

## 序章

Interarchy は、そのインターフェイスを外観、機能ともに自由にカスタマイズすることが可能です。Interface はこれを達成するための方法です。あなたは、ボタンが使用できるかどうか、そしてそれらの役割と設置位置を決定するInterface の目に見える外観を指定することが可能です。これは楽しいだけでなく、役に立ちます。というのは、一つの Interface 内に関連したインターネット操作や URL をまとめることができるからです -- あなたに最適な Interface を選ぶか、既存のものに見当たらない場合は自作することが可能です。

Interface を作成するには、最もよく使用するコマンドと、最もしばしば利用するサーバ、ディレクトリ、そしてサービスを選びます。それからそれらを一つのInterface に集めます。または、Interarchy は一度に複数の Interface を使用できるので、お望みなら何個かの Interface を作成して下さい。

Interface はあなた(または他のInterarchy ユーザ)が構築した特殊なウィンドウと考えて下さい。一度、Interface が完成し起動されると、Interarchy は常時それを表示し、あなたはシングルクリックで数多くのインターネット操作を実行することができます。

### Interface って何？

Interface は独自のフォーマットを持つテクニカルなMacintosh ファイルです。それぞれのInterface は幾つもの画像、ボタン、そしてURL を内包します。Interarchy は、これらのファイルを読み込んで、それらの部品を指令パネルとして機能する特別なインターフェイスウィンドウを表示するために使用します。

あなたは 'Interfaces' とも呼ばれる、これらのウィンドウを通して Interarchy を運用するのです。

### Interface Builder って何？

そして Interarchy Interface Builder は何の役に立つの？ Interface ファイルをツールなしで作成するのは不可能ではありませんが、殆どの人々にとっては辟易する作業でしょう。Interarchy Interface Builder は Interface を視覚的に、他の Macintosh アプリケーションを使うのと同じく容易に作成するためのツールです。

Interarchy Interface Builder を使用してあなた独自の Interarchy 用の Interface を簡単に作成できます。Interarchy Interface Builder は Interface の構築と変更を可能にします -- あらゆるサイズと形状の独自画像の読み込み、ボタンの追加と除去、それらの位置の自由な変更、それぞれのボタンが実行するコマンドの指定、が可能です。

完成したら、あなたの作品を Interarchy サイトにある特別エリアで公開することができます。それに他の人達が作成した Interface を手に入れて使用する事も可能です。

## でも、簡単なの？

はい！ なぜ Interarchy Interface Builder がそんなに使いやすかった？ それは、Interarchy がするのと同じように表示しながら Interface を視覚的にシミュレートするからです。これは、作業結果を逐次、直ちに確認することを可能にします。つまり Interface を Interarchy で読み込んでテストし、また Interarchy Interface Builder に戻って変更することを繰り返す必要はありません。

Interarchy Interface Builder が唯一できないのは、あなたのために Interface をデザインすることです。これに関しては残念です (^\_^)

---

## Interface とメニュー

### Interface の動作

最初の起動時に Interarchy Interface Builder は Interarchy:Custom Interfaces:Interface Folder: 内で、Work Interface という名のファイルを探して自動的に開きます；Interface Builder は Interface Folder が存在しない場合は自動的にこれを作成します。その後は最後に作業したファイルを記憶して、Interarchy Interface Builder の毎起動時にそれを開きます。同じ Interface で作業する限り、Interarchy Interface Builder をただ開くだけで、直ちに中断していた作業が再開されます。

別の Interface で作業したいと思えば、Interarchy Interface Builder に読み込むだけで、自動的にそれが新しい**作業用** Interface となります。

もし、移動されたために**作業用** Interface が見つからない場合、Interarchy Interface Builder は不平を言わずに、ただ他のものを開くのを待ちます。

### Interarchy メニュー

通常 Interarchy は Interface とその他のファイルを（Interarchy の Home メニューから簡単にアクセス可能な）**システムフォルダ:初期設定:Interarchy:Home Items** 内に保持します。

(： 半角コロンはパスを示します、ポー太)

Interarchy Interface Builder は、これとよく似たメニュー、Interarchy、を提供します。これは、**システムフォルダ:初期設定:Interarchy:** とその（Home Items: を含む）サブフォルダ内の全ての Interface とフォルダをリスト表示します。このメニューから Interface を選択すると、Interarchy Interface Builder でそれを開きます。Interarchy:Home Items **フォルダ** を指定すると、直ちにこのフォルダを Finder で開きます。これは Interarchy の Home:Home Items フォルダメニューを指定するのと全く同じです。それに加えて便利なことに、Shift キーを押しながら Interarchy メニューから Interface を選んだ場合、それは Interarchy Interface Builder の代わりに Interarchy で開かれて、Interarchy Interface Builder はそれを新規の**作業用** Interface にはしません。

また、Interarchy メニューはフォルダの**再スキャン**と、Finder 上でフォルダを開くオプションを提供します。メニューは、“Home Items **フォルダ内のみを表示**”と“Interarchy **フォルダ内の全てを表示**”に交互に切り替わり、Home Items フォルダとそのサブフォルダ内の Interface のみを表示するか、全ての Interarchy ファイル機構内の Interface を全部表示するかを選ぶことが可能です。

## 作業フォルダメニュー

あなたの**作業フォルダ**は、Interarchy Interface Builder 自身のフォルダとその中に存在する全てのフォルダです。これはHome Items フォルダとよく似た、しかしHome Items フォルダに入れる準備がまだできていないInterface用のフォルダです。Interarchy Interface Builder はInterfaceがこのスペースに保存されると予想します。Interarchyのパッケージには、あなたが全てのInterfaceを便利にまとめておけるようにInterface Folder が用意されています。アクセスに便利のように、**作業フォルダ**:Interface Folder を指定すると、直ちにこのフォルダをFinderで開きます。

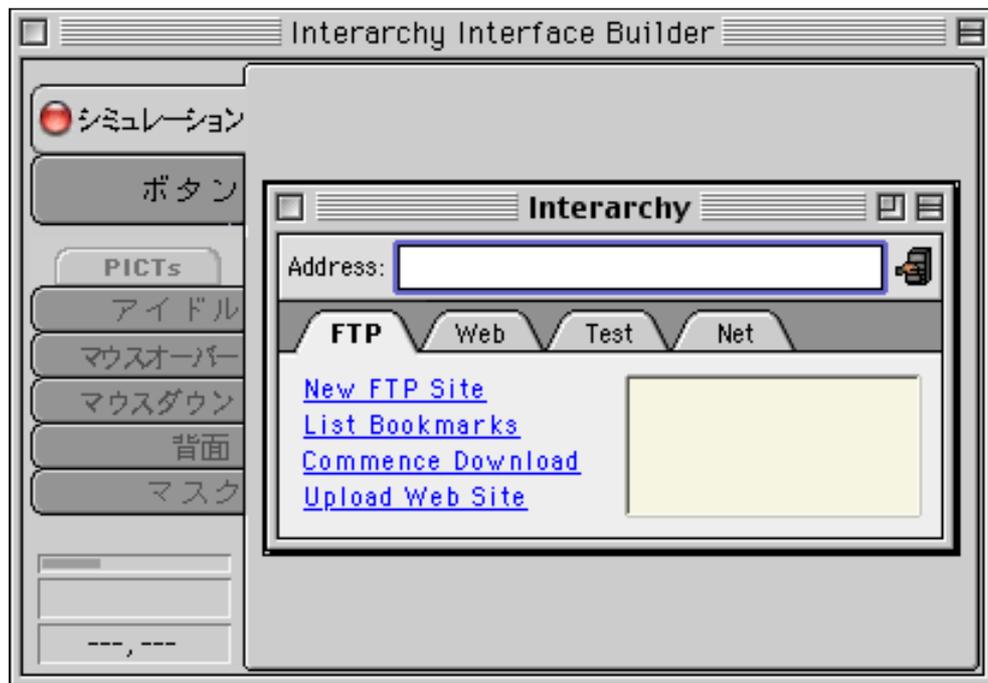
**作業フォルダ**で見つけた全てのInterfaceは**作業フォルダ**メニュー内にリスト表示されるので、どれでも簡単に開くことが可能です。それに加えて便利なことに、Shift キーを押しながら**作業フォルダ**メニューからInterfaceを選んだ場合、それはInterarchy Interface Builderの代わりにInterarchyで開かれて、Interarchy Interface Builderは次回の起動まで自動的にそれを開くことはしません。**作業フォルダ**メニューはInterarchyのHomeメニューとよく似ていますが、これは完成したInterface用のHome Items フォルダ及びHomeメニューとは対照的に制作途中のInterfaceを置いておくためのものです。メニューは、“Interface **フォルダ内のみを表示**”と“**作業フォルダ内の全てを表示**”に交互に切り替わり、Interface Folderとそのサブフォルダ内のInterfaceのみを表示するか、全てのInterarchy Interface Builder ファイル機構内のInterfaceを全部表示するかを選ぶことが可能です。

もし、Interfaceが多量にあって、**作業フォルダ**メニューにそれらの全てを表示したくない場合は、任意の名前のフォルダを作成して、その名前をこのように半角丸括弧でくくります: ‘(その他のInterfaces)’。括弧でくくられたフォルダ内のInterfaceは**作業フォルダ**メニューには現れません。

また、**作業フォルダ**メニューはフォルダの**再スキャン**と、Finder上でフォルダを開くオプションを提供します。

[よくある質問](#):Interarchyは何かInterfaceを持ってリリースされたの？ それらは、どこで見つかるの？

## Interface の構成



シミュレーションモードでのInterface Builderのメインウィンドウ

### Interface PICTs

Interfaceの作成には、5枚のPICT画像（アイドル、マウスオーバー、マウスダウン、背面、そしてマスク）が必要です。それぞれは、以下の[PICTパネル](#)セクションで説明される特定の目的を持っています。

PICTsは、それらがもたらす視認性と楽しさの点で非常に重要ですが、Interfaceで能動的な役目は果たしません。Interfaceは、マウスでクリックする場所を表示するのとポインタがボタンの上にあるかを知らせるのに画像を利用しますが、その他には使用しません。

### マスクの使用

InterfaceでのPICTは常に矩形内に描画、表示されます。しかし、マスクを使用すればInterfaceを規定外で面白い形状に作成することが可能です。

Multi Interfaceは、この仕組みを実際に説明してくれます（Interarchy 3.8にはMulti Wandとして添付されていますが、4.0には付属していないようです：ポー太）。そのマスクを見ると：黒色の部分は、それぞれ対応した画像部分をInterarchyにおいて可視にしていることが分かるでしょう。そして白色の部分は、それぞれ対応した部分をInterarchyにおいて不可視にしています。

注意：マスクはInterarchyにおいてのみInterfaceに影響を与えます。Interface Builderは常にPICTを全て表示します。

## Interface アイテム

アイテムはInterfaceの能動的な構成部品です。 アイテムには3種類あります：

### ボタン（任意の数）

クリックされると、Interarchyに指令（URL）を送ります。

### URL（一つ以下）

もし存在すれば、テキスト入力を受け入れます。

### 記述（一つ以下）

もし存在すれば、現在マウスの下にあるボタンの記述内容を表示します。

アイテムの作成または編集時に **種別**ポップアップで、そのアイテムが **ボタン**かURLか**記述**かを知ることができます。

Interface内では（常識範囲内で）幾らでもボタンを使用できます。 また最大で、一つのURLと一つの**記述**を設置できますが、これは必ずしも必要ではありません。

Interfaceアイテムは[ボタンパネル](#)で操作します。

## パネル

全てのInterface作成作業は単独の操作ウィンドウで生み出されます。

幾つかのパネルタイプが存在します：

### 一つの[シミュレーションパネル](#)

これはInterfaceが動作する様子（マウスオーバー、マウスクリック）をシミュレートし、またボタンを使ってInterarchyに送る指令の作成ができます。

### 一つの[ボタンパネル](#)

ここで、アイテムの作成、移動、サイズ変更、そして削除ができます。

### 五つの[PICTパネル](#)

Interfaceを構成する複数の画像を表示します。

パネルは、ウィンドウの左にあるタブをクリックするか、ショートカットで切り替わります。

## キーボードショートカット

スペースバーで、上の二つのモード（**シミュレーション**と**ボタン**）と四つのPICTの最初（アイドル）を交互に切り替えます。

**タブキー**で、現在のグループ内の次のパネルに切り替えます（マスクPICTはスキップ）。



# PICT パネル

それぞれのパネルは Interface で使用する画像の内の一つを表示します

## アイドル

アイドル状態（何も外的影響を受けていない状態）での画像です

## マウスオーバー

全てのボタンが‘ 停空 ’モード（マウスポインタがボタン上にあるが、押されていない状態）でのウィンドウ画像

## マウスダウン

全てのボタンが押し込まれた状態のウィンドウ画像

## 背面

バックグラウンドにある時のウィンドウ画像

## マスク

この‘ ピクチャー ’は透過に使用されます -- マスクの黒色のピクセルにおいては、対応する画像のピクセルは表示され; 白色のピクセルにおいては、対応する画像のピクセルは無視されます。このように Interface の一部分を‘ 切り抜く ’ことで、どのような背景でも表示したり、不定形の縁を作成して四角くない Interface を作り出すことができます。 デフォルトの Interface は、丸いエッジのために白色の角を持つマスクを使用しています。

## Interface サイズの決定

アイドル状態の PICT が Interface のサイズを決定します。 その他の PICT はそれに従って拡大 / 縮小されます。 ただ、**マスク**の PICT は Interface の表示サイズと形状を決定するのでサイズ変更されることはありません。

殆どの Interface においては、それぞれの PICT は同じサイズでしょう。 でも、これは必ずしも必要ではありません; これがより簡単なだけです -- ; 違ったやり方もアリです (^\_^  
あなたに便利なように、Interarchy Interface Builder は**アイドル**状態と異なるサイズの PICT を使用しようとする度に警告を表示しますが、無視することも可能です。

Interarchy Interface Builder のウィンドウは、パネルコントロールに必要な最低限の大きさは保ちながら、**アイドル** PICT のサイズに自動的に調整されます。

## コピー / ペーストと PICT の読み込み

パネルは、グラフィックソフトとの間で、パネル同士で、またインターフェイス間で、相互にコピー / ペーストで簡単に入れ替えることができます。

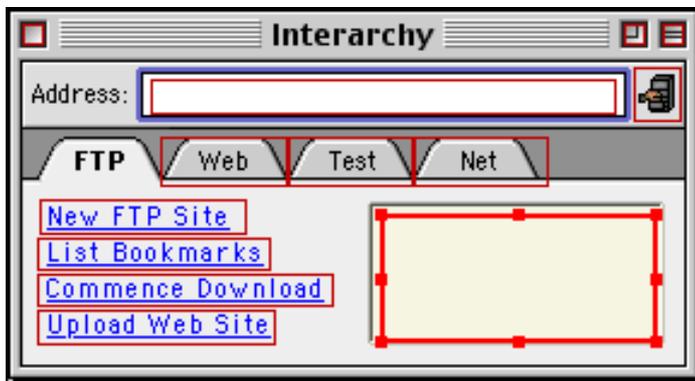
利用したい PICT ファイルがあれば、**ファイルメニュー:PICT を読み込む** を使ってそれらの読み込みが可能です。

## PICT 操作での複数回の取り消し

Interarchy Interface Builder は 10 回までの PICT 操作の取り消しを提供します（十分なメモリがある場合）。これは、例えば、Interface で各 PICT を二度ずつ取り換えてもまだ最初の状態に**取り消し**て戻れると言うことを意味します。もし、あなたが非常に大きなInterfaceを作成しているなら、全ての**取り消し**を使用するには、Interarchy Interface Builder への割り当てメモリを増やす必要があるでしょう。

**重要な注意**：PICT 操作の**取り消し**は PICT パネルでのみ動作します！ シミュレーションやボタンパネルでは、PICT 操作の**取り消し**は使用できません。 PICT 操作の**取り消し**を使用するには PICT を一つ選択して下さい。

## ボタンパネル



Interface Builder はボタンの枠を表示します。  
**参考**：Option キーを押すと、全ての枠に連番が表示されて参照に便利です。

このパネルは項目の殆どがボタンなので“ボタン”という名前ですが、実際は全てのアイテムタイプ（**ボタン**、URL、**記述**）を管理できます。

これは、アイテムの作成、削除、位置決め、そしてサイズ変更のなされる場所です。

## アイテムの作成と削除

新規に Interarchy のアイテムを作成するには、対応するパレットボタンをクリックします。新規アイテムのサイズと位置は、現在選択されているアイテムか、または（何も選択されていない場合は）以前作成されたアイテムに基づきます。もし、あなたがデザインしたInterfaceが非常に小さい場合は、Interarchy Interface Builder が適切なサイズと位置を設定するのを助けるために、既存のアイテムを選択してから新規のものを作成することをお勧めします。

一つのアイテム、またはアイテムグループを削除するには、**選択ボタン**を**削除ボタン**をクリックします。

## アイテムの移動とサイズ変更

アイテムは“ハンドル”（画像処理ソフトでよく知られた並んだ8個の小さな四角形）のドラッグ、またはキーボードの矢印キーで操作することができます：

矢印キーで移動

Shift + 矢印キーで高速移動

Command + **矢印キー** でサイズ変更

Command + Shift + **矢印キー** で高速サイズ変更

キーボードではマウスよりも、きめの細かいコントロールが可能です。 精密な作業ではハンドルを**不可視**にできます：これは、1 ピクセルの精度でアイテム矩形を背景 PICT に配置することを可能にします。

## アイテムの選択

アイテムの外から **ドラッグ** すると、そのドラッグ領域内に囲まれたアイテムが選択されます。

アイテムの外から Command **キー** を押しながら **ドラッグ** すると、そのドラッグ領域内に掛かった全てのアイテムが選択されます。

Shift **キー** を押しながら **クリック** すると、アイテムの選択を追加 / 解除します；これは複数のアイテムをドラッグで囲むときにも適用されます。

## アイテムの不可視オプション

下にある PICT のカラーとコントラストにアイテムの外枠を適応させることで、その矩形の視認性を良くするための幾つかのオプションが使用可能です。 合わせて、それらは多くの表示組み合わせを提供します：

アイテムの枠は、**アウトラインカラーを変更ボタン** をクリックすることで変更可能なカラーで縁取られます。 これは、**オプションメニュー** から直接指定することができます。

**ハンドル** のカラーもまたカスタマイズ可能です。

**ボタン** パネルで PICT **を隠す**：PICT の彩度がとても高い場合、カラー枠で縁取っているにも関わらずアイテム矩形が見にくくなります。 PICT **を隠す** ボタンをクリックして、一時的に PICT を隠すことが可能です。

視認性をより良くする：Control キーを押すと一時的にアイテム矩形を暗くして、より見やすくします。

Option キーを押すと、アイテム番号を表示します。

ハンドルの操作では、視認性とカラーの設定がまだ三つあります( **オプションメニュー**、キーボードショートカットは Command + H )：

### 同一のカラー、太いアウトライン

通常の太さのアウトラインおよび、それと同じ色をアイテム矩形に使用

### 異なるカラー、細いアウトライン

別の対照的なカラーを使用する：これは、アウトラインとハンドルが重なるような非常に小さなアイテムに効果的です；また、細いアウトラインを使用します。

### 不可視、細いアウトライン

下にある PICT にアイテムを整列するような細かい作業にとっても便利です；また、細いアウトラインを使用します。

複数のアイテムの移動とサイズ変更を一度に行えます。 複数アイテムのサイズ変更をするとき、最後に選択されたアイテムはハンドルを表示していて、単独にサイズ変更されます；その後、他の選択中のアイテムに同じ絶対的(相対的ではなく)サイズ変更が適応されます。 全ての場合、アイテムは Interface の範囲を超えて移動またはサイズ変更をすることはできません。

## アイテム操作の複数取り消し

全てのアイテム操作：移動、サイズ変更、整列、作成、そして削除は、32 回までの**取り消し**が可能です。

## シミュレーションパネル

### Interarchy Interface のシミュレート

このパネルは、あなたの作業結果を表示します：これは、マウスクリックやボタン上での‘ 停空 ’操作に対応した目に見える反応を伴うことを可能にしながら、実際のInterarchy Interface を(マスク効果 / 透過性を除いて) 厳密に再現します。 **参考**：背面状態のInterface (すなわち、Interarchy アプリケーションがバックグラウンドにある時) はInterarchy Interface Builder では‘ シミュレート ’されません。Interface が背面状態で、どのように見えるかは、**背面**パネルの PICT を参照して下さい。

### アイテム情報の編集

アイテムのクリックはInterarchy がするようには、それを起動しません；代わりにアイテムの属性の編集を可能にします。

#### 種類

**ボタン**、URL または**記述**のどれか (URL- アイテムと、ボタン - アイテムの URL 属性の区別に注意)

#### フラグ

URL と**記述**では、フラグは整列を指定します：**左辺**、**中心**、または**右辺**

#### 座標

アイテムを数値指定する方を好む人のための位置です。

ボタンはそれぞれ独自の属性セットを持っています：

#### 記述

簡単な説明です

#### URL

Interarchy に送られる指令で、Interarchy [スキーム](#)や他の種類のスキームを使用可能です。

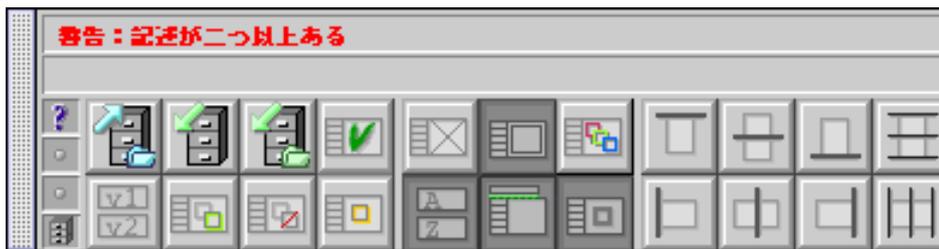
#### フラグ

Interarchy 、**ヘルパー**、そして(ウィンドウの)**閉鎖**との何れかの組み合わせです。

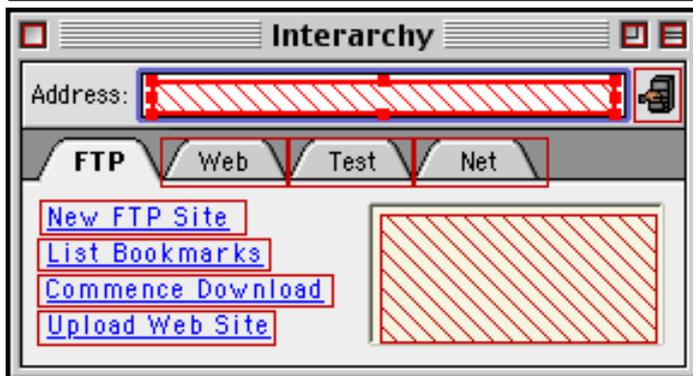
## アイテムでの警告

Interface はボタンを幾つでも（常識範囲内で）持つことができますが、必ず URL と記述は一つ以下でなければなりません。これは、一つの URL アイテムを挿入したりそれを除去することはできるが、二つの URL アイテムを挿入することはできないということです。そして、これは記述アイテムにも同様に言えます。

**参考：**もし、一つ以上の URL または記述が存在すれば、Interarchy Interface Builder は警告を表示します。このような Interface は正常に動作しないので、保存する前に修正しなければなりません。どのアイテムに間違いがあるか判らない場合でも、この修正はたやすくできます。このような問題を簡単に修正するために、重複している二つの範囲上に網かけパターンが描かれます。



重複ボタンを警告している、Interface Builder のツールパレット



重複ボタンを警告している、Interface Builder

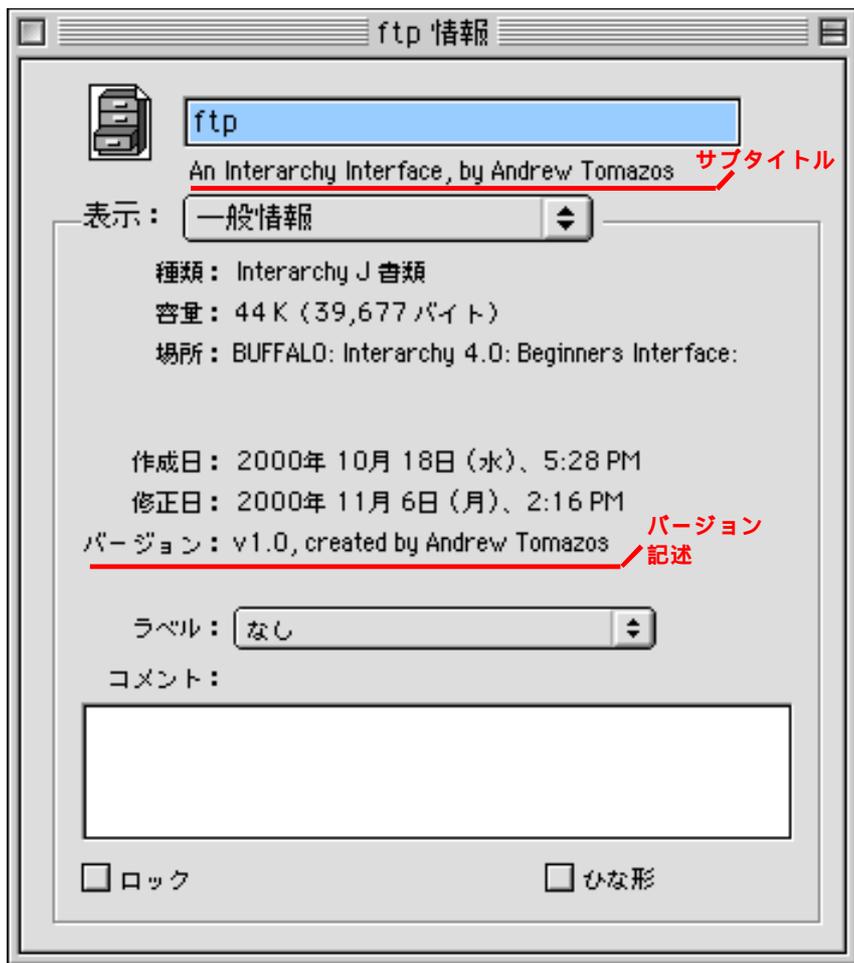
編集ウィンドウ内の種類およびフラグフィールドでは明示的で適切な値を持つポップアップメニューを使用します。しかし、数値を入力する必要がある場合は、Command + Option キーを押しながらアイテムをクリックすると、ポップアップメニューは代わりに数値を表示します。

## あなたの作品にサインをする

Interface の作成と編集を終えた時（または、テストのため他人に渡す前）、**アクション:Interface の署名とバージョン** であなたの作品にサインして下さい。呼び出されたウィンドウで以下のバージョン情報を入力できます：

- バージョンナンバー
- リリースタイプ：開発途中、アルファ、ベータ、またはファイナル
- Finder バージョン記述
- サブタイトル（パッケージネームとも呼ばれる）

最後の二つのフィールドは Finder **情報** ウィンドウ内に表示されます。



Interface の Finder **情報** ウィンドウ

# スキーム

Interarchy Interface は URL を起動する特別なウィンドウです。 これらの URL は広範囲のインターネット操作と、Interarchy の殆どの内部操作も同様に遂行することが可能です。

URL の最初の部分は 'スキーム' と呼ばれ、どのプログラムが URL を処理するのかをコントロールし、そしてそれが何をすることを明らかにします。 インターネットコントロールパネルはどのスキームがどのプログラムを呼び出すかの記録を保持していますが、特定のプログラムはこれらのマッピングを使用しないかも知れません。

Interarchy は以下の URL スキームを全て取り扱えます。 指名されると、これら全てを引き受けませんが、http だけは通常 Web ブラウザを指定するので除きます

- bbftp
- dns (ドメイン参照)
- finger
- icedit -- インターネットコントロールパネル設定の編集
- interarchy
- netsearch
- ping (ICMP でテスト)
- tcpecho (TCP でテスト)
- traceroute (ルート追跡)
- whois
- udpecho (UDP でテスト)
- ftp
- interarchy (特別、以下を参照)
- http (デフォルトで Interarchy に割り当てられていない)

interarchy: (単独の半角コロン) は Home Items 内の特別なファイルを開きます。 以下の内部機能とウィンドウには interarchy:: (コロン二つ) でアクセスできます :

- interarchy::web interarchy::http == interarchy::httplist
- interarchy::httpdownload
- interarchy::httpmirrordownload
- interarchy::ftp == interarchy::ftplist
- interarchy::ftpdownload
- interarchy::ftpupload
- interarchy::ftpmirrorupload
- interarchy::netsearch
- interarchy::home
- interarchy::history
- interarchy::finger
- interarchy::whois
- interarchy::report
- interarchy::status
- interarchy::connections
- interarchy::traffic

interarchy::dns  
interarchy::traceroute  
interarchy::ping  
interarchy::tcpecho  
interarchy::udpecho  
interarchy::transcript  
interarchy::interface (4.0 での新機能：もし Interface が最前面にある場合、Interface Builder を起動してそれを開く)

## 環境設定

全ての継続的な指定は自動的に初期設定ファイル：**システムフォルダ:初期設定:Interarchy:Interarchy Builder Prefs** に保存されます。もし、初期設定ファイルが存在しなければ作成されません。

もし、古い初期設定ファイル（すなわち、旧バージョン番号を持つ）が見つかったら、警告なしにその内容は新しい形式のデフォルト設定で置き換えられます。あなたは、**作業する** Interface も含めて環境設定を再度指定する必要があります。

## よくある質問

Interarchy **は何か Interface を持ってリリースされたの？** それらは、**どこで見つかるの？**

はい、されました。それらには、**作業フォルダ**と Interarchy メニューからアクセスできます。全てのバンドルされた Interface はここにリストされます -- どれかを選ぶだけで開きます。

もし、Finder からそれらを直に操作したいのなら、Interarchy を（必要な場合）起動して Home:Home Items **フォルダ**項目を選びます。これで Interarchy が自分自身の Interface を保持しているフォルダが開きます。Interface を Home Items から **作業フォルダ**へ作業用のコピーを取ります。あなたの新しい作品が用意できたら、それを Interarchy で使用するために Home Items フォルダに戻します。

また、<http://interface.interarchy.com/> で多くの Interface を見つけることができます。

## 更新箇所：Interarchy とのシームレスな統合

この機能は、Interface に加えられた全ての変更を保ちながら Interarchy と Interface Builder とをシングルクリックまたはメニューコマンドによって相互に切り替えることを、双方向に容易にします。

### Interarchy から Interface Builder へ

Interarchy からは、最前面の Interface を Interface Builder で開くために 諸設定:インターフェイスを編集 を使用します。 変更後に Command + S をタイプすると、直ちに Interarchy にその変更と共に戻ります。 こんなに簡単なことはないでしょう？

お望みなら、メニューコマンドと同じ働きをする Interface ボタンを作成することが可能です。 このボタンは、メニューコマンドよりもマウスクリックの方を好む人々のため用です。 Interface Builder はこのボタンをあなたのために作成することもできます：ツールパレット上の “この Interface を編集” ボタンを作成ボタン をクリックして、好きな場所に新しいボタンを設置して下さい。

参考：自動的に Interarchy に戻る機能はオプションです：あなたが暫くこの Interface 上の作業を続行したいと分かっている場合は、**オプションメニューからファイルの保存時に Interarchy の表示を更新**のチェックを外します。 また、覚えておいて欲しいのは、このメニューアイテムは Interface を Interarchy の **諸設定:Interface を編集** コマンドまたは同等のボタンから開いたときのみ使用でき、その他の場合はグレー表示で使用できません。

### Interface Builder から Interarchy へ

通常、Interface Builder の **ファイル:保存** コマンドは、ただあなたの Interface を保存するだけです。 しかしオプションで、いま保存した Interface を Interarchy で開くことができます。 これを遂行するには、**保存** コマンド（または **ツールパレットの保存ボタン**）を使用する際に Shift ボタンを併用するだけです。 思い出せない場合は、**ファイル:保存して Interarchy でテスト** も使用できます。

Interarchy Interface Builder は Daniel Bondard bondard@tnuclip.com がデザインとプログラミングをしました。

©2000 Daniel Bondard and Stairways Software Pty Ltd