ADAMS

Advanced **D**ata mining **A**nd **M**achine learning **S**ystem

Module: adams-yaml



Peter Reutemann

November 21, 2026

©2019



THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato



Except where otherwise noted, this work is licensed under
http://creativecommons.org/licenses/by-sa/4.0/

Contents

1 Flow **5**

Bibliography **7**

Chapter 1

Flow

Flows can process YAML[2] files and strings as long as they are in map notation, which is commonly used for configuration files.
The following transformers are available:

- *GetMapKeys* – returns all keys from the key.
- *GetMapValue* – extracts a value associated with the specified key from a *java.util.Map* object.
- *MapToVariables* – turns the key-value pairs of a *java.util.Map* into variables with their associated values.
- *SetMapValue* – sets the value of a *java.util.Map* object (either specified value or obtained from a callable actor or storage item).
- *YamlFileReader* – reads the specific YAML file and forwards a Map object.

The following sinks are available:

- *YamlFileWriter* – writes the *java.util.Map* object to disk.

The following conversion are available:

- *ArrayToYamlString* – turns an object array into a YAML string.
- *ListToYamlString* – turns a *java.util.List* object into a YAML string.
- *PropertiesToMap* – turns a *java.util.Properties* object into a *java.util.Map* object.
- *ReportToMap* – turns a *Report* object into a *java.util.Map* object.
- *MapToReport* – turns a *java.util.Map* object into a *Report* object.
- *MapToKeyValuePairs* – turns the map into actual key-value pair objects.
- *MapToYamlString* – turns a *java.util.Map* object into a YAML string.
- *YamlStringToMap* – generates a *java.util.Map* object from the YAML string.

Bibliography

[1] *ADAMS* – Advanced Data mining and Machine learning System `https://adams.cms.waikato.ac.nz/`

[2] *YAML* – YAML Ain't Markup Language, a human friendly data serialization standard for all programming languages. `http://yaml.org/`