



個人でもコンテンツマネジメント

日記サイトを超越る

Blogサイトを作りたい!

日記サイトはかねてからウェブサイトを作る人たちの定番だ。そこに拍車をかけるように“ Blog ”がやってきた。Blog は日記サイトのような形態を持った個人ジャーナルだ。それよりも素晴らしいのはそのツールだ。誰もが日記を日記以上のクォリティーで作れるのだ。この新しい波に乗らない手はない。

協力: Japan Blog Association
photo: Tsushima Takao



Blog とは何か? 歴史と現状から紐解く『Blog 序説』

実践編に入る前に「そもそもBlogってなんなの」という話をまとめてみよう。「わざわざBlogなんて舶来言葉を持ち込まなくても、日本には日記サイトの文化があるし、ツールもある」と言う人もいる。米国でもBlogの定義に関しては諸説紛々だ。

Blogをとリまく現状とこれまでの流れ、そしてその本質と可能性を、ケビン・ワーバック氏のレポートから読み解く。

text: Kevin Werbach『Weblogs: A Powerful Emerging Communications Mechanism』より 注1
構成・翻訳: 先田千映

ウェブログに明確な定義はない 何でもアリだから面白い

Weblog ウェブログ^{注2}という言葉の定義については、誰もが納得するようなピットリとくるものは存在しない。それはそれで、むしろ結構なことだ。大事なのは「ウェブログを使ってユーザーが何ができるようになるか」ということなんだから。

実のところ、ウェブログという言葉自体が誤解を招く原因になっているのかもしれない。ときどきウェブサーバーのログファイルとごっちゃにする人もいるくらいだ。

「log」というのが「時系列の記録」としての機能を強調しすぎているんだね。それはそれで重要なんだけど、ほかにもたくさんの重要な使い方がある。「log」という言葉の響きには、シンプルで平易なテキスト、エピソード風の語り口を思わせるようなところがあるけど、実際にウェブログってのは、携帯から送信した写真を並べても、コンファレンスのリアルタイム報告でも、ニュース速報でも、なんでもいいわけで。どちらにせよ、今日広く使われている言葉でもあることだし、ここではウ

ェブログという呼び方で通すとしてしよう。

ウェブログには主に3つの特徴がある。

- ・一時的な方向性
- ・短くて頻繁な投稿
- ・個人または話題にフォーカスした視点

一時的な方向性:ウェブログ形式の本質は、時間の流れにある。典型的なウェブログには、すべての投稿(エントリー、あるいはポスト)に日付スタンプがあり、同じ日の投稿はその日付のもとにグループ化されている。時間というのは、情報を並べるのに便利な基準だけど、たとえばヤフーはカテゴリで情報を分類しているし、アマゾンの書評はタイプ(編集部発かユーザー発か)や評価(ランクの高い評者のものが上にくる)などの要素で並べられている。ウェブログの場合は時系列で並べることによって情報が自然にそして新鮮に保たれ、関連性も生まれるし、書き手が常に新しいものを追加し続けようという気になる効果もある。ほとんどのウェブログには過去の投稿の検索機能やアーカイブがあるけど、本当の面白いことはたいていその

時その場で起きている。

短くて頻繁な投稿:ウェブログでは、1つの投稿が2、3行とか数パラグラフということが多く、活動の盛んなところでは1日に何個も新規投稿がある。一般的にウェブログは、詳細な解説とか長編の物語を伝えるのに適した媒体ではないが、読み手に「そのときその場で書き手が何を考えているのか」をパッと理解させることができる。また、文章を精巧に磨き上げてから投稿するよりは、現時点で話題性のあることを投稿しやすいようにできている。別のサイトへのハイパーリンクだけという投稿もあり得る。たいていは部分引用とか書き手のコメントが付いているものだけど、さらに、投稿がテキストでなければならぬということもなく、HTMLで可能なことなら何でもOK。画像だけ、音声ファイルだけ、ビデオクリップだけの投稿ということもあり得る。

個人または話題にフォーカスした視点:ウェブログに命を吹き込むのは、作者の個性だ。万人向けのウェブログを作ることにあんまり

筆者紹介:

ケビン・ワーバック Kevin Werbach
テクノロジアナリスト、コンサルタント。
米国でもっとも定評のあるコンピュータ業界専門のニュースレター『Release 1.0』の前編集長で、現在もコントリビューティングエディターとして寄稿している。また、ハーバードビジネスレビュー誌や、フォーチュン誌、レッドヘリング誌など多数の雑誌、ウェブサイトなどでも執筆。コロラド州アスペンで毎年1度開催され、コンピュータ業界のエグゼクティブが一堂に会する「PCフォーラム」の共同創設者でもある。

Jump werbach.com

注1:この記事はネオテニーの協力で作られたケビン・ワーバック氏の英語による原文を日本語に訳したものに、編集部でタイトルや本文中の見出しなどを独自に付けた。

注2: Weblog

weblogはweb + logが語源となっており、これが縮まってblogと言われるようになった。ここでは原文中のweblogまたはblogをウェブログと、webloggerまたはblogger(ウェブログする人)をウェブログ作者と表記する。

BoingBoing

Jump www.boingboing.net

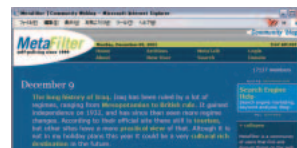
他の有名ウェブログでもしよっちゅうネタ元として登場する大御所サイト。扱分野はテクノロジーから馬鹿ネタまで幅広い。1988年創刊のミニコミ誌に端を発し、現在は主に4人のコントリビューターとゲストによるウェブログ形式をとっている。



MetaFilter

Jump www.metafilter.com

BoingBoingと双璧をなす一大ウェブログ。コミュニティウェブログと銘打つだけあって、登録ユーザーからの自由投稿を中心に活発な議論(コリイムダ話もアリ)が行われている。1日二十数本の投稿があり、話題によっては多少荒れがち。



意味はない。CNN.comとMSNBC.comが覇権を競っているところでさらに勝負を挑もうとするようなものだ。なぜ人がウェブログを見に来るかといえば、書き手のことを気にかけているから、あるいは自分が面白いと思うトピックを扱っているからなのだ。個性といっても、1人で作らなければならぬということではなくて、BoingBoingKJumpのようなところは複数の寄稿者によるものだし、MetaFilterKJumpのように登録すれば誰でも投稿できるところもある。

この3つ以外にも、スレッド形式の議論など、ウェブログを構成する要素はさまざまだが、それらは必須というわけではない。上記の3つの要素が相まって、ウェブログに深い意味でのコンテクスト性を与えている。ウェブログは個々人の存在がデジタルに発露したもので、そういう意味ではこれまでのWWWには足りなかったものだとはいえる。文章の洗練とか思想の深みには欠けるけど、その分スピードと多様性、語り口で補うのがウェブログだ。

元祖は『NCSAのWhat's New』？ウェブログの歴史をおさらい

ウェブログ的なサイトの例を探していくと、いくつかはWWWの黎明期までさかのぼることができる。Scripting NewsKJumpは1997年に始まっているけど、明らかに今僕らが理解している形でのウェブログだ。だけ

どジョン・バージャーのRobot WisdomKJumpについてはどうだろう？ これも1997年に始まっているし、weblogという言葉を作ったのも彼だとされているけど、リンク以上のオリジナルなコメントがないんだよね。96年に始まったマイケル・シッピーのThe Filter（現在は閉鎖）は？ または1994年に始まったジャスティン・ホルのLinks from the UndergroundKJumpは？ あとは、1993年に始まったNCSAのWhat's Newページ（現在は閉鎖）なんてのも、初期のWWWにおける新しいサイトをリストにしているという意味ではウェブログといえないだろうか？

これらのサイトのどれが史上初のウェブログか、なんてことを議論することにはあんまり意味がない。しかし、これらのサイトがみな手打ちのHTML、あるいは自作のツールで作られていたということから、現在のウェブログとの間に明らかな境界線が存在することがわかる。以前は、かなりの時間と専門知識が必要だったのだ。

1999年はウェブログにとってのターニングポイントとなった。PyraのBloggerや、現在のRadio Userlandの前身となるUserlandのManilaなど、ウェブログの管理を自動化するエンドユーザー向けの使いやすいツールがリリースされた。また、Slashdot これもウェブログと呼んでいいだろうが、Linuxやオープンソースへの関心が高まる中、爆発的な人気を獲得した年でもあった。Slashdotは使用しているソフトウェアのソースコードを無料で公開したので、派生的なオープンソースのツールがいくつか生まれるこ

とになった。さらにこの年には、ネットスケープがウェブサイト間でコンテンツをシンジケートするための規格としてRSSを開発している。瞬間にRSSはウェブログコンテンツのXML版を出力して情報を共有するための共通フォーマットとなった。多くの有名ウェブログが台頭してきたのもこの年で、ちょうどネットバブルがはじける直前でもあった。

2000年以降、現在ウェブログ上級者御用達となっているMovable Typeなどの新しいツールが開発され、Live Journal、BloggerのBlogSpotなど、新しい形のウェブログコミュニティサイトも登場してきた。

2001年、ウェブログはメインストリームを直撃した。9月11日のテロ事件が、格好の題材となったのだ。従来のニュースソースではウェブログのスピードと守備範囲の広さにはかなわなかった。なにしろ、元記事からウェブ上のどこに行けば最上の情報が得られるかのアドバイス、独自のコメントまでひとまとめにできたのだから。9月11日以降、ニューズウィーク誌などのメジャーな出版物で取り上げられたことにより、さらにウェブログのメディア露出は増えていく。

現在では100万を超えるウェブログサイトが存在し、想像しうる限りのトピックをカバーしている。ジャーナリストが自著の副読本として執筆しているウェブログもあれば、社外的なフォーラムあるいは社内用の情報源として使われているビジネス用のブログもある。また、ウェブログで収入を生み出している人もいる。

とはいえ、ウェブログはまだ成長の途

Scripting News

KJump www.scripting.com

ManilaやRadioUserlandなどのウェブログツールを提供しているUserland社のCEOでもあるディブ・ワイナー氏のウェブログ。氏はRSSやXML-RPC、SOAPなどの標準策定にも参加している。



Robot Wisdom

KJump www.robotwisdom.com

作者がニュースサイトを巡回して集めてきた記事タイトルとリンクのみを列記したシンプルな老舗ウェブログ。扱う分野はおもにテクノロジー・政治など。最近はおちょと精神世界に入りすぎとの評判。

The Filter

KJump www.theobvious.com

KJump sippey.com

作者によるウェブログは現在も運営中（Movable Type使用）。

Justin's Links

KJump www.links.net

開設以来膨張してきた雑多なコンテンツが現在はウェブログを中心にまとめられている。作者のホル氏は日本とカリフォルニアを行き来しているフリーランスジャーナリスト（その他いろいろ）。

NCSA's What's New Page

KJump archive.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/old-whats-new/whats-new-0693.html

現在もアーカイブ化されたものが残っている。内容は主に大学や軍の研究所！ などにお

けるウェブサーバーの運用開始情報、Mosaicなど初期ウェブブラウザの開発状況など。歴史を感じます。

Doc Searls' Weblog

KJump doc.weblogs.com

作者のシールズ氏はインターネット時代のマーケティングを考察する名著『The Cluetrain Manifesto』の共著者でもある。個人運営だが、他のウェブログの反応や読者のフィードバックを反映しつつ、ものすごい勢いでアップデートされているのが特徴。






上にある。ツールを作っているのは小さな会社かタダでソフトを提供しようという人たちばかりだし、いまだにおおむねテキストベースだ。ウェブログ内、あるいはウェブログ間でコンテンツを管理するソフト的なインフラも、いまだ限定的なものだ。ウェブログはメインストリームの意識に浸透してきたが、まだ「聞いたこともない」という人も多し、ただのおもちゃだと思って無視している人もいる。

Blogの主要なジャンルは3つ：技術・ニュース・日記

現在ウェブログの牽引役となっているのは3つのカテゴリーだ。ウェブログは本質的に個人的で分散型のものなだけで、今のところ、むらなく分散型というわけではない。大半は、Techblog「技術ブログ」、Warblog「戦争(ニュース)ブログ」、Journal「日記」のどれかに当てはまる。

最初期のブログの多くは現在Techblogと呼ばれているものに相当する。初期のウェブログ作者にはインターネット業界に自らかかわっていた人が多く、そういう人たちが興味のあることについて書いていたのだから、これはごく当然のことだ。インターネットの草創期と同じで、以前は比較的高度な知識を持った人にしかウェブログを管理することはできなかった。コミュニティの構成要員も必然的に偏ることになった。通常Techblogにはインターネット業界の動向に関する記事や、ほかのウェブログへのリンクが張られている。有名な例としては、Linux

Journal誌の編集者ドク・シールズのウェブログ  やキャメロン・バレットのCamworld  などが挙げられるだろう。

9月11日以後、突出してきたのが、ニュース報道に力を入れる新しいグループのウェブログだ。中でもテロ事件を取り上げることから始まったウェブログが多かったため、このタイプをWarblogと呼ぶようになった。Warblogは幅広い層の読み手を惹きつけることができるし、書き手の方も頻りにアップデートする傾向にあるので人気がある。中でもずば抜けているのは、法学者グレン・ライノルズのInstapundit  だろう。背後で動いている技術を意識しなくてもツールを使いこなすことは可能なので、Warblog作者はウェブログを新しいタイプのニュースチャンネルと見なす傾向にある。

3つ目に挙げられるのがJournal、日記タイプのウェブログで、これは書き手個人の生活を時系列で記録するもの。このカテゴリーから連想されがちなのは、お馬鹿な妄想を画面上で吐き散らかすティーンエージャーだ。この連想は一面真実を突いてはいるんだけど、でも決定的にフェアじゃない。ウェブログは、人々が自分の関心事について語るための強力なメカニズムだし、語りたがっている人はたくさんいる。それを聞きたがっている読み手だって大勢いるんだから。

以上の3カテゴリーは、ウェブログが、現実の世界や既存のオンラインコミュニティの動きと互いに影響を与え合いながら進化してきた過程を反映するものだ。

ある意味、ウェブログはウェブそのものと

同じように成長してきたとも言える。最初期のサイトは個人的かつ愛のこもった手作りの成果だったが、新しいツールが手に入りやすくなるにつれ、コンセプトも広まった。みんなに見てもらうことも簡単になった。またウェブログにも商業価値を潜在的に秘めたアプリケーションがたくさんある。

しかし、ウェブログが離陸したのは、インターネットバブルがはじけた瞬間だった。ようやくクリティカルマスに達した時点では、ホットで新しいテクノロジーに対する投資も需要も豊富とはいえない状況。そのうえ、ウェブログは必然的にパーソナルなものだから、ますますビジネスとの関連性が見出しにくい。ウェブにとつてのネットスケープやアマゾンのように、ウェブログを広く一般向けのビジネスストーリーに変えるような存在は現れないだろう。それでもウェブログの重要性は今後も増すことになるはずだ。

まとめ：

ウェブログという言葉の明確な定義は存在しない。実のところ、ウェブログを狭義で意味づけることは、かえってその真の価値を曖昧にしてしまうことになる。ウェブログであるかないかは、その一時的な方向性、短く頻繁な投稿、個人または話題にフォーカスした視点で見分けることができる。ウェブログが人気を集めたのは、使いやすいツールやRSSがリリースされた1999年のことだ。現在では、100万を超えるブログが存在し、ありとあらゆるトピックを網羅している。

Camworld

 www.camworld.com

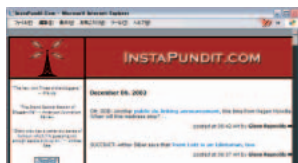
現在は小休止中。2003年の年明けにリアルオープン予定とのこと。ちなみにwebcamに関する話題はない。



Instapundit

 www.instapundit.com

テネシー大学法学部教授によるニュースクリップ。「短くて頻繁な投稿」の見本のようなウェブログだ。「ブロッガー村のグランドセントラル駅」と評されている(らしい)。



RSS(RDF Site Summary)

ネットスケープがRSS(当初はRich Site Summary または Really Simple Syndication)を開発したのは、もともとは同社のポータルNetCenter用にコンテンツを分類するためだった。Userland社のデイブ・ワイナー氏などのウェブログ作者たちは、ウェブログ間でのニュース記事の相互配送を自動化するのにこの規格が便利だと気がついた。2000年、ネームスペースやRDF(Resource Definition Format)を用いて標準を正式化しようとする動きにワイナー氏が異を唱えたことにより、RSS標準は二派に分

岐する。おそらくこのことにより、RSSの利用は比較的限定的なものにとどまってきた。主要ウェブログツールや商用ニュースサイトのポータルNetCenter用にコンテンツは自動的にRSSフィードを出力するようになり、スタンドアローンのアプリケーションとしてのRSSニュースリーダーもいくつが存在する。しかし、現状では、RSS構造の利点を活かした“セマンティックウェブ”(意味を持つウェブアプリケーションを作るにあたって、その潜在的な可能性を活かしている)とは言えない。

【インストール編】

Movable TypeでBlogを始めよう

今回は「Movable Type」という高機能なツールを使ってBlogサイトを構築してみる。このツールを使えば、Blogのような日記型のコンテンツを簡単に管理できるようになるのだ。まさに、パーソナルなコンテンツマネージメントシステムと言えるだろう。ここではMovable Typeのインストール方法を紹介しよう。

text: Hirata Daiji

Movable Typeはこんなソフト

Movable Typeはベンジャミン・トロットとメナ・G・トロット夫妻が2001年10月に発表したBlogのためのソフトウェアです。CGIを使ったいわゆるウェブアプリケーションで、ホスティングサービスなどにインストールするとブラウザを介して簡単にBlogコンテンツの生成、編集、更新ができます。

具体的には、Blog用に記述したテキストデータ(タイトルや本文など)がデータベースに保存され、リビルド(再構築)という操作を行うことで、HTMLのテンプレートファイルに沿ってBlogのHTMLファイルが生成されます。いったん、BlogのデータがHTMLファイルとして生成されたとしても、テキストを書き換えて(たとえば本文だけを書き直して)リビルドするだけで、書き換えたBlogのHTMLファイルが生成されます。また、テンプレートファイルを書き換えてデザインを変更する場合、すでに生成してあるHTMLファイルの1つ1つを書き直さなくても、サイト全体のデザインを変更することもできます。こういった「コンテンツマネージメント」としての機能が非常に注目を集めています(図1)。

Movable Typeは現在まで多くの改良が重ねられ、最新版はバージョン2.51になっています。普通のBlogやオンラインジャーナルだけでなく、フォトアルバムなど、工夫次第でいろいろな使い方ができるようになっています。Movable TypeはPerlで書かれており、拡張性も優れていて、作者以外の人でも機能を拡張できます。

個人利用もしくは非商用目的であれば無料で使えますが、寄付を募っているのも、もし気に入ったらいくら支払うといいでしょ

う。20ドル以上寄付すると、Movable Typeの自家ウェブサイト、自分のBlogのアップデート情報を載せてくれるようになります。なお、商用で使う場合は150ドルを支払わなければなりません。

Movable Typeは英語での利用が中心であるため、国際化についてはまだまだ不十分です。しかし、Perlで書かれているので、簡易的に日本語で利用できるように工夫することはできます。また、作者に対して技術的なフィードバックが行われており、最新版では、日本語やその他の言語の利用を意識した設定も加えられています。ここでは、日本語で使うことを前提にしたMovable Typeのインストール方法を紹介します。

Step 1*

Movable Typeを使うための準備をしよう

Movable Typeをインストールするサーバーなどの条件については図2にまとめておきました。ホスティングサービスを利用する場合は図2の条件を満たしているかどうかをチェックしてみてください。

次に、Movable Typeのセットアップに必要なプログラムをダウンロードします。まずは、Movable Typeの本体を自家サイトからダウンロードします。図2に示したダウンロードのページから「Full version, With Libraries」を選ぶと「MT-2.51-full-lib.tar.gz」というファイルがダウンロードされます。

図1. Movable Typeの仕組み

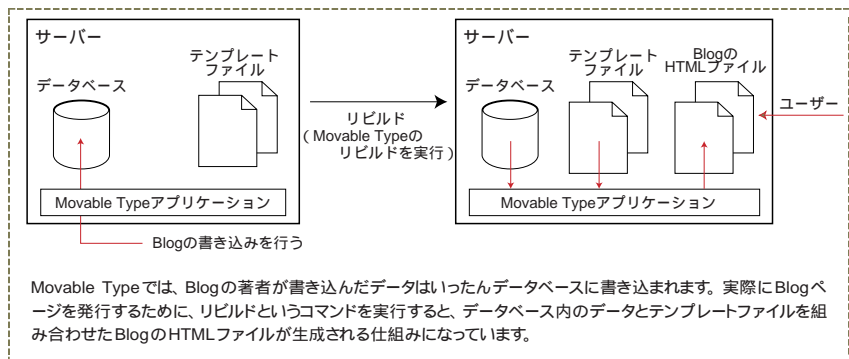


図2. インストールに必要なものを用意する

Movable Typeをインストールするサーバーとクライアントの条件	必要なソフトウェアのダウンロード先URL
・独自のCGIを動かせるウェブサーバー ・Perl 5.004_04以上がウェブサーバーにインストールされていること ・FTPなどファイルをサーバーにアップロードできる環境 ・Javascriptとcookieが利用できるウェブブラウザ	Movable Type(本体) www.movabletype.org/download.shtml Movable Type Japanese Language Pack(日本語化ツール) www.movabletype.org/resources.shtml#languages www.movabletype.org/downloads/mt-ja.tar.gz

同じく、Japanese Language Packもダウンロードしておきましょう。これはMovable Typeの設定画面を日本語化するためのものです。ファイル名は「mt-ja.tar.gz」です。

Step2**

ファイルを手入して置き場所を作ろう

今回のインストールでは、漢字コードに UTF-8(ユニコード)を利用して Movable Typeをホスティングサービスなどのサーバーに設置します。Movable Type自体は漢字コードを内部で処理しないため、EUC-JPやシフトJISのような漢字コードでも利用できますが、Language Packを利用するためには、UTF-8かEUC-JPのいずれかにする必要があります。EUC-JPを使う場合は、これ以降の設定で「UTF-8」と書かれている部分を「EUC-JP」と読み代えて設定してください。

また、Blogのデータを蓄積しておくデータベースシステムとして、今回は手軽に利用できる Berkeley DB を選びました(Movable

TypeではMySQLも利用できます)。実際のインストール手順は次のように行います。

① ダウンロードした MT-2.51-full-lib.tar.gz という圧縮ファイルを、Lhasaなどの解凍ツールを使って展開します。

② 上記のファイルを展開して現れるファイルは図3のとおりです。「mt.cgi」をはじめとする拡張子が「.cgi」のファイル(以下、CGIファイル)は、Perlのスクリプトです。したがってCGIファイルすべてに、インストール先のサーバーのPerlのパスを設定しなければなりません。初期設定では「/usr/bin/perl」になっていますので、自分が使っているホスティングサービスなどでこれとは違う環境であれば、エディターを使ってすべてのCGIファイルを書き換えます(図4)。

③ 展開したファイルのサーバーへのインストール先として以下の4つのディレクトリーを決める必要があります。

A CGIファイルとライブラリーを置くディレクトリー

B Movable Typeのシステムが利用する画像やスタイルシートといったファイルを置くディレクトリー

C データベースのディレクトリー

D 実際にBlogとして表示するHTMLファイルを置くディレクトリー

ここでは、例として図5のような環境を考えます。ディレクトリーの対応として以下のように設置することにします。

Aを「/www/cgi-bin/」

Bを「/www/htdocs/movabletype/」

Cを「/www/db」

Dを「/www/htdocs/mt/」

なお、サイトにアクセスするためのURLはそれぞれ以下のような対応になります。

Aは「http://サイト名/cgi-bin/」

Bは「http://サイト名/movabletype/」

Dは「http://サイト名/mt/」

なお、「サイト名」はMovable Typeをインストールするサーバーのドメイン名です。また、CのデータベースのディレクトリーにはBlogの重要な情報が置かれるため、ブラウザで直接アクセスできないようにしておく必要があります。上記のサイトのパスは、必ずしもこのようなディレクトリー構造になっている必要はないので、自分の環境に合わせて設定してください。

図4. 各CGIファイルのPerlのパス

```
#!/usr/bin/perl -w
```

上記は各CGIファイルの1行目に書かれているPerlのパスです。通常は初期設定のままでもいいですが、ホスティングサービスによってはPerlのパスが違うことがあるので、サービスのマニュアルなどで確認して各CGIファイルの設定を変更しましょう。

図3. Movable Type のインストールファイルの内容

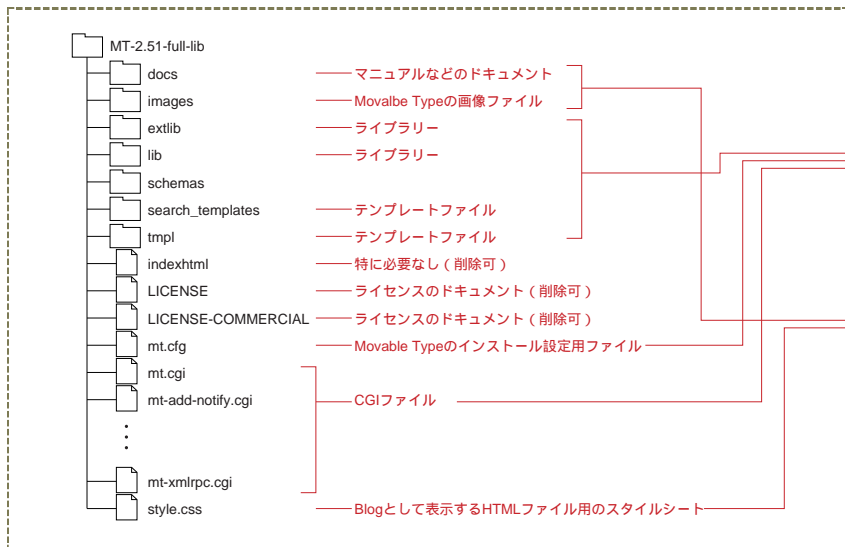
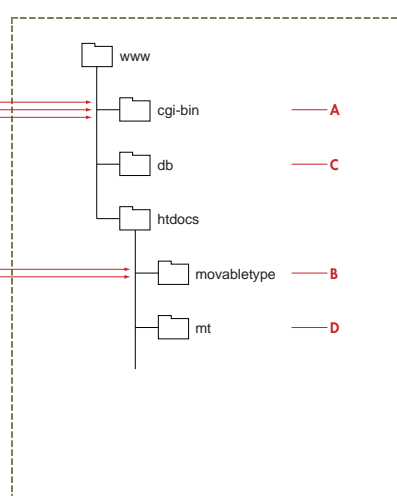


図5. Movable Type をインストールするサーバーのディレクトリー



Step 3***

mt.cfg ファイルを書き換える

次にエディターを使って、Movable Type のインストール設定ファイル「mt.cfg」を図6の①~⑧のように変更します。変更が終わったら、mt.cfg ファイルを保存しておきます。

① 14行目の「CGIPath」ではCGIファイルを置くディレクトリ(図5のA)をURLの形式で指定します。

② 22行目の「DataSource」ではデータベースのファイルを置くディレクトリ(図5のC)を指定します。mt.cfgを置くディレクトリ(図5のA)から相対パスで指定します。

③ 35行目の「StaticWebPath」ではMovable Typeのシステムで利用する画像やスタイルシートを置くディレクトリ(図5の)を指定します。URLで指し示す場合の「http://サイト名」以下のパスとしてこのディ

レクトリを指定します。図6のように行頭の「#」を外しておきましょう。

④ 63行目の「MailTransfer」は、Movable Type自身がメールを送信する機能を使う場合に設定しておくものです。Movable TypeをインストールするサーバーからSMTPが使えるときは63行目と64行目の「#」を消して、64行目に自分のサイト内のSMTPサーバー名(たとえばsmtp.example.jpのようなサーバーの名前)を設定します。

⑤ 114行目から117行目までは、suexecというセキュリティを強化した機能がサーバーにある場合に、Movable Typeが利用するデータベースファイルへのパーミッション(アクセス権と実行権)を設定しておくものです。サーバーでsuexecが実装されていれば、行頭にある「#」を外しておきましょう。suexecが実装されているかどうかは、サーバー管理者に確認してください。

⑥ 198行目の「NoHTMLEntities」は日本語の文字化けを避けるため、行頭にある「#」を外しておきましょう。

⑦ 216行目の「ImageDriver」は、サーバーにNetPBMもしくはImageMagickといった画像用のライブラリーがあるときに設定するものです。どちらかのライブラリーがあると、Movable Typeを使って画像ファイルをアップロードする際に、自動的にサムネイル画像を作成することができます。この機能を使う場合には、行頭の「#」を外しましょう(ImageMagickを使う場合には図6の「NetPBM」の部分で「ImageMagick」に書き換えます)。

NetPBMがインストールされているときは、インストールされているディレクトリを確認しましょう。Movable Typeは実行時に自動的に「/usr/local/netpbm/bin」「/usr/local/bin」「/usr/bin」を探すのですが、もし違うディレクトリにインストールされているときは、「pnmscale」というコマンドがあるディレクトリを227行目の「#」を外して「NetPBMPATH」に指定します。NetPBMがインストールされているかどうかは、サーバーの管理者に確認してください。

⑧ 252行目の「PublishCharset」はMovable Typeが生成するHTMLファイルの文字コードを設定します。文字コードをUTF-8にするには、行頭の「#」を外してShift_JISとなっているのを「UTF-8」に変更します。

図 6. mt.cfg の記述の変更

```
ライン
1  ## Movable Type configuration file [mt.cfg] ##
:
14 CGIPath http://サイト名/cgi-bin/ ①
:
22 DataSource ./db ②
:
34 #
35 StaticWebPath /movabletype/ ③
36 #
:
63 # MailTransfer smtp
64 # SMTPServer SMTPサーバー名 ④
65 #
:
114 # DBUmask 0022
115 # HTMLUmask 0022
116 # UploadUmask 0022
117 # DirUmask 0022 ⑤
:
198 NoHTMLEntities 1 ⑥
:
216 # ImageDriver NetPBM ⑦
:
227 # NetPBMPATH /home/watashi/netpbm/bin ⑧
:
252 PublishCharset UTF-8 ⑨
:
```

—— は変更か所
..... は該当する場合の変更か所

Step 4****

Movable Typeの日本語環境を設定する

英語のみで使うのであれば、以上の設定であとはファイルをサーバーにアップロードすればいいのですが、日本語を使えるようにするために、いくつかのファイルを変更する必要があります。変更する設定ファイルについては図7を参照してください。

① step2の①で MT-2.51-full-lib.tar.gzを

展開したフォルダーの中にある「lib」フォルダーの下にある「MT」フォルダーにある「default-templates.pl」ファイルを書き換えます。7、169、199、237、302、375、506、818、917、997、1189行目にある「iso-8859-1」という記述をすべて「UTF-8」に置換します。また、176、514行目の「en-us」という記述を「ja」に変更します。これにより、BlogのHTMLファイルはすべてUTF-8のコードで出力されるようになります。

2 1 の default-templates.plと同じフォルダーの中にある「App.pm」ファイルも変更します。47行目の記述を図8のように変更します。これによりMovable Typeをサーバーで実行したときに、「mt-comments.cgi」などのCGIが出力するHTMLファイルのCharset(文字コード設定)が iso-8859-1からmt.cfgで設定したものの「UTF-8」に変更されます。

3 Mac OSを利用している場合は、IE 5.2などのブラウザでは、標準のフォント設定のままだと編集画面やコメントの入力画面で文字化けが起きることがあります。styles.cssに記述されているfont-familyの設定をすべて削除したうえで、textareaについてののみ、

日本語のフォントを設定します。具体的には、styles.cssのfont-familyの記述を削除して、

最終行のあとに図9の記述を追加します。

図8. App.pmの記述の変更(47行目)

```

if (my $charset = $app->{charset}) {          変更前
}

if (my $charset = $app->{cfg}->PublishCharset) {  変更後
}
    
```

図9. style.cssの記述の追加(最終行)

```

textarea {
    font-family: 'Hiragino Kaku Gothic Pro W3', 'Osaka';
}
    
```

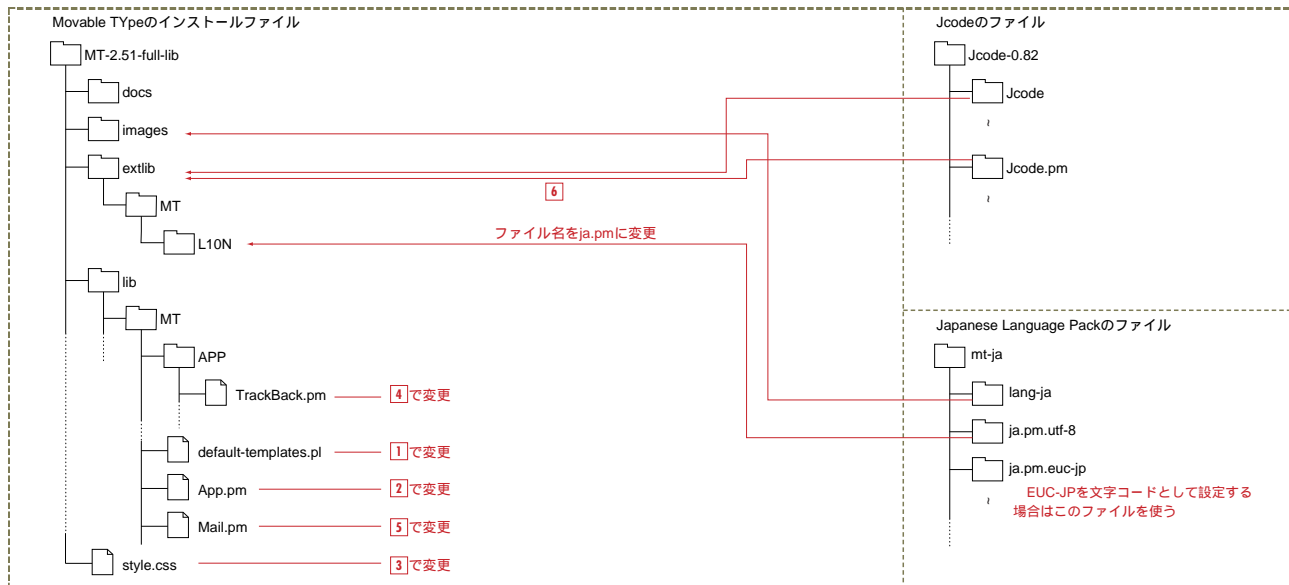
図10. Mail.pmの記述の追加

```

ライン
1      # Copyright 2001, 2002 Benjamin Trott. This co
:
:
53     sub no_utf8 {
:         require Jcode;
:         for (@_) {
56             next if !defined $_;
:                 $_ = pack 'COA*', $_;
:                 $_ = Jcode->new($_)->utf-8
:         }
71     }
:     <?xml version="1.0" encoding="UTF-8"?>
:
79     <?xml version="1.0" encoding="UTF-8"?>
:
:
266    <language>ja</language>
    
```

□ は追加か所
— は変更か所

図7. style.cssの記述の追加(最終行)



4 Movable Typeでは異なる文字コードを利用しているサイトからTrackBack(後述)を受けると文字化けします。このため、「TrackBack.pm」ファイルの53行目の次の行と56行目の次の行に図10のように文字列を追加します。また、71行目と79行目の「iso-8859-1」を「UTF-8」に置換して、266行目の「en-us」を「ja」に置換します。

5 内部コードがUTF-8のままだと、SMTPを使ってメールを送信する場合、文字化けするため、「Mail.pm」ファイルの20行目の次の行に図11のような記述を追加します(詳しくはみらの氏のサイトを参照 rebecca.ac/milano/mt/)

6 TrackBackとMailの文字化け対策には「Jcode.pm」という日本語処理のためのPerlのモジュールを利用しているため、もしサイトにJcode.pmが無いときはインストールする必要があります。下記のURLなどからダウンロードした「Jcode-0.82.tar.gz」ファイルを展開し、「Jcode.pm」ファイルと「Jcode」フォルダー(中身をそのまま)を図7のように「extlib」

フォルダーの下に置いてください。

search.cpan.org/CPAN/authors/id/D/DANKOGAI/Jcode-0.82.tar.gz

7 Japanese Language Packをセットアップします。まず、step1でダウンロードした「mt-ja.tar.gz」ファイルを展開します。展開した中にある「ja.pm.utf8」ファイルを「ja.pm」というファイル名に変更し、図7のように「L10N」ディレクトリーの下に置きます。同様に「lang-ja」フォルダー(中身はそのまま)を図7のように「images」ディレクトリーの下に置きます。

Step 5***** ファイルをアップロードする

最後にFTPツールなどを使ってサーバーにMovable Typeのファイルをアップロードします。step2に決めたおりのディレクトリーにファイルを置いていきます(図5参照)。

1 imagesとdocの両フォルダーおよびstyles.cssを除くすべてのファイルをサーバ

側の「cgi-bin」ディレクトリーにアップロードします。CGIファイルは実行属性が必要なので、パーミッションの変更が必要になります。パーミッションの変更は「755」にします(下囲み参照)。

2 次にimages、docの両フォルダーとstyles.cssをstep2の3で決めたBのディレクトリーにアップロードします。今回の例の場合は、「/www/htdocs/movabletype/」ディレクトリーになります。また、BlogのHTMLファイルが生成されるディレクトリー(/www/htdocs/mt/)とデータベースファイルを置くディレクトリー(/www/db/)をあらかじめ作っておきます。この2つのディレクトリーは、Movable TypeのプログラムであるCGIファイルから書き込みができるようにパーミッションを設定しておきます。

3 すべてのファイルのアップロードとパーミッションの設定ができればブラウザから、初期設定をします。初期設定はCGIファイルの1つ「mt-load.cgi」を起動して行います。今回の例ですと下記のURLをブラウザで指定して開きます。

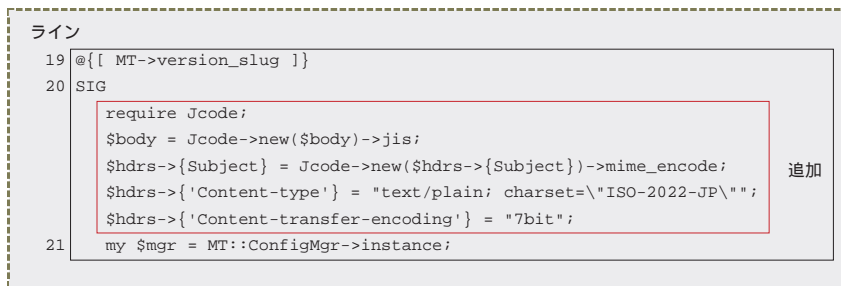
<http://サイト名/cgi-bin/mt-load.cgi>

うまく実行されれば、Movable Typeの設置は完了です。安全のため、mt-load.cgiとmt-check.cgiをサーバーから削除しておきます。

4 mt-load.cgiが正常に終了しないときは、以下のポイントを確認し直しましょう。

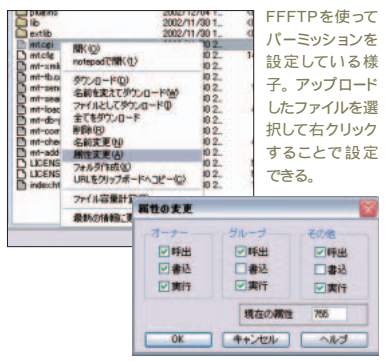
- ・CGIのパーミッションの問題などが考えられます。実行属性が正しいか(パーミッションが「755」)よく確認してください。
- ・データベースのディレクトリーを作成し忘れていないかを確認してください。このディレクトリーはCGIファイルから書き込めるように設定してください。
- ・システム環境などは、mt-check.cgiを実行すると一部がわかります。システム環境の設定がmt.cfgファイルと異なっている可能性があるため、このCGIファイルをブラウザで開いて確認してみてください。

図11. Mail.pmの記述の追加



パーミッションを設定しよう

パーミッションとはサーバー上でのユーザーに対するファイルのアクセス権や実行権を設定するものです。パーミッションが「755」と数字3つで表現されているのは、それぞれが、ファイルの「オーナー」、「オーナーが属するグループのユーザー」、「その他のユーザー」のそれぞれのファイルのアクセス(実行)権を表しています。数字はファイルの「実行:4」「書き込み:2」「読み出し:1」を加算することで表現します。つまり、7は4+2+1でファイルのすべての操作が可能で、5は4+1で実行と読み出しが可能であることを表しています。パーミッションはFTPツールなどでファイルをアップロードした際に設定できます。



【セットアップ&利用編】

Movable Type で Blog を実際に始めよう

Movable Type はセットアップした1つのサイトで複数のユーザーがログインできる。しかも1ユーザーで複数の Blog を管理することもできる。テンプレートファイルを書き換えることで自由に Blog のページのデザインも一度に変えられるのだ。今回は Movable Type のたくさんの機能の中から重要なものをピックアップして紹介していこう。

text: Hirata Daiji

Movable Type を使うための初期設定をしよう

1 インストールが済んだらすぐに Blog の書き込みを始めたいところですが、初期設定が必要です。まずは下記の URL をブラウザに入力して、Movable Type のログイン画面を表示させます。

`http://サイト名/cgi-bin/mt.cgi`

2 ログイン画面でユーザー名に「Melody」をパスワードに「Nelson」を入力して「Log In」を押します(大文字と小文字に注意)

3 ログインを済ませて最初に行うのは、ユ

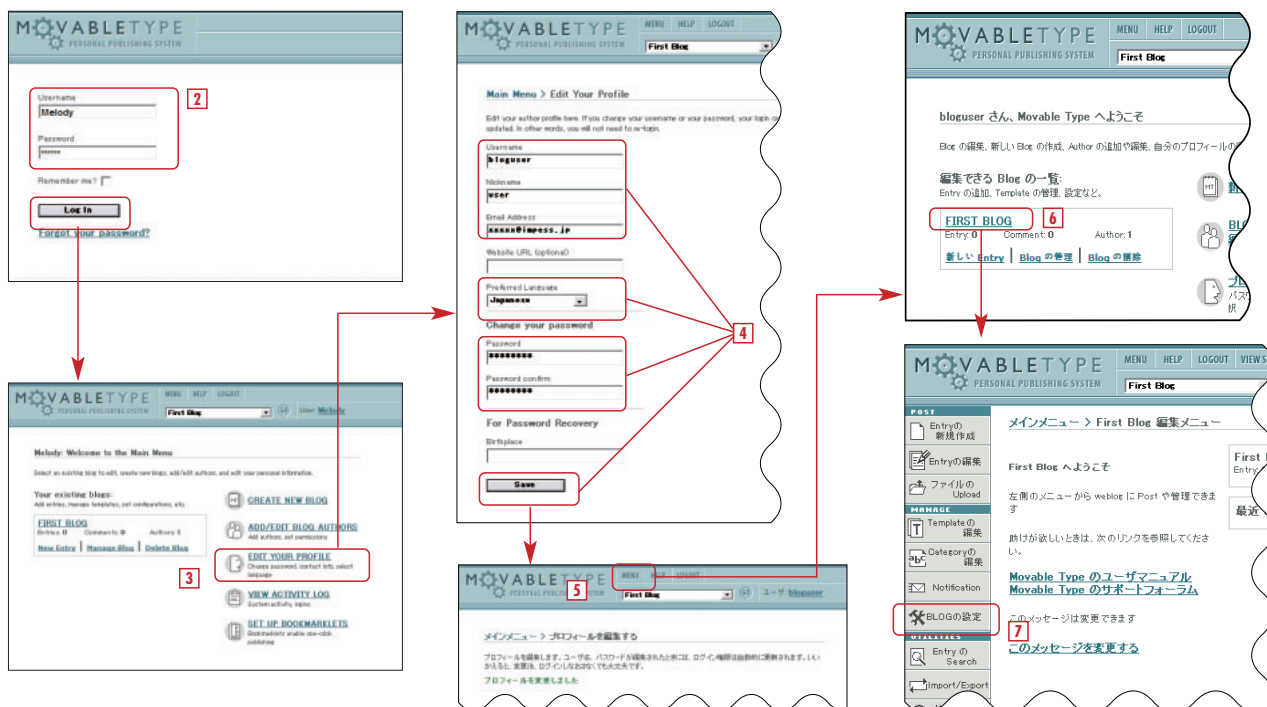
ーザーアカウントの設定です。「EDIT YOUR PROFILE」を押して次の画面に移ります。

4 表示された画面で「Username」を Medoly から自分が使いたいアカウント名(ここでは bloguser)に書き換えます。「Nickname」「Email Address」なども入力しておきましょう。「Preferred Language」では Movable Type の編集画面を日本語に設定するので、「Japansene」を選択します。「Password」「Password confirm」に新しいパスワードを入力して「Save」を押して保存します。

5 保存が終わると、画面が日本語に切り替わります。次に画面上の「MENU」を押して最初の画面に戻ります。

6 Movable Type では最初から1つ「First Blog」という Blog が用意されています。最初はこの基にして Blog を立ち上げてみましょう。「編集できる Blog の一覧」から「First Blog」をクリックします。

7 First Blog の編集メニューが表示されるので、左側のメニューから「BLOG の設定」を押します。



8 Blogの「基本設定」の画面が表示されます。まず「Blogの名前」を「First Blog」から自分が付けたい名前に変更します。この名前はBlogの各ページに表示されます。

「Local Site Path」は111ページの図7で決めたBlogのトップディレクトリーを指定します(例では「/www/htdocs/mt」)。

「Site URL」はLocal Site PathのURLを指定します(例では「http://サイト名/mt/」)。

「Local Archive Path」はBlogの各ページをまとめて置くためのディレクトリーを指定します。今回はLocal Site Pathの下に「archives」ディレクトリーを作って置くことにしますので「/www/htdocs/mt/archives」と指定します。

「Archive URL」はLocal Archive PathのURLを指定します(例では「http://サイト名/mt/archives/」)。

「Timezone」は「UTC+9 (Japan Times)」を指定してください。

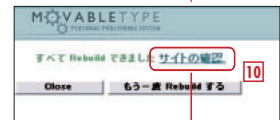
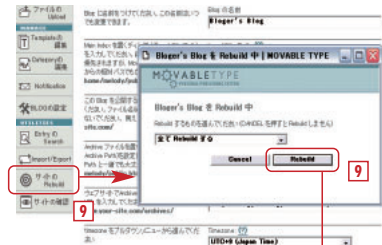
最後に「Save」を押して保存します。

ディレクトリーの設定を間違えると意図しないところにファイルが作られてしまうので十

分注意してください。

9 うまく設定されているかどうか確認するために、左側にある「サイトのREBUILD」を押します。小さなウィンドウが現れるので、このウィンドウの「全てをREBUILDする」を選んで「REBUILD」を押します。表示が切り換わるので「サイトを表示する」をクリックします。

10 カレンダーとちょっとしたリンクだけしかありませんが、Blogサイトのトップ画面が表示されます。



Blogのエントリーを書いてページを作ろう

1 ここからがBlogの始まりです。まず管理画面の左のメニューから「Entryの新規作成」を選びます。

2 Movable Typeでは書き込みをエントリーと呼びます。エントリーの「タイトル」「内容」「追記」などを記述します。入力が終わったら「Post Status」を「Draft」から「Publish」にし、「Save」を押して内容を保存します。

3 これで、最初のエントリーが登録されました。さっそくページを見てみましょう。画面の上の方「VIEW SITE」を押します。

4 入力したエントリーの内容が、サイトに載っているのがわかります。このページが設定したBlogのトップページです。URLは以下ようになります。

http://サイト名/mt/

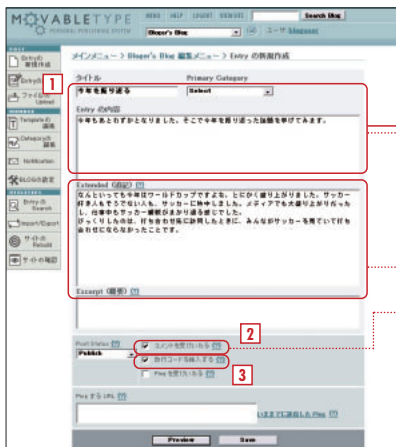


エントリーの内容を理解しよう

エントリーを書き始めると、長い文章や画像などを盛り込みたいこともあります。大きなエントリーの場合、トップページには触りだけを載せておきたいこともあります。

- 1 このようなときは、エントリーのうち、トップページに載せたいところを「内容」のところに、残りを「Extended(追記)」のところに入力します。
- 2 また「コメントを受け入れる」にチェックを入れておくと、各エントリーページにコメントを受け付けるボックスが出力されます。
- 3 「改行コードを挿入する」にチェックを入れておくと、実際にエントリーの各項目に入力したとおりに改行が入ります。
- 4 実際に作られた Blog のページを見ると、1 2 3 の設定が反映されているのがわかります。「追記」の部分はトップページな

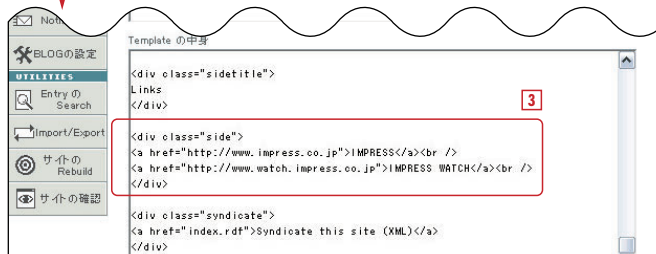
どには表示されず、「Continue reading ...」という各エントリーページへのリンクが表示されます。



表示内容やデザインを変更する

Movable Type の Blog の各ページの表示内容を変更するには、テンプレートを変更します。テンプレートは HTML や CSS で記述されていますが、一部「MT」で始まる Movable Type 専用の特殊なタグが使われています。今回は詳しく触れませんが、Movable Type のマニュアル(英語)に説明が出ていますので研究してみてください。ここではテンプレートの書き換え例としてトップページにリンクを追加してみることにします。

- 1 Blog の編集メニューにある「Template の編集」を選びます。
- 2 トップページのテンプレート「Main Index」を選びます。これを書き換えるとトップページの内容が変わります。
- 3 Main Index の 144 行目を画面のように変えます。変更したら「Save」を押して「Rebuild」を実行します。
- 4 Rebuild が終了したら実際にページを見てみましょう。リンクが加わっているのがわかります。



画像ファイルをアップロードしよう

Movable Typeでは、簡単にですが画像を扱う機能があります。といっても画像ファイルをアップロードすると、アップロード先のHTMLコードを自動的に作成してくれるものです。ここでは、Movable Typeでの画像のアップロード方法について簡単に触れておきます。

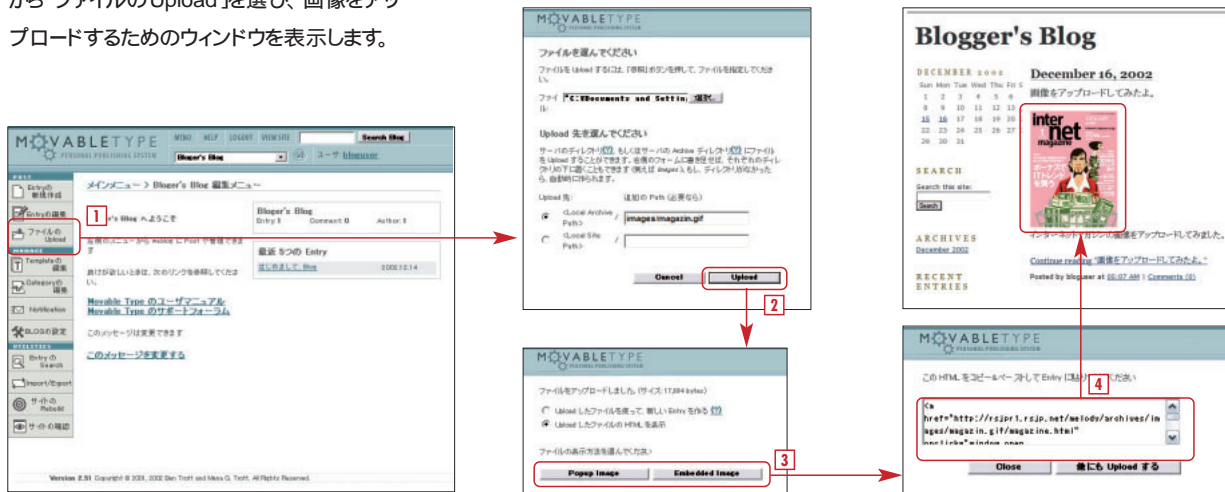
1 画像ファイルをサーバーにアップロードするには、Blogの編集画面の左側のメニューから「ファイルのUpload」を選び、画像をアップロードするためのウィンドウを表示します。

2 アップロードするファイルを指定し、アップロード先として、114ページの8に出てきた Local Site Pathもしくは Local Archives Pathを選びます。「images」のようなサブディレクトリも作成できます。最後に「Upload」を押します。

3 画面が切り替わるので、「UploadしたファイルのHTMLを表示」を選んでその下のボタンのいずれかを押します。「Pop up Image」

と「Embedded Image」の違いは、生成するHTMLコードの内容が、サムネイル画像をクリックしたときにフルサイズの画像を別ウィンドウで表示するか(110ページの7)の手順を済ませていることが必要)同じウィンドウで表示するかという点です。

4 HTMLコードが表示されるので、これを自分が編集中のエントリーに貼り付けて終わりです。



コメントの一部をトップページに表示する

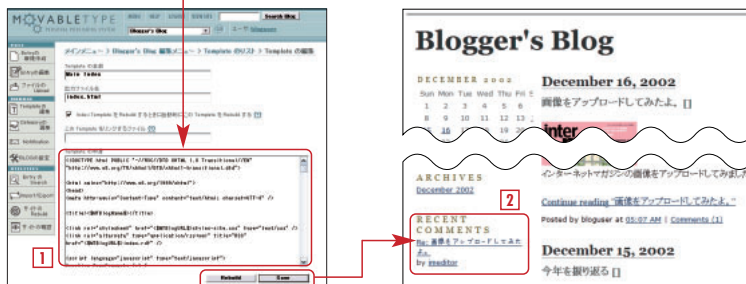
Movable Typeでは、それぞれのエントリーごとにコメントを付けられるようになっていきます。最新のコメントを左側に一覧で出るように、テンプレートを変更してみましょう。

1 Main Index Templateの120行目に図12の記述を追加します。「Save」してRebuildします。

2 実際にBlogのトップページを見ると、左側にコメントがあったかどうかかわかるような表示がなされています。

図12. コメントをトップページに表示するためのHTMLソース

```
<div class="sidetitle">Recent Comments</div>
<div class="side">
  <MTEEntries recently_commented_on="6">
    <a href="<MTEntryLink$>">Re: <MTEEntryTitle></a><br />
    <MTCComments lastn="1">by <a href="<MTEntryLink$>#<MTEntryID pad="1">">
      <MTCCommentAuthor></a>
    </MTCComments><br />
  </MTEEntries>
</div>
```





【活用編】

Blogをもっと楽しむためにツールを使おう

ここまでで、Movable Typeをセットアップして自分でBlogを始めるまでの一通りの説明をしてきた。スタイルシートとテンプレートを編集することで、簡単に自分のウェブサイトが更新できることを理解してもらえただろう。さあ、Blogを本当に楽しむのはこれからだ。基本を理解したところで、次にBlogをもっと活用するウェブサイトやツールを紹介しよう。

text: Hirata Daiji

Blogのためのお気に入り管理サービス「Blogrolling」

はじめに紹介する「Blogrolling」^{KJump}は、Blogのために作られたようなリンクリスト集の管理サービスです。無料で1つのリンク集（Blogroll）を作成できますが、10ドル寄付すると10個のリンク集を作成することができます。ブックマーク（お気に入り）のようにウェブサイトを見ていて気に入ったページがあれば、ページのタイトルとURLを簡単に登録しておけます。どんなサイトも登録できますが

インターネットエクスプローラの「お気に入り」とは違って、Blogを登録した場合はいちいちページを見に行かなくても、自動的にそのページが更新されたかどうかをチェックする機能も付いています。自分のBlogに組み込んでリンク集として表示させることが可能で、登録したリンクページが更新されていた場合に[NEW!]などと表示させることもできます（左画面） 残念ですが、タイトルに日本語を使っているサイトの場合、自分のBlogの文字コード以外ですと文字化けしますので、タイトルを英字で打ち直す必要があります。それでは、Blogrollingの使い方について説明しましょう。

^{KJump} www.blogrolling.com

Step 1*

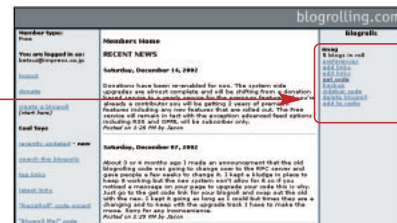
Blogrollを作ろう

1 1 まず、ユーザー登録をします。Blogrollingのトップページにある「Get an account」のリンクから登録ページに移ります。

2 2 メールアドレスとパスワードを入力して「Register me!」をクリックします。その後、入力したメールアドレスに確認のメールが届きますので、メールに示されているURLを開けば登録完了です。メールアドレスとパスワードを入力してログインしましょう。

3 3 ログインしたら「create a blogroll」をクリックします。

4 4 「Roll name」は自分のBlogrollに付ける名前です。「Blog URL」は自分のBlogのURLです。それぞれを入力してから「Make me a new Blogroll」をクリックすると、Blogrollが作成されます。



Step2**

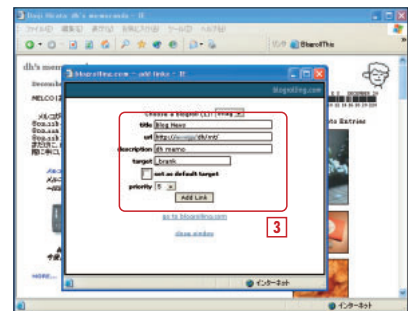
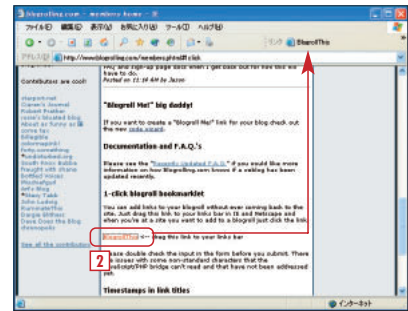
お気に入りのページを登録しよう

1 ログイン後のページにある「1 click "Blogroll This" bookmarklet」をクリックします。

2 「Blogroll This」をインターネットエクスプローラのリンクバーにドラッグ&ドロップします。「Blogroll This」のURLを「お気に入り」に追加してもいいです。

3 Blogrollに登録したいページを見つけたときに「Blogroll This」のリンクバーかお気に入りをクリックすると別ウィンドウが開きます。登録するBlogのタイトルとURLは自動的に取得しますが、日本語のタイトルは文字化けするので、英字で打ち直します。「description」

にサイトの説明を入れておくと、マウスカーソルがリンクの上にあるとき、入力した説明文をツールチップで表示してくれます。また「target」に「_brank」と入力すると、リンクをクリックしたときに新しいウィンドウで表示します。「priority」は、表示順序を決めるときに使います。数字が小さいものから順に表示されます。以上を入力したら、確認して「Add Link」をクリックすれば登録完了です。



Step3***

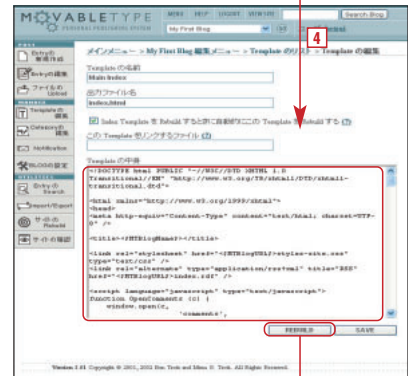
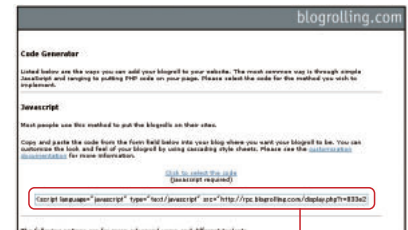
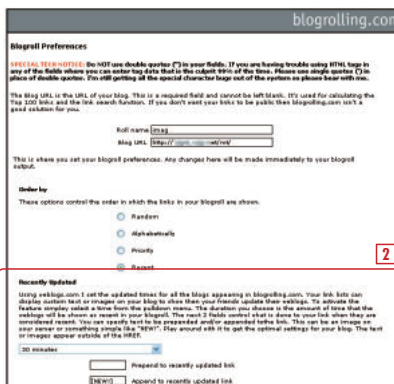
Blogrollを表示しよう

1 Blogrollのログイン後のページにある「preference」をクリックします。

2 「Recent Updated」の項目で、登録したblogページの更新を確認する時間を30分～48時間の間で選び、それぞれのリンクの前(Prepend)もしくは後(Append)に[NEW!]など更新されていたときに表示したい文字を入力します。

3 ログイン後のページに戻り、右側にある「get code」をクリックします。

4 Javascriptのコードをコピーして、自分のBlogのindex.htmlなどのテンプレートに貼り付けてリビルドすれば、貼り付けた場所に自分のBlogrollが表示されます。




RSSを利用してウェブサイトの概要をblogに取り込む

RSS (RDF Site Summary) は、ウェブサイトの概要をメタデータとして簡潔に記述するXMLフォーマットのことです。RSSを使うと簡単にウェブサイトの概要(サマリー)を見出し、更新情報を取得することが可能です(下画面)。注目しているサイトがRSSを提供していれば、これを自分のBlogに取り込んで表示できます。そうすれば、ほかのサイトとの一体感が高まるような気がしますし、コミュニケーションが広がるでしょう。また、自分でウェブサイトを更新しなくてもサイトに変化がでるのでおもしろいです。


このRSSは、もともとはネットスケープ社が開発して公開したRSS 0.9がベースになっています。現在一般に使われているRSSは、ウェブログの老舗であるユーザーランド社の規格を取り入れたRSS 0.91で、見出しだけではなくウェブサイトの概要や更新日時などの情報も登録できます。また、ネットスケープがRSSから手を引いた後、RSSの開発者グループ「RSS-DEVワーキンググループ」が、RSSを容易に拡張できるように、XMLを交換構文として使用するRDF (Resource Description Framework) を基にして改めて仕様をまとめたRSS 1.0があります。

Movable Typeでは、標準の状態では

RSS 0.91に対応したテンプレート「index.xml」とRSS 1.0に対応したテンプレート「index.rdf」の2つが用意されています。これらはエントリーが増えてリビルドされるときに同時に更新されて、常に新しいものが用意されるようになってます。しかし、Movable Typeでは標準でRSSの取り込み機能を用意していません。そこで、Timothy Appnel氏が提供しているプラグイン「mt-rssfeed」を利用します。そうすると、新しくRSSを取り込むための<MTRSSFeed>という独自のタグをテンプレート内で使えるようになりますが、Movable Typeのindex.htmlにこのタグを直接書き込むと文字化けしますので工夫します。この使い方を説明しましょう。

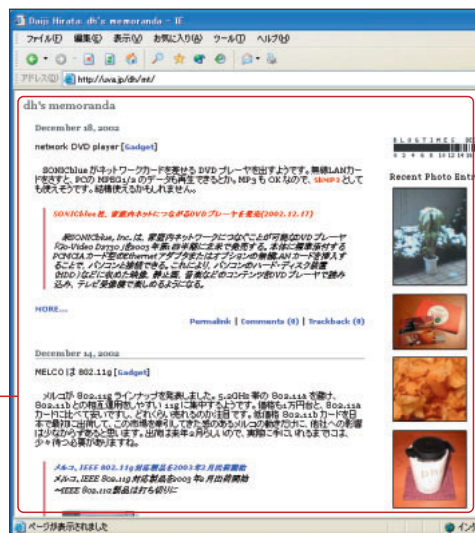
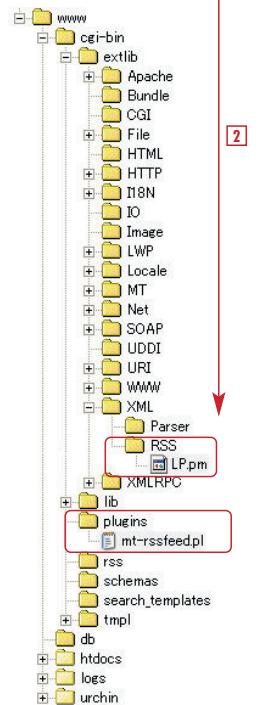
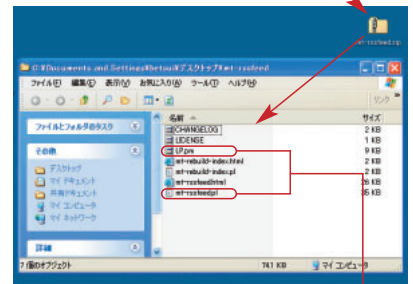
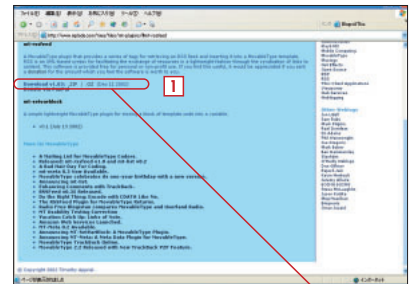
 www.mplode.com/tima/files/mt-plugins/#mt-rssfeed

Step 1* プラグインをセットアップしよう

1 まず、 から「mt-rssfeed」をダウンロードして解凍します。

2 「mt-rssfeed.pl」を、Movable Typeの

CGIファイルを置いたディレクトリの中にある「plugins」ディレクトリにアップロードします。また、「extlib/XML」ディレクトリの中に「RSS」ディレクトリを作成します。ここに「LP.pm」をアップロードすればセットアップは完了です。



Step 2**

「mt-rssfeed」のタグを使おう

- 1 RSSファイルのURLを取得します。ここでは筆者のBlogページから「http://uva.jp/dh/mt/index.rdf」をコピーしました。
- 2 Movable Typeの「Templateの編集」で「新しいIndex Templateを作る」をクリックします。
- 3 ここでは、Templateの名前を「mt-rssfeed result」、出力ファイル名を

「syndication.html」とします。そして「Index Template を Rebuild するとき自動的にこの Template を Rebuild する」をチェックします。

- 4 MTRSSfeedのタグを書き込みます。Aには自分のBlogに表示させるRSSのタイトルを書き込みます。Bには1で取得したURLを記述します。

6 保存が終わりましたら、再び2の「Templateの編集」に戻って、「Main Index」をクリックします。

- 7 「Templateの中身」には、RSSを表示させたい場所に以下のタグを書き込みます。

<\$MTInclude file="作成したテンプレートのファイル名">

ここでは「作成したテンプレートのファイル名」に先ほど作成した「syndication.html」とします。書き込んだら「SAVE」をクリックして保存します。



```

4 <div class="sidetitle">
  syndicate from dh A
</div>
<div class="side">
  <MTRSSFeed file="RSSのファイル名もしくはURL"> B
  <$MTRSSFeedTitle$><br/>
  <MTRSSFeedItems lastn="5">
    <a href="<$MTRSSFeedItemLink$>"><$MTRSSFeedItemTitle$></a><br />
  </MTRSSFeedItems>
</MTRSSFeed>
</div>

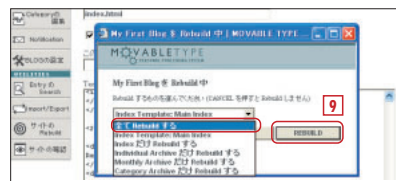
```



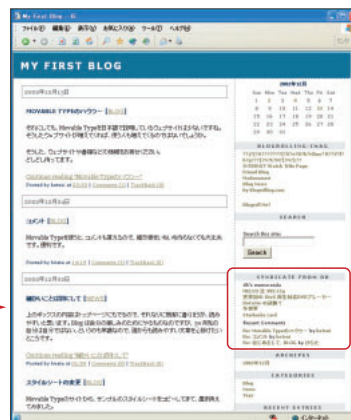
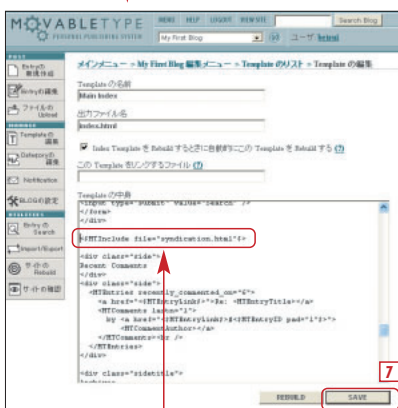
8 保存し終わって「変更した Template を保存しました。変更をサイトに反映させるためには Rebuild してください」と表示されたら、続けて「REBUILD」をクリックします。



9 すると別ウィンドウが開きます。プルダウンメニューから「全て Rebuild する」を選択して「REBUILD」をクリックします。



10 「全て Rebuild できました」と表示されたら「CLOSE」をクリックします。これで、自分のトップページにほかのサイトの RSS が表示されるようになりました。

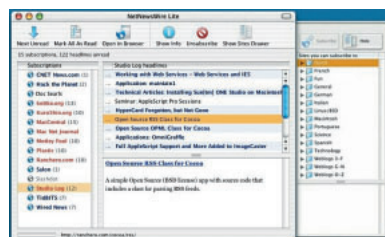


```
<MTInclude file="作成したテンプレートのファイル名">
```

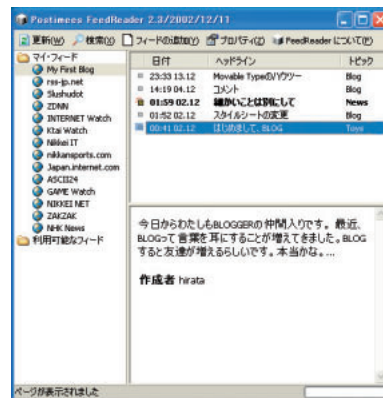
RSSを表示する専門ツール

ブラウザのようにRSSを直接読み込んで表示したり、一定間隔ごとにRSSを取得して更新していたら報せたりする専用ソフトウェアがあります。ウィンドウ用には「Feedreader」^{Jump01}、Mac OS X用には、「NetNewsWire Lite」^{Jump02} がありまして、両方ともフリーウェアです。標準のインターフェイスは英語ですが、Feedreaderには各国後のインターフェイスが用意されており、日本語のインターフェイスもダウンロードできます^{Jump03}。また、両ソフトウェア共に海外のソフトのためか、文字コードがUTF-8で書かれているRSSのみタイトルが日本語で表示されます。ほかの文字コードを利用しているサイトのタイトルには文字化けしたり、表示されなかったりする場合もあります。RSSはウェブサイト自体が提供しているもののほかにも、実験中ながら一般のニュースサイトのRSSを独自に配布しているウェブサイトもあります^{Jump04}。

- Jump01 www.feedreader.com
- Jump02 ranchero.com/software/netnewswire/
- Jump03 www.feedreader.com/translation/get.php?lang=japanese
- Jump04 rss-ijp.net



Ranchero Software社が提供している「NetNewsWire Lite」



i-system社が提供しているWindows用の「Feedreader」

今回の記事はホスティングサービスのラピッドサイトにご協力いただきました。



[インターネットマガジン バックナンバーアーカイブ] ご利用上の注意

このPDFファイルは、株式会社インプレスR&D(株式会社インプレスから分割)が1994年～2006年まで発行した月刊誌『インターネットマガジン』の誌面をPDF化し、「インターネットマガジン バックナンバーアーカイブ」として以下のウェブサイト「All-in-One INTERNET magazine 2.0」で公開しているものです。

<http://i.impressRD.jp/bn>

このファイルをご利用いただくにあたり、下記の注意事項を必ずお読みください。

- 記載されている内容(技術解説、URL、団体・企業名、商品名、価格、プレゼント募集、アンケートなど)は発行当時のものです。
- 収録されている内容は著作権法上の保護を受けています。著作権はそれぞれの記事の著作者(執筆者、写真の撮影者、イラストの作成者、編集部など)が保持しています。
- 著作者から許諾が得られなかった著作物は収録されていない場合があります。
- このファイルやその内容を改変したり、商用を目的として再利用することはできません。あくまで個人や企業の非商用利用での閲覧、複製、送信に限られます。
- 収録されている内容を何らかの媒体に引用としてご利用する際は、出典として媒体名および月号、該当ページ番号、発行元(株式会社インプレス R&D)、コピーライトなどの情報をご明記ください。
- オリジナルの雑誌の発行時点では、株式会社インプレス R&D(当時は株式会社インプレス)と著作権者は内容が正確なものであるように最大限に努めましたが、すべての情報が完全に正確であることは保証できません。このファイルの内容に起因する直接のおよび間接的な損害に対して、一切の責任を負いません。お客様個人の責任においてご利用ください。

このファイルに関するお問い合わせ先

株式会社インプレスR&D

All-in-One INTERNET magazine 編集部

im-info@impress.co.jp