# JET Extension Tags

Bradford Matthew Arant Sr.

November 1, 2024

# Contents

# Chapter 1

# Introduction

JET will pass through the untagged areas to the output without any modi-fications. Data contained within the tags may modify their containers before placing any output. The space taken by the tag itself is not passed to the output and will not appear in the output.

Skip Blank Lines options on containers will skip passing any blank lines or line containing only whitespace to the output.

# Chapter 2

# Tags and Attributes

XML style tagging is used to insert functions and other behaviors into the script. As the document contents are output a tag may be inserted to perform the desired function and affect the output.

Tag attributes are used to pass parameters into the operations and functionality for each tag.

Outputting a database from mysql may be performed using the ¡mysql¿ tag in combination with the ¡ifrow¿, ¡whilerow¿ and ¡sql¿ tags.

# Chapter 3

# Variables and Variable Types

Variables are used to contain dynamic content values and can be sourced from several locations and limited in scope depending on use. Some variable types can be set to reflect script state or are read only from other outside sources of data. The following is a list of variable types and a brief description of their source:

- Global - values can be initiaized using the set tag and specifying the scope of global.

- Local -

## 3.1 Global Variables

Global is the default if the scope is not specified for any function that writes to a variable.

## 3.2 Local Variables

## 3.3 Environment Variables

## 3.4 CGI Variables

# Chapter 4

# Expressions

Expressions are used to perform complicated calculations (arithmetic), perform comparisons between values (boolean), manipulate strings (string) or perform operations on dates (date).

## 4.1 Operators

Operators are used to perform arithmetic or boolean operations on arguments to derive a particular result. There are two types of operators, as follows:

- Arithmetic -
- Boolean -

### 4.1.1 Arithmentic Operators

### 4.1.2 Boolean Operators

## 4.2 Function Reference

### 4.2.1 left(string, number-of-characters)

### 4.2.2 random()

Use the random function to return a random number between 0 and 1.

### 4.2.3 substring(string, start-position, number-of-characters

Use the substring operation to extract a portion of a string and return the value as a string.

# Chapter 5

# Common Gateway Interface Features

JET extension tags provides options for enabling the processing of data using the Common Gateway Interface (CGI) specification used by Apache and NGINX as well as other HTTP server applications. This makes JET a powerful tool for creating web interfaces and API handler environments for various javascript frameworks as well as just native HTML style interaction.

To enable the CGI features within a requested document you can specify the cgi="true" attribute on a ¡jet¿ tag containing the document or ¡jet /¿ tag contained within the document.

When you enable the CGI interface the script will look for specific environment variables used to retrieve the input data from the browser interface and automatically provide these values to the JET scripting using the variables interface. There are currently three supported data formats or content types through this interface. These content types are:

formdata/urlencoded mutipart/formdata text/json

# Chapter 6

# Tag Reference

This chapter will cover all the tags and parameters required to perform the operations that JET provides.

## 6.1 call

Use the call tag to evoke an executable file from a jet script.

The atrributes are:

pgm

argn

name

input

```
<call pgm="ls" arg1="-al" name="listing" />
```

will call the ls command with parameter -al and place the output into a variable named 'listing'.

## 6.2 comment

Use the comment tag to create a section in the jet script that can be used for making comments and is ignored by the jet reader.

## 6.3 for

Use the for tag to iterate a tag container for a logical number of times.

## 6.4 header

Use the header tag to output a header prior to outputting the process buffer to the requester.

## 6.5   if/else

Use the if tag to perform a conditional output on the tag container. An optional else container provides alternate output in the event the condition is not met.

## 6.6   ifrow/else

Use the ifrow tag to output the tag container if a row exists in the mysql tag.

## 6.7   jet

Use the jet tag to specify parameters for the jet parsing session and to control the output options.

## 6.8   mysql

Use the mysql tag to specify parameters for connecting to a mysql server. The container of the tag is where you can specify the sql statement and method of processing the result output of the executed sql statement.

## 6.9   read

Use the read tag to read the contents of a file contained on the local file system into a variable for further output or processing.

## 6.10   set

Use the set tag to store initialize a variable to a value contained in a value attribute, the result of an expression (expr) attribute or the contents of the set tag container.

## 6.11   sql

Use the sql tag to specify an sql statement to run on the sql server specified in the mysql tag.

As long as sql statements are executed within the same mysql containing tag then the same mysql session is used for each sql statement.

## 6.12   stream

Use stream tag to output data from the server without waiting for the standard output buffering and processing of the JET script. This is useful for outputting

images and streams for audio and video without the server having to load the whole thing into RAM first.

## 6.13   system

Use the system tag to execute a bash shell command within the JET script.

## 6.14   tag

Use the tag tag to define a new tag definition for use within the script.

Tags are normally defined globally and can be used anywhere within the script.

Tags can also be defined within other tags and the scope of those tags is only within the defining tag's container. The enclosed tag definition will not be available globally.

## 6.15   until

Use the until tag to process a container at least once and continue to process the container until the specified condition is met.

Note that this tag always processes the container at least once.

Be careful not to create a never ending loop by not modifying any of the condition parameters specified in the condition clause.

## 6.16   while

Use the while tag to process a container if the specified condition is true.

Note that the container may not be processed if the condition is never met.

Be careful not to create a never ending loop by not modifying any of the condition parameters specified in the condition clause.

## 6.17   whiledir

Use the whiledir tag to loop a directory path to the container for processing.

## 6.18   whilerow

Use the whilerow tag in combination with mysql and sql tags to provide a container to process for each row retrieved by the sql statement.

## 6.19   write

Use the write tag to write data to the local file system.