

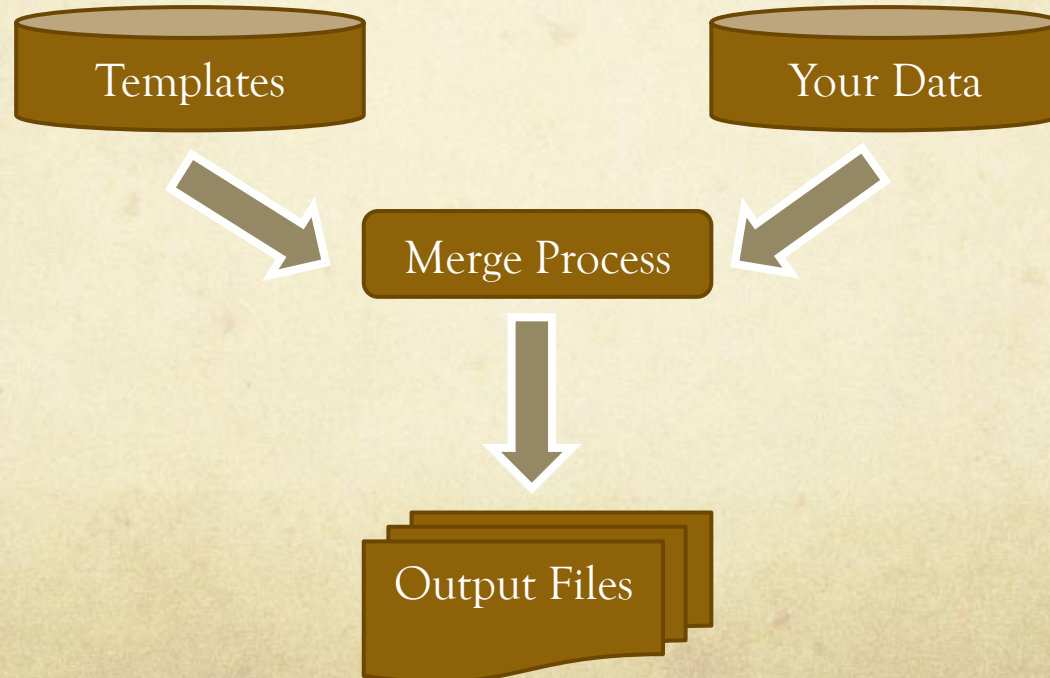


DragonFly

A SQL and Template based Merge Tool

Overview

DragonFly is a powerful web based tool that allows you to generate any type of output files from simple text templates and a SQL data source.



Terminology

- Template – The template Content and options.
- Directive – Instructions for the Merge Process, associated with a Template
- Merge – The process of replacing tags with data and inserting sub-templates to generate output
- HashTable – An associative array with unique Keys associated with a corresponding value. When a duplicate key is added, it replace the existing value for that key.

Merge Process - Replace

The Merge process builds a hash of replace from/to values and then processes them

From	To
{name}	Mr. Smith
{address}	1234 Main St.
{company}	Blue Widgets

Template

Dear {name}
 {company}
 {address}

We are writing to invite your staff to
enroll in our special training class.

Thank you

Intense Training Specialist



Output

Dear Mr. Smith
 Blue Widgets
 1234 Main St.

We are writing to invite your staff to
enroll in our special training class.

Thank you

Intense Training Specialist

Merge Process

Merge Sub-Templates

The Merge process will insert sub-templates at Bookmarks



Merge Sub-Templates Collections

Sub-Template selected
based on contact
preference column

template1

Dear {name}
{company}
{address}
We are writing to invite your staff to
enroll in our special training class.
We would like to extend an
invitation to:
<tkBookmark name="template2"/>
Thank you

Template2 - eMail

- {contact} eMail: {eMail}

Template2 - Mail

- {contact} Address: {addr}

Template2 - Phone

- {contact} Phone: {phone}

Merge Process

Output

Dear Mr. Smith
Blue Widgets
1234 Main St.
We are writing to invite your staff to
enroll in our special training class.
We would like to extend an
invitation to:
- Fred eMail: fred@gmail.com
- Betty Address: Betty Boop, 123 St.
- Kara Phone: 555-121-3312
Thank you

Select * from contacts

contact	preference	eMail	addr	phone
Fred	eMail	fred@gmail.com		
Betty	Mail		Betty Boop, 123 st.	
Kara	Phone			555-121-3312

alist

4 Ways to add Replace Values

- The replace stack is built in the following order:
 - Parameters to the html request
 - Directives associated with the Template
 - Replace Value directive
 - Replace Column directive
 - Replace Row directive

Replace Parameters

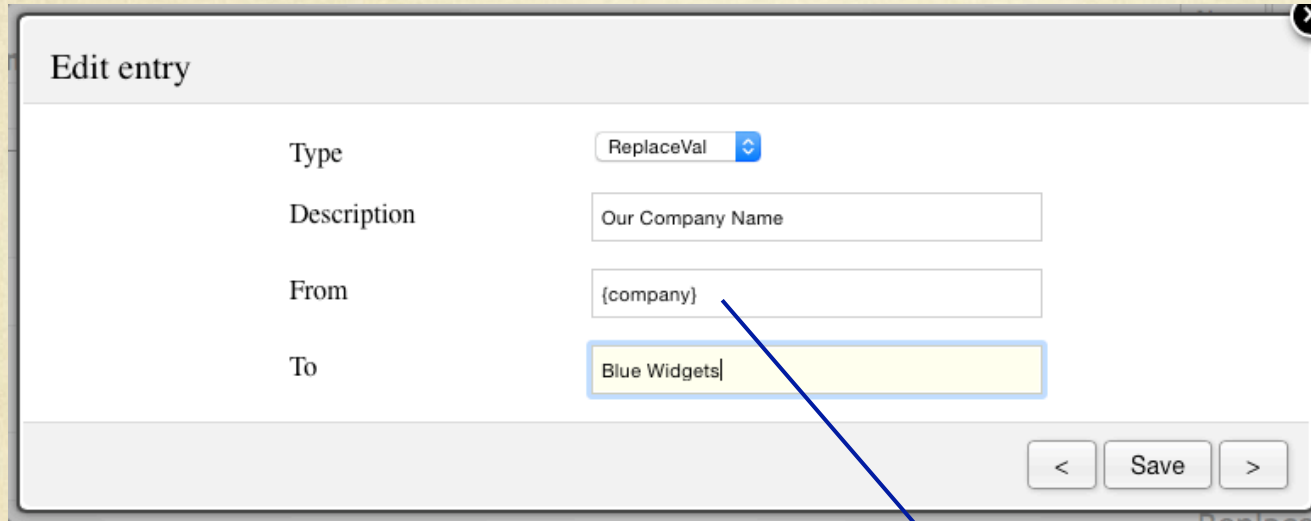
- <http://www.you.com/DragonFly/Merge?name=Mike&age=21>



From	To
{name}	Mike
{age}	21

- Note that { and } are added to the From value
- The parameter “CacheReset” is reserved, with a value of “Yes” will clear the Template Cache on the server.

Replace Values Directive




The screenshot shows a dialog box titled "Edit entry" with a close button in the top right corner. It contains four fields: "Type" with a dropdown menu set to "ReplaceVal", "Description" with the text "Our Company Name", "From" with the text "{company}", and "To" with the text "Blue Widgets". The "To" field is highlighted with a blue border. At the bottom right, there are three buttons: "<", "Save", and ">". A blue arrow points from the "{company}" text in the "From" field to the third row of the table below.

Note that { and } are NOT added to the From value

From	To
{name}	Mike
{age}	21
{company}	Blue Widgets

Replace Column Directive


Type	ReplaceCol 
Description	Get Special Replace Values
Select Columns	fromValue, toValue
From Tables	reportReplace
Where Condition	idreport = '1'

Note { and } are added to fromValue

Select fromValue, toValue from reportReplace where idreport='1'	
fromValue	toValue
Division	Sales
CostCenter	East
Date	June 1 st , 2015

From	To
{name}	Mike
{age}	21
{company}	Blue Widgets
{Division}	Sales
{CostCenter}	East
{Date}	June 1 st , 2015

Replace Row Directive

Type	ReplaceRow 
Description	Get Employee Record
Select Columns	*
From Tables	employee
Where Condition	employeeName = {name}

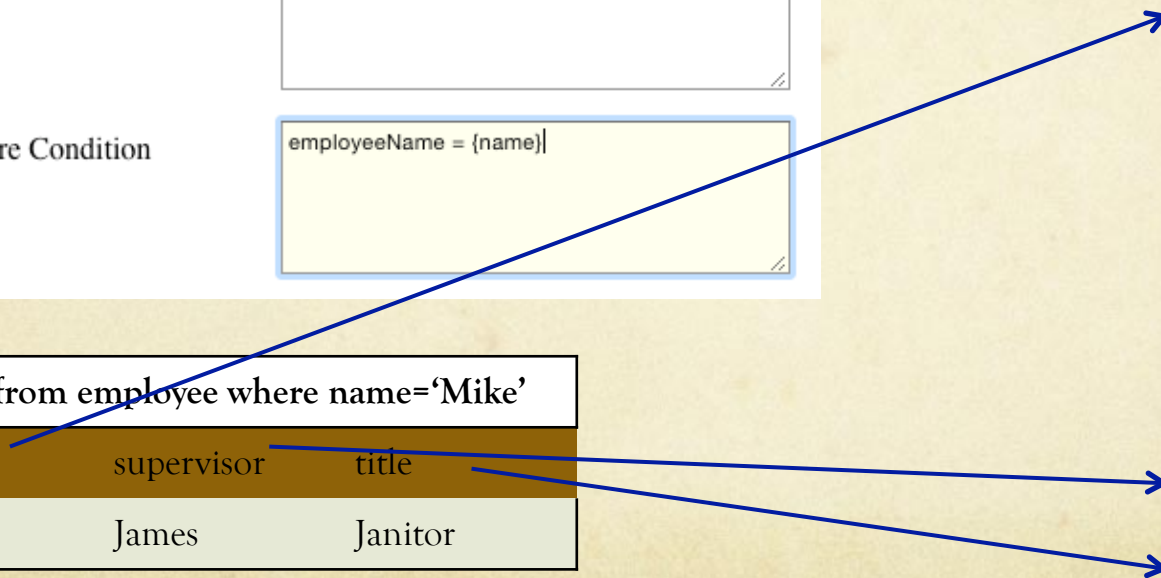
Note: { and } are added to Column Name

All where conditions are “Replace Processed” during Merge.

A result set with more than 1 row will result in an error and merge failure.

Select * from employee where name='Mike'		
name	supervisor	title
Mike	James	Janitor

From	To
{name}	Mike
{age}	21
{company}	Blue Widgets
{Division}	Sales
{CostCenter}	East
{Date}	June 1 st , 2015
{supervisor}	James
{title}	Janitor



The Insert Directive

Type	<input type="text" value="Insert"/>
Description	<input type="text" value="Insert Contact Info"/>
Insert From Collection	<input type="text" value="testDirective"/>
Select Columns	<input type="text" value="."/>
From Tables	<input type="text" value="customerContacts"/>
Where Condition	<input type="text" value="repName = {Name}"/>
Not Last Tags:	<input type="text" value="{and} "/>
Only Last Tags	<input type="text" value="{and},{or},{coma}"/>

The insert directive will insert sub-templates for each row of the result set.

Sub-templates are inserted at `<tkBookmark name="subTemplate" />` tags. Multiple bookmarks are supported.

The current replace hash, along with a replace row of the current result set is the starting set for the sub-template.

Not Last tags are set to “” on the last inserted sub-template

Only Last tags are set to “” on all but the last sub-template.

Sub-Template Collections

- Vary the sub-template inserted based on a column in the result set.
- Sub-Template Collection specifies the Column used to select sub-templates to be inserted.
- Template for Collection, Name, Column Value is inserted if found.
- If not found, the template for a blank column value is inserted.
- If the blank value is not found, the merge fails.

Saving Output Files

- By default a template is either inserted into the parent template, or written to the servlet response. A template can specify an output file name to be created.
- If a file is created, nothing is inserted into the parent template.
- File names are “replace processed” before the file is saved.