

Movable Type 4 的 テンプレート管理



Six Apart 株式会社 Movable Type 製品企画

上ノ郷谷太一

- テンプレートタグの書き方に自由度
- テンプレートの管理方法を変更
- カテゴリー・日付アーカイブテンプレートの統合

MT3

<\$MTEntryTitle\$>

<\$MTENtryTitle\$>



MOVABLE TYPE®

<\$MTEntryTitle\$>

<\$MT**E**NtryTitle\$>

<\$**m**t**e**ntry**t**itle\$>

<\$**M**t**e**ntry**T**itle\$>

<mt:**e**ntrytitle />

タグの書き方のまとめ

- 大文字か小文字かのミスはなくなる
- タグの種類によって書き方を変更できる
- なんかちょっとカッコイイと思う書き方にできる

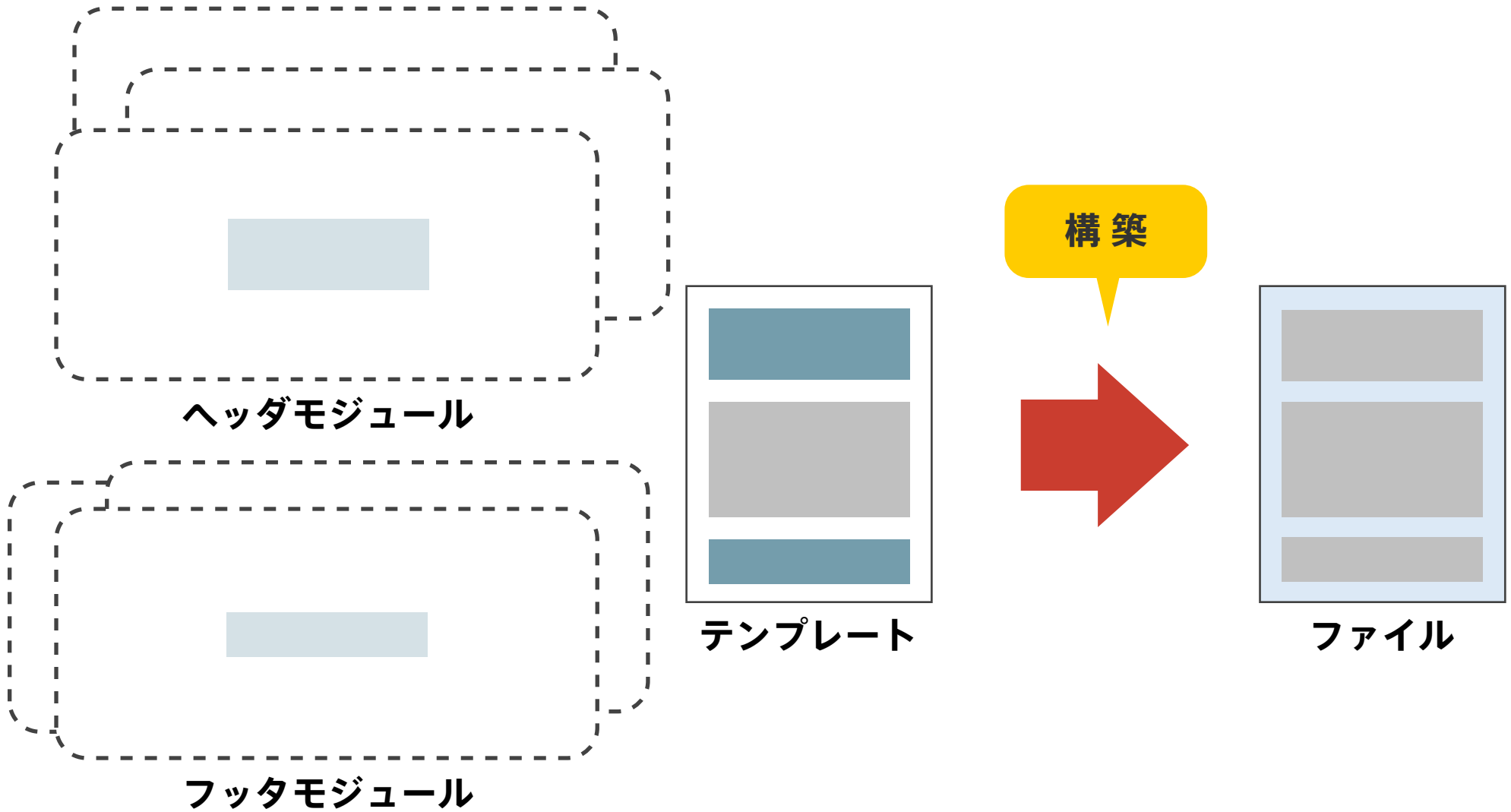
メリット

- 使い回しのできるものを部品に
- 大きいテンプレートの場合編集しやすい

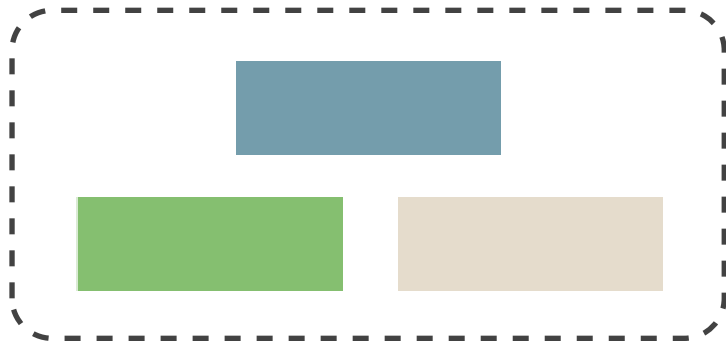
デメリット

- モジュールが増える可能性がある
- 管理が大変になってしまい本末転倒

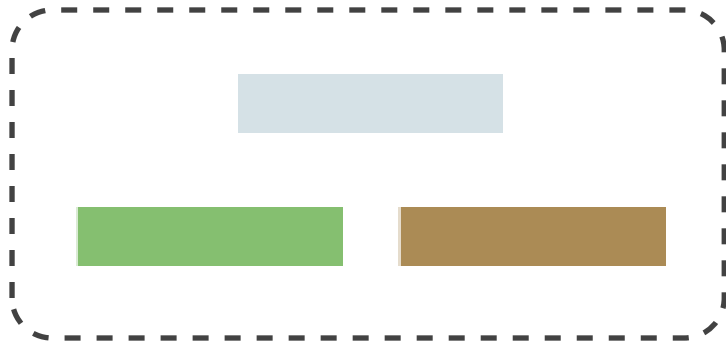
テンプレートモジュール



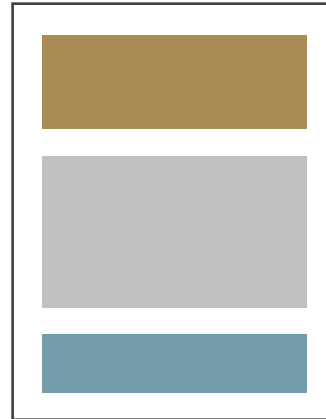
- 編集しやすいというだけでは分ける意味が無い
- モジュール各々が柔軟なのがいい
- ひとつのモジュールで
いくつものテンプレートに合わせた内容を出力



ヘッダモジュール

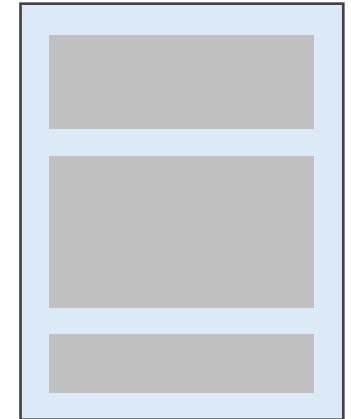
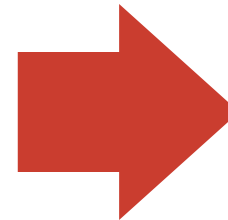


フッタモジュール



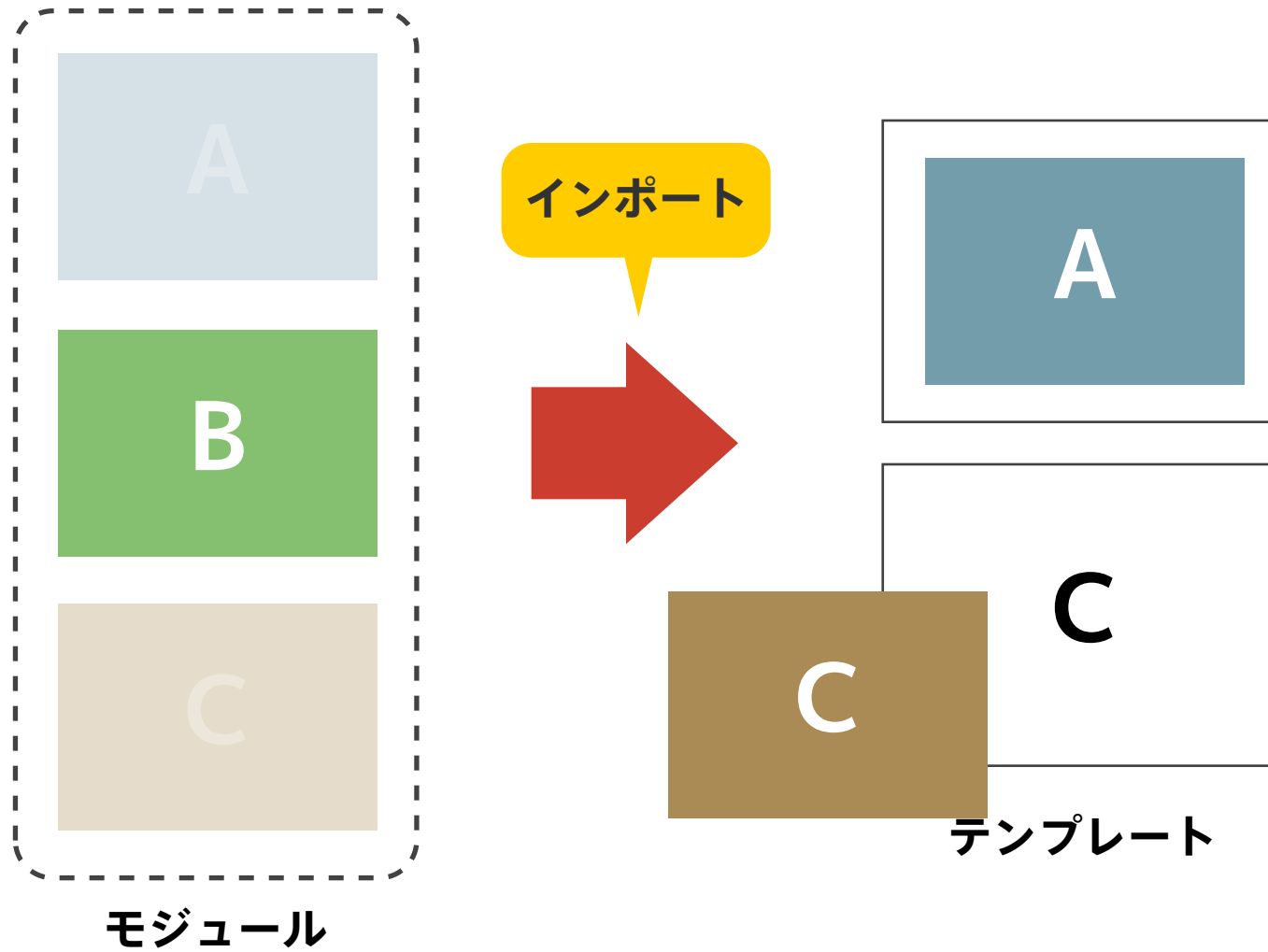
テンプレート

構築



ファイル

柔軟なモジュール



- SetVar* テンプレートタグでテンプレートに名前
- If / Unless / Else 条件文で出し分け
- 1テンプレートモジュールをマルチテンプレート対応
- ほかに盛りだくさん

- 名前 (name) をつけた箱に値 (value)をセットする
- セットした値は参照可能

```
<mt:setvar name="main_index" value="1" />
```

main_index という箱に 1 という値をセットするよ～

```
<mt:setvar name="elephant" value="love" />
```

elephant という箱に love という値をセットするよ～

```
<mt:setvar name="main_index" value="1" />
```

main_index という箱に 1 という値をセットするよ～

- Movable Type ではテンプレートを判別するのに利用

```
<mt:setvar name="elephant" value="love" />
```

elephant という箱に love という値をセットするよ～

```
<mt:getvar name="elephant" />
```



構築

love

- 名前 (name) をつけた箱に値 (ブロックした内容)をセットする
- いろんなものをセットしておける
- セットした値は参照可能

```
<mt:setvarblock name="entry_title">  
  <mt:entrytitle />  
</mt:setvarblock>
```

entry_title という箱に
このテンプレートで処理する <mt:entrytitle /> の結果をセットするよ～

SetVarBlock

```
<mt:setvarblock name="blog_name">  
  <mt:blogname /> だよ！  
</mt:setvarblock>
```

blog_name という箱に
<mt:blogname /> の結果をセットしますよ～

```
<mt:getvar name="blog_name" />
```



構築

ゾウラヴぶろぐ だよ！

Movable Type テンプレートタグリファレンス

Movable Type で利用できるテンプレートタグの一覧です。Movable Type 3.x 及び Enterprise 1.0 のタグリファレンスに掲載の無いものには **NEW** アイコンが付いています。また、new in mt4 タグを設定していますので、以下の検索結果ページで一覧可能です。

- new in mt4 を含むテンプレートタグ

A

- MTAdminCGIPat **FUNCTION**
- MTAdminScript **FUNCTION** **NEW**
- MTApp:ActionBar **FUNCTION** **NEW**
- MTApp:Form **BLOCK** **NEW**
- MTApp:Link **FUNCTION** **NEW**
- MTApp:ListFilters **FUNCTION** **NEW**
- MTApp:Listing **BLOCK** **NEW**
- MTApp:PageActions **FUNCTION** **NEW**
- MTApp:Setting **BLOCK** **NEW**
- MTApp:SettingGroup **BLOCK** **NEW**
- MTApp:StatusMsg **BLOCK** **NEW**
- MTApp:Widget **BLOCK**
- MTArchiveCategory **FUNCTION**
- MTArchiveCount **FUNCTION**
- MTArchiveDate **FUNCTION**
- MTArchiveDateEnd **FUNCTION**
- MTArchiveFile **FUNCTION**

タグリファレンス

If / Unless / Else

- もし、A だったら…
- もし、A じゃなかったら…
- SetVar* で設定した値を利用して条件分岐

- もし A だったら
- どの変数を評価するか (name)
- どう比較するか (eq, ne, lt, gt, le, ge, like)

```
<mt:if name="main_index"></mt:if>
```

main_index という変数の値が 0 以外の場合は実行するよ～

```
<mt:if name="elephant" eq="love"></mt:if>
```

elephant という変数の値が love と等しかったら実行するよ～

- もし A じゃなかったら
- どの変数を評価するか (name)
- どう比較するか (eq, ne, lt, gt, le, ge, like)

```
<mt:unless name="main_index"></mt:unless>
```

main_index という名前の値が 0 場合 (main_index じゃない) は実行するよ～

```
<mt:unless name="elephant" eq="love"></mt:unless>
```

elephant という名前の値が love と等しいから実行しないよ～

```
<mt:setvar name="hoge" value="foo" />
```

モディファイア	働き
eq="foo" (equal)	foo と等しい
ne="foo" (less than)	foo と等しくない
lt="foo" (less than)	foo より小さい
gt="foo" (greater than)	foo より大きい
le="foo" (less than or equal)	foo 以下
ge="foo" (greater than or equal)	foo 以上
like="foo"	foo を含む (正規表現可)

MTIf BLOCK NEW

条件文の対象となる MTSetVer ファンクションタグに設定した value モディファイアの値、または MTSetVarBlock, MTSetVarTemplate ブロックタグの持つデータを利用して条件文を作成します。MTIf ブロックタグは条件に一致した場合にのみ実行します。MTUnless は MTIf の逆で条件に一致しない場合にのみ実行します。

MTIf ブロックの中で MTUnless の結果と同じ結果を得たい場合は MTElse ブロックタグと組み合わせることで、条件に一致しない場合の実行内容も設定できます。

クイックリンク

- [block](#)
- [conditional](#)
- [new in mt4](#)
- [template tag](#)

使い方

タグリファレンス

```
<MTIf name="foo">
  <!-- do something -->
<MTElse>
  <!-- do something else -->
</MTElse>
</MTIf>
```

モディファイア

name="*foo*"

条件文の対象となる MTSetVer ファンクションタグに設定した name モディファイアの値を設定します。このモディファイアの値でどの値で条件分岐を行うかを決定します。このモディファイアは必須です。

```
<mt:if name="elephant" eq="love">  
  <mt:else></mt:else>  
</mt:if>
```

elephant という名前の値が love と等しくなかったら実行するよ～

```
<mt:unless name="elephant" eq="love">  
  <mt:else></mt:else>  
</mt:unless>
```

elephant という名前の値が love と等しかったら実行するよ～

テンプレートを区別する

<mt:setvar>

A

<mt:setvar>

B

<mt:setvar>

C

```
<mt:setvar name="A" value="1" />
```

このテンプレートは A ですよ～。というのを示すのに利用

`<mt:if name="A">`  `</mt:if>`

`<mt:if name="B">`  `</mt:if>`

`<mt:if name="C">`  `</mt:if>`

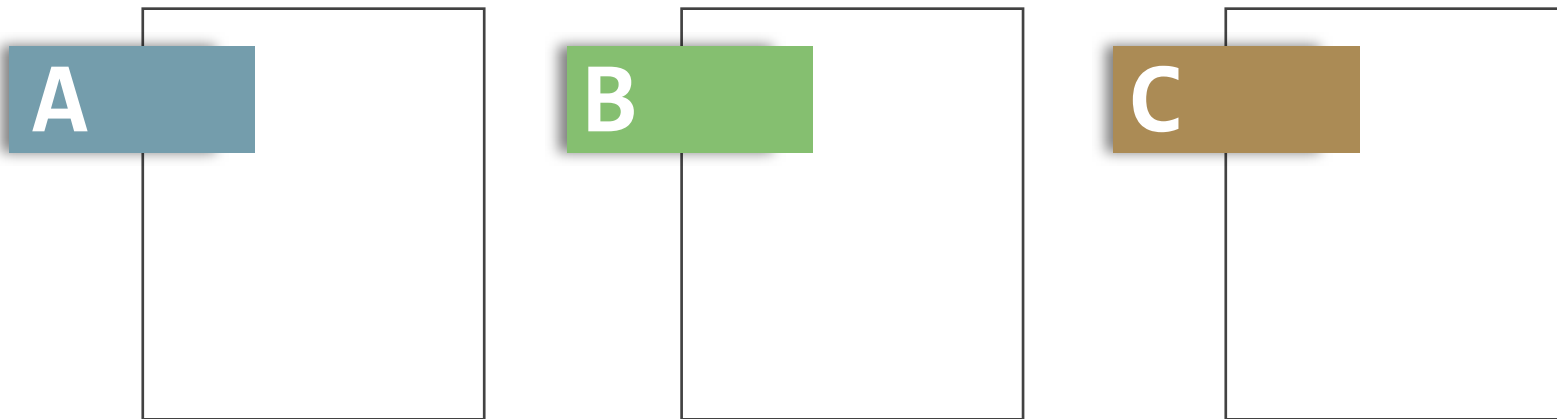
モジュール “hoge”

hoge モジュールの読み込み

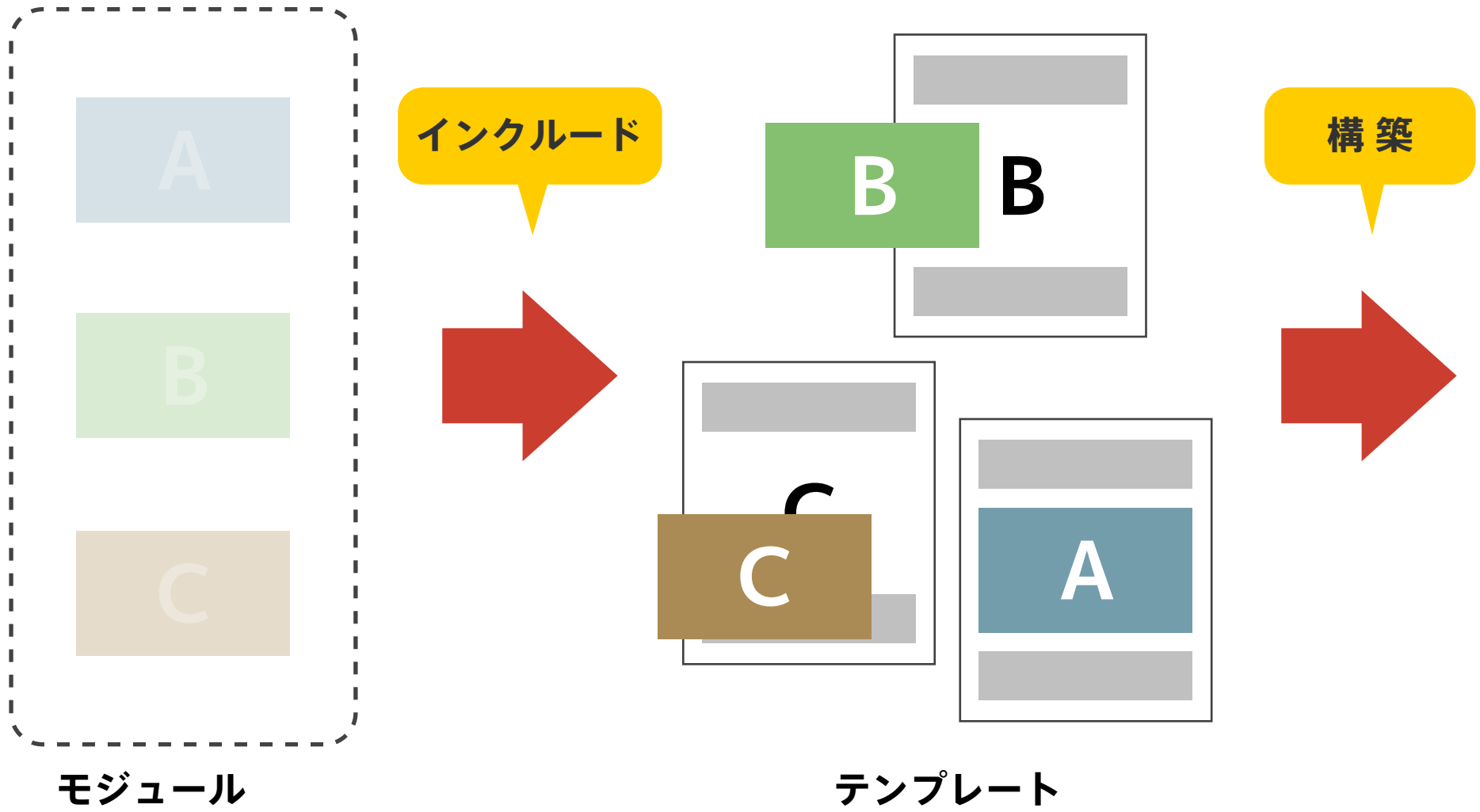


```
<mt:include module="hoge" />
```

hoge というモジュールを読み込みますよ～



テンプレート



- テンプレートごとに名前をつける（グループでも可）
- モジュールの中で各々のテンプレート用に条件文
- ひとつのモジュールで複数のテンプレートに対応
- 少ないテンプレートでいろいろ

利用するテンプレートタグ

- SetVar
- If / Unless / Else
- IfArchiveType
- Include

MTIf BLOCK NEW

条件文の対象となる MTSetVer ファンクションタグに設定した value モディファイアの値、または MTSetVarBlock, MTSetVarTemplate ブロックタグの持つデータを利用して条件文を作成します。MTIf ブロックタグは条件に一致した場合にのみ実行します。MTUnless は MTIf の逆で条件に一致しない場合にのみ実行します。

MTIf ブロックの中で MTUnless の結果と同じ結果を得たい場合は MTElse ブロックタグと組み合わせることで、条件に一致しない場合の実行内容も設定できます。

クイックリンク

[block](#)
[conditional](#)
[new in mt4](#)
[template tag](#)

使い方

タグリファレンス

```
<MTIf name="foo">  
  <!-- do something -->  
<MTElse>  
  <!-- do something else -->  
</MTElse>  
</MTIf>
```

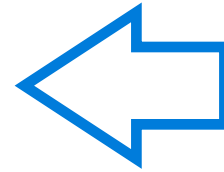
モディファイア

name="*foo*"

条件文の対象となる MTSetVer ファンクションタグに設定した name モディファイアの値を設定します。このモディファイアの値でどの値で条件分岐を行うかを決定します。このモディファイアは必須です。

```
<mt:setvarblock name="archive_title">
```

```
  <mt:archivetitle />
```



ファンクションタグ

```
</mt:setvarblock>
```

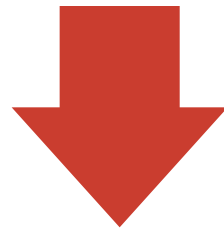
このテンプレートで処理する `<mt:archivetitle />` の結果は
`archive_title` という名前で記憶しておきますよ～

テンプレートごとに値が変わるようなデータもOK

SetVarBlock 時間を記憶

```
<mt:setvarblock name="hour">  
  <mt:date format="%H" />  
</mt:setvarblock>
```

現在の時刻を hour という名前で記憶しておきますよ～



ダイナミックパブリッシング

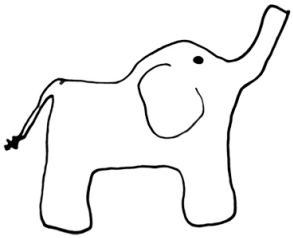

```
<mt:if name="hour" like="[01-04]">
  <p>夜更かしたら明日にひびくYO!自重してね><</p>
</mt:if>

<mt:if name="hour" eq="12">
  <p>おひるだよー \(^o^)/</p>
</mt:if>
```

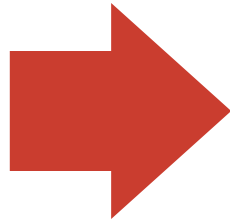
⋮

モジュール "message"

アクセスした時間にあわせて



アクセス



03:10

夜更かしは明日にひびくYO!自重してね><

10:50

仕事中にネットかよ!自重してね><

12:23

おひるだよー \(^o^)/

15:30

そろそろおやつの時間?和菓子作れ!

メッセージを変えて出力できる

それ。

javascript でできるよ

SetVarBlock まとめ

- SetVarBlock + Date で時間をセット
- モジュールに時間を利用した条件文
- ダイナミックパブリッシング

- SetVarBlock
- If / Unless / Else
- Include

ダイナミックパブリッシング ++

MTUnless BLOCK NEW

条件文の対象となる MTSetVer ファンクションタグに設定した value モディファイアの値、または MTSetVarBlock, MTSetVarTemplate ブロックタグの持つデータを利用して条件文を作成します。MTUnless ブロックタグは条件に一致しない場合にのみ実行します。MTIf は MTUnless の逆で条件に一致する場合にのみ実行します。

MTUnless ブロックの中で MTIf の結果と同じ結果を得たい場合は MTElse ブロックタグと組み合わせることで、条件に一致しない場合の実行内容も設定できます。

クイックリンク

- [block](#)
- [conditional](#)
- [new in mt4](#)
- [template tag](#)

タグリファレンス

使い方

```
<MTUnless name="foo">
  <!-- do something -->
<MTElse>
  <!-- do something else -->
</MTUnless>
```

モディファイア

name="*foo*"

条件文の対象となる MTSetVer ファンクションタグに設定した name モディファイアの値を設定します。このモディファイアの値でどの値で条件分岐を行うかを決定します。このモディファイアは必須です。

- テンプレートモジュールは各々を柔軟に
- SetVar* で値を設定
- If / Unless /Else で値をあやつる
- 少ないテンプレートで柔軟なページ作り
- 条件にあわせてコンテンツをあやつる

夢... ひろがりんぐ。