



Director Article



Director MX と Macromedia Flash Communication Server MX

最近まで、デベロッパーの多くは、Director でマルチユーザー・アプリケーションを作成するのに、Shockwave Multiuser Server を使って来ましたが、今は Macromedia Flash Communication Server MX が使用可能な状況になっています。

この記事の中で、私は Macromedia Flash Communication Server MX の力強さを考察し、Director を併用したサンプルを示します。その中で Macromedia Flash Communication Server MX と Shockwave Multiuser Server の何が同じで何が異なるかを語っていきます。

Macromedia Flash Communication Server MX を使う理由

Macromedia Flash Communication Server MX は、一から作られた全く新しい製品です。それは、Multiuser Server を直接的に受け継ぐものではなく、Flash Communication Server では不可能なこともいくつかあります。もし、UDP(User Data Protocol)やピア・トゥ・ピアを必要とするなら、最新の Shockwave Multiuser Server を選択する方がよいでしょう。

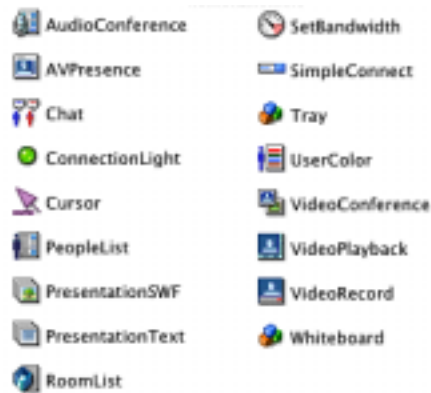
しかし、例えばクライアント-サーバー間でのメッセージのやりとりや、サーバー・サイド・スクリプティングなど、Shockwave Multiuser Server が提供したサービスの核の部分は、Flash Communication Server でも可能です。シンタックスこそ違え、基本的な機能は変わりません。

では Flash Communication Server の何が新しいのでしょうか？ 最も明確な違いはビデオとオーディオです。これらは確かに期待の大きい要素ではありますが、しかし Macromedia Flash Communication Server は、ただビデオ会議を実現させるだけの道具ではありません。

Macromedia Flash Communication Server MX は、Shockwave Multiuser Server よりヘビーなユーザーをサポートすべくデザインされています。同時に何万ものユーザーが接続するテストに成功し、サーバーの状態や統計を見るための管理者ツールや API が予め作成されるなど、管理者へのサービスも向上しています。Macromedia Flash Communication Server MX はまた、.NET や ColdFusion といったアプリケーション・サーバーに接続する新しい可能性を開く Macromedia Flash Remoting もサポートしています。

Macromedia Flash Communication Server MX にはまた、ドラッグ・アンド・ドロップ方式のコンポーネントが提供

されています。コンポーネントとはビヘイビアのようなもので、作成するアプリケーションに、共有ホワイトボードやビデオ、チャットといったコミュニケーション要素を容易に追加できるようにするものです。



Macromedia Flash ムービーでコンポーネントを配置し、Director MX のスコアにそのムービーを置くだけでいいのです。

最後になりましたが、SharedObject と呼ばれる新機能があります。これは私の個人的なお気に入りです。次のサンプル・アプリケーションでは、Director で Macromedia Flash Communication Server MX を使用した SharedObject やクライアント-サーバー・メッセージング、他の詳細についてご紹介します。

サンプル・アプリケーション

このサンプルでは、それぞれのユーザーはアバター(アイコン)、この場合は魚、になります。ユーザーは、魚の大きさや速さ、回転を変更でき、その変化が他のユーザーが見ているそれぞれの画面上でも現れます。コード自体に焦点を当てて平易にするため、機能などはごくシンプルにまとめてあります。


これは何人かのユーザーが泳いでいる画面です。



スタート サンプルを動作させるには、Macromedia Flash Communication Server をインストールするか、それが常に稼働しているマシンにアクセスできる環境が必要になります。また、Macromedia Flash Communication Server MX をサポートしている Web ホスティング会社と契約する方法もあります。

ソース・コードのファイルはここからダウンロードできます。

 fish.zip(561k)

 fish.sit(476k)

サーバー・サイド アプリケーションのサーバー・サイドから始めましょう。**main.asc** ファイルをサーバーの以下の場所にコピーします。

```
/Flash Communication Server MX /flashcom /applications /fish /main.asc
```

サーバー・サイドでアプリケーションが必要とするのは、ユニークな名前のつけられたフォルダだけです。main.asc ファイルは空であってもかまいませんし、全く飛ばしてもかまいません。しかし今の場合、main.asc ファイルには、Fish アプリケーションの管理をサポートするサーバー・サイドの ActionScript が含まれています。OnAppStart という最初のファンクションは、アプリケーションがスタートしたとき、通常初めてのユーザーが入ってきたとき、実行されます。

以下の行を見てください。

```
Application.users = SharedObject.get("users",false);
```

ここでは、アプリケーション内で全ユーザーを追跡する SharedObject を作成しています。SharedObject はすぐに見慣れたものになるので心配は無用です。

Fish のサーバー・サイド・スクリプトの次のファンクションは、application.onConnect()です。名前から分かるように、このファンクションは、新しいユーザーがアプリケーションに接続してきたらいつでもコールされます。このファンクションでは多くのことが進行していますが、2, 3の重要なコードを見てください。

ファンクションの最初の行では3つの引数があります。

```
Application.onConnect = function(client, username, userinfo)
```

クライアント・オブジェクトである第一の引数は、自動的に接続ファンクションに渡ります。次のユーザー名とユーザー情報の引数は、クライアント上で Director の Lingo コードから渡すカスタムの引数です。これは、接続が許可される前でも一方的にサーバーに情報を取りに行きます。

次に、ユニークな数値とユーザーの ID を作成します。この情報は、本人であることを確認する目的で、ユーザーに戻されます。

```
//ユーザーのIDが何であるか伝えるためにクライアントをコールする
```

```
var vbSuccess = client.call("setUserNumber",0,userNumber);
```

これはサーバー-クライアント間でのメッセージングの簡単な例です。Shockwave Multiuser Server を使って同じようなコールを作成していたことがあるなら、その経験が役立ちます。

最後に各ユーザーの SharedObject を作成します。

```
UserSO = SharedObject.get(userid,false);
```

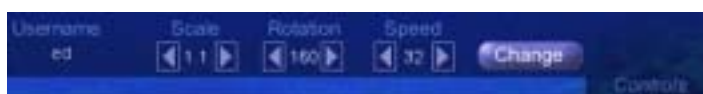
この SharedObject は各ユーザー情報を追跡するために使用します。パラメータの"false"は、ユーザーが切断した後、サーバーにこのオブジェクトを保有しないよう告げるものです。

SharedObject の何がすばらしいか

SharedObject はデータのコンテナです。それぞれの SharedObject は情報を運ぶ多くの名前をついたスロットを持っています。あるスロットのデータが変化したら、サーバーは参加している全クライアントにその変化を知らせます。単純なコンセプトに思えますが、マルチユーザーの設定では非常に有用です。変化を記述するメッセージを作成し、グループの全員にそれを送る必要はないのです。

Macromedia Flash Communication Server MX の SharedObject は、Shockwave Multiuser Server のグループ属性と似ています。しかし SharedObject はグループ属性の複雑さをなくし、それ自体に新しい機能を加え、アプリケーションで使用するときにも Shockwave Multiuser Server より簡単なものになっています。

このアプリケーションの各ユーザーは、自分の魚を現わす自分自身の SharedObject を持つこととなります。SharedObject は、速さ、スケール、回転といったいくつかのプロパティ(スロット)を持ちます。クライアントのムービーがそのスロットを更新すると、他の全てのクライアントは自動的にその変化を知るのです。



クライアント・サイド

いよいよ Director ムービーの fish.dir を開けてみましょう。クライアント・サイドでのコードのほとんどは Lingo で書かれています。ステージで最初に気づくことは、次の URI を含むフィールド部分でしょう。

```
rtmp://www.your_fcs_domain/fish
```

このアドレスをご使用の Flash Communication Server のドメイン名に合うよう変更してください。

ルームとアプリケーション・インスタンス

上記 URI の最後の "fish" に注目してください。これは、Macromedia Flash Communication Server MX に、ここで Fish アプリケーションを起動させたい、アプリケーションのデフォルトのインスタンスを加えたいという場所を伝えるものです。クラシックなマルチユーザー環境では、インスタンスは時に "ルーム" と呼ばれます。ユーザーは "ロビー" で出会い、2人ひと組になってルームに入り、ユーザーの組に出会います。

/fish は、Macromedia Flash Communication Server MX が、前の Shockwave Multiuser 構造から少し発展した別のエリアです。Macromedia Flash Communication Server MX を使うと、ルームを作るのにグループを作成する必要も、異なるユーザーに異なる "joinGroup" メッセージを送る必要もありません。その代わりにアプリケーションの URI の最後に単語を加えるだけでよいのです。

言いかえれば、このルーム:./fish/PacificOcean は、このルーム:./fish/AtlanticOcean とは異なる Fish アプリケーションのインスタンスとなる訳です。

両方のルームとも同じサーバー・サイド・スクリプト(main.asc)の新しいインスタンスを生み出し、スクリプトの全ての変数はそれぞれのインスタンス固有のものとなります。これが、ルームをセットアップしユーザーを異なるグループに配分するという過程を簡単なものになっているのです。

URIs のチップ: 管理者の中には、サーバーのドメインに合致しないアドレスからの接続は拒否するよう、Macromedia Flash Communication Server MX をセットアップしたいと考える方もいらっしゃるでしょう。これはセキュリティ上

の理由であり、ホスティング会社ではよくあることです。残念ながら、テストするにはサーバーにムービーをアップロードしなければならず、それが開発やデバッグを難しくしています。もしローカル・マシンから Macromedia Flash Communication Server MX に接続できないと分かったら、管理者に頼んで、Macromedia Flash Communication Server MX のセットアップの”allow”セクションに、お使いの IP アドレスを追加してもらってください。

接続の作成

Shockwave Multiuser Server のコールは全て Multiuser Xtra を通してのものでしたが、Macromedia Flash Communication Server MX は、クライアントとサーバーを結ぶ道筋として Macromedia Flash Player を使います。接続には、Macromedia Flash スプライトを使用するか、グローバルな Macromedia Flash オブジェクトを作成します。このサンプル・アプリケーションでは、Macromedia Flash スプライトを使用しています。

Note: Macromedia Flash スプライトは Director のステージの見える範囲内にあることを確認してください。そうしないとスプライトはサーバーからコールバックを受け取るのに必要なサービスを受けられなくなります。この Macromedia Flash スプライトは、グラフィクスを含む必要はなく、私はインクをバックグラウンド・トランスペアレントに設定することでユーザーから隠しています。

Fish での Macromedia Flash Communication Server MX Lingo のほとんどは、”Fishcom behavior”という名前のスクリプト・メンバー内にあります。Macromedia Flash Communication Server MX での最初のステップは、Shockwave Multiuser Server でもそうであったように、接続を作成することです。これは”connectToServer”というメソッドの中で行われます。

```
on connectToServer (me, aUri, aUsername)
    --[一部: いくつかの初期化コード]
    pConnection = pSprite.newObject(“NetConnection”)
    pSprite.setCallback(pConnection, “onStatus”, #connectionStatusHandler, me)
    pSprite.setCallback(pConnection, “setUserNumber”, #setUserNumber, me)
    pConnection.connect(aUri, aUsername)
end
```

ここでは Macromedia Flash スプライトである pSprite を使用して接続オブジェクトを作成し、2つのコールバックを設定しています。この過程は Shockwave Multiuser Server を使用した netMessageHandlers の設定に非常によく似ています。

初めのコールバックは、接続についてのステータス・メッセージとともにサーバーがコールするメソッドを定義しています。これは接続が許可されたか拒否されたかを知らせるためにサーバーがコールバックするとすぐに分かります。

このメソッドの2番目のコールバックは、setUserNumber のためのものです。サーバー・サイド・スクリプトがユーザー ID を返すためにクライアントにコールさせたことを思い出すかも知れません。ここで setCallback を使って、クライアント上の適切なメソッドにメッセージを送ったのです。

このコールバックは、サーバーからクライアントに送られるメッセージの完全かつ簡単な例です。このように簡単にコールするほかの方法は今この場では詳細に踏み込んでしまうのでやめますが、これは基本的なシンタックスです。

クライアント上の Lingo:

```
pConnection.call("someMethod","someArgument")
```

サーバー上の ActionScript:

```
Client.prototype.someMethod = function(someArgument){  
    //ここで動作を定義  
}
```

接続を作成しサーバーからユーザー ID を受け取れるようになり、アプリケーション・インスタンス内の全ユーザー間で、この接続を使ってデータを受け渡す準備は整えられました。

SharedObject を読む

やろうと思えばメッセージを使ってデータを戻したり先へ送ったりできます。これは Shockwave Multiuser Server ユーザーにはなじみのある手法です。しかしこのサンプルでは、代わりに SharedObject を使います。

“Flashcom behavior”には newUser というメソッドが含まれています。ここでは各ユーザー、つまり魚それぞれを表す SharedObject を読みます。

SharedObject を読むには、単純な過程で事足ります。オブジェクトを取得し、それに接続するのです。

```
tUserSO = tSharedObject.getRemote(("U" & tUserNumber),pConnection.uri,FALSE)  
tUserSO.connect(pConnection)
```

初めに読むのは、各ユーザー自身の魚のデータを保持する SharedObject です。その後は新しいユーザーが認知されるごとに新しい SharedObject を読みます。これは通常ユーザーのリストが変化すると発生します。SyncUsers というメソッドには、ユーザー・リストを監視するコードが書かれています。

データを受け取る SharedObject を使う

例えば魚のスピードを遅くするなどして、他のユーザーの魚が変化すると、それはクライアント・ムービーに伝えられ”同調”します。setCallback を使ってこの変化をスプライトに伝え、そのユーザーの魚を表示させます。

“User sprite”というスクリプト・メンバーを開いてください。その中に syncUserObject というメソッドがあります。ここでは変化をユーザーの SharedObject に同調させる処理を行っています。

SharedObject が状態を変化させると、そのイベントがコード・タイプの1つになります。コード・タイプ“change”で発生するイベントを見てください。以下ではこのコード・タイプを探し、変化のあったデータ・スロットの名前を取得し、そのスロットの新しい値を取得しています。

```
Case tCode of  
    “change”,“success”:  
        tSlotName = symbol(tNextInList.name)  
        tSlotValue = value(“tData” & tSlotName)
```

その後新しい値を使ってそのユーザーの魚を変化させます。例えば、tSlotName が”speed”で tSlotValue が”10”だったとしたら、ユーザーの魚スプライトに速さを 10 に設定するよう伝えるのです。

スロットが変化すると、ShareObject 全部でなくそのスロットのデータだけが送られます。もしパフォーマンスや帯域幅の扱いに心配があるなら、データ構築を設計するときこのことに留意してください。

データを送る SharedObject を使う

そろそろ逆方向にデータを送って、他のユーザーに自分の魚が変化したことを伝えたいと思われるでしょう。そうするには、SharedObject 上の適切なスロットを変化させるだけです。

“Flashcom behavior”には以下のメソッドがあります。

```
On setUserProperty(me, aSharedObject, aSlotName, aSlotValue)
```

```
    tSOsetter = pSprite.getValue(“gSOsetter”, FALSE)
```

```
    tSOsetter.setDataSlot(aSharedObject, aSlotName, aSlotValue)
```

```
end
```

setDataSlot は Macromedia Flash スプライトのファンクションで、SharedObject を変更できます。例えば、スロット名が”rotation”でスロット値が”180”にしたこのメソッドを魚の SharedObject に渡すと、魚の回転を反転できます。

必要なことはデータを送ることです。他の全クライアントはイベントを受け取り、画面上の魚の回転が変化します。

このイベントを確認するには、メッセージ・ウィンドウを開いて、そこに表示される”put”メッセージを見てください。またマウスを魚の上に乗せるとその魚のユーザー名が表示されます。

無論、Flash Communication Server は、魚の分身の群れ以上にできることはあります。この記事が Director MX で Flash Communication Server を必要とする際の基本要素を理解する助けとなることを願います。

追加リソース

デバッグと管理者ツール : Flash Communication Server MX にはよくできた管理者ツールとデバッグ・ツールがあります。Communication App Inspector は始めるのに最適の場所です。サーバーにインストールされている”Getting Started”上にあります。



サーバー・サイド・コード中に trace メッセージを埋め込んで、Communication App Inspector でそのメッセージを確認することができます。Macromedia Flash にはまた、クライアント・サイドでのオーサリング環境から使用できるデバッグ・ツールが提供されています。

– **Macromedia Flash Communication Server MX Development Center:** Flash Communication Server には、Application Development Center が用意され、そこではサーバーやコンポーネント、Flash Remoting や他のトピックスに関する多くの記事を読むことができます。ここではまたサーバーやコミュニケーション・コンポーネントのアップデートを調べることができます。

– **Macromedia Flash Communication Server MX forums:**Macromedia の Flash Communication Server フォーラムではトピックスや経験レベルでの話題を扱っています。Chatty Flashcom forum は、一歩進んだ Macromedia Flash や ActionScript 開発者の発言が盛んですが、経験レベルにかかわらず良い情報源となっています。

– **ActionScript:**ActionScript にまだなじんでいないなら、checking out ActionScript は有効です。特に JavaScript の知識があれば、書けるようになるまで時間はかからないでしょう。Director MX 内のクライアント・サイドから多くの ActionScript を使用する必要はなく、サーバー・サイドで少量の ActionScript を書くことで、Macromedia Communication Server MX アプリケーションは大いなる変貌をとげます。

著者について

ジョン・テイラーは Zingtime 社の創立者にしてマネージング・ディレクター。10年以上にわたって Electronic Arts や Lucas Learning、NFL などのクライアントに対し、ゲームや教育ソフト作成の経験を持つ。shockwave.com サイト初のマルチユーザー・ゲームのデザイン、開発をリードし、Shockwave Multiuser Server を使った Macromedia's PowerApp サンプルを多数制作している。MIT Media Lab でソフトウェア・デザインを、Harvard Graduate School of Education で教育技術を学んだ。

この記事は Macromedia 社サイトの DevNet に掲載されている “Director Article” を(有)ヒム・カンパニーが勝手に翻訳したものです。