

基本からしっかり学ぶ

 blog cms の  
カスタマイズ

基礎編



appleple

Web System&Design

## はじめに

---

a-blog cms（エーブログ・シーエムエス）は2009年6月にリリースされました。当初はWeb制作者のために、より便利に簡単にカスタマイズできるようにという事で開発されましたが、バージョンアップを重ねサイトを更新するクライアント様にも使いやすいインターフェースとなり、双方に嬉しいシステムに進化してきています。

これまで、各地での勉強会やオンラインセミナーの2-3時間コースで紹介してきた事を文章にまとめました。a-blog cmsに興味を持った人に、まず読んで頂く入門書になったらと思い書きました。

a-blog cms 開発元 有限会社アップルアップル 代表 山本 一道

## 1. ユニット

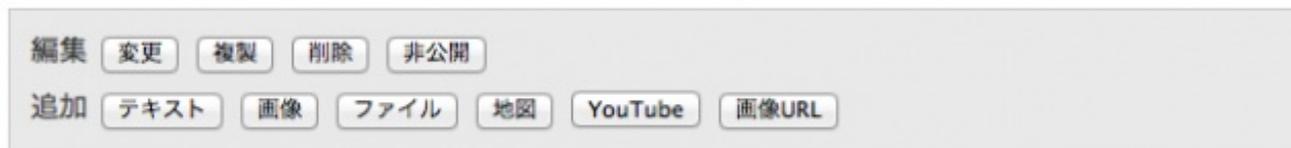
a-blog cms の特徴的なエントリーの記述方法として、WYSIWYG（ウィジウィグ）を利用しないユニット毎のテキスト投稿があります。主にHTMLのブロック要素単位にテキストエリアを用意しテキストを登録したり、画像を登録したりする事ができます。



ここでは、テキストのユニット、画像のユニット、地図のユニットを表示させていますが、この他にはファイルやYouTube、画像URLを指定するユニットがあります。これらを駆使する事で綺麗なHTMLで複雑なレイアウトのエントリーを作成していく事が可能になります。

また、このユニットはドラッグして順番を入れ換えたりする事もでき、細かいユニットの初期値の設定や、ユニットグループ機能というものを使っての段組レイアウトも簡単に設定する事が可能です。

## 1-1.ユニット追加ボタン



エントリーを編集・追加等を行うボタンをこちらで設定する事ができます。このボタンについては、Entry\_Bodyモジュール内に以下のように記述されています。

```
<!--#include file="/admin/entry/edit.html"-->
<!--#include file="/admin/entry/add.html"-->
```

そして、管理ページのコンフィグのユニット設定に、ボタンの画像と同様の並び順で設定されている事が確認できます

### ユニット追加ボタン

モード ?	ラベル ?	
↑ ↓	<input type="text" value="text"/>	<input type="text" value="テキスト"/> <input type="button" value="削除"/>
↑ ↓	<input type="text" value="image"/>	<input type="text" value="画像"/> <input type="button" value="削除"/>
↑ ↓	<input type="text" value="file"/>	<input type="text" value="ファイル"/> <input type="button" value="削除"/>
↑ ↓	<input type="text" value="map"/>	<input type="text" value="地図"/> <input type="button" value="削除"/>
↑ ↓	<input type="text" value="youtube"/>	<input type="text" value="YouTube"/> <input type="button" value="削除"/>
↑ ↓	<input type="text" value="eximage"/>	<input type="text" value="画像URL"/> <input type="button" value="削除"/>
<input type="button" value="追加"/>		

こちらで並び順を変更するとボタン側も変更する事ができます。また、こちらで設定を追加する事でボタンを追加する事も可能です。

- モード photo
- ラベル 写真を3枚追加

のような設定を追加し、保存します。

## 1-2.ユニット設定

上記の続きになります。ユニット追加ボタンで設定を追加後は、編集設定の横にあるユニット設定を行います。



このユニット設定のページでは、新規エントリーの追加ボタンや、テキスト、画像といったユニットの追加ボタンを押した際の初期状態の設定が行えます。そして、上記で「写真を3枚追加」というボタンを追加した事から以下のようなエリアが用意される事になります。



ここで、画像を追加ボタンを3回押し、それぞれの画像ユニットに画像の配置を左に設定し、小さな画像のサイズを設定する事で、ボタンを押した際に3つの画像ユニットを表示させるボタンを作る事ができます。

このようにユニット設定で、事前に利用するようなセットのボタンを作ってみたり、画像のサイズを事前に適切なサイズにしておきましょう。

過去作ったサイトで、このユニット設定を再度確認してみてください。画像のサイズが「そのまま」になっている場合には、どれかサイズを指定しておきましょう。きっと、クライアントは毎回サイズを手で設定しているハズです。

## 1-3.テキストタグセレクト

テキストユニットのタグを指定するための設定になります。基本的なHTMLのタグに加えて、none, markdown, wysiwyg という特別なタグが設定できます。この中がないものを追加する際には /include/column.html をカスタマイズする事で表示が可能になります。

```
<!--#include file="/include/column.html"-->
```

この column.html は、a-blog cms のユニットそのものであり、ユニットで表示されるタグについては、このファイルでコントロールされています。このファイルを理解する事がユニットのカスタマイズを攻略する近道かもしれません。

### テキストタグセレクト

	タグ <input type="text"/>	ラベル <input type="text"/>	
↑ ↓	<input type="text" value="p"/>	<input type="text" value="本文"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="h2"/>	<input type="text" value="見出し大"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="h3"/>	<input type="text" value="見出し中"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="h4"/>	<input type="text" value="見出し小"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="ul"/>	<input type="text" value="リスト"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="ol"/>	<input type="text" value="番号付きリスト"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="table"/>	<input type="text" value="テーブル"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="pre"/>	<input type="text" value="ソース"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="blockquote"/>	<input type="text" value="引用"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="none"/>	<input type="text" value="自由入力"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="markdown"/>	<input type="text" value="マークダウン"/>	<input type="button" value="削除"/>
↑ ↓	<input type="text" value="wysiwyg"/>	<input type="text" value="WYSIWYG"/>	<input type="button" value="削除"/>
	<input type="button" value="追加"/>		

そして、管理ページのテキストタグセレクトを編集する事で、以下のようなテキストユニットのプルダウンをカスタマイズする事ができます。



このテキストタグセレクトのタグエリアには、通常のHTMLタグだけでなくCSSのクラスを指定する事が可能です。

p.price

のように指定する事で<p class="price">~</p>のようにする事もできます。あとはラベル側をクライアントが理解しやすいラベルを指定し、"商品価格"のようにする事で、商品の値段を表示させたい時には、この指定を利用するといった事が可能になります。



## 1-5. サイズセレクト

ユニットグループと同様に、画像や地図、YouTubeの動画のサイズを事前に設定しておき、クライアントには選んで頂くだけでサイズを手動で入れないのが a-blog cms のエントリー上に貼られるコンテンツの作り方になります。

サイズセレクト

	基準	サイズ	ラベル		
イメージサイズ選択肢 <input type="checkbox"/>	↑ ↓	長辺 :	<input type="text"/>	そのまま	削除
	↑ ↓	横 :	220	幅1/3 ( 220px )	削除
	↑ ↓	横 :	340	幅1/2 ( 340px )	削除
	↑ ↓	横 :	700	幅1/1 ( 700px )	削除
		<input type="button" value="追加"/>			
マップサイズ選択肢 <input type="checkbox"/>	サイズ		ラベル		
	↑ ↓	<input type="text" value="210x210"/>	<input type="text" value="小 (210x210)"/>	<input type="button" value="削除"/>	
	↑ ↓	<input type="text" value="320x320"/>	<input type="text" value="中 (320x320)"/>	<input type="button" value="削除"/>	
↑ ↓	<input type="text" value="430x430"/>	<input type="text" value="大 (430x430)"/>	<input type="button" value="削除"/>		
		<input type="button" value="追加"/>			
	サイズ		ラベル		
	↑ ↓	<input type="text" value="320x205"/>	<input type="text" value="小 (320x205)"/>	<input type="button" value="削除"/>	
	↑	<input type="text" value="480x295"/>	<input type="text" value="中 (480x295)"/>	<input type="button" value="削除"/>	

オススメの設定としては、メインコンテンツの幅で大画像、2つの画像のマージンをメインのコンテンツ全体から引いて、2で割ったサイズを中画像、3つの画像を並べる際のマージンx2を引いて3で割ったサイズを小画像と設定します。

例 :

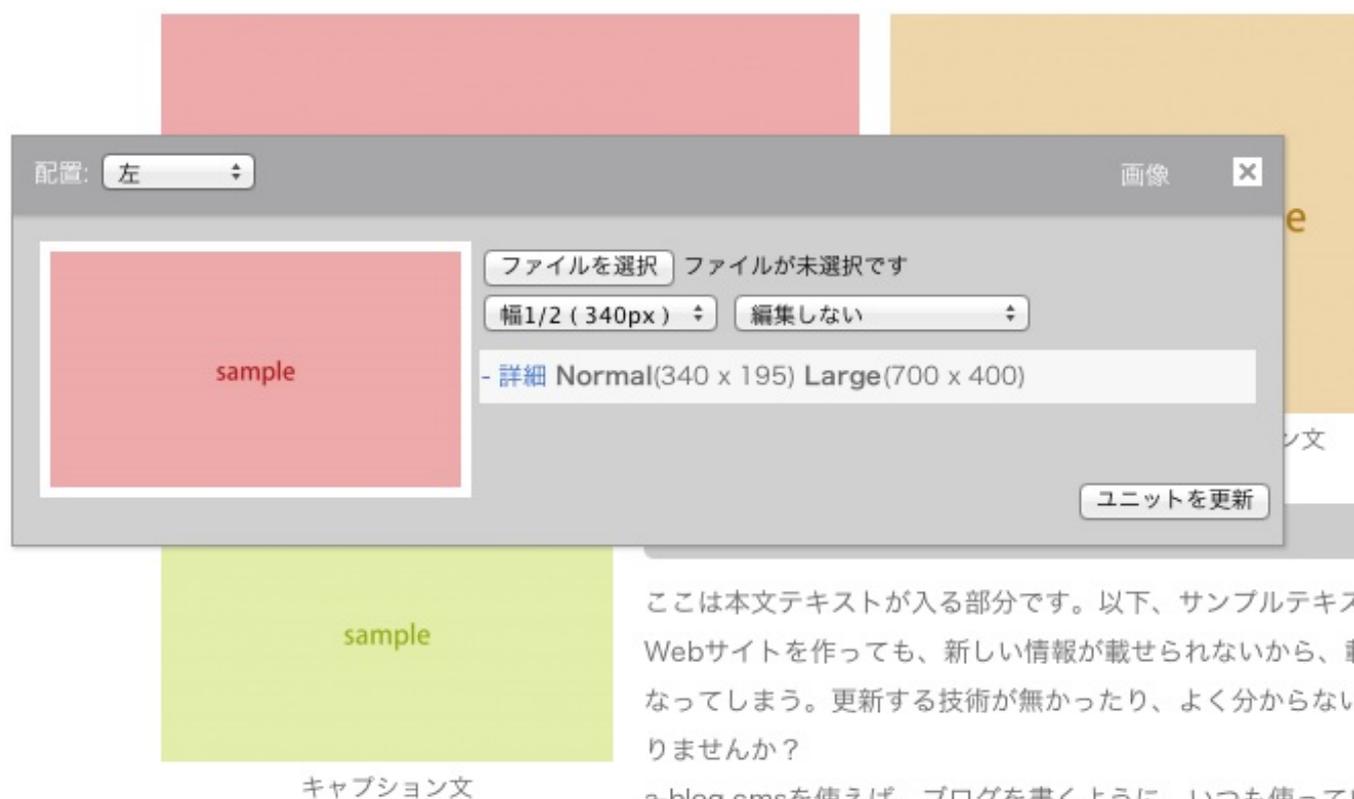
- 大 700px
- 中 340px
- 小 220px

## 1-6.ユニットのダイレクト編集

ユニットを沢山設定して、画像やテキストのユニットをしっかり用意すると、1つのエントリーで50ユニットになったりすることもあります。そうなると、変更ボタンを押してフォームになった時にどこのユニットを更新するのかが分からなくなる事も多々ある事です。



そこでダイレクト編集機能を利用する事で、1つのユニットのみフォームを表示させ更新する事が可能になります。



画面のリロード無しに画像やテキストを変更する事ができます。注意する点としては、1記事表示の時にしか使えない事と、そのユニットを削除する事ができません。

## 1-7.複数画像の一括アップロード

---

1.5からの新しい機能として、複数画像の一括アップロードする機能が追加されました。



この機能を利用する事で、簡単に何枚もの画像を1つの画像ユニットからアップロードする事が可能です。これはHTML5の機能を利用している事から対応しているブラウザである必要があり、今のところ残念ながら Internet Explorer がバージョン9でも利用する事ができませんので注意が必要です。

## 2.モジュール

a-blog cms では、ブラウザ上から更新できる部分には、モジュールというパーツがテンプレートのHTMLに貼り付けられています。どのようなモジュールがあるのかを知る事が、a-blog cms のカスタマイズの上達の近道ではないでしょうか。



モジュールには大きく分けると、ビルトインモジュール・フィールドモジュール・タッチモジュールの3種類のモジュールがあります。

モジュールをテンプレートにいくつでも貼り付ける事は可能です。たくさんのモジュールが動作すると、それだけ実行する時間がかかる事も頭の片隅に入れておいて下さい。例えば0.1秒で動くモジュールが100個テンプレートに貼ってあると表示に10秒かかる事になります。

モジュールの貼り過ぎには注意しましょう！でも、標準搭載のキャッシュ機能で最初に10秒かかるページも、2回目以降は瞬時に表示できますので、ご安心下さい。

## 2-1.ビルトインモジュール

---

基本的にモジュールと言えば、このビルトインモジュールの事になります。多くはエントリー系の情報を表示させるモジュールや、カテゴリー、ナビゲーション、バナー、トピックパス等いろいろなモジュールを利用する事になります。

```
<!-- BEGIN_MODULE Entry_List -->
<ul>
  <!-- BEGIN entry:loop -->
  <li><a href="{url}">{title}</a></li>
  <!-- END entry:loop -->
</ul>
<!-- END_MODULE Entry_List -->
```

<!-- BEGIN\_MODULE モジュール名 --> から <!-- END\_MODULE モジュール名 --> を1つのモジュールとして動作します。モジュールの中にモジュールを書く事もでき、モジュールの実行順としては、内側のモジュールが実行され、その後に外側のモジュールが動作するようになっています。

## 2-2.フィールドモジュール

---

登録したカスタムフィールドのデータを表示させるためには、基本的にはフィールドモジュールを利用する必要があります。例外として、ビルトインモジュール内で利用できるカスタムフィールドもあります。

- Blog\_Field
- Category\_Field
- Entry\_Field
- User\_Field

※1.5.0 から Entry\_Body , Entry\_Summary の entry:loop 内でも、カスタムフィールドが利用できるようになりました。モジュールの設定画面でカスタムフィールドを編集するかどうかをチェックボックスで指示する事ができるようになりました。使わない時には、動作させない方が、処理を軽くさせる事ができます。

## 2-3. タッチモジュール

---

a-blog cms には IF文 のような条件分岐が用意されていません。ですのでプログラミングが苦手とか、分からない方でも、a-blog cms のカスタマイズが簡単と頂ける1つの理由ではないかと思えます。

```
<!-- BEGIN_MODULE Touch_Entry -->
  <!-- BEGIN_MODULE Entry_List -->
    (中略)
  <!-- END_MODULE Entry_List -->
<!-- END_MODULE Touch_Entry -->
```

のように、このような時に表示させたいというモジュールで囲む事で、表示させたり、消したりをコントロールします。また、タッチモジュールは40以上あり、例えば、以下のようなものが用意されています。

## 2-4.各種モジュールの設定

モジュールの細かい設定については、テンプレート上ではなく管理ページ上で設定します。例えば、一覧の表示件数や表示順、画像を表示する際のサイズの指定等の各種設定が可能です。

1.5より数が多くなってきたモジュールを整理し、カテゴリ毎に分けて表示するようになりました。



モジュールの設定で、例えばエントリーフォトの設定をしてみると、ここで表示順の設定や、表示件数、画像のサイズ設定等を行う事ができるようになっており、テンプレート上では管理ページ内で設定する項目を設定する事はありません。

## コンフィグ

表示順 ?	日時 (降順) ▾
表示件数 ?	<input type="text" value="3"/>
オフセット ?	<input type="text" value="0"/>
インデキシング ?	<input checked="" type="checkbox"/> インデキシングされているエントリーのみ表示する
シークレットエントリー ?	<input type="checkbox"/> ログインしていなくてもシークレットブログのエントリーを表示する
ユニット ?	<input type="text" value="1"/>
エントリーがない場合の表示 ?	<input type="checkbox"/> 指定された条件のエントリーが一つもない場合にnot foundを表示する (チェックなしで何も表示しない)

## イメージ

サイズ ?	<input type="text" value="160"/> x <input type="text" value="160"/>
トリミング ?	<input type="checkbox"/> 画像を指定されたサイズにトリミングする
拡大 (トリミングがoffの場合のみ) ?	<input type="checkbox"/> 画像が指定されたサイズに満たない場合に拡大する
表示位置 (トリミングがoffの場合のみ) ?	<input checked="" type="checkbox"/> 画像を指定されたサイズの中心に表示する

## 2-5.オリジナルモジュールを作る

---

a-blog cms ではモジュールが暗号化されていて修正する事もできませんが、オリジナルのモジュールを作れないわけではありません。

```
<!-- BEGIN_MODULE Aaa_Bbb -->
```

(この部分に各種HTMLを記述する)

```
<!-- END_MODULE Aaa_Bbb -->
```

のようなモジュールの記述がある際には、以下のようなプログラムを動作させる事になります。

```
/php/ACMS/GET/Aaa/Bbb.php
```

オリジナルのモジュールの作り方につきましては、今回は割愛し、別の機会に紹介する事とします。

### 3.URLコンテキスト

---

a-blog cmsでは、表示しているページがWebサイトの何処なのか、どのような状態であるかをURL上のパスとして表現しています。これをURLコンテキスト（URLによって表現されるページの文脈情報）と呼んでいます。個々のモジュールは、このURLコンテキストで表す条件を元に何を表示するかを決定する事になります。

### 3-1.URLコンテキストと3つの基本ページ

---

詳しくはテンプレートのところでも紹介しますが a-blog cms の基本ページとして、トップページ・一覧ページ・詳細ページに分けられます。

#### トップページ

一般的な運用として Webサーバのドキュメントルートに a-blog cms が設置された場合には、この状態でトップページが表示されます。また、個別のディレクトリ内に設置する事も可能で、その際には、設置先のディレクトリまで指定した状態がシステムとしてのトップページという扱いになります。

```
http://www.example.com/
```

#### 一覧ページ

URLがスラッシュで終わっている状態の時、基本的には a-blog cms では一覧ページが表示されます。

```
http://www.example.com/news/
```

#### 詳細ページ

URLが.htmlで終わるような時には、エントリーの詳細ページが表示されます。

```
http://www.example.com/news/entry-1.html
```

## 3-2.URLコンテキストとモジュールの関係

---

### 場所的なコンテキスト

ブログコード、カテゴリコード、エントリーコードのような情報から、どの情報を表示するかを指示する事で、表示させたい情報をモジュール内に編集します。

www.example.com というサーバの appleple というブログの news というカテゴリの一覧を表示する場合には以下のようなURLになります。

```
http://www.example.com/appleple/news/
```

www.example.com というサーバの appleple というブログの news というカテゴリの entry-1.html という一記事を表示する場合には以下のようなURLになります。

```
http://www.example.com/appleple/news/entry-1.html
```

www.example.com というサーバのルートブログの news というカテゴリの campaign という子カテゴリの一覧を表示する場合には以下のようなURLになります。

```
http://www.example.com/news/campaign/
```

### 状況的なコンテキスト

キーワードやタグによる検索、日付、ページ等の情報が含まれていた場合にも、モジュールが表示する情報が影響される事になり、情報を絞り込んで表示する事になります。

2ページ目を表示する場合には以下のようなURLになります。

```
http://www.example.com/news/campaign/page/2/
```

「2009年8月」を表示する場合には以下のようなURLになります。

```
http://www.example.com/news/campaign/2009/06/
```

ここでは、簡単なURLコンテキストのみを紹介しておきますが、カスタムフィールドの検索を含むURLコンテキストについては、カスタムフィールドを説明するところで紹介する事にします。

## 4. コンフィグ

4.

a-blog cms の設定をコンフィグと呼びます。このコンフィグはブログ毎にあり、子ブログでは個別の設定が必要になります。

下記のyamlファイルがコンフィグの初期値を設定しているファイルになり、子ブログを追加した際には、このファイルに設定してあるブログが追加される事になります。

```
/private/config.system.yaml
```

このコンフィグ画面では、主に以下の3つのグループに分かれた設定がブラウザ上でできるようになっています。

- ブログの設定
- エントリーやユニットの設定
- モジュールの設定

後半で紹介するルールやモジュールIDでもコンフィグの設定は出てきますが、ここで紹介するコンフィグについては、ブログ（システム）の基本設定という位置づけになります。

コンフィグ管理

a-blog cms > コンフィグ一覧

コンフィグ一覧

ブログ

- 機能設定
- 出力設定
- プロパティ設定
- メール設定
- アクセス設定
- テーマ設定
- 外部認証設定

エントリー

- 編集設定
- ユニット設定

モジュール

- エントリー
- ユニット・カテゴリ
- ユーザー・ブログ

- Entry\_Body  
エントリー本文
- Entry\_List  
エントリーリスト
- Entry\_Photo  
フォト
- Entry\_Headline  
ヘッドライン

## 4-1. ブログ設定

---

機能設定に、キャッシュの設定があります。大事な機能ですので、ここで紹介しておきます。a-blog cms では、ページにアクセスがあった部分を、アクセスがあった時にページの情報を作ります。この作ったページの情報をサーバ上に保存しておき、同じページへのアクセスがあった際には、このデータをそのまま使うようにします。それがキャッシュ機能といえます。

このキャッシュ機能はサイト制作中は変更した部分をすぐに確認できない事から、通常オフになっています。config.server.php の最後に以下のような設定があります。この1と設定されている部分を0にする事で、デバッグモードがオフの状態になり、キャッシュ機能が利用できるようになります。

```
define('DEBUG_MODE', 1);
```

キャッシュの有効時間や、子ブログが更新されたときにもキャッシュをリフレッシュするかどうかを指定するキャッシュの感度設定、クライアントのキャッシュの時間を設定すると、この時間内はクライアント側のキャッシュが使用されサーバにアクセスしない事になります。そのためページが更新された場合にも反映されないことがあり、管理ページにログインした時にブラウザの再読込ボタンを押したり、キャッシュを消したりしないとページが切り替わらないことがあります。

## 4-2. エントリー設定

---

<p style="margin-bottom:1em">エントリー関連については、最初の章である 1.ユニット で細かく紹介しています。</p>

### 4-3. モジュール設定

モジュールの個々の初期設定が可能です。初期設定をブラウザから変更する際にはこちらを修正するのですが、実際にサイトを構築していく際には、利用するモジュールにモジュールIDという個々の名前をつけて運用する事が多くなり、こちらの設定はあまり利用しない事が多いです。

また、モジュールの設定画面の右上には、ショートカットに追加というリンクがあります。この設定をクライアントに設定しやすくするためのリンクをダッシュボード上に用意する事ができます。こちらを設定する事で、どこを更新できるのか分かりやすくする効果があります。



## 5. テンプレート

---

a-blog cms で更新ができるページを作るためのHTMLファイルをテンプレートファイルといいます。テンプレートには、一番最初にお伝えしたようにモジュールを貼り付けていく事で、データベースに保存されている情報をHTMLに配置していきます。

エントリーの一覧の例：

```
<!-- BEGIN_MODULE Entry_List -->
<ul>
  <!-- BEGIN entry:loop -->
  <li><a href="{url}">{title}</a></li>
  <!-- END entry:loop -->
</ul>
<!-- END_MODULE Entry_List -->
```

<!-- BEGIN entry:loop -->から<!-- END entry:loop -->の間を繰り返し表示する事になります。また、モジュールのコメントタグについては実行後には削除され通常のHTMLソース側には表示されません。

また、a-blog cms ではテンプレートファイルはテキストファイルであれば、どのようなファイルでもテンプレートにする事が可能です。

## 5-1.インクルード機能

---

いくつかのテンプレートファイルが出来てくると、共通部分を1つにまとめて管理ができるようにしたいという事になるかと思えます。以下のような記述でインクルード（外部ファイルを読み込む）を設定します。

```
<!--#include file='/include/sample.html'-->
<!--#include file='http://www.example.com/include/sample.txt'-->
```

上記の表記については、SSIと同様の記述になりますが、a-blog cms がSSIを使ってインクルードをしているという事ではありません。Adobe Dreamweaver のデザインビューでインクルード後の画面が表示されるように同じ表記としています。

また、インクルードの記述をする際には、相対パスで記述する事も可能ですが、絶対パスで設定する事をオススメします。絶対パスのルートについてはテーマのディレクトリがルートになり、SSIとは違います。

## 5-2. グローバル変数

---

{title}のような変数はモジュールの中に記述されている必要がありますが、グローバル変数はテンプレートのどこに記述しても編集されます。テンプレートの編集する順序としては、グローバル変数、インクルード、モジュールの順で実行されます。

例：

- **{BLOG\_NAME}** 表示ページが属するブログの名前
- **{CATEGORY\_NAME}** 表示ページが属するカテゴリーの名前
- **{KEYWORD}** URLコンテキスト上で、指定されたキーワード
- **{CID}** カテゴリーID
- **{EID}** エントリーID

また、グローバル変数を利用したインクルード機能を使うと、表示ページのカテゴリー毎に違うテンプレートファイルを読み込むような事も可能になります。

```
<!--#include file='/include/category%{CID}.html'-->
```

このような組み合わせでテンプレートのカスタマイズをする事は多くありますので、大事なテクニックの1つになります。

### 5-3.校正オプション

---

{data}のような変数に{data}[option]のような表記でデータをプログラムで処理する事ができる機能を校正オプションと呼びます。いろいろな校正オプションがありますが、いくつかの例を紹介します。

文字数を指定した数で足切りします。（半角の幅は1、全角の幅は2となります。）

```
{text}[trim(13, '...')]
```

数字を千位毎にカンマ区切りにします。

```
{number}[number_format]
```

日付から和暦の年を算出して表示します。（1985/08/26[wareki] -> 昭和60年）

```
{date}[wareki]
```

日付を誕生日としたときの現在の年齢を表示します。（1985/08/26[age] -> 23）

```
{date}[age]
```

a-blog cms のプログラムは基本的には暗号化されていてプログラムを見る事はできませんが、この校正オプションのプログラムは暗号化されずに以下のファイルで見える事ができるようになっています。

```
/php/ACMS/Corrector.php
```

また、オリジナルのファイルを修正せずに、以下の場所にオリジナルの Corrector.php を作る事もできます。 /omake/php の中にサンプルファイルが入っていますので、こちらもチェックしてみてください。

```
/php/ACMS/User/Corrector.php
```

## 6. テーマ

---

テンプレートファイルやイメージファイル、CSSファイル、JavaScriptのファイル等を1つのフォルダにまとめたものをテーマと呼びます。a-blog cmsのカスタマイズをする際に作るものがテーマであり、そのテーマを利用した設定（コンフィグ）をする事でサイトが出来上がります。

## 6-1. テーマの設定

オリジナルのテーマを作る際に、一番最初に設定を変更する部分になります。カスタマイズ管理 > コンフィグ > ブログ > テーマ設定 にアクセスして下さい。

テーマ設定	
<b>テーマ</b>	
テーマ ディレクトリ名	site
<b>テンプレートファイル</b>	
トップページ	top.html
一覧ページ	index.html
詳細ページ	entry.html
エラーページ(404 Not Found)	404.html
管理ページ	admin.html
エントリー編集ページ	entry.html
ユニット追加ページ	entry.html
ログインページ	login.html
保存	

上記のような画面が表示されます。

### テーマ ディレクトリ名

themes ディレクトリ内にあるテーマディレクトリを指定します。

### テンプレートファイル

テンプレートのコントロールについては、テンプレートの章で紹介します。トップページ・一覧ページ・詳細ページ・エラーページ・管理ページ・エントリー編集ページ・ユニット追加ページ・ログインページの8種類のテンプレートを指定します。

一般的には、詳細ページとエントリー編集ページ・ユニット追加ページのテンプレートは同じに

設定しておくようにした方がいいでしょう。

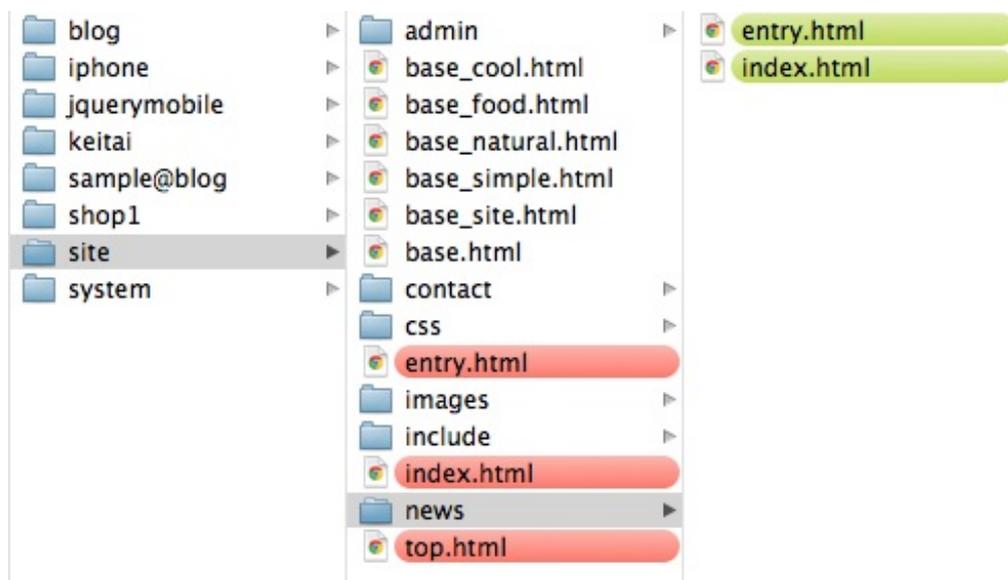
## 6-2. テーマとテンプレート

このテーマを複数用意する事によってコンテンツのデザインを切り換えて表示させる事は可能です。しかし、テーマではブログ単位に設定するか、この後に紹介するルール機能を利用して設定する事になります。

a-blog cms では、もっと簡単にカテゴリ毎にテンプレートを設定する事が可能です。テーマディレクトリ内にカテゴリコードのフォルダと、その中に必要なテンプレートファイルを用意します。また、エントリーコードのファイル名を用意する事で特定のエントリー専用のテンプレートも設定できます。

### news カテゴリの時

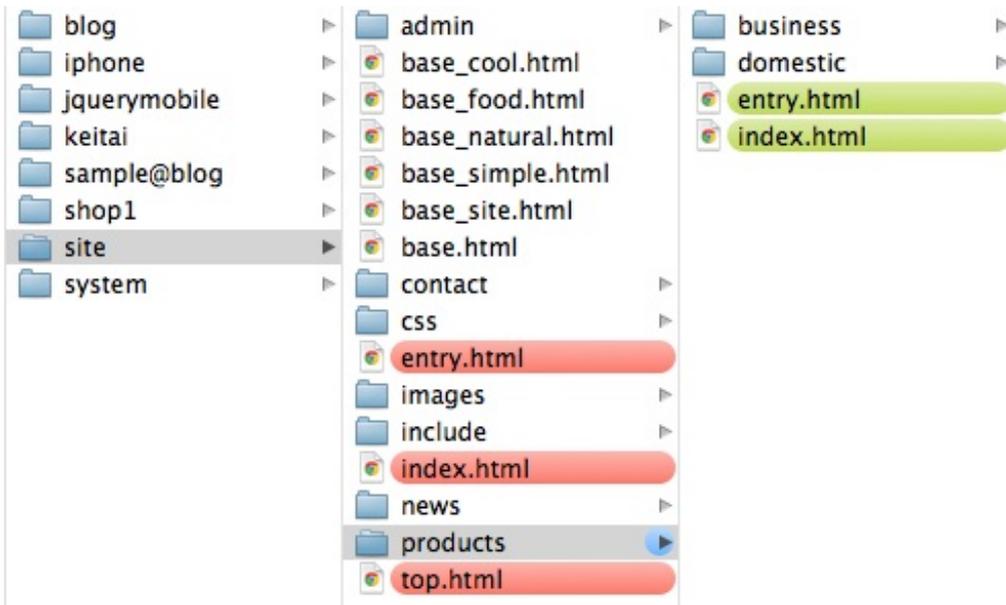
news というカテゴリには、site/news/index.html (一覧) と site/news/entry.html (詳細) のテンプレートが用意されています。この場合であれば、news のカテゴリの際には、これらのテンプレートファイルを利用する事になります。



もし、詳細ページのテンプレートが他のページと共用の設定にするのであれば、site/news/entry.html (詳細) のファイルを削除します。すると、テーマディレクトリの直下にあるファイルを参照するようになり site/entry.html (詳細) を利用するようになります。

### products カテゴリを作った時

注意する点としては、以下のように site/products/index.html (一覧) と site/products/entry.html (詳細) のテンプレートがあり、さらにサブカテゴリでいくつかのディレクトリが用意されています。



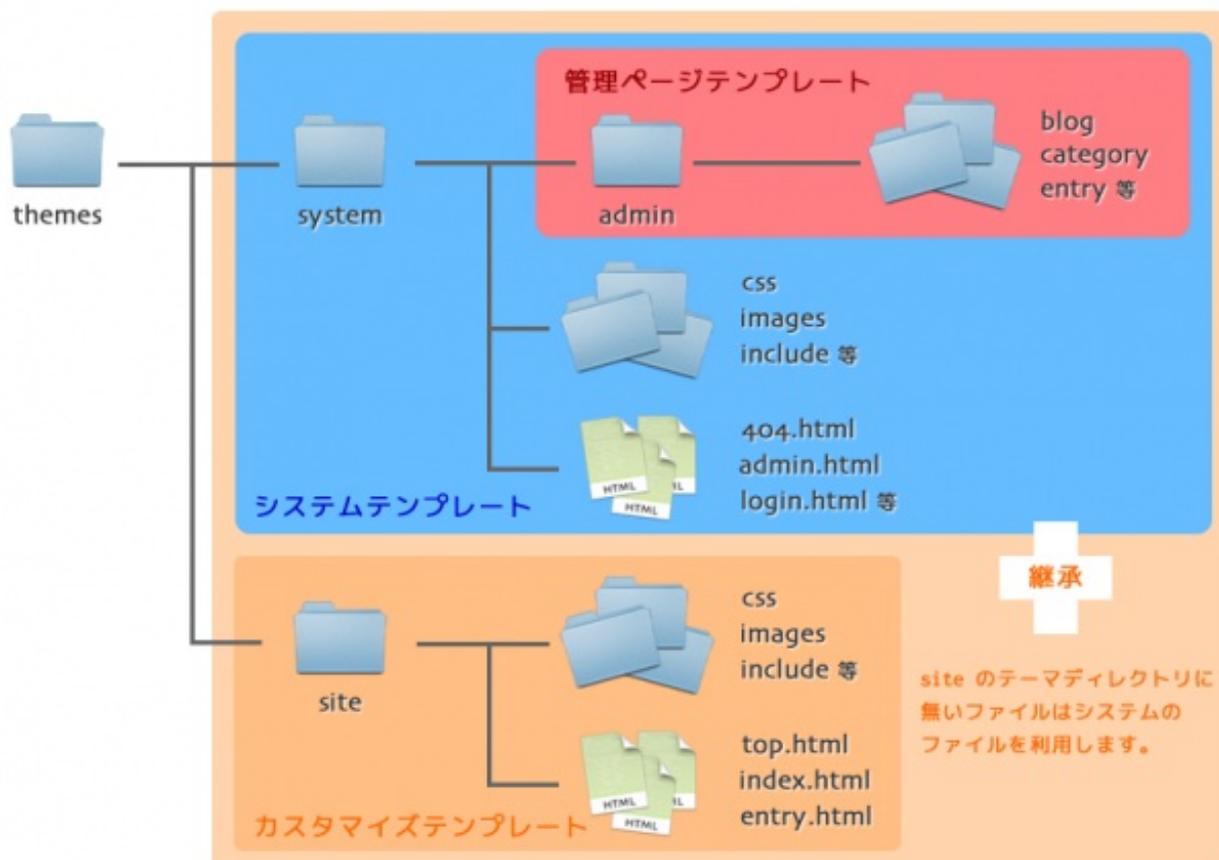
この場合に、クライアント側でサブカテゴリを追加した際には、テーマ内にそのカテゴリコードのディレクトリが存在しない事になる事から site/entry.html（詳細）を利用してしまふ事になり、希望するテンプレートでなくなる可能性があります。ご注意ください。

#### 表示テンプレートの確認

デバッグモード（config.server.php の DEBUG\_MODE）が 1 の時のみ上記のように、現在表示しているテンプレートがどのファイルなのかを表示させています。この部分で表示させるファイルの確認を行って下さい。



## 6-3. テーマの構成



テーマディレクトリ (themes) には、テーマ設定で指定するテーマディレクトリ以外に、system という a-blog cms のシステムで利用しているテーマが入っています。さらに system の中に admin というディレクトリがあり、ここに全ての管理ページのテンプレートファイルが格納されています。ですから a-blog cms の管理ページをカスタマイズする際には、admin 中のファイルを修正すればいい事になります。

テーマ設定で設定されているテーマ (site) と、system のテーマは継承関係にあり、テーマ設定で設定されているテーマに呼び出すファイルが存在しない時には、system のファイルをチェックし、存在していればそのファイルを表示させる事になります。例えば、login.html や 404.html は site のテーマの中には存在していませんが、表示される様になっています。

また、テーマ作成の作法として、system のファイルを直接修正するのではなく、テーマ設定で設定されているテーマにコピーして修正するようにして下さい。システムのアップデートの際に system ディレクトリはアップデートされる事になり、カスタマイズ部分が消えてしまう事になります。

## 6-4. テーマの継承

---

site テーマに無いファイルは、system のファイルを利用するようになっていますが、同様に子ブログ用のテーマを作る際に以下のようなテーマディレクトリを作成する事で、system との関係のようにテーマを継承する事が可能です。

```
blog@site
```

すぐには利用する事は無いかもしれませんが、複数のブログを1つのシステム内に設定してカスタマイズするようなサイトになった時には利用する事になります。

## 7. モジュールID

---

「モジュールID」は基本的には「モジュール」と表示内容が同じですが、モジュールにIDを付けることで、特定のモジュールを指定することができます。それにより、ルールや状況によって変化する内容を固定することができます。

URLコンテキストの情報を元にモジュールは、何を表示させるかを決定すると、これまで紹介してきましたが、このモジュールIDを利用する事で、URLコンテキストを利用せず、管理ページのモジュールIDで指定した条件を元にモジュールを動作させる事ができるようになります。

## 7-1.モジュールIDの設定

```
<!-- BEGIN_MODULE Entry_List id="topEntryList" -->
<ul>
  <!-- BEGIN entry:loop -->
  <li><a href="{url}">{title}</a></li>
  <!-- END entry:loop -->
</ul>
<!-- END_MODULE Entry_List -->
```

指定方法としては、テンプレート上のモジュールの BEGIN\_MODULE モジュール名の後に id="モジュールID" のように指定をします。

### モジュールID作成

#### ID情報

モジュール ?	<input type="text" value="▼ モジュールを選択"/>
id ?	<input type="text"/>
名前 ?	<input type="text"/>
説明 ?	<input type="text"/>
グローバル ?	<input type="checkbox"/> 下の階層のブログが利用することを許可する

#### 引数 (チェックするとURLコンテキストが優先されます)

ブログID (bid) ?	<input type="checkbox"/>	<input type="text"/>	(ID参照)
ユーザーID (uid) ?	<input type="checkbox"/>	<input type="text"/>	(ID参照)
カテゴリID (cid) ?	<input type="checkbox"/>	<input type="text"/>	(ID参照)
エントリーID (eid) ?	<input type="checkbox"/>	<input type="text"/>	
キーワード (keyword) ?	<input type="checkbox"/>	<input type="text"/>	
タグ (tag) ?	<input type="checkbox"/>	<input type="text"/>	
フィールド (field) ?	<input type="checkbox"/>	<input type="text"/>	
開始日時 (start) ?	<input type="checkbox"/>	日付: <input type="text"/>	時刻: <input type="text"/>

管理ページ側の指定については、どのモジュールなのかをプルダウンメニューから指定し、テン

プレート上に記述したモジュールIDを指定します。

モジュールID化する事で、ブログやカテゴリの階層の細かいコントロールが可能になります。

- 下層のブログやカテゴリを含めない
- 下層のブログやカテゴリも含める
- 下層のブログやカテゴリのみを対象とする

また、子ブログでも利用するようなモジュールについては、グローバルのチェックボックスにチェックをする事で、グローバルモジュールIDとして利用できるようになります。



id	名前	適用モジュール	グローバル	アクション
banner	サイドエリアのバナー	Banner	-	<a href="#">詳細 / モジュール</a>
topImage	トップページのメイン画像	Banner	-	<a href="#">詳細 / モジュール</a>
topHeadline	トップページお知らせ用	Entry_Headline	-	<a href="#">詳細 / モジュール</a>
entryList	エントリーリスト (サイド用)	Entry_List	-	<a href="#">詳細 / モジュール</a>
photoDefault	photoDefault	Entry_Photo	-	<a href="#">詳細 / モジュール</a>

モジュールIDで指定したモジュールについては、個別にモジュールの設定ができます。

## 7-2. モジュールIDの引数

---

Entry\_Body と Entry\_Headline については、URLコンテキストの値を自動で取得するような仕様になっていますが、他のモジュールについては指定する必要があります。そのためモジュールIDを設定しないとURLコンテキスト上のpageが反映されない事が起ります。

キーワード (keyword)	<input type="checkbox"/>	
タグ (tag)	<input type="checkbox"/>	
フィールド (field)	<input type="checkbox"/>	
開始日時 (start)	<input type="checkbox"/>	日付: <input type="text"/>
終了日時 (end)	<input type="checkbox"/>	日付: <input type="text"/>
ページ (page)	<input checked="" type="checkbox"/>	<input type="text"/>
表示順 (order)	<input type="checkbox"/>	

## 8.ルール

---

ルールという機能を利用する事で、条件を指定してコンフィグを設定する事ができるようになります。例えば、特定のデバイスやブラウザにのみコンフィグを設定したり、特定のカテゴリーの時のみコンフィグ設定したいという時に、このルールを設定します。

## 8-1.スマートフォンに対応する



a-blog cms では、スマートフォン専用プラグインとか、専用のプログラムがあるという事ではなく、このルール機能によって、ブラウザの名前（USER-AGENT）をチェックし設定を切り換えて表示されます。

### 詳細情報

エイリアスID (aid)	指定しない	
ユーザーID (uid)	指定しない	( ID参照 )
カテゴリID (cid)	指定しない	
エントリーID (eid)	指定しない	
ユーザーコード (ucd)	指定しない	CODE参照)
カテゴリコード (ccd)	指定しない	CODE参照)
エントリーコード (ecd)	指定しない	
ユーザーエージェント	=	

携帯電話  
docomo  
i-mode2.0  
au  
SoftBank  
WILLCOM  
スマートフォン  
iPhone  
Androidフォン  
タブレット  
iPad  
Androidタブレット  
✓ PC  
Internet Explorer  
6.x  
Safari  
Firefox

作成

このルールを設定を行った後に、一覧のコンフィグのリンクから、スマートフォンの時のコンフィグについて設定を行います。最低限、テーマの設定を変更する事になります。

この場合、PCとスマートフォンでは同じURLで違ったテーマを表示させる事ができるようになります。a-blog cms で標準搭載のキャッシュ機能もルールによって別で処理されますので、キャッシュの事は気にする必要はありません。

## 8-2.特定の 카테고리의時にユニット設定を変更する

---

細かな設定を進めていく際に、特定の 카테고리의時のみユニットの細かい設定を変更するといった事ができます。例えば、「商品紹介」と「お知らせ」では、ユニット設定の内容が違ったり、ユニット追加ボタン自体が違う、テキストタグセレクトの内容が違うという事までカスタマイズする事で、使わない場所では使わない設定を表示させない事を実現する事ができるようになります。

## 9. フォーム

---

サイトを公開しているのは、お客様からの注文や問い合わせを受けるためといった目的の事も多いと思います。a-blog cms では標準でフォーム機能も搭載しております。機能としては、以下のようなものが上げられます。

- サイトのどこにでも設置が可能
- メールの送信には2種類のメールを送れます
- 入力項目のチェック機能
- 入力項目のコンバート機能
- 複数ステップのフォームを作る事ができる
- DBに蓄積&CSVのダウンロード

## 9-1. フォームモジュール

---

フォームについても、モジュールとして提供しております。フォームの基本ステップとしては、新規・修正・確認・完了の4つのステップに分かれており、このステップを増やす事で複数ステップのフォームを用意できる事になります。

```
<!-- BEGIN_MODULE Form -->

<!-- BEGIN step-->
<!-- 新規 -->
<!--#include file="/include/form/insert.html"-->
<!-- END step -->

<!-- BEGIN step#reapply -->
<!-- 修正 -->
<!--#include file="/include/form/update.html"-->
<!-- END step#reapply -->

<!-- BEGIN step#confirm -->
<!-- 確認 -->
<!--#include file="/include/form/confirm.html"-->
<!-- END step#confirm -->

<!-- BEGIN step#result -->
<!-- 完了 -->
<!--#include file="/include/form/result.html"-->
<!-- END step#result -->

<!-- END_MODULE Form -->
```

## 9-2. フォームの基本

---

ここではテキストの入力欄についての表記は、以下のようになります。他のフォーム項目についても機能としては用意されておりますが、この章では割愛させていただきます。

表示されるフォームタグ(メールアドレスの入力欄)

```
<input type="text" name="mailaddress" value="" />
```

システムに mailaddress という項目がある事を伝える指定

```
<input type="hidden" name="field[]" value="mailaddress" />
```

必須入力を指示するための指定

```
<input type="hidden" name="mailaddress:validator#required" />
```

## 9-3. フォームの管理ページ

サイト管理 > フォーム にフォームを管理するページが用意されています。こちらに、フォームIDを設定し、そのフォームIDをテンプレート上に設定する事で、データベース上にフォームの入力情報を保存し、CSVダウンロードを可能にしています。また、メールのテンプレートの設定や、From, To等の設定も管理ページ上で設定します。

### 基本情報

フォームID <input type="text"/>	contact
フォーム名 <input type="text"/>	お問い合わせ

### 一般メール設定

SubjectTpl <input type="text"/>	/contact/form/subject.txt
BodyTpl <input type="text"/>	/contact/form/body.txt
To <input type="text"/>	
From <input type="text"/>	info@a-blogcms.jp
Cc <input type="text"/>	
Bcc <input type="text"/>	
Reply-To <input type="text"/>	
文字コード <input type="text"/>	ISO-2022-JP

※カンマ区切りで複数指定出来ます。

### 管理者宛メール設定

AdminSubjectTpl <input type="text"/>	/contact/form/adminsubject.txt
AdminBodyTpl <input type="text"/>	/contact/form/adminbody.txt
AdminTo <input type="text"/>	info@a-blogcms.jp
AdminFrom <input type="text"/>	
AdminCc <input type="text"/>	
AdminBcc <input type="text"/>	
AdminReply-To <input type="text"/>	

## 10. カスタムフィールド

---

a-blog cms のカスタムフィールドについては、テンプレートを手で書く必要がありますが、その分、自由度が高いフォームを作る事ができます。テーマの章で少し書いたように、a-blog cms では管理ページのテンプレートファイルを、カスタマイズしているテーマのディレクトリにコピーする事でオリジナルの管理ページを作る事ができます。

カスタムフィールドには以下の4種類のもので設定する事ができます。

- ブログ
- カテゴリー
- エントリー
- ユーザー

## 10-1.カスタムフィールドのタグの基本

---

以下のようにカスタムフィールドを作る事ができます。どこかでご覧になった事があるかと思います。フォームのHTMLと同様に書く事で a-blog cms のカスタムフィールドを作る事ができると理解して下さい。

表示されるフォームタグ(メールアドレスの入力欄)

```
<input type="text" name="mailaddress" value="{mailaddress}" />
```

システムに mailaddress という項目がある事を伝える指定

```
<input type="hidden" name="field[]" value="mailaddress" />
```

必須入力を指示するための指定

```
<input type="hidden" name="mailaddress:validator#required" />
```

## 10-2. ブログのカスタムフィールド

まず、どこのファイルを修正すればいいかを知る方法からご紹介します。まず、ブログの管理ページのフォームを開いてみます。

ブログ管理  
a-blog cms > ブログ詳細

ブログ編集

ID	1
ステータス	公開
名前	a-blog cms (日本語可)
ドメイン	localhost (英数字)
コードネーム	(英数字)
インデキシング	<input checked="" type="checkbox"/> リストに出す

保存

戻る

以下はローカルPC上にインストールした場合のアドレスになります。レンタルサーバにインストールした際には localhost の部分がオリジナルのドメインになります。

```
http://localhost/bid/1/admin/blog_edit/?edit=update&step=reapply
```

このテンプレートファイルは以下のファイルになります。

```
/themes/system/admin/blog/edit.html
```

これを site テーマ内で修正するのであれば、以下の場所にコピーして修正する事になります。

```
/themes/site/admin/blog/edit.html
```

これが基本ではありますが、この edit.html が次のバージョンアップでメンテナンスが入ってしまう事があるかもしれません。もちろん、このファイルを修正し、しっかりカスタマイズする事も方法としてはあります。

今回、より簡単にカスタムフィールドをカスタマイズ頂けるように、edit.html の中に書かれているインクルードの記述をご紹介します。この記述上の場所は何もファイルが存在していません。ですので、カスタマイズ前の段階では何も表示されません。

```
<!--#include file="/admin/blog/field.html" -->
```

この場所に field.html を置く事で、あたかも edit.html を修正したかのような効果がある事になります。

```
/themes/site/admin/blog/filed.html
```

テキストの入力欄を1つ、プルダウンメニューを1つ追加する際のサンプルは以下のようになります。

```
<h3>カスタムフィールド サンプル</h3>

<input type="text" name="sample1" value="{sample1}" />
<input type="hidden" name="field[]" value="sample1" />

<select name="sample2">
  <option value="aaa"{sample2:selected#aaa}>aaa</option>
  <option value="bbb"{sample2:selected#bbb}>bbb</option>
  <option value="ccc"{sample2:selected#ccc}>ccc</option>
</select>
<input type="hidden" name="field[]" value="description" />
```

このようにして多くのカスタムフィールドを設定する事で、柔軟にフォームを作成でき、クライアントに分かりやすいインターフェースを実装する事ができるようになります。

このカスタムフィールドのデータの表示には、Blog\_Field というモジュールを利用し、以下の様にテンプレートに記述します。

```
<!-- BEGIN_MODULE Blog_Field -->
<p> sample1 : {sample1}</p>
<p> sample2 : {sample2}</p>
<!-- END_MODULE Blog_Field -->
```

### 10-3. エントリーのカスタムフィールド

---

エントリーのカスタムフィールドについては、いろいろなコントロールが可能です。例えば、このカテゴリーの時とか、このブログの時、このエントリーの時のみというように、同じエントリーのフォームで違うフォームを表示させる事ができれば、必要ところで必要なカスタムフィールドのみを表示させる事が可能になります。

ブログのカスタムフィールドの時と同様にエントリーのカスタムフィールド用の field.html を用意します。

```
/themes/site/admin/entry/field.html
```

内容としては、以下のように記述します。

```
<!--#include file="/admin/entry/blog%{BID}.html" -->
<!--#include file="/admin/entry/category%{CID}.html" -->
<!--#include file="/admin/entry/entry%{EID}.html" -->
```

グローバル変数とインクルードを利用して、特定のブログや、カテゴリー、エントリーの時のみ読み込むファイルを用意します。例えば、カテゴリーIDが1の時には以下のようなファイルを用意します。

```
/themes/site/admin/entry/category1.html
```

エントリーのカスタムフィールドについては、エントリーについては修正時点でないとIDが確定していませんし、カテゴリーのプルダウンで変更してもカスタムフィールドは変化しませんので、注意する必要があります。