



Technical Note 04-32

4D Client 自動アップグレード

By Jonathan Le, Technical Support, 4D, Inc.
Technical Note 04-32

(原題: Client Self-Upgradable)

概要

バージョン 2004 の目玉機能のひとつは、クライアント/サーバ版のファイナルアプリケーションをビルドできるようになったことです。ビルドされたクライアント/サーバ版のファイナルアプリケーションは、4D Engine を組み込まれた従来のファイナルアプリケーションと同じように、ダブルクリックで起動可能で、4D からは独立したアプリケーションです。

ビルドされたクライアント/サーバアプリケーション

4D のクライアント/サーバアプリケーションを配布しようとする場合、従来の方式では、ストラクチャファイルを 4D Server で開かなくてはなりません。新しい機能を利用すれば、ユーザはただビルドされたサーバアプリケーションをダブルクリックするだけで済みます。

用意するもの

クライアント/サーバ版のファイナルアプリケーションをビルドするために必要なものは、以下の 3 点です。

- ・コンパイルされたデータベースストラクチャ
- ・ 4D Server
- ・ 4D Client

利点

前述のように、ユーザはサーバアプリケーションをダブルクリックするだけでよく、ストラクチャを特定する手間が省かれます。クライアントアプリケーションをダブルクリックすると、次のいずれかの動作をします。

- ・特定のサーバに自動的に接続
- ・接続ダイアログを表示

通常の結果は、前者であり、ユーザはサーバの所在地を意識することはありません。これはビルド時にサーバ情報が組み込まれているからです。後者は、サーバからの反応が得ら

れない時、あるいは所定のサーバが見つからない時に生じます。

自動アップグレード

ビルドされたクライアントアプリケーションは、自動アップグレード機能の恩恵にあずかることができます。これにより、各クライアントステーションを更新してまわらなくても、クライアントアプリケーションが自ら更新を検出して自動アップグレードを実施できるので、配布する側の負担が大幅に軽減されます。

この機能を利用するためには、後述するサーバ側の設定が必要です。

アプリケーションビルド

「アプリケーションビルド」ツールは、デザインモードの「デザイン」メニューから「アプリケーションビルド」を選択すると表示されます。

4D Client Win には、Windows 版の 4D Client の置かれたフォルダ、4D Client Mac には、Macintosh 版の 4D Client の置かれたフォルダを指定します。（※それぞれのクライアントアプリケーションをビルドするためには、各プラットフォームの 4th Dimension を使用します。）クロスプラットフォームのデータベースをビルドする際は、環境設定の「デザインモード/コンパイラ/コード生成」で「両プラットフォーム用にコンパイルする」チェックボックスを有効にしてからコンパイルを実行します。

プロジェクトファイル

アプリケーションをビルドするには、プロジェクトファイルが必要です。プロジェクトファイルには、アプリケーションビルドの際に考慮される設定項目が XML 形式で記述されます。ビルドを実行すると、デフォルトのプロジェクトファイル「BuildApp.xml」が、ストラクチャと同じ階層の Preferences/BuildApp フォルダに保存されます。

クライアント/サーバアプリケーションのビルドに関係があるのは、以下の XML キーです。（すべて XML パス Preferences4D/BuildApp/CS 以下のキー）

| | |
|------------------------|---------------------|
| BuildServerApplication | true=サーバアプリケーション |
| BuildCSUpgradable | true=自動アップグレード |
| CurrentVers | 数値=現在のバージョン |
| IPAddress | (任意) サーバの IP アドレス |
| PortNumber | (任意) 上記のポート番号 |
| HardLink | 署名=指定しなければアプリケーション名 |
| RangeVersMin | 接続を許可する最小バージョン |
| RangeVersMax | 接続を許可する最大バージョン |

コマンドでアプリケーションビルド

バージョン 2004 では、コマンドでアプリケーションビルドを実行できるようになりました。コマンドには任意のパラメータ（プロジェクトファイルの場所）を渡すことができ、

渡さない場合はデフォルトのプロジェクトファイルが使用されます。

Preferences/BuildApp/BuildApp.xml を使用する

BUILD APPLICATION

プロジェクトファイルを指定する

BUILD APPLICATION(“C:¥¥My Project¥¥MyProj.xml”)

プロジェクトファイルが見つからない場合は、標準の「開く」ダイアログが表示されます。

ビルド前のコンパイルに失敗した場合は、関連するメッセージが表示されます。
アプリケーションビルドに成功すると、システム変数 OK に 1 が代入されます。

サンプルデータベース

起動するとデザインモードになるので、「デザインメニュー」から「アプリケーションビルド」を選択し、次のようにビルドの設定をします。

クライアント/サーバを選択

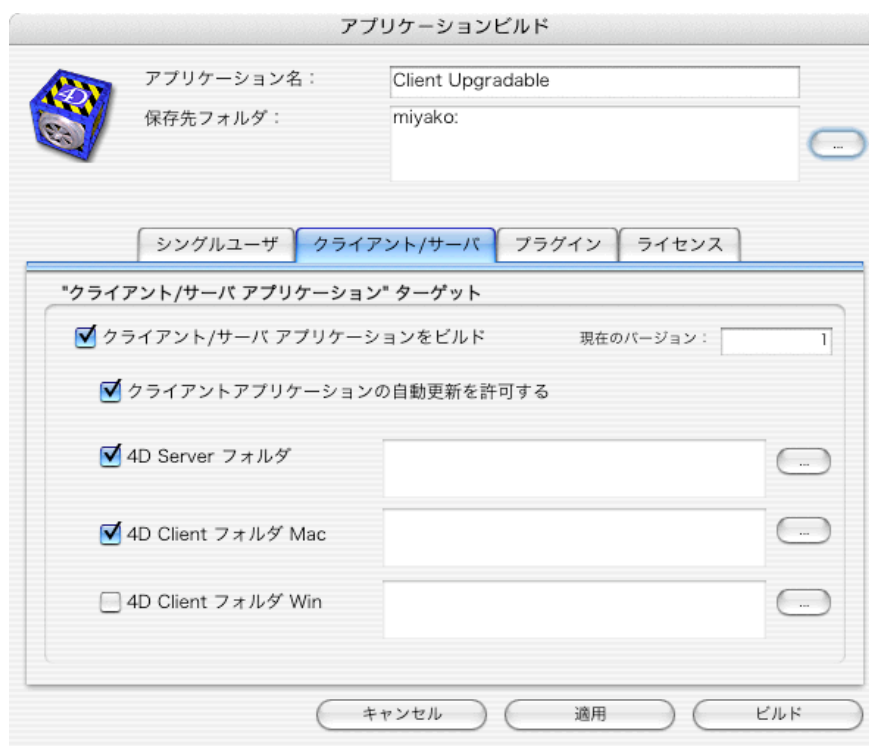
「クライアント/サーバアプリケーションをビルド」有効

現在のバージョン=1

「クライアントアプリケーションの自動更新を許可する」有効

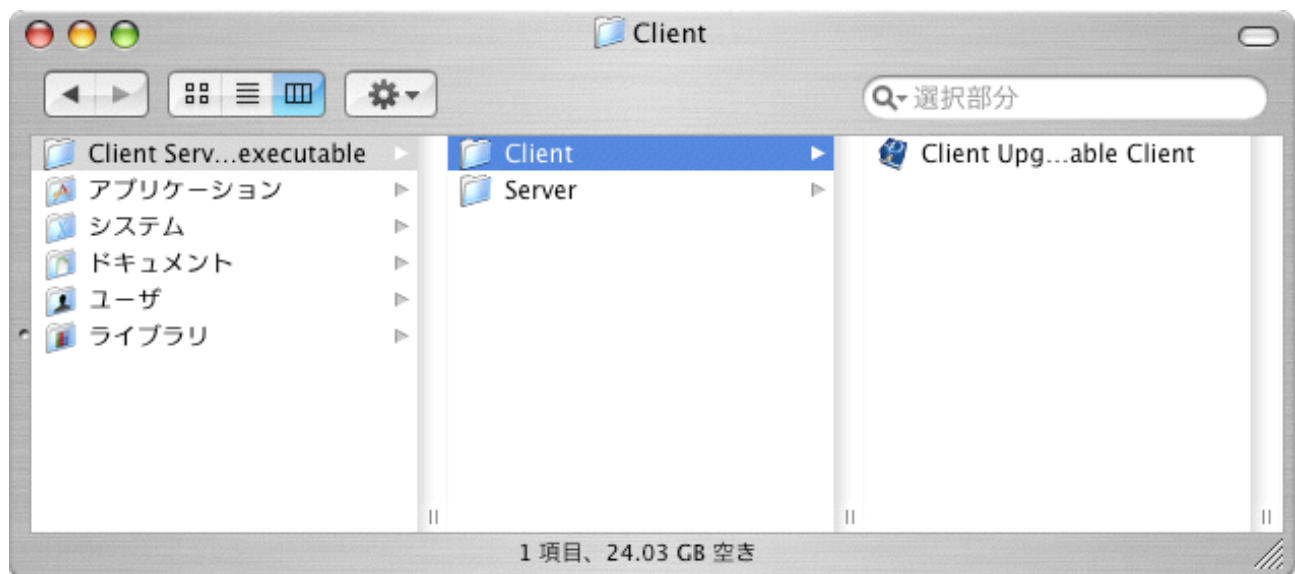
4D Server フォルダの場所を指定

4D Client フォルダの場所を指定



設定が済んだら、「ビルド」ボタンをクリックします。

ビルドに成功すると、「Client Server executable」フォルダにクライアントアプリケーション、サーバアプリケーションがそれぞれ作られます。

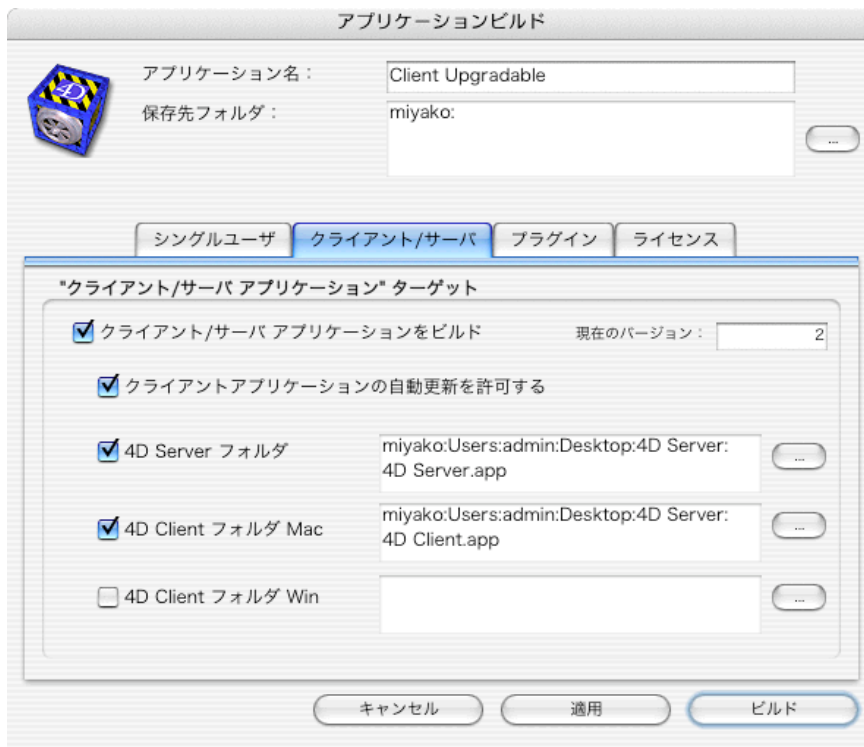


試しに「Client Server executable Server」をダブルクリックで起動し、次いで「Client Server executable Client」を起動してみましょう。接続ダイアログを経ないで、直接、サーバへ接続することができます。

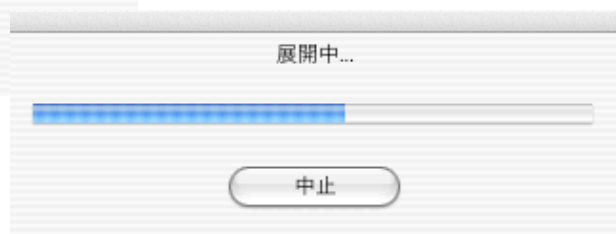
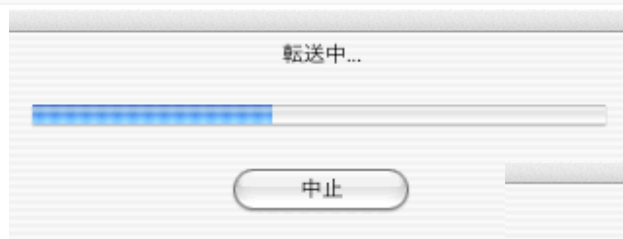
自動アップグレード

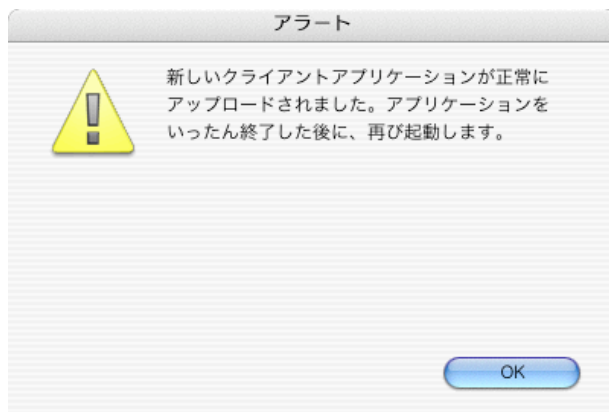
前述の手順で作成した「Client Server executable Client」のバックアップを取ります。(再ビルドによって古いバージョンが上書きされてしまうため)

アプリケーションビルドを再度、起動し、今度は「現在のバージョン」を2に設定します。再ビルドされたサーバアプリケーションをダブルクリックで起動します。バックアップしておいた古いクライアントアプリケーションを起動します。



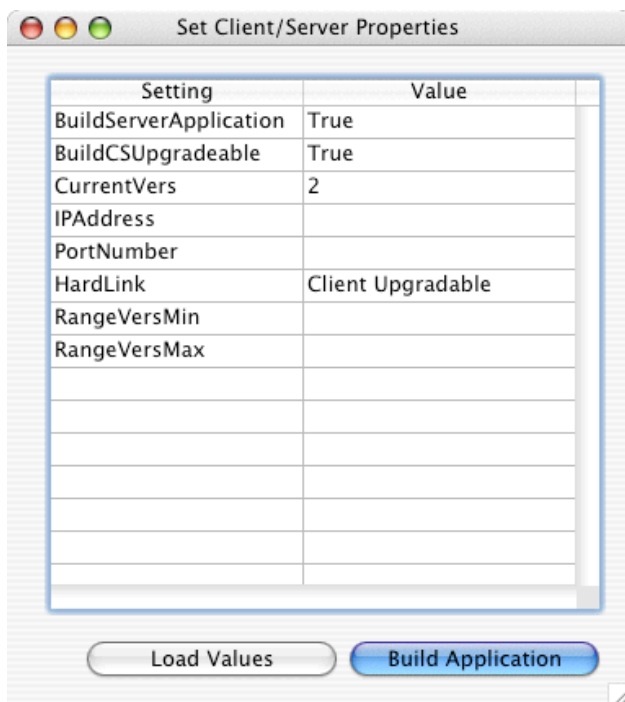
自動更新を行なうかを確認するダイアログが表示されます。更新をクリックすると、クライアントアプリケーションの自動更新が行なわれた後、4D Client が再起動します。





XML キーの設定

サンプルデータベースをカスタムモードで実行し、「ファイル」メニューから「Set CS Properties」を選択すると、デフォルトプロジェクトファイルの XML キーを編集するためのダイアログが表示されます。



設定例

バージョン「2」のクライアントアプリケーションのバックアップを取ります。

CurrentVers を 3 に設定

Setting RangeVersMin を 2 に設定

RangeVersMax を 3 に設定

「ビルド」ボタンをクリックします。

バージョン「2」のクライアントでバージョン「3」のサーバへの接続を試みます。

バージョンが古くても、許容範囲内であれば、自動更新のダイアログは表示されず、すぐにサーバへ接続することが確認できます。