

KNIME File Handling Guide

KNIME AG, Zurich, Switzerland Version 4.4 (last updated on 2022-12-01)



Table of Contents

Introduction
Basic concepts about file systems
Working directory
Hidden files
Path syntax
KNIME Analytics Platform and file systems
Standard file systems
Connected file systems9
Read and write from or to a connected file system
Reader nodes
Writer nodes
Path data cell and flow variable
Creating path data cells
Manipulating path data cells
Creating path flow variables
String and path type conversion
File Folder Utility nodes
Table based input Utility nodes
Compatibility and migration
How to work with workflows that contain both old and new file handling nodes $\ldots \ldots$ 31
How to migrate your workflows from old to new file handling nodes

Introduction

With the move towards cloud and hybrid environments we had to rethink and rewrite the existing file handling infrastructure in KNIME Analytics Platform to provide our users with a better user experience.

With KNIME Analytics Platform release 4.3 we introduced a new file handling framework thanks to which you can migrate workflows between file systems or manage various file systems within the same workflow in a more convenient way.

In this guide the following topics are covered:

- Basic concepts concerning file systems
- · How to access different file systems from within KNIME Analytics Platform
- How to read and write from and to different file systems and conveniently transform and adjust your data tables when importing them into your workflow
- The new Path type and how to use it within the nodes that are build on the basis of the file handling framework
- Finally, in the Compatibility and migration section you will find more detailed information about:
 - How to distinguish between the old and new file handling nodes
 - How to work with workflows that contain both old and new file handling nodes
 - How to migrate your workflows from old to new file handling nodes.

Basic concepts about file systems

In general, a file system is a process that manages how and where data is stored, accessed and managed.

In KNIME Analytics Platform a file system can be seen as a forest of trees where a folder constitutes an inner tree node, while a file or an empty folder are the leaves.

Working directory

A working directory is a folder used by KNIME nodes to disambiguate relative paths. Every file system will have a working directory whether it is explicitly configured or implicitly.

Hidden files

Within KNIME Analytics Platform hidden files and folders are not displayed when browsing a file system. However they can be referenced knowing the path. Hidden files currently only exist for the local file systems in KNIME Analytics Platform:

- On Linux and MacOS their filename starts with a dot "."
- On Windows instead they are treated as regular files and folders

Path syntax

A path is a string that identifies a file or folder position within a file system. The path syntax depends on the file system, e.g. a Windows local file system might look like C:\Users\username\file.txt, while on Linux and most other file systems in KNIME Analytics Platform might look like /folder1/folder2/file.txt.

Paths can be distinguished in:

- Absolute: An absolute path uniquely identifies a file or folder. It starts always with a file system root.
- Relative: A relative path does not identifies one particular file or folder. It is used to identify a file or folder relative to an absolute path.

KNIME Analytics Platform and file systems

Different file systems are available to use with KNIME Analytics Platform. Reader and writer nodes are able to work with all supported file systems.

File systems within KNIME Analytics Platform can be divided into two main categories:

- Standard file systems
- Connected file systems

Standard file systems

Standard file systems are available at any time meaning they do not need a connector node to connect to it.

Their working directory is pre-configured and does not need to be explicitly specified.

To use a reader node to read a file from a standard file system drag and drop the reader node for the file type you want to read, e.g. CSV Reader for a .csv file, into the workflow editor from the node repository.

Right click the node and choose *Configure...* from the context menu. In the *Input location* pane under the tab *Settings* you can choose the file system you want to read from in a drop-down menu.

A Dialog	- 10:63 - CSV Rea	der									_		×
File													
		Advanced Settings	Limit Rows	Encoding	Flow Variables	Job Manager Selection	Memory Po	olicy					
-Input loc													^
Read from	n Local File Syster Local File Syster												
Mode	Mountpoint	·											
File	Relative to Custom/KNIME	JRL								~	Browse	V.	
	🔀 Please speci											0	
-Reader o		iy a paul.											
Format													
Au	todetect format	\$											
	Column delimiter	\n F	low delimiter										
	Column delimiter	Vi I	cow demniter										
•	Quote char	- (Quote escape d	har									
	1 .												
#	Comment char												
Ha	is column header	Has rov	v ID										
Su Su	pport short data ro	ws Prepen	d file index to re	w ID									
Preview													
🚫 Plea	se specify a path.												
													~
								ОК	Apply	Car	ncel		

The following standard file systems are available in KNIME Analytics Platform:

- · Local file system
- Mountpoint
- Relative to
 - Current workflow
 - Current mountpoint
 - Current workflow data area
- Custom/KNIME URL

Local file system

When reading from *Local File System* the path syntax to use will be dependent on the system on which the workflow is executing, i.e. if Windows or UNIX operative system.

The working directory will be implicit and will correspond to the system root directory.

tings Transf	ormation Ad	vanced Settings	Limit Rows	Encoding Flo	w Variables J	ob Manager Se	election Men	nory Policy		
put location —										
ad from Loca	al File System	\sim								
ide 💿 Fi	le 🔵 Files in fo	older								
e C:\µ	lsers \Daria \Desk	top\adult.csv						✓ Browner Stresser Stresse	wse	
										_
eader options -										_
ormat										
Autodetect	format 🕴	🔅 📀 Succe	essfully autode	etected!						
, Column	n delimiter	\n Ro	w delimiter							
Quote	char		unto occorro d							
Quote	criar	Q	uote escape d	la						
# Comme	ent char									
# Comme										
# Comme	and choir									
# Comme		Has row 1	ID							
Has colum	n header									
Has colum			ID file index to ro	ow ID						
Has column	n header			ow ID						
Has column	n header ort data rows	Prepend	file index to ro							
Has column	n header ort data rows		file index to ro		ed Settings' tab					
Has column	n header ort data rows	Prepend	file index to ro		ed Settings' tab					
Has column	n header ort data rows	Prepend	file index to ro		ed Settings' tab		S occupa	S relation	S race	
Has column Support sh eview The suggest	n header ort data rows ed column type:	Prepend	file index to ro	only. See 'Advanc	-			S relation	S race White	
Has column Has column Support sh eview The suggest Row ID	n header ort data rows ed column type:	Prepend s are based on the S workclass State-gov	file index to ro first 50 rows o	only. See 'Advanc	educati	S marital Never-married				
Has column Has column Support sh eview The suggest Row ID Row0	n header ort data rows ed column types 1 age 39	Prepend s are based on the S workclass State-gov	file index to ro first 50 rows o	only. See 'Advanc S education Bachelors	educati	S marital Never-married	Adm-clerical	Not-in-family Husband	White	
Has column Has column Support sh eview The suggest Row ID Row0 Row1	n header ort data rows ed column types 1 age 39 50	Prepend s are based on the S workclass State-gov Self-emp-no	file index to ro first 50 rows o 1 fnlwgt 77516 83311	only. See 'Advanc S education Bachelors Bachelors	educati 13 13	S marital Never-married Married-civ Divorced	Adm-clerical Exec-manag	Not-in-family Husband Not-in-family	White White	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2	n header ort data rows ed column types 1 age 39 50 38	Prepend s are based on the S workclass State-gov Self-emp-no Private	file index to ro first 50 rows of 1 fnlwgt 77516 83311 215646	only. See 'Advanc S education Bachelors Bachelors HS-grad	educati 13 13 9	S marital Never-married Married-civ Divorced	Adm-clerical Exec-manag Handlers-cle Handlers-cle	Not-in-family Husband Not-in-family	White White White	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3	ed column types a header ed column types 1 age 39 50 38 53 28 37	Prepend s are based on the S workclass State-gov Self-emp-no Private Private	file index to ro first 50 rows of 1 fnlwgt 77516 83311 215646 234721	only. See 'Advanc S education Bachelors Bachelors HS-grad 11th	educati 13 13 9 7 13 14	S marital Never-married Married-civ Divorced Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Handlers-cle	Not-in-family Husband Not-in-family Husband	White White White Black	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3 Row4	n header ort data rows ed column types 1 age 39 50 38 53 28 37 49	Prepend s are based on the S workclass State-gov Self-emp-no Private Private Private	file index to ro first 50 rows of 1 fnlwgt 77516 83311 215646 234721 338409	only. See 'Advanc S education Bachelors Bachelors HS-grad 11th Bachelors	educati 13 13 9 7 13 14 5	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife	White White White Black Black	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3 Row4 Row5 Row6 Row7	a header nort data rows ed column types 39 50 38 53 28 37 49 52	Prepend s are based on the S workclass State-gov Self-emp-no Private Private Private Private Private Self-emp-no	file index to ro first 50 rows o [1] fnlwgt 77516 83311 215646 234721 338409 284582 160187 209642	only. See 'Advanc S education Bachelors Bachelors HS-grad 11th Bachelors Masters	educati 13 13 9 7 13 14 5 9	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Married-spo	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family	White White White Black Black White	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3 Row3 Row4 Row5 Row6 Row7 Row8	a header ort data rows ed column types 39 50 38 53 28 37 49 52 31	Prepend s are based on the S are based on the S tate-gov Self-emp-no Private Private Private Private Private Private	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781	only. See 'Advance Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad Masters	educati 13 13 9 7 13 14 5 9 14	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Married-spo	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family	White White White Black Black White Black White White	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3 Row3 Row4 Row5 Row6 Row7 Row8 Row9	a header ort data rows ed column types 39 50 38 53 28 37 49 52 31 42	Prepend s are based on the s are based on the S workclass State-gov Self-emp-no Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449	only. See 'Advanc S education Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad	educati 13 13 9 7 13 14 5 9 14 13	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Married-civ Never-married Married-civ	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Not-in-family Husband	White White White Black Black White Black White White White	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3 Row3 Row4 Row5 Row6 Row7 Row8 Row9 Row10	a header ort data rows ed column types 39 50 38 53 28 37 49 52 31 42 37	Prepend s are based on the s are based on the S workclass State-gov Self-emp-no Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 280464	only. See 'Advance Bachelors Bachelors HS-grad 11th Bachelors 9th HS-grad Masters 9th HS-grad Masters Bachelors Some-college	educati 13 13 13 13 14 5 9 14 13 14 13 14 10	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Never-married Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Husband Husband	White White Black Black White Black White White Black Black	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3 Row3 Row3 Row4 Row3 Row4 Row5 Row6 Row7 Row6 Row7 Row8 Row9 Row10 Row10 Row10	a header ort data rows ed column types 39 50 38 53 28 37 49 52 31 42 37 30	Prepend s are based on the S are based on the S tate-gov Self-emp-no Private Private Private Private Private Private Private Private Private Self-emp-no Private Self-emp-no	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 280464 141297	See 'Advance' Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad Masters Bachelors Some-college Bachelors	educati 13 13 7 13 14 5 9 14 13 10 13	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-spo Married-civ Never-married Married-civ Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag Exec-manag Prof-specialty	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Husband Husband Husband	White White Black Black White Black White White Black Asian-Pac-1	
Has column Support sh eview The suggest Row ID Row ID Row 1 Row 2 Row 3 Row 4 Row 5 Row 4 Row 5 Row 6 Row 7 Row 8 Row 7 Row 8 Row 9 Row 10 Row	a header ort data rows ed column types 39 50 38 53 28 37 49 52 31 42 37 30 23	Prepend s are based on the S are based on the S tate-gov Self-emp-no Private Private Private Private Private Self-emp-no Private Private Self-emp-no Private Self-emp-no Private Self-emp-no	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 280464 141297 122272	Inly. See 'Advance Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad Masters Bachelors Some-college Bachelors Bachelors	I educati 13 13 13 9 7 13 14 5 9 14 13 10 13 13	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Married-civ Never-married Married-civ Married-civ Married-civ Married-civ Married-civ Never-married	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag Prof-specialty Adm-clerical	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Husband Husband Husband Own-child	White White Black Black White Black White White Black Asian-Pac-1 White	
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3 Row4 Row5 Row4 Row5 Row6 Row7 Row6 Row7 Row8 Row9 Row10 Row10 Row12 Row13	a header ort data rows ed column types 39 50 38 53 28 37 49 52 31 42 37 30 23 32	Prepend s are based on the S are based on the S tate-gov Self-emp-no Private Private Private Private Self-emp-no Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 280464 141297 122272 205019	Ith Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad Masters Bachelors Some-college Bachelors Bachelors Bachelors	I educati 13 13 13 14 5 9 14 13 10 13 13 12	S marital Never-married Married-civ Married-civ Married-civ Married-civ Married-civ Never-married Married-civ Married-civ Married-civ Married-civ Never-married Never-married	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag Prof-specialty Adm-clerical Sales	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Husband Husband Husband Own-child Not-in-family	White White Black Black White Black White White Black Asian-Pac-1 White Black	_
Has column Support sh eview Row ID Row0 Row1 Row2 Row3 Row4 Row5 Row6 Row7 Row8 Row9 Row10 Row11 Row11 Row12 Row13 Row14 Row13 Row14 Row1	n header nort data rows ed column types 39 50 38 53 28 37 49 52 31 42 37 30 23 30 23 32 40	Prepend s are based on the S are based on the S are based on the State-gov Self-emp-no Private Private Private Private Private Self-emp-no Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 280464 141297 122272 205019 121772	Inly. See 'Advance Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad Masters Bachelors Bachelors Bachelors Bachelors Bachelors Assoc-acdm Assoc-voc	I educati 13 13 13 14 5 9 14 13 10 13 13 12 11 12	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag Prof-specialty Adm-clerical Sales	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Husband Husband Husband Own-child	White White Black Black White Black White White Black Asian-Pac-1 White	_
Has column Support sh eview The suggest Row ID Row0 Row1 Row2 Row3 Row4 Row3 Row4 Row5 Row6 Row7 Row8 Row9 Row10 Row10 Row11 Row12 Row13	a header ort data rows ed column types 39 50 38 53 28 37 49 52 31 42 37 30 23 32	Prepend s are based on the S are based on the S tate-gov Self-emp-no Private Private Private Private Self-emp-no Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 280464 141297 122272 205019	Ith Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad Masters Bachelors Some-college Bachelors Bachelors Bachelors	I educati 13 13 13 14 5 9 14 13 10 13 13 12	S marital Never-married Married-civ Married-civ Married-civ Married-civ Married-civ Never-married Married-civ Married-civ Married-civ Married-civ Never-married Never-married	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag Prof-specialty Adm-clerical Sales	Not-in-family Husband Not-in-family Husband Wife Not-in-family Husband Not-in-family Husband Husband Husband Own-child Not-in-family Husband	White White Black Black White Black White White Black Asian-Pac-1 White Black Asian-Pac-2	_
Has column Support sh eview Row ID Row0 Row1 Row2 Row3 Row4 Row5 Row6 Row7 Row8 Row9 Row10 Row11 Row11 Row12 Row13 Row14 Row13 Row14 Row1	n header ort data rows ed column types 39 50 38 53 28 37 49 52 31 42 37 30 23 30 23 32 40	Prepend s are based on the S are based on the S are based on the State-gov Self-emp-no Private Private Private Private Private Self-emp-no Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private Private	file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 280464 141297 122272 205019 121772	Inly. See 'Advance Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad Masters Bachelors Bachelors Bachelors Bachelors Bachelors Assoc-acdm Assoc-voc	I educati 13 13 13 14 5 9 14 13 10 13 13 12 11 12	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag Prof-specialty Adm-clerical Sales	Not-in-family Husband Not-in-family Husband Wife Not-in-family Husband Not-in-family Husband Husband Husband Own-child Not-in-family Husband	White White Black Black White Black White White Black Asian-Pac-1 White Black Asian-Pac-2	Is

1

You can also have access to network shares supported by your operating system, via the local file system within KNIME Analytics Platform.

Please be aware that when executing a workflow on KNIME Server version 4.11 or higher the local file system access is disabled for security reasons. The KNIME Server administrators can activate it, but this is not recommended. For more information about this please refer to the KNIME Server Administration Guide.

Mountpoint

With the *Mountpoint* option you will have access to KNIME mount points, such as LOCAL, your KNIME Server mount points, if any, and the KNIME Hub. You have to be logged in to the specific mountpoint to have access to it.

The path syntax will be UNIX-like, i.e. /folder1/folder2/file.txt and relative to the implicit working directory, which corresponds to the root of the mountpoint.

tings T	e	1 10 11	11.00	E 11 EL			1 12 14			
out location -	stormation Ac	lvanced Settings	Limit Rows	Encoding Flo	w Variables J	ob Manager Se	lection Mem	lory Policy		
ad from Mo	untpoint	✓ <mou< p=""></mou<>	ntpoint ID>							
		-								
de 💿 F	File O Files in f	older								
/Us	ers/daria.tombo	lelli/demo/adult.csv						~	Browse	
Ľ										
ader options										
ormat										
Autodeteo	t format	¢								
	[
, Colum	n delimiter	\n Ro	ow delimiter							
T Ouetr	char	· 0	into oncono d							
• Quote	e char	" Qu	uote escape d	har						
Quote	e char	* Qu	uote escape d	har						
	e char Ient char	* Qi	uote escape d	har						
		*Q	uote escape d	har						
	ent char	Qu Has row		har						
# Comm	ent char In header	Has row	ID							
# Comm	ent char	Has row								
# Comm	ent char In header	Has row	ID							
# Comm Has colur Support s	eent char nn header hort data rows	Has row	ID file index to re	ow ID	ed Settinne' tak					
# Comm Has colur Support s	eent char nn header hort data rows	Has row	ID file index to re	ow ID	ed Settings' tab					
# Comm Has colum Support s eview The sugges	ient char in header hort data rows ited column type	Has row Prepend	ID file index to ro first 50 rows o	ow ID only. See 'Advanc						
# Comm Has colum Support s eview The sugges Row ID	ent char In header hort data rows ited column type	Has row Prepend s are based on the S workclass	ID file index to ro first 50 rows o	ow ID only. See 'Advanc	educati	S marital	S occupa	S relation	S race	
# Comm Has colum Support s eview The sugges Row ID Row0	ent char In header hort data rows ited column type	Has row Prepend s are based on the S workclass State-gov	ID file index to ro first 50 rows o	ow ID only. See 'Advanc S education Bachelors	educati	S marital Never-married	Adm-clerical	Not-in-family	White	
# Comm Has colum Support s view The sugges Row ID Row0 Row1	ent char In header hort data rows ited column type 1 age 39 50	Has row Prepend s are based on the S workclass State-gov Self-emp-no	ID file index to ro first 50 rows of 1 fnlwgt 77516 83311	ow ID only. See 'Advanc S education Bachelors Bachelors	educati 13 13	S marital Never-married Married-civ	Adm-clerical Exec-manag	Not-in-family Husband	White White	N
# Comm Has colum Support s view The sugges Row ID Row0 Row1 Row2	ent char In header hort data rows ited column type 1 age 39 50 38	Has row Prepend s are based on the S workclass State-gov Self-emp-no Private	ID file index to ro first 50 rows of [] fnlwgt 77516 83311 215646	ow ID only. See 'Advanc S education Bachelors Bachelors HS-grad	educati 13 13 9	S marital Never-married Married-civ Divorced	Adm-clerical Exec-manag Handlers-cle	Not-in-family Husband Not-in-family	White White White	N
# Comm Has colur Gupport s View The sugges Row ID Row0 Row1 Row2 Row3	ent char In header hort data rows ited column type 1 age 39 50 38 53	Has row Prepend s are based on the S workclass State-gov Self-emp-no Private Private	ID file index to ro first 50 rows of 1 fnlwgt 77516 83311 215646 234721	ow ID only. See 'Advance S education Bachelors Bachelors HS-grad 11th	educati 13 13 9 7	S marital Never-married Married-civ Divorced Married-civ	Adm-clerical Exec-manag Handlers-cle Handlers-cle	Not-in-family Husband Not-in-family Husband	White White White Black	N
# Comm Has colur Gupport s View The sugges Row ID Row0 Row1 Row2 Row3 Row4	ent char in header hort data rows sted column type [] age 39 50 38 53 28	Has row Prepend s are based on the S workclass State-gov Self-emp-no Private Private Private	ID file index to ro first 50 rows of 1 fnlwgt 77516 83311 215646 234721 338409	ow ID only. See 'Advance S education Bachelors Bachelors HS-grad 11th Bachelors	educati 13 13 9 7 13	S marital Never-married Married-civ Divorced Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Handlers-cle Prof-specialty	Not-in-family Husband Not-in-family Husband Wife	White White White Black Black	N N F
# Comm Has colum Support s view The sugges Row ID Row0 Row1 Row2 Row3 Row4 Row5	ent char in header hort data rows sted column type 39 50 38 53 28 37	Has row Prepend s are based on the S workclass State-gov Self-emp-no Private Private Private Private	ID file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582	ow ID only. See 'Advance S education Bachelors HS-grad 11th Bachelors Masters	educati 13 13 9 7 13 14	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Handlers-cle Prof-specialty Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife	White White Black Black White	N N F
# Comm Has colum Has colum Support s eview The sugges Row ID Row0 Row1 Row2 Row2 Row3 Row4 Row5 Row6	ent char nn header hort data rows ited column type 39 50 38 53 28 37 49	Has row Prepend s are based on the S are based on the Private Private Private Private Private	ID file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187	ow ID only. See 'Advance S education Bachelors HS-grad 11th Bachelors Masters 9th	educati 13 13 9 7 13 14 5	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Married-spo	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family	White White Black Black White Black	N N F F
# Comm Has colum Has colum Support s view The sugges Row ID Row0 Row1 Row2 Row3 Row3 Row4 Row5 Row6 Row7	ent char in header hort data rows ited column type 39 50 38 53 28 37 49 52	Has row Prepend s are based on the S workclass State-gov Self-emp-no Private Private Private Private Private Self-emp-no	ID file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642	ow ID only. See 'Advanc S education Bachelors Bachelors HS-grad HS-grad	educati 13 13 9 7 13 14 5 9	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-civ Married-civ	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband	White White Black Black White Black White	
# Comm Has colum Has colum Support s view The sugges Row ID Row0 Row1 Row2 Row3 Row4 Row5 Row4 Row5 Row4 Row5 Row6 Row7 Row8	ent char in header hort data rows ited column type 1 age 39 50 38 53 28 37 49 52 31	Has row Prepend s are based on the S workclass State-gov Self-emp-no Private Private Private Private Private Self-emp-no Private	ID file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781	ow ID only. See 'Advance S education Bachelors Bachelors HS-grad Masters 9th HS-grad Masters	educati 13 13 13 7 7 13 14 5 9 14 14	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-spo Married-civ Never-married	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Not-in-family	White White White Black Black White Black White White	N N F F
# Comm Has colum Has colum Support s eview The sugges Row ID Row0 Row1 Row2 Row3 Row4 Row5 Row5 Row6 Row7 Row8 Row9	ent char nn header hort data rows ted column type 1 age 39 50 38 53 28 37 49 52 31 42	Has row Prepend s are based on the S workclass State-gov Self-emp-no Private Private Private Private Self-emp-no Private Private Private	ID file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781 159449	ow ID only. See 'Advance Bachelors Bachelors HS-grad 11th Bachelors Masters 9th HS-grad Masters Bachelors	educati 13 13 13 9 7 7 13 14 5 9 14 13 14 13	S marital Never-married Married-civ Married-civ Married-civ Married-civ Never-married Married-civ	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Not-in-family Husband	White White White Black Black White Black White White White	
# Comm Has colum Has colum Support s view The sugges Row ID Row0 Row1 Row2 Row3 Row4 Row5 Row4 Row5 Row4 Row5 Row6 Row7 Row8	ent char in header hort data rows ited column type 1 age 39 50 38 53 28 37 49 52 31	Has row Prepend s are based on the S workclass State-gov Self-emp-no Private Private Private Private Private Self-emp-no Private	ID file index to ro first 50 rows of 77516 83311 215646 234721 338409 284582 160187 209642 45781	ow ID only. See 'Advance S education Bachelors Bachelors HS-grad Masters 9th HS-grad Masters	educati 13 13 13 7 7 13 14 5 9 14 14	S marital Never-married Married-civ Divorced Married-civ Married-civ Married-spo Married-civ Never-married	Adm-clerical Exec-manag Handlers-cle Prof-specialty Exec-manag Other-service Exec-manag Prof-specialty Exec-manag Exec-manag	Not-in-family Husband Not-in-family Husband Wife Wife Not-in-family Husband Not-in-family Husband	White White White Black Black White Black White White	

Please note that workflows inside the mountpoints are treated as files, so it is not possible to read or write files inside a workflow.

Relative to

With the Relative to option you will have access to three different file systems:

- *Current mountpoint* and *Current workflow*: The file system corresponds to the mountpoint where the currently executing workflow is located. The working directory is implicit and it is:
 - Current mountpoint: The working directory corresponds to the root of the mountpoint
 - *Current workflow*: The working directory corresponds to the path of the workflow in the mountpoint, e.g. /workflow_group/my_workflow.
- *Current workflow data area*: This file system is dedicated to and accessible by the currently executing workflow. Data are physically stored inside the workflow and are copied, moved or deleted together with the workflow.

All the paths used with the option *Relative to* are of the type folder/file and they must be relative paths.

ttings Transf	ormation A									
nput location —		dvanced Settings	Limit Kows	Encoding Flo	w Variables J	lob Manager Se	election Mem	ory Policy		
·										
ead from Rela	tive to	 Current wor 	rkflow data are	ea 🗸						
	0 =1 -1									
ode 💿 Fi	le 🔘 Files in f	folder								
ie adult	t.csv						· · · · · · · · · · · · · · · · · · ·	Browse.		1
									•	
leader options –										
Format										٦
Autodetect	format	\$								
, Column	delimiter	\n Ro	ow delimiter							
, Column	delimiter	\n Ro	ow delimiter							
, Column		••	ow delimiter uote escape d	har						
		••		har						
T Quote		••		har						
Quote	char ent char	• Q	uote escape d	har						
T Quote	char ent char	••	uote escape d	har						
Quote Quote # Comme Has column	char ent char	The second secon	uote escape d							
Quote Quote Has column Support sh	char ent char n header	The second secon	uote escape d							
# Comme # Comme Has column Support sh	char ent char header ort data rows	The second secon	uote escape d ID file index to ro	ow ID						
# Comme # Comme Has column Support sh	char ent char header ort data rows	The second secon	uote escape d ID file index to ro	ow ID	ed Settings' tab					
# Comme # Comme Has column Support sh	char ent char header ort data rows	The second secon	ID file index to ro first 50 rows o	ow ID	ed Settings' tab).				
# Comme # Comme Has column Support sh	char ent char header ort data rows	The second secon	uote escape d ID file index to ro	ow ID		o.	S occupa	S relation	S race	
Quote of the second secon	char ent char header ort data rows ed column type []] age 39	Has row Prepend es are based on the S workclass State-gov	ID file index to ro first 50 rows o	ow ID only. See 'Advanc S education Bachelors	educati	S marital Never-married	Adm-clerical	Not-in-family	White	^
Quote of the second secon	char ent char n header ort data rows ed column type []] age 39 50	Has row Has row Prepend es are based on the S workclass State-gov Self-emp-no	ID file index to ro first 50 rows o 1 fnlwgt 77516 83311	ow ID only. See 'Advanc S education Bachelors Bachelors	educati 13 13	S marital Never-married Married-civ	Adm-clerical Exec-manag	Not-in-family Husband	White White	^
Quote of the second secon	char ent char n header ort data rows ed column type 39 50 38	Has row Has row Prepend es are based on the S workclass State-gov Self-emp-no Private	ID file index to ro first 50 rows of 1 fnlwgt 77516 83311 215646	ow ID only. See 'Advance S education Bachelors Bachelors HS-grad	I educati 13 13 9	S marital Never-married Married-civ Divorced	Adm-clerical Exec-manag Handlers-cle	Not-in-family Husband Not-in-family	White White White	^
Quote of the second secon	char ent char n header ort data rows ed column type []] age 39 50	Has row Has row Prepend es are based on the S workclass State-gov Self-emp-no	ID file index to ro first 50 rows o 1 fnlwgt 77516 83311	ow ID only. See 'Advanc S education Bachelors Bachelors	educati 13 13	S marital Never-married Married-civ Divorced	Adm-clerical Exec-manag Handlers-cle Handlers-cle	Not-in-family Husband Not-in-family Husband	White White	

In the example above you will read a .csv file from a folder data which is in:

Please note that workflows are treated as files, so it is not possible to read or write files inside a workflow.

When the workflow is executing on KNIME Server the options *Relative to* \rightarrow *Current mountpoint* or *Current workflow* will access the workflow repository on the server. The option *Relative to* \rightarrow *Current workflow data area*, instead, will access the data area of the **job copy** of the workflow. Please be aware that files written to the data area will be lost if the job is deleted.

Custom/KNIME URL

This option works with a pseudo-file system that allows to access single files via URL. It supports the following URLs:

- knime://
- http(s):// if authentication is not needed
- ssh:// if authentication is not needed
- ftp:// if authentication is not needed

For this option, you can also set up manually a timeout parameter (in milliseconds) for reading and writing.

The URL syntax should be as follows:

- scheme:[//authority]path[?query][#fragment]
- The URL must be encoded, e.g. spaces and some special characters that are reserved, as ?. To encode the URL you can use any available online URL encoder tool.

Using this option you can read and write single files, but you would not be able to move and copy files or folders. However, listing files in a folder, i.e. browsing, is not supported.

Connected file systems

Connected file systems instead require a connector node to connect to the specific file system. In the connector nodes configuration dialog it is possible to configure the most convenient working directory.

The file system Connector nodes that are available in KNIME Analytics Platform can be divided into two main categories:

Connector nodes that need an Authentication node:

- Amazon S3 Connector node with Amazon Authentication node
- Google Cloud Storage Connector node with Google Authentication (API Key) node
- Google Drive Connector node with Google Authentication node
- SharePoint Online Connector node with Microsoft Authentication node
- Azure Blob Storage Connector node with Microsoft Authentication node
- Azure Data Lake Storage Gen2 node with Microsoft Authentication node

Connector nodes that do not need an Authentication node:

- Databricks File System Connector node
- HDFS Connector node
- HDFS Connector (KNOX) node
- Create Local Bigdata Environment node
- SSH Connector node
- HTTP(S) Connector node
- FTP Connector node
- KNIME Server Connector node

File systems with external Authentication

The path syntax varies according to the connected file system, but in most cases it will be UNIX-like. Information on this are indicated in the respective Connector node descriptions.

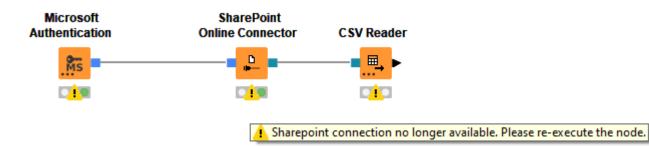
Typically in the configuration dialog of the Connector node you will be able to:

- Set up the working directory: In the *Settings* tab type the path of the working directory or browse through the file system to set up one.
- Set up the timeouts: In the *Advanced* tab set up the connection timeout (in seconds) and the Read timeout (in seconds).

🛕 Dialog - 10:29 - Google Drive Connector	-		\times	🛕 Dialog - 10:29 - Google Drive Connector	-		\times
File				File			
Settings Advanced Flow Variables Job Manager Selection				Settings Advanced Flow Variables Job Manager Selection			
File System settings				Connection timeout (seconds): 30 🌩			
				Read timeout (seconds): 30 -			
Working directory: /	~	Browse					
OK Apply Can	icel (2		OK Apply Cancel	(D	

Most connectors will require a network connection to the respective remote service. The connection is then opened when the Connector node executes and closed when the Connector node is reset or the workflow is closed.

It is important to note that the connections are not automatically re-established when loading an already executed workflow. To connect to the remote service you will then need to execute again the Connector node.



Amazon file system

To connect to Amazon S3 file system you will need to use:

- Amazon Authentication node
- Amazon S3 Connector node



Amazon S3 file system normalizes paths. Amazon S3 allows paths such as /mybucket/.././file, where "..." and "." must not be removed during path normalization because they are part of the name of the Amazon S3 object. When such a case is present you will need to uncheck *Normalize paths* option from the Amazon S3 Connector node configuration dialog.

Please be aware that each bucket in Amazon S3 belongs to an AWS region, e.g. eu-west-1. To access the bucket the client needs to be connected to the same region. You can select the region to connect to in the Amazon Authentication node configuration dialog.

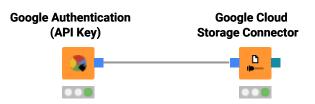
뇌 Dialog ile	- 4:26 - Amazon	Authenticat	ion				_		×
Options	Flow Variables	Job Manag	jer Seleo	tion					
	Au	thentication							
		Anonymous (Credenti	als					
		Credentials							
		Access Key I							
		Default Cred	ential Pr	ovider Cha	in		1		
		Switch Role							
		Account: Role:		9008251 IUserAcces	-D-l-				
	Rea	on eu-west			skole				
		eu-west-	1 2	^					
	Te	eu-west- eu-centra eu-north	al-1						
		ap-east- ap-south ap-south	-1						
		OK	edst-1	Apply		Cancel	(2	

Google file systems

We support two file systems related to Google. However, even though they both belong to Google services the corresponding Connector nodes use a different authentication type, and hence Authentication node.

To connect to Google Cloud Storage you will need to use:

- Google Cloud Storage Connector node
- Google Authentication (API Key) node



Also the Google Cloud Storage Connector node, as the Amazon S3 Connector node, normalizes the paths.

To use the Google Authentication (API Key) node You will need to create a project at console.developers.google.com.

i

The specific Google API you want to use has to be enabled under APIs.

After you create your Service Account you will receive a p12 key file to which you will need to point to in the Google Authentication (API Key) node configuration dialog.

To connect to Google Drive, instead, you will need to use:

- Google Drive Connector node
- Google Authentication node



The root folder of the Google Drive file system, contains your Shared drivers, in case any is available, and the folder My Drive. The path of your Shared drivers will then be /shared_driver1/, while the path of your folder My Drive will be /My Drive/.

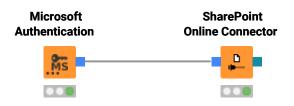
Microsoft file systems

We support three file systems related to Microsoft.

To connect to SharePoint Online, Azure Blob Storage, or to Azure Data Lake Storage Gen2, you will need to use:

- SharePoint Online Connector node, or Azure Blob Storage Connector node, or Azure ADLS Gen2 Connector node
- Microsoft Authentication node

The SharePoint Online Connector node connects to a SharePoint Online site. Here document libraries are represented as top-level folders.



In the node configuration dialog you can choose to connect to the following sites:

- Root Site: Root site of the your organization
- Web URL: https URL of the SharePoint site (same as in the browser)
- Group site: Group site of a Office365 user group
- · Subsite: Connects to subsite or sub-sub-site of the above

The Azure Blob Storage Connector node connects to an Azure Blob Storage file system.

The path syntax will be UNIX-like, i.e. /mycontainer/myfolder/myfile and relative to the root of the storage. Also Azure Blob Storage Connector node performs paths normalization.



The Azure ADLS Gen2 Connector node connects to an Azure Blob Storage file system.

The path syntax will be UNIX-like, i.e. /mycontainer/myfolder/myfile and relative to the root of the storage.



The Microsoft Authentication node offers OAuth authentication for Azure and Office365 clouds.

It supports the following authentication modes:

 Interactive authentication: Performs an interactive, web browser based login by clicking on Login in the node dialog. In the browser window that pops up, you may be asked to consent to the requested level of access. The login results in a token being stored in a configurable location. The token will be valid for a certain amount of time that is defined by your Azure AD settings.

- Username/password authentication: Performs a non-interactive login to obtain a fresh token every time the node executes. Since this login is non-interactive and you get a fresh token every time, this mode is well-suited for workflows on KNIME Server. However, it also has some limitations. First, you cannot consent to the requested level of access, hence consent must begiven beforehand, e.g. during a previous interactive login, or by an Azure AD directory admin. Second, accounts that require multi-factor authentication (MFA) will not work.
- Shared key authentication (Azure Storage only): Specific to Azure Blob Storage and Azure Data Lake Storage Gen2. Performs authentication using an Azure storage account and its secret key.
- Shared access signature (SAS) authentication (Azure Storage only): Specific to Azure Blob Storage and Azure Data Lake Storage Gen2. Performs authentication using shared access signature (SAS). For more details on shared access signatures see the Azure storage documentation.

🛕 Dialog - 4:10 - Microso File	oft Authentio	cation			_		×
Settings Flow Variables							
Username/Passwo Use Sha	eractive auth ername/passw ared key auth	entication vord authenticat entication (Azur	tion e Storage only)) Azure Storage of	v nly)		
Username: Password:							
Credentials flow varial	ble						
 Sharepoint files (Read Sharepoint files (Read User Groups (Read) A Azure Blob Storage/A Storage account: 	l/Write) <i>Note: Require</i> :						
		ОК	Apply	r Ca	incel	0	

File systems without external Authentication

All the Connector nodes that do not need an external Authentication node will connect upon execution to a specific file system. This allows the downstream nodes to access the files of the remote server or file system.

KNIME Server Connector node

With the KNIME Server Connector node you are able to connect to a KNIME Server instance.

When opening the KNIME Server Connector node configuration dialog you will be able to either type in the URL of the KNIME Server you want to connect to, or to *Select...* it from those available among your mountpoints in KNIME Explorer. You do not need to have the KNIME Server mountpoint set up or already connected to use this node.

🛕 Dialog - 4:36 - k ïle	NIME Server Connector			_		×
Settings Advanc	ed Settings Flow Variable	s Job Manager Selection				
URL:	https://server:8443/knime/			~	Select	
Authentication:	Username:	username				
	Password:	•••••				
	Use credentials:	~				
Working directory:	/myfolder/myfile			~	Browse	
		OK Appl	y Cancel	0	2	

You can authenticate either by typing in your username and password or by using the selected credentials provided by flow variable, if any is available. Pleas be aware that when authenticating typing in your username and password the password will be persistently stored in an encrypted form in the settings of the node and will be then saved with the workflow.

SMB Connector node

With SMB Connector node you can connect to a remote SMB server (e.g. Samba, or Windows Server). The resulting output port allows downstream nodes to access the files in the connected file system.

This node generally supports versions 2 and 3 of the SMB protocol. It also supports connecting to a Windows DFS namespace.

When opening the SMB Connector node configuration dialog you can choose to connect to a

File server host or a Windows *Domain*. Choosing *File server* specifies that a direct connection shall be made to access a file share on a specific file server. A file server is any machine that runs an SMB service, such as those provided by Windows Server and Samba.

Choosing *Domain* specifies that a connection shall be made to access a file share in a Windows Active Directory domain.

Read and write from or to a connected file system

When you successfully connect to a connected file system you will be able to connect the output port of the Connector node to any node that is developed under the File Handling framework.

To do this you will need to activate the corresponding dynamic port of your node.



You can also enable dynamic ports to connect to a connected file system in Utility nodes.

To add a dynamic port to one of the node where this option is available, right-click the three dots in the left-bottom corner of the node and from the context menu choose *Add File System Connection port*.



Reader nodes

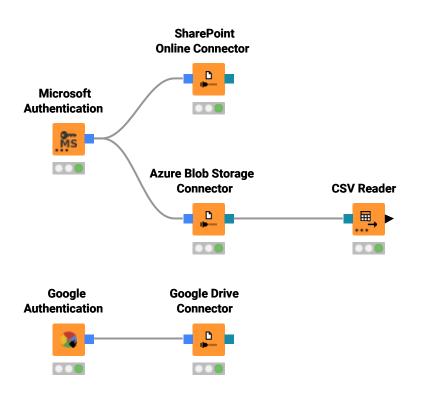
A number of Reader nodes in KNIME Analytics Platform are updated to work within the File Handling framework.

1

See the How to distinguish between the old and new file handling nodes section to learn how to identify the reader nodes that are compatible with the new File Handling framework.

You can use these Reader nodes with both Standard file systems and Connected file systems. Moreover, using the File System Connection port you can easily switch the connection between different connected file systems.

In the following example a CSV Reader node with a File System Connection port is connected to an Azure Blob Storage Connector node and it is able to read a .csv file from an Azure Blob Storage connected file system. Exchange the connection with any of the other Connector nodes, i.e. Google Drive and SharePoint Online, will allow to read a .csv file from the other file systems.



Transformation tab

The new table based reader nodes e.g. Excel Reader or CSV Reader allow you to transform the data while reading it via the *Transformation* tab. To do so the node analyzes the structure of the file(s) to read in the node dialog and stores the file structure and the transformation that should be applied. When the node is executed this information is used to read the data and apply the transformations before creating the KNIME data table. This allows the nodes to return a table specification during configuration of the node and not only once the node is executed, assuming that the file structure does not change. The benefits of this is that you can configure downstream nodes without executing the reader node first and improves execution speed of the node.

1

To speedup the file analysis in the node dialog by default the reader node only reads a limited amount of rows which might lead to mismatching type exceptions during execution. To increase the limit or disable it completely open the *Advanced Settings* tab and go to the *Table specification* section.

/101	ug - t		SV Reader									— C	1	
ting	gs Tr	ansfor	mation Advan	ced Settings Limi	t Rows Enco	ding Flow Varial	bles Job Mana	ger Selection	lemory Policy					
ans	forma	tions												
Re	set ac	tions	1 Move	e up 🚽 Mo	ve down	Enforce types	s Take columns	s from: 💿 Unio	n 🔘 Interse	tion				
		Colur	n			New name			Type					
										mber (integer)				
			orkdass						S St				$\overline{}$	
•										mber (long)			Ť	
		L] fn	lwgt										$^{\circ}$	
1	\checkmark	S ec	lucation											
:			lucation-num							G image				
	_								S Str	-				
		Sm	arital-status											
1	\checkmark	S oc	cupation						XML XM	L				
		Sre	lationship						🐼 Zo	ned Date Time			$\overline{}$	
		S ra	ce						S Str	ing				
:		S se	v						S Str	ina			\neg	
	_									-			-	
		E	pital-gain							mber (integer)			-	
			pital-loss						L Nu	mber (integer)				
	\checkmark	ho	ours-per-week						I Nu	mber (integer)				
1	\checkmark	S na	tive-country						S Str	ing				
:		S in	come						S Str	ing				
		? <	any unknown n	ew column>					?					
		<u> </u>	,											
2Vi	ew —													٦
)1	The su	ggeste	d column types	are based on the	first 50 rows (only. See 'Advanc	ed Settings' tab) .						
	Row	ID	📘 age	S workclass	fnlwgt	S education		S marital	S occupa	S relation	S race	S sex		
	ow0		39	State-gov	77516	Bachelors	13	Never-married		Not-in-family	White	Male	^	
_	ow1		50		83311	Bachelors	13	Married-civ	Exec-manag		White	Male	_	
_	ow2		38 53	Private	215646	HS-grad	9	Divorced		. Not-in-family	White	Male	-	
_	ow3 ow4		28	Private	234721 338409	11th Rachalara	13		Handlers-de		Black	Male	-	1
	ow4 ow5		37	Private Private	284582	Bachelors Masters	13	Married-civ Married-civ			Black White	Female Female	-	
_	ows ow6		49	Private	160187	Masters 9th	5	Married-civ			Black	Female	-	
_	ow0 ow7		52	Self-emp-no	209642	HS-grad	9	Married-civ			White	Male	-	
_	ow8		31	Private	45781	Masters	14	Never-married	-		White	Female	-	
	DIAIO		47	Drivata	150440	Pachalora	13	Marriad civ	Evec manage	Husband	Mbito	Mala	-	

To change the transformation in the updated reader nodes configuration dialog, go to the *Transformation* tab after selecting the desired file. This tab displays every column as a row in a table that allows modifying the structure of the output table. It supports reordering, filtering and renaming columns. It is also possible to change the type of the columns. Reordering is done via drag and drop. Just drag a column to the position it should have in the output table. Note that the positions of columns are reset in the dialog if a new file or folder is selected.

If you are reading multiple files from a folder, i.e. with the option *Files in folder* in the *Settings* tab, you can also choose to take the resulting columns from *Union* or *Intersection* of the files you are reading in.

The *Transformation* tab provides two options which are important when dealing with changing file structures if you control the reader via flow variables.

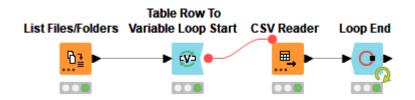
-		ansformation Advanced Settings Flow Variabl		
	sforma set ac		Enforce types Take columns from:	Union O Intersection
		Column	New name	Туре
		D Universe_0_0		D Number (double)
		D Universe_0_1		D Number (double)
		D Universe_1_0		D Number (double)
H		D Universe_1_1		D Number (double)
		S Cluster Membership		S String
		? <any column="" new="" unknown=""></any>		?

The *Enforce types* option controls how columns whose type changes are dealt with. If selected, the node attempts to map to the KNIME type you configured in the transformation tab and fails if that is not possible. If unselected, the KNIME type corresponding to the new type is used whether it is compatible to the configured type or not. This might cause downstream nodes to fail if the type of a configured column has changed.

The <any unknown new column> option is a place holder for all previously unknown columns and specifies whether and where these columns should be added during execution.

Controlling readers via flow variables

As described in the *Transformation* tab section, the reader assumes that the structure of the file will not change after closing the node dialog. However when a reader is controlled via a flow variable the resulting table specification can change. This happens if settings that affect which data should be read are changed via a flow variable. One example is the reading of several CSV files with different columns in a loop where the path to the file is controlled by a flow variable.



During configuration the node checks if the settings have changed. If that is the case the node will not return any table specification during configuration and will analyze the new file structure with each execution. In doing so it will apply the transformations to the new discovered file structure using the <any unknown new column> and Enforce types option.

Currently the new reader nodes only detect changes of the file path. Other settings via flow variables that might influence the structure such as a different sheet name in the *Excel Reader* or a different column separator in the *CSV Reader* are not monitored with the current version. We are aware of this limitation and plan to change this with KNIME Analytics Platform 4.4. Until then you need to enable the *Support changing file schemas* option which is described in the next section if you encounter this problem.

Reading the same data files with changing structure

As described in the *Transformation* tab section, the reader assumes that the structure of the file will not change after closing the node dialog. However the structure of the file can change (e.g. by adding additional columns) if it gets overwritten. In this case you need to enable the *Support changing file schemas* option on the *Advanced Settings* tab. Enabling this option forces the reader to compute the table specification during each execution thus adapting to the changes of the file structure.



i

The Support changing file schemas option will disable the *Transformation* tab and the node will not return any table specification during configuration.

Append a path column

In the Advanced Settings tab you can also check the option Append path column under Path column pane. If this option is checked, the node will append a path column with the provided name to the output table. This column will contain for each row which file it was read from. The node will fail if adding the column with the provided name causes a name collision with any of the columns in the read table. This will allow you to then distinguish the file from which a specific row is read from in case you are reading multiple files and are concatenating them into a single data table.

Writer nodes

Also Writer nodes can be used in KNIME Analytics Platform to work within the File Handling Framework.

You can use these Writer nodes with both <u>Standard file systems</u> and <u>Connected file systems</u>. Moreover, using the File System Connection port you can easily switch the connection between different connected file systems.

An output File System Connection port can be added to Writer nodes and this will allow them

to be easily connected to different file systems and will be able to write persistently files to them.

In the following example a CSV Writer node with a File System Connection port is connected to an Azure Blob Storage Connector node and it is able to write a .csv file to an Azure Blob Storage connected file system. A CSV Reader node read a .csv file from a SharePoint Online File System, data is transformed, and the resulting data is written to the Azure Blob Storage file system.

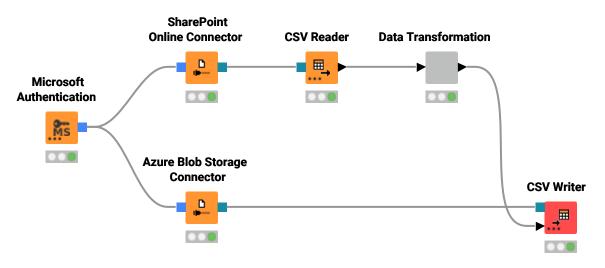


Image Writer (Table) node

The Image Writer (Table) node is also available with a table based input.

You can give to this node an input data table where a column contains, in each data cell, images. Image Writer (Table) is able to process the images contained in a column of the input data table and write them as files in a specified output location.

Path data cell and flow variable

Files and folders can be uniquely identified via their path within a file system. Within KNIME Analytics Platform such a path is represented via a path type. A path type consists of three parts:

- 1. **Type:** Specifies the file system type e.g. local, relative, mountpoint, custome_url or connected.
- 2. **Specifier:** Optional string that contains additional file system specific information such as the location the relative to file system is working with such as workflow or mountpoint.
- 3. **Path:** Specifies the location within the file system with the file system specific notation e.g. C:\file.csv on Windows operating systems or /user/home/file.csv on Linux operating systems.

Path examples are:

- Local
 - (LOCAL, , C:\Users\username\Desktop)
 - (LOCAL, , \\fileserver\file1.csv)
 - (LOCAL, , /home/user)
- RELATIVE
 - (RELATIVE, knime.workflow, file1.csv)
 - (RELATIVE, knime.mountpoint, file1.csv)
- MOUNTPOINT
 - (MOUNTPOINT, MOUNTPOINT_NAME, /path/to/file1.csv)
- CUSTOM_URL
 - (CUSTOM_URL, , https://server:443/my%20example?query=value#frag)
 - (CUSTOM_URL, , knime://knime.workflow/file%201.csv)
- CONNECTED
 - (CONNECTED, amazon-s3:eu-west-1, /mybucket/file1.csv)
 - (CONNECTED, microsoft-sharepoint, /myfolder/file1.csv)
 - (CONNECTED, ftp:server:port, /home/user/file1.csv)
 - o (CONNECTED, ssh:server:port, /home/user/mybucket/file1.csv)

(CONNECTED, http:server:port, /file.asp?key=value)

A path type can be packaged into either a Path Data Cell or a Path Flow Variable. By default the Path Data Cell within a KNIME data table only displays the path part. If you want to display the full path you can change the cell renderer via the context menu of the table header to the *Extended path* renderer.

Creating path data cells

In order to work with files and folders in KNIME Analytics Platform you can either select them manually via the node configuration dialog or you might want to list the paths of specific files and/or folder. To do this you can use the *List Files/Folders* node. Simply open the dialog and point it to the folder whose content you want to list. The node provides the following options:

- *Files in folder*: Will return a list of all files within the selected folder that match the *Filter options*.
- *Folders*: Will return all folders that have the selected folder as parent. To include all sub folders you have to select the *Include subfolders* option.
- *Files and folders*: Is a combination of the previous two options and will return all files and folders within the selected folder.

🛆 Dialog - 6:21 - List Files/Folders (workflow)	_	×
File		
Settings Flow Variables Job Manager Selection Memory Policy		
Read Read from Relative to V Current workflow data area v		
Mode Files in folder Files and folders Filter options Include subfolders		
Folder .	Browse	 V
() Selected 1 of 1 files		
Indude parent folder		
Add folder identifier column		
OK Apply Cancel	0	

Manipulating path data cells

Since KNIME Analytics Platform version 4.4 you can also use the Column Expressions node which supports now the Path type data cells. This node provides the possibility to append an arbitrary number of columns or modify existing columns using expressions.

Creating path flow variables

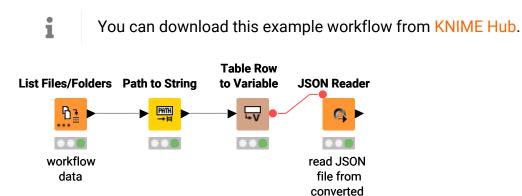
There are two ways to create a path flow variable. The first way is to export it via the dialog of the node where you specify the path. This could be a *CSV Writer* node for example where you want to export the path to the written file in order to consume it in a subsequent node. The second way is to convert a path data cell into a flow variable by using one of the available variable nodes such as the *Table Row to Variable* or *Table Row to Variable Loop Start* node.

String and path type conversion

Until now not all nodes that work with files have been converted to the new file handling framework and thus do not support the path type. These nodes require either a String or URI data cell or a string flow variable.

From path to string

The *Path to String* node converts a path data cell to a string data cell. By default the node will create a String representation of the path that can be used in a subsequent node that still requires the old string or URI type, e.g. *JSON Reader*.



If you only want to extract the plain path you can disable the _Create KNIME URL for 'Relative to' and 'Mountpoint' file system option in the Path to String node configuration dialog.

path location

▲ Dialog - 6:22 - Path to String File	_		×
Settings Flow Variables Job Manager Selection Memory Policy Column selection P Path ~			
Output Output Append column: Create selected column Create KNIME URL for 'Relative to' and 'Mounpoint' file systems 			
OK Apply Cancel		2	

Similar to the *Path to String* node, the *Path to String* (*Variable*) node converts the selected path flow variables to a string variable.

If you want to use a node that requires a URI cell you can use the *String to URI* node after the *Path to String* node.

From string to path

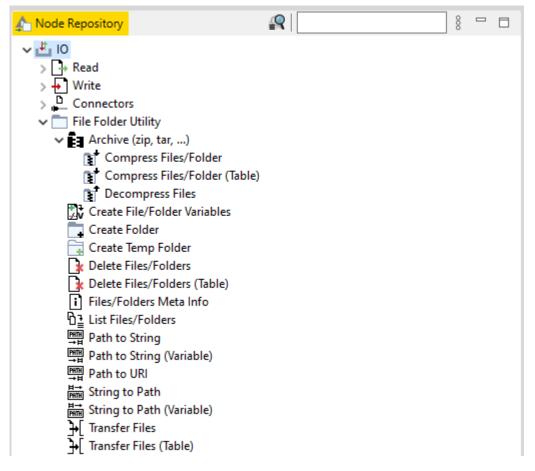
In order to convert a string path to the path type you can use the *String to Path* node. The node has a dynamic File System port that you need to connect to the corresponding file system if you want to create a path for a connected file system such as Amazon S3.

Similar to the *String to Path* node the *String to Path* (*Variable*) node converts a string flow variable into a path flow variable.

File Folder Utility nodes

With the introduction of the new file handling framework with KNIME Analytics Platform release 4.3, we also updated the functionality of the utility nodes.

You can find the File Folder Utility nodes in the node repository under the category *IO* > *File Folder Utility*.



You can add a dynamic port to connect to a connected file system directly with the File Folder Utility nodes. In this way you can easily work with files and folders in any file system that is available.

In the example below the Transfer Files node is used to connect to two file systems a source file system, Google Drive in this example, and a destination file system, SharePoint Online in this example, to easily transfer files from Google Drive to SharePoint Online.

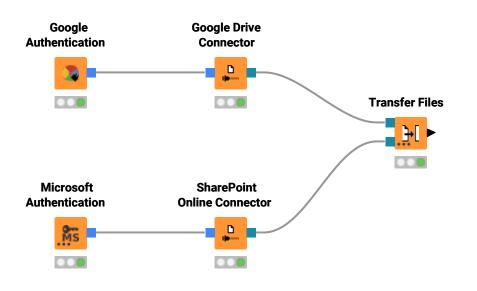


Table based input Utility nodes

Some of the File Folder Utility nodes are also available with a table based input. Those are the ones that have (*Table*) in their name, e.g. Compress Files/Folder (Table), Delete Files/Folders (Table), and Transfer Files (Table) nodes. You can give to these nodes an input data table where a column contains, in each data cell, Paths to files and these are processed by the File Folder Utility (Table) nodes.

For example, with Compress Files/Folder (Table) and Delete Files/Folder (Table) nodes you can respectively compress or delete the files whose Paths are listed in a column of the input data table. With Transfer Files (Table) you can transfer all the files whose Paths are listed in a column of the input data table.

Since the new utility nodes (e.g. *Compress Files/Folder, Delete Files/Folders* and *Transfer Files*) do not provide the full functionality of the old nodes yet, as of now the nodes, including the required connector nodes for the different file systems, are still available but marked as *(legacy)*. You can find them in the node repository under the *IO/File Handling (legacy)* category. They will be deprecated once the new nodes have the full functionality. If you are missing some functionality in the new nodes please let us know via the KNIME Forum.

i

Compatibility and migration

With the 4.3 release of the KNIME Analytics Platform we introduced the new file handling framework. The framework requires a rewrite of the existing file reader and writer nodes as well as the utility nodes. However not all nodes provided by KNIME or the Community have been migrated yet which is why we provide this section to help you to work with old and new file handling nodes side by side.

How to work with workflows that contain both old and new file handling nodes

How to distinguish between the old and new file handling nodes

You can identify the new file handling nodes by the three dots for the dynamic port on the node icon that allow you to connect to a connected file system.



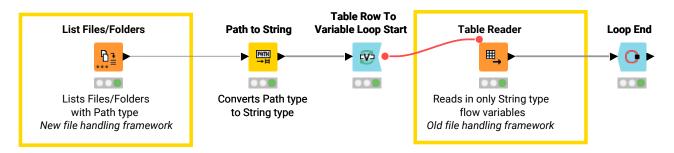
Another way to identify a new file handling node is by the *Input location* section in the node configuration dialog. Here, for nodes that have been migrated to the new file handling framework you will typically find a drop-down menu that allows you to specify the file system that should be used. The following image shows the *Input location* section of the configuration dialog of a typical reader node in the new and old file handling framework.

🛕 Dialog - 0:3 - CSV Reader	– 🗆 X
File	
Settings Transformation Advanced Settings Limit Rows Encoding Flow Variables Memory Policy	
Input location	
Read from Local File System	
Local File System	
Mode Mountpoint - Relative to	
File Custom/KNIME URL	V Browse
	•
▲ Dialog - 3:2 - CSV Reader	- 🗆 ×
File	
1.1%	
Settings Limit Rows Encoding Flow Variables Memory Policy	
Input location:	
	→ Browse
Input location section	
old file handling framework Custom connection timeout [s]:	

Working with old and new flow variables and data types

Using flow variables you can specify the location of a file to read or write automatically. Because not all nodes provided by KNIME or the Community have been migrated yet you might face the problem that the newer file nodes support the new Path type whereas older nodes still support a string flow variable which can be either a file path or a URI.

The List Files/Folders node returns a table with a list of files/folders in the new Path type. To use the Path type in a node that has not been migrated yet (e.g. in the Table Reader), you need to first convert the new Path type using the Path to String node into a string.



Some older nodes require a URI as input. To convert the Path type to an URI you first need to convert it to a string using the Path to String node and then convert the string to an URI using the String to URI node.

If you already have a Path type flow variable you can use the Path to String (Variable) node to convert the Path type flow variable into a string flow variable.

If you want to convert an existing file path or string to the new Path type you can either us the String to Path node to convert data columns or the String to Path (Variable) node to convert string variable.

How to migrate your workflows from old to new file handling nodes

This section should help you to migrate your workflows from the old file handling framework to the new file handling framework. The new framework provides lots of advantages over the old framework which come at the cost of some changes in the usage which we will address here.

In order to standardize the node names and better capture the functionality of the actual node we have renamed some of the nodes. In addition we have also removed some nodes whose functionality has been integrated into another node such as the *Excel Writer* which replaces the original *Excel Writer* (*XLS*) as well as the *Excel Sheet Appender* node.

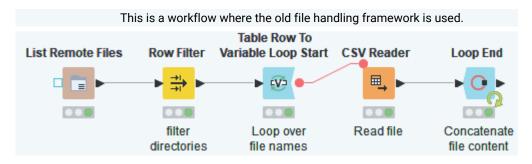
Table 1. Old (<4.2) and new (4.3+) file handling framework nodes conversion table

Node Icon	4.2 Old		4.3 New
B	Zip Files	→	 Compress Files/Folder Compress Files/Folder (Table)
	Unzip Files	\rightarrow	Decompress Files
0	Create Directory	\rightarrow	Create Folder
	Delete Files	→	 Delete Files/Folders Delete Files/Folders (Table)
<mark>.}•</mark> [►	 Copy/Move Files Download Upload 	\rightarrow	 Transfer Files Transfer Files (Table)
<u>ि</u> ⊒ ≡	List Remote FilesList Files	→	List Files/Folders
0	Create Temp Dir	→	Create Temp Folder

Node Icon	4.2 Old		4.3 New
	Connection (e.g. SSH Connection)	<i>→</i>	Connector (e.g. SSH Connector)
	Excel Reader (XLS)	\rightarrow	Excel Reader
×	Read Excel Sheet Names (XLS)	\rightarrow	Read Excel Sheet Names
	 Excel Writer (XLS) Excel Sheet Appender 	\rightarrow	Excel Writer
 +	File Reader	\rightarrow	 File Reader File Reader (Complex Format)

Reading multiple files into a single KNIME data table

Using the old file handling framework you had to use loops in order to read multiple files into a single KNIME data table. So your workflows looked like the following image.



i

With the new file handling framework you no longer need to use loops since the reading of multiple files into a single KNIME data table is now built into the reader nodes them self. All you have to do is to change the reading mode to *Files in Folder*. You can than open the Filter options dialog by clicking on the *Filter options* button. Here you can filter the files that should be considered e.g. based on their file extension. In addition you can specify if files in subfolders should be included or not.

🛕 Dialog - 3:30 - CSV Reader	▲ Filter options	× – – ×
File	File filter options	
Settings Transformation Advanced Settings Limit Rows Encoding Flow Variables	File extension(s) csv	
Input location Read from Local File System V	Case sensitive File name *	
Mode O File I Files in folder Filter options I Indude subfolders	Case sensitive Wildcard Regular expression	
Folder c:\temp	Include hidden files	V Browse
Celected 4 of 7 files Reader options	Folder filter options	
Format	Folder name *	
Autodetect format	Case sensitive	
, Column delimiter V/n Row delimiter	Include hidden folders	
Quote char Quote escape char	OK Cancel	
# Comment char		

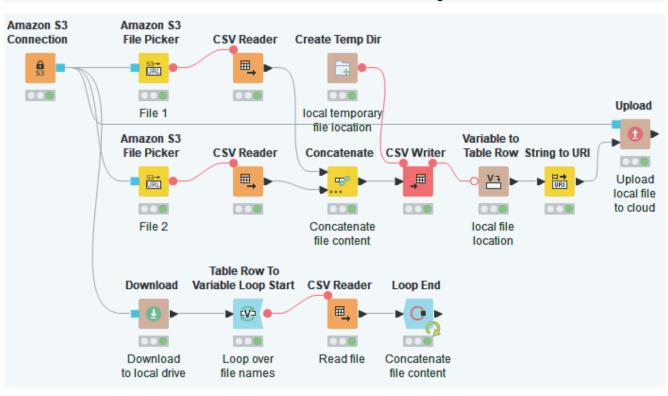
If you continue to use a loop to read multiple files and encounter problems during the execution have a look at the Controlling readers via flow variables section.

Reading from or writing to remote file systems

With the old file handling framework, when working with remote file systems such as Amazon S3 or HDFS you had two options when reading files:

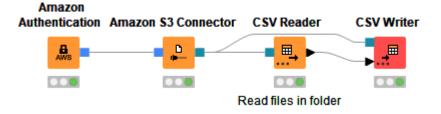
- 1. Download the file to a local folder and then point the reader node to the local copy
- 2. If available, use one of the File Picker nodes (e.g. Amazon S3 File Picker, Azure Blob Store File Picker or Google Cloud Storage File Picker) to create a signed URL to pass into the reader node.

In order to write a file to a remote file system you had to first write the file to your local hard drive and then upload it to the remote file system.



This is a workflow where the old file handling framework is used

With the new file handling framework you no longer need to use any additional nodes to work with files in remote file systems but simply connect the reader and writer nodes to the corresponding connector node via the dynamic ports.



Working with KNIME URL

When working with the nodes of the new file handling framework you no longer need to create KNIME URLs but can use the built-in convenience file systems Relative to and Mountpoint. The following table lists the relation between the KNIME URL and the two new file systems.

KNIME URL	Convenient file system
knime://knime.node/	No direct replacement due to security reasons but check out Relative to → Current workflow data area

KNIME URL	Convenient file system
knime://knime.workflow/	Relative to \rightarrow Current workflow*
knime://knime.mountpoint/	Relative to \rightarrow Current mountpoint
knime:// <mountpoint_name>/</mountpoint_name>	Mountpoint → <mountpoint_name></mountpoint_name>

i

i

Due to security reasons KNIME workflows are no longer folders but considered as files. So you can no longer access data within a workflow directory except for the newly created workflow data area.

If you need the KNIME URL for a *Relative to* or *Mountpoint* path you can use the Path to String and Path to String (Variable) nodes with the *Create KNIME URL for 'Relative to' and 'Mountpoint' file systems* option enabled, as it is by default.

> We encourage you to use the new convenience file systems Relative to and Mountpoint in favor of the KNIME URLs. For more details on how to convert from the new file systems to the old KNIME URLs see the Working with old and new flow variables section.

Excel Appender

The functionality of the *Excel Appender* node has been integrated into the new *Excel Writer* node. The node allows you to create new Excel files with one or many spreadsheets but also to append any number of spreadsheets to an existing file. The number of sheets to write can be changed via the dynamic ports option of the node.

File Reader and File Reader (Complex Format) nodes

With KNIME Analytics Platform release 4.4.0 we:

- Improved the File Reader node
- Introduced a new File Reader (Complex Format) node

The **File Reader node** now supports the new File Handling framework and can use path flow variables and connect to file systems. The File Reader node is able to read the most common text files.

The newly introduced **File Reader (Complex Format) node** also supports the new File Handling framework, and it is able to read complex format files. We recommend to use the

File Reader (Complex Format) node only in case the File Reader node is not able to read your file.

With the File Reader node you can also enable the *Support changing file schemas* option in the *Advanced Settings* tab of its configuration dialog. This allows the node to support eventual input file structure changes between different invocations. This is instead not possible with the File Reader (Complex Format) node that does not support files changing schemas. Thus, if the File Reader (Complex Format) node is used in a loop, you should make sure that all files have the same format (e. g. separators, column headers, column types). The node saves the configuration only during the first execution. Alternatively, the File Reader node can be used.





KNIME AG Talacker 50 8001 Zurich, Switzerland www.knime.com info@knime.com

The KNIME® trademark and logo and OPEN FOR INNOVATION® trademark are used by KNIME AG under license from KNIME GmbH, and are registered in the United States. KNIME® is also registered in Germany.