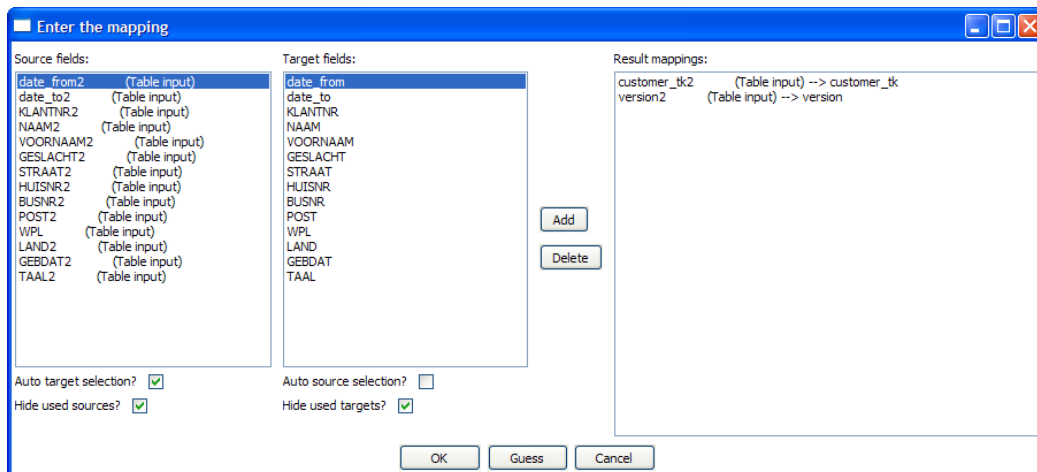


3.6. Generate mapping against target step

In cases where you have a fixed target table, you want to map the fields that go into the table output:

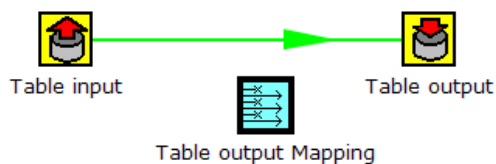


A dialog is then shown that helps you to determine which input fields goes to which table field:

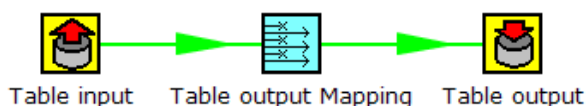


The selection of the target fields are done semi-automatic based on (parts of) the field name.

As a result of this dialog, a new Select Values step is generated:



We can simply place this step between the two and the mapping from input to output is done:



3.7. Safe mode

In cases where you are mixing the rows from various sources, you need to make sure that these row all have the same layout in all conditions. For this purpose, we added a “safe mode” option that is available in the Spoon logging window. When running in “safe mode”, the transformation will check every row that passes and will see if the layouts are all identical.

If a row is found that does not have the same layout as the first row, an error is thrown and the step and offending row are reported on.

***Note:** this option is also available in Pan with the safemode option.*

4. Chef

Although it may appear at first glance that not too much has changed, a lot of things were fixed and improved upon in version 2.3.0.

For a list of the changes that were done in the job entries, please see the corresponding chapter below.

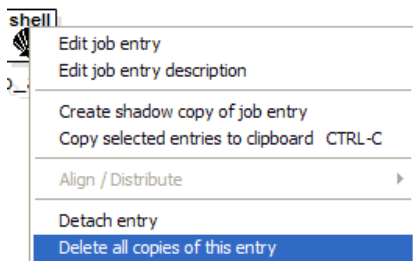
4.1. New logging window

As you can see below, the execution of the job entries is now shown in a hierarchical way using a tree. This allows you to track the processing more easily.

Job / Job Entry	Comment	Result	Reason	Nr	Log date
Process and move file	Start of job execution		start		2006/06/06 16:17:42
Job: Process and move file	Job entry started		start		2006/06/06 16:17:42
START	Job entry ended	Success		0	2006/06/06 16:17:42
Set TODAY variable	Job entry started		Followed unconditional link		2006/06/06 16:17:42
Set TODAY variable	Job entry ended	Success		1	2006/06/06 16:17:43
Process files from TODAY	Job entry started		Followed link after success!		2006/06/06 16:17:43
Process files from TODAY	Job entry ended	Success		2	2006/06/06 16:17:44
Send files by mail	Job entry started		Followed link after success!		2006/06/06 16:17:44
Send files by mail	Job entry ended	Success		2	2006/06/06 16:17:45
move_to_archive.bat	Job entry started		Followed link after success!		2006/06/06 16:17:45
move_to_archive.bat	Job entry ended	Success		4	2006/06/06 16:17:45
Job: Process and move file	Job execution ended	Success	end	4	2006/06/06 16:17:45

4.2. Delete job entry copies

To allow you to delete created job entries faster, you now also have a new “Delete all copies off this entry” feature.



5. Steps

5.1. Text file input

- Accept filenames from previous steps, together with the new step “Get file names” you can compose your own filenames, or get filenames from text-files, databases, etc.

Text file input

Step name: Text file input

File | Content | Error Handling | Filters | Fields

Accept filenames from previous step ☐

Field in the input to use as filename

Step to read filenames from

File or directory

Regular Expression

Selected files:

#	File/Directory	Wildcard	Required
001			

Show filename(s)... Show file content Show content from first data line

OK Preview rows Cancel

- Scroll-bars appear in this step to prevent options from being hidden

Text file input

Step name: Text file input

File | Content | Error Handling | Filters | Fields

Accept filenames from previous step ☒

Field in the input to use as filename

Step to read filenames from

File or directory

OK Preview rows Cancel

- Date parsing can be set to “lenient” or strict. In lenient mode, February 31st is converted to the corresponding day in March. In not in lenient mode, an error is thrown.

Text file input

Step name: Text file input

Error Handling

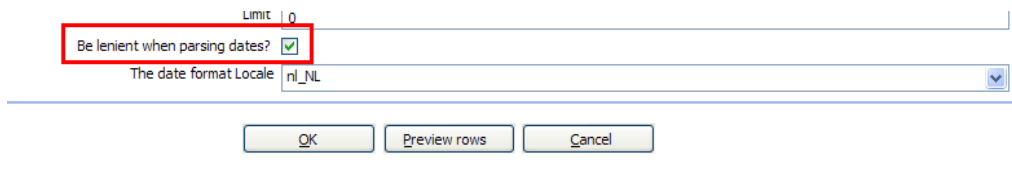
Limit: 0

Be lenient when parsing dates? ☒

The date format Locale: nl_NL

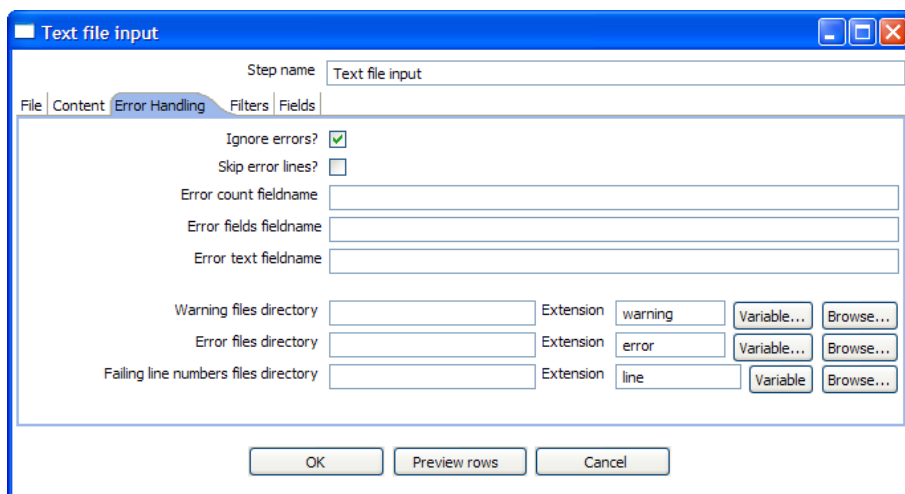
OK Preview rows Cancel

- The locale can be set for date formats, this is needed when parsing dates written in full in different locale:



LIMIT | 0
 Be lenient when parsing dates? ☒
 The date format Locale nl_NL
 OK Preview rows Cancel

- Then we added a whole list of extra features to make error handling better for automated text-file handling:



Step name Text file input
 File Content Error Handling Filters Fields
 Ignore errors? ☒
 Skip error lines? ☐
 Error count fieldname
 Error fields fieldname
 Error text fieldname
 Warning files directory Extension warning Variable... Browse...
 Error files directory Extension error Variable... Browse...
 Failing line numbers files directory Extension line Variable Browse...
 OK Preview rows Cancel

These features allow you to create ignore errors, send the failing field names to next steps or even create files with line numbers in it on which the error(s) occurred. When the transformation is replayed, the other lines can be ignored.

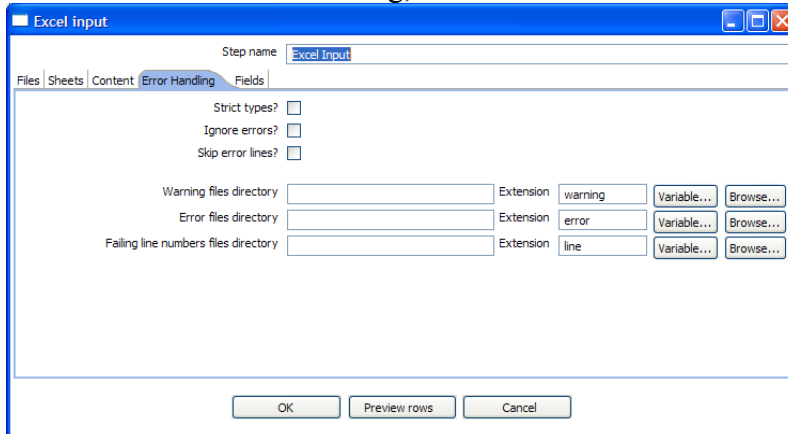
5.2. Get System Info

Even though at first glance nothing has changed here, it is now more convenient to add values from this step to a stream by simply giving this step input. The selected values will be added to the output.

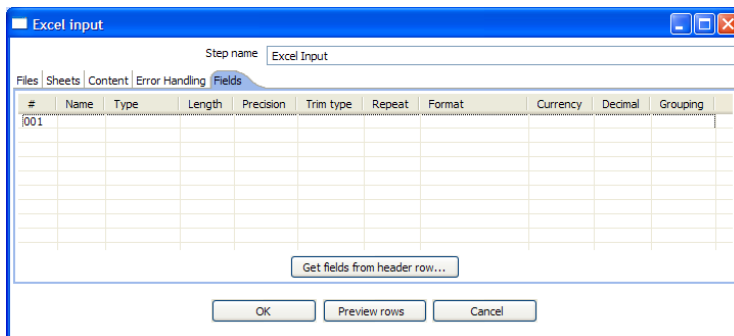
5.3. Excel Input

As reading excel files is something that's used a lot, this step got some attention as well.

- First there is the error handling, modeled after the Text File Input step:



- Because any data type can be found in a cell in a sheet in a worksheet, you need some kind of conversion logic. This is covered in the new Fields tab:



- Last but not least, Sven Boden added support for Formula fields. As it seems, the result of the formula is stored in the Excel file and you grab it without too much problems.

5.4. XML Input

- While reading excel files, stored in XML, James Dixon wanted to skip the header row(s). So this option was added just for that:

The screenshot shows the 'XML Input' dialog box with the 'Fields' tab active. The 'Nr of header rows to skip' field is highlighted with a red rectangle and contains the value '0'. Other fields include 'Include filename in output?', 'Filename fieldname', 'Rownum in output?', 'Rownum fieldname', and 'Limit'.

Location	#	Elements
001		

- Support was added for grabbing information from XML documents without repeating elements. In that case, you need to specify a single Element in "Location".
- Support was added for XML documents where all the information is stored in the Repeating (or Root) element. The special R= locator was added to allow you to grab this information. The "Get fields" button finds this information if it's present.

5.5. Table output

- When a row of data is written to a table, and a new key is generated in an auto-increment field, you might want to grab that generated key. You can do so now with this new feature:

The screenshot shows the 'Table output' dialog box with the following fields and options:

- Step name:
- Connection:
- Target table:
- Commit size:
- Truncate table: ☐
- Ignore insert errors: ☐
- Use batch update for inserts: ☒
- Partition data over tables: ☐
- Partitioning field:
- Partition data per month: ☒
- Partition data per day: ☐
- Is the name of the table defined in a field?: ☐
- Field that contains name of table:
- Store the tablename field: ☒
- Return auto-generated key:** ☐ (highlighted with a red rectangle)
- Name of auto-generated key field: (highlighted with a red rectangle)

Buttons at the bottom:

5.6. Insert / Update

- Sometimes you don't want to perform an update in a database table if the row is already present. That's the reason this option was added:

Step name: Insert / Update

Connection: [dropdown] [Edit...] [New...]

Target table: lookup table [Browse...]

Commit size: 100

Don't perform any updates: ☐

The key(s) to look up the value(s):

#	Table field	Comparator	Stream field1	Stream field2
001				

Get fields

Update fields:

#	Table field	Stream field	Update
001			Y

Get update fields

Ok SQL Afbreken

- The updates in the table can now also be specified on a field level.
- A number of smaller performance updates were done to this step as well. Also, a serious dead-lock issue was removed on certain databases because this step used more than 1 connection to do inserts, updates and lookups. If the same key passed more than once, this caused dead-locks in 2.2.2

5.7. Delete (new step)

- The delete step allows you to delete data from a database table:

[illegible]

5.8. Sort rows

- To speed up the sorting, you can specify the number of rows in the buffer, the more rows you can keep in the buffer, the faster sorting gets (it reduces I/O):

Sort rows

Step name: Sort rows

Sort directory: %%java.io.tmpdir%% Variable... Browse...

TMP-file prefix: out

Sort size (rows in memory): 5000

Fields:

#	Fieldname	Ascending
---	-----------	-----------

OK Get Fields Cancel

5.9. Value Mapper (New step)

- If you simply want to change a certain value to a different string value:

[illegible]

5.10. Dimension Lookup

- Mostly GUI changes here, trying to make it easier to use:

[illegible]

5.11. Combination Lookup

- Mostly GUI changes here, trying to make it easier to use.
- Making support for auto-increment fields more configurable.
- Support for caching rows of data.

The screenshot shows the 'Combination Lookup / Update' dialog box. The 'Step name' is 'Combination lookup/update 2'. The 'Connection' is set to a dropdown menu with 'Edit...' and 'New...' buttons. The 'Target table' is 'dim table name' with a 'Browse...' button. The 'Commit size' is '0' and the 'Cache size' is '9999'. Below these is a table for 'Key fields (to look up row in table):' with columns '#', 'Dimension field', and 'Field in stream'. The table has one row with '# 001'. Below the table is the 'Technical key field' section, which is 'technical/surrogate key field'. This section has a 'Creation of technical key' sub-section with three radio buttons: 'Use table maximum + 1' (selected), 'Use sequence', and 'Use auto increment field'. There are also checkboxes for 'Remove lookup fields?' and 'Use hashcode?'. At the bottom are buttons for 'OK', 'Get Fields', 'SQL', and 'Cancel'.

#	Dimension field	Field in stream
001		

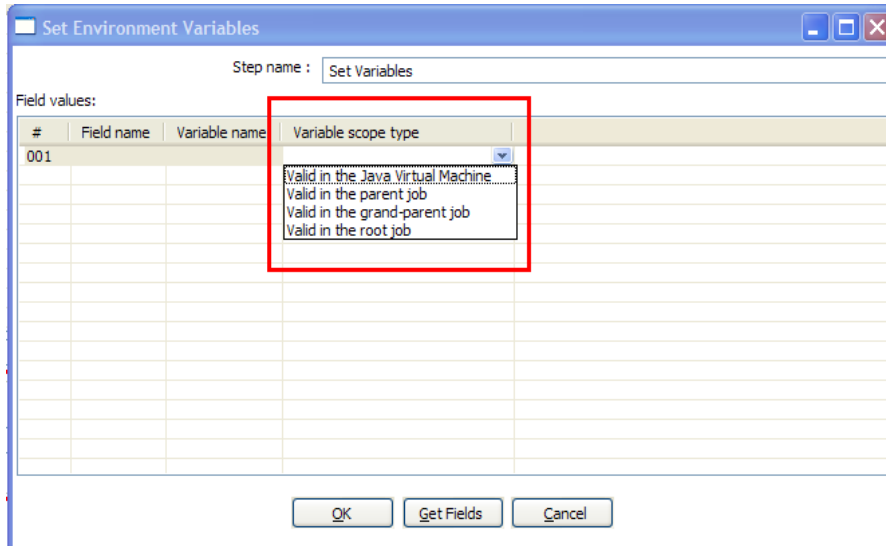
5.12. Get rows from result

- To make it easier to specify the meta-data of rows rows of data from previous transformations / jobs, you can simply specify the fields you are expecting. A Nicholas Goodman idea:

[illegible]

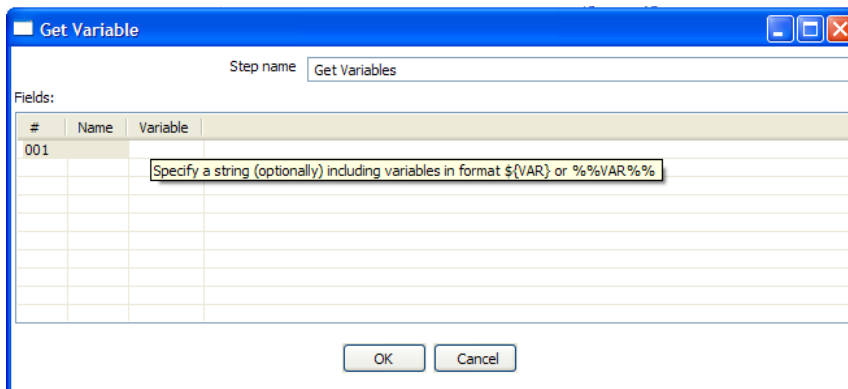
5.13. Set Variable (New step)

- Because of the dynamic nature of certain requirements (deployments etc), you might want to set variables also dynamically in a job. You now can do this with the Set Variables step:



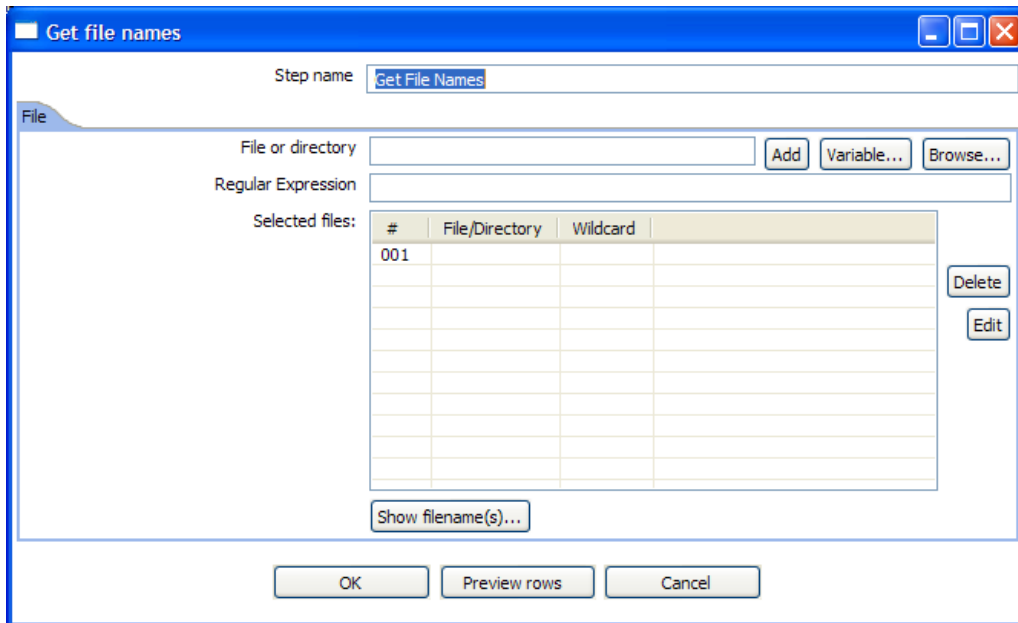
5.14. Get Variable (New step)

- If you want to add information from a variable to a stream of data:



5.15. Get file names (New step)

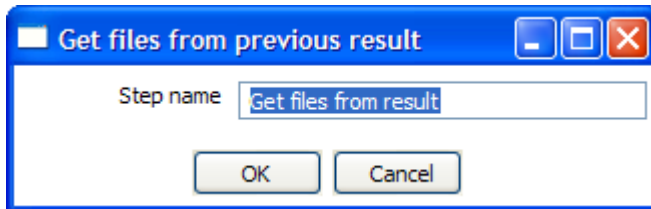
- File names are information too, so in case you want to grab a filename, just use this step:



- This code was “stolen” from Text File Input and works just the same. With the except that the generated filenames are simply sent to the next steps in the field named “filename”.
- You can use this to send the file names directly to Text File Input *or* (a more dynamic approach) send the filenames to the next “Job” job entry (in a Job in Chef) that sets a variable “FILENAME” and then you can process the file, move the file to an archive directory, mail the file, etc. This kind of functionality is very useful to replace shell scripting. Those have typically a high maintenance cost and are often platform specific.

5.16. Get files from result (New step)

- Every time a file gets processed, used or created in a transformation or a job, the details of the file, the job entry, the step, etc. is captured and added to the result. You can access this file information using this step:

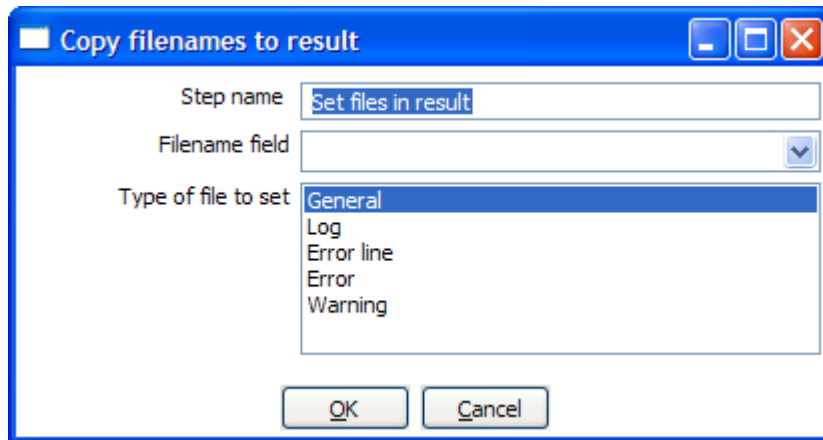


- These are the output fields:

Field name	Type	Example
type	String	Normal, Log, Error, Error-line, etc.
filename	String	somefile.txt
path	String	C:\Foo\Bar\somefile.txt
parentorigin	String	Process files transformation
origin	String	Text File Input
comment	String	Read by text file input
timestamp	Date	2006-06-23 12:34:56

5.17. Set files in result (New step)

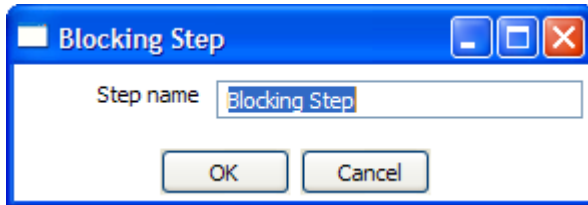
- So, in certain cases, to steer the list of files in the result ourself, we can use this step:



- For example the Mail job entry can use this list of files to attach to a mail, so perhaps you don't want all files sent, but only a certain selection. For this, you can create a transformation that sets exactly those files you want to attach.

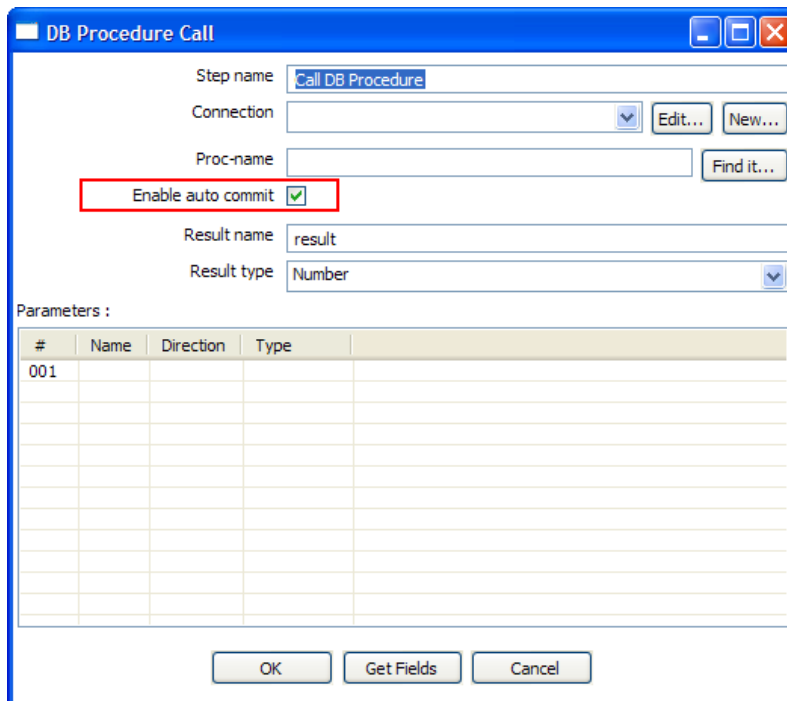
5.18. Blocking step (New step)

- This is again a very simple step. It blocks all output until the very last row was received from the previous step. This last row is then sent off to the next step. This step then knows that all previous steps have finished. You can use this for triggering custom plugin, stored procedures, java scripts, etc. :



5.19. Call Database Procedure

- We added a new flag here to help those people out that are modifying the database in the stored procedures. The connection in the previous versions was always set in auto-commit, meaning that any table update was persisted immediately. However, certain database types can't handle this. That's why we added the "Enable auto commit" flag, so that it can be optionally turned off. The default value is set to enabled to be backward compatible. If the flag is turned off, a single commit is executed after the last row was processed.



5.20. Injector (new step)

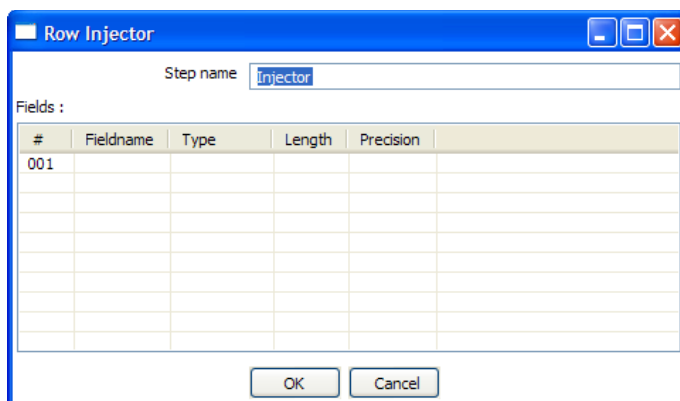
- Injector was created for those people that are developing special purpose transformations and want to “Inject” rows into the transformation using the Kettle API and Java. Among other things you can build “headless” transformations with it: transformations that have no input at design time: don't read from file or database.
- Here is some information on how to do it:
 - You can ask a Trans object for a RowProducer object
 - Also see the use case test in: package be.ibridge.kettle.test.rowproducer
 - Use this type of code:

```
Trans trans = new Trans(... TransMeta ...);
trans.prepareExecution(args);
RowProcuder rp = trans.addRowProducer(String stepname, int stepCopy);
```

- After that you start the threads in the transformation. Then you can inject the rows while the transformation is running

```
trans.startThreads();
...
rp.putRow(Row SomeRowYouHaveToInject);
...
```

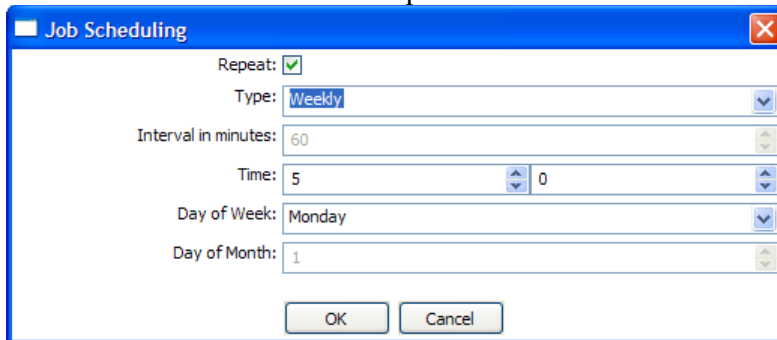
- As you can see from the screenshot below, you can specify the rows you are expecting to be injected. This makes it easier to build transformations because you have the meta-data at design time.



6. Job Entries

6.1. Start

- You can set the schedule of the job by simply specifying until when the “Start” job entry should block or if it needs to repeat:



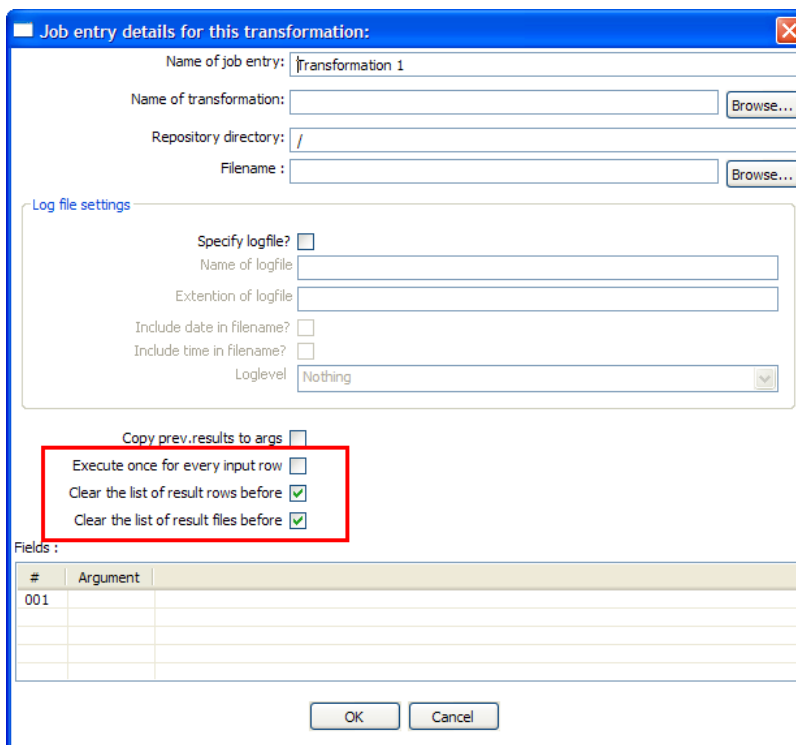
The "Job Scheduling" dialog box is shown with the following settings:

- Repeat: ☒
- Type: Weekly
- Interval in minutes: 60
- Time: 5:00
- Day of Week: Monday
- Day of Month: 1

Buttons: OK, Cancel

6.2. Transformation

- Support for “looping” has been added by allowing a transformation to be executed once for every input row:



The "Job entry details for this transformation" dialog box is shown with the following settings:

- Name of job entry: Transformation 1
- Name of transformation: (empty)
- Repository directory: /
- Filename: (empty)
- Log file settings:
 - Specify logfile: ☐
 - Name of logfile: (empty)
 - Extension of logfile: (empty)
 - Include date in filename: ☐
 - Include time in filename: ☐
 - Loglevel: Nothing
- Copy prev. results to args: ☐
- Execute once for every input row: ☐
- Clear the list of result rows before: ☒
- Clear the list of result files before: ☒

Fields:

#	Argument
001	

Buttons: OK, Cancel

- The (file-) name of the transformation can be specified using variables as well. This allows for easier deployment logic.

6.3. Job

- Support for “looping” has been added by allowing a job to be executed once for every input row:

[illegible]

- The (file-) name of the job can be specified using variables as well. This allows for easier deployment logic.

6.4. Shell

- Support for “looping” has been added by allowing a shell script to be executed once for every input row:

Job entry details for this shell script:

Name of shell script: Shell 1

Filename : Browse...

Log file settings

Specify logfile? ☐

Name of logfile

Extension of logfile

Include date in filename? ☐

Include time in filename? ☐

Loglevel: Nothing

Copy prev.results to args ☐

Execute once for every input row ☐

Fields:

#	Argument
001	

OK Cancel

- Also, output generated by the shell script (console output and error output) are captured and sent to the logging. This no longer blocks the execution of the job entry.
- On windows, scripts are now preceded by “CMD.EXE /C” (NT/XP/2000) or “COMMAND.COM /C” (95,98).
- The (file-) name of the shell script can be specified using variables as well. This allows for easier deployment logic.

6.5. Mail

- Support for authentication has been added, as well as (zipped) attachments.

Job mail details

Name of mail job entry: Send processed files by e-Mail

Destination address: dude@foobar.com

SMTP Server: smtp.foobar.com

Use authentication? ☒

Authentication user: auth.user

Authentication password: *****

Reply address: no-reply@foobar.com

Subject: [Data Warehouse : processed files]

Include date in message? ☒

Attach files to message? ☒

Select the result file types to attach: General, Log, Error line, Error, Warning

Zip files into a single archive? ☒

The zip filename: attachment.zip

Contact person: Mr. Dude Foobar

Contact Phone: 555-1234

Comment: If you receive this file, it means that all is well.

OK Cancel

- The full trace of the job execution prior to arriving at the mail job entry is also included in the mail as well as the content of the result. (number of lines read, written, etc.)
- All fields can be specified using environment variables as well. For example: `${DESTINATION_ADDRESS}`
- Specify multiple destination addresses separated with spaces.

7. Source code improvements

7.1. A few extra lines of code

Version 2.2.2 contains 177,450 lines of code.

Version 2.3.0 contains 213,489 lines of code, an increase of 36,039 lines.

7.2. Trackers

In the period between the release of version 2.2.2 until June 6th, 112 bugs were fixed and 21 feature requests from users were implemented.

7.3. Unit testing

To guard quality and backward compatibility in this release and for the future ones, we started with the implementation of several unit tests. Sven Boden, fellow Belgian is implementing tests for all core classes and has so far covered 5% of the complete Kettle code base.

7.4. Committers

ID / Commits	e-Mail	Name	Country	Work
audinchan 10	Audinchan (at) gmail.com	Chan Audin	CH	Repository importer fixes Various Text File Input fixes
berarma 129	Bernardo (at) tsolucio.com	Arlandis Bernardo	ES	A lot of i18n fixes. Various postgres database fixes Better architecture detection in .sh scripts
DennisVR 11	Dennis (at) iqit.be	Dennis Van Roeyen	B	Basic scheduler for Jobs Insert/update step: update field based
hoezx6r 40	Ahoesley (at) skyroadasp.com	Andrew Hoesley	USA	Reworked Job Entry Plugin system Fixed plugin loading actually Text File Input: added default for null values Text File Input: file handling for jobs
jbleuel 132	Jens.bleuel (at) proratio.de	Jens Bleuel	D	Database wizard fixes Database reserved words: DB2, AS/400, MS-Access, MS-SQL Server, Postgres Various bug fixes German translations and i18n fixes
jvanhent 455	johnny.vanhentenryk (at) ixor.be	Johnny Vanhentenryk	B	Error handling for Text File Input / Excel Input as well as replay functionality, various fixes, data lenient (TFI), log window history refresh...
qinhui99 472	Qinhui99 (at) hotmail.com	Tom Qin	CH	String externalization Translation into Simplified Chinese Various bug fixes
sboden 120	Svenboden (at) hotmail.com	Sven Boden	B	Wrote lots of Unit tests Code review, verifications & hardening Dimension & Combination lookup improvements and GUI changes Various bug fixes
vitoecn	Through: Qinhui99 (at)	Iv Vitoe	CH	Various i18n improvements

ID / Commits	e-Mail	Name	Country	Work
	hotmail.com			Translation into Simplified Chinese
wconroy 31	wconroy (at) gmail.com	Bo Conroy	USA	New Blocking step FTP: allow variables for regex and target dir. fix timeout. Added ValueSerializable data type Various bug fixes General Java advice and proposals
wdeclerc 47	Wim.DeClercq (at) ixor.be	Wim De Clercq	BE	Parameterizing database connections Implementation of kettle.properties file General java advice
MattCasters 2,434	mcasters (at) pentaho.org	Matt Casters	BE	Miscellaneous
TOTAL 3,919	12 developers		6 countries	

7.5. Other contributors

Disclaimer: *The list below is by no means complete. Many people files bug fixes and feature requests anonymously. Others were probably shamelessly bypassed. Please remember that this is not done on purpose but rather a result of the limited time we can spend on these type of documents. If you were one of the victims of this horrible policy, simply contact the maintainer of this document and you will be added immediately.*

The same is true for everyone that DOESN'T want to be on this list: just let us know.

- The entire crew at Pentaho for the many suggestions for improvements, bug reports and continued support.
- Etienne Dubé (etdube)
 - Fixed a complex rounding problem for numbers with double data type. Proposed the java code fix to solve it as well. (Tracker #1876)
 - Found out that the Select Values doesn't save the type correctly to XML (must have been in there forever) and proposed the fix as well. (Tracker #1875)
 - Proposed a fix for a problem with the Mapping step. (Tracker #1874)
- Christopher Fahey (chrostpher.fahey (at) comcast.net)
 - Proposed code to enhance Text File Input: look at the header to determine the *order* in which fields occur in the text file
 - Bug fix in Normaliser step: first=false flag was forgotten
- lokvin (Ren Yufei, purcr (at) hotmail.com)
 - Created a flash demo in Chinese
- Kettle partner Ixor, Johnny Vanhentenryk [jvanhent (at) gmail.com], Dennis Vanroeyen [dennis.vanroeyen (at) gmail.com]
 - Changes paid for by Mondial Assistance Group released into the Kettle code base for a project in Japan, I'm verifying the changes.
 - Most of the text file input and excel input error handling
 - various bug fixes
 - Replay functionality
 - JUnit tests and use case library for text file input and excel input steps with test files and transformations.

- Bernardo Arlandis Mañó [bernardo (at) tsolucio.com]
 - Proposes to help with localization
 - Wants to hire a professional company to do the Kettle translation to Spanish
 - I gave him an account and I'm helping him find his way around in the codebase.
 - Also wants to help with migration of localization from 3.0.1 to the trunk
- Pentaho partner Proratio: Jens Bleuel [jens.bleuel (at) proratio.de]
 - Registration of Kettle presentation for Linuxtag
 - Question answering on forum
 - Bug reporting
- Steven Lehr [lehrs (at) erau.edu]
 - Wrote most of next weeks Kettle tip: several gotcha's for first time users
- Ben Stroud [ben.stroud (at) waypointdatasolutions.com]
 - Fixing TextFileInputDialog GUI bug on GTK (Linux)
- Craig Moran [cmohan (at) shoppertrak.com]
 - Bug reporting and testing on the OSX platform (Spoon and particularly Chef)
- Catalin Pitis [cpitis (at) gmail.com]
 - Various feedbacks on the forum, suggestions and bug reports
- David Hurst [dthurst (at) cogniscape.com]
 - XSL Transform Suggestion
- Frank Schunig [Frank.Schunig (at) shd-online.de]
 - Bug reporting on exceptions in Spoon
- Gergő Bácskai [gbacscai (at) gmail.com]
 - Big help entering bug reports, feature reports and doing suggestions!
- Giyora Megido [giyoramegido (at) gmail.com]
 - Various bug reports and questions on the forum.
- Gunter Rombauts [gunter_rombauts (at) chicago-metallic.be]
 - Suggestions and bug reporting
- Nicholas Guzaldo [guzaldon (at) gmail.com]
 - Big help in all areas: feedback, bug reporting, forum, ...
- Hans Baier [hans.baier (at) soluzione.de]
 - Bug reporting on Copy Tables Wizard & Hypersonic
- Gavin Higgs [Gavin.Higgs (at) auspost.com.au]
 - Reporting DBCache bug & testing solution
- Andrew Hoesley [hoezx6r (at) gmail.com]
 - Various suggestions and bug reports
- Jaime Madrid [jaime.madrid (at) gmail.com]
 - Problem reporting and testing
- James Bradt [javaforge.x.bradtwurst (at) xoxy.net]
 - Reporting on MS-Access alter table problem & solution
- Jens Bleuel [jbleuel (at) proratio.de]
 - Bug reporting, testing, fixing, documenting, forum participation, ..., being a nice guy.
- Jesper Villadsen [JSV (at) maconomy.dk] (and colleagues :-))
 - Various suggestions, feedback and bug reports
- Jules Grengbondai [jgrengbondai (at) cincom.com]
 - Forum questions, suggestions, feedback
- Jocelyn Bernier [jbernier (at) dalcom-intl.com]
 - Enter options dialog bug report
- Jonathon Coombes [jon (at) cybersite.com.au]
 - Lots of feedback and bug reports on OSX, MySQL (5.1.x), Linux 64bit, etc.
 - Forum support, questions and feedback

- Mark Nuttall [mknuttall (at) gmail.com]
 - Suggestions on OpenAdapter stuff that we should probably pick up in the near future.
- Mark Tetrode [mark.tetrode (at) altitude.com]
 - Suggestions on GUI looks, sample dialogs, etc.
- Mason Sharp [msharp (at) extendb.com]
 - Added ExtenDB support (patch) as well as the JDBC driver for this database.
 - Suggestions on batch processing in the Insert/Update step.
- Matthew Walnock [matthew.walnock (at) gmail.com]
 - Suggestion (feature request) on extending user rights to repository directories.
- Matt Harbert [mharbert (at) divx.com]
 - Feature request for Mail job entry with secure authentication support.
- Youssef Mbaret [mrabetyoussef (at) gmail.com]
 - Big help on the forum
 - Plenty of feedback and bug reports
 - Wrote new “XML parsing by Value” plug-in that makes reading XML docs a lot faster.
 - Performance testing on PostgreSQL
- Olivier Clemens [olivier.clemens (at) mondial-assistance-group.com] and colleagues at Mondial
 - Feature requests, bug reports & much appreciated continued support!
- Paul Lewen [Plewin (at) s-und-n.de] and colleagues at S+D
 - Helping detect and solve a serious performance problem
- Raoni Guerra [raoniguerra (at) hotmail.com]
 - Bug reporting on PostgreSQL
- Rich Cristy [rchristy (at) skyroadasp.com]
 - Feedback on Text File Input, compilation issues, testing and bug reporting on the forum.
- Roland Bouman [[roland.bouman (at) gmail.com]
 - Dutch translator for Kettle
 - Documentation updates & reviewing
 - Doing a very nice blog on Kettle & Pentaho:
 - <http://rpbouman.blogspot.com/2006/06/pentaho-data-integration-kettle-turns.html>
 - Helping out on the forum, suggestions, bug reports, etc.
- Hassan Samatar [sahass78 (at) yahoo.fr]
 - French translator for Kettle
 - Huge help on the forum
 - Bug reporting, feature requests, etc.
- Sergio Ayuso Iniesta [sayuso (at) satec.es]
 - Bug reporting on the Combination Lookup step

- Seth Ladd [seth (at) ehawaii.gov]
 - Feature request on hard returns in enclosed Text Files fields.
 - Javascript step feedback on the forum
- Shuo Wang [shuo_wang_2000 (at) yahoo.com]
 - Various feedback on the forum, bug reports and testing
- Sven Boden [Sven.Boden (at) pandora.be]
 - Wrote unit test cases
 - Did java code reviews
 - bug reporting & fixing
 - Wrote a lots of improvements to the code.
 - Documentation review & updates
 - Big help on the forum answering questions
 - Java advice & test code
- Marc D Swingler [Swingler.MarcD (at) ensco.com]
 - Bug & feature report on data type handling on SQL Server
 - MySQL testing, Feature request for identity fields, ...
- Tom Qin [qinhui99 (at) hotmail.com]
 - Simplified Chinese translator for Kettle
 - Big help on the project, the forum, ...
- Zachary Zeus [zachary (at) polonious.com.au]
 - Helping out on OSX and reporting on the Sorting a column Kettle crash on that platform.
- Lee Bruise [workformydream (at) yahoo.com]
 - Found a problem in Table Input step with variable replacement and suggested the fix for it.