

原文は[こちら](#)をご覧ください。

wxConnection

wxConnection オブジェクトは、クライアント - サーバ間の接続を表現する。[wxClient](#) オブジェクトを使用した接続の確立、または [wxServer](#) オブジェクトにより接続が受け入れられることにより作成される。[wxConnection](#) のメンバを呼ぶことにより、またはそのメンバをオーバーライドすることにより、DDE (Dynamic Data Exchange) ライクな通信の大部分を制御できる。wxDDEConnection を使用した実際の DDE ベースの実装は、Windows でのみ利用可能である。しかし、同じ API を持つ wxTCPConnection を使用すると、プラットフォームに依存しない、ソケットベースの API の使用が可能である。

A [wxConnection](#) object represents the connection between a client and a server. It is created by making a connection using a [wxClient](#) object, or by the acceptance of a connection by a [wxServer](#) object. The bulk of a DDE-like (Dynamic Data Exchange) conversation is controlled by calling members in a [wxConnection](#) object or by overriding its members. The actual DDE-based implementation using [wxDDEConnection](#) is available on Windows only, but a platform-independent, socket-based version of this API is available using [wxTCPConnection](#), which has the same API.

アプリケーションは、コミュニケーションイベントハンドラをオーバーライドするために、通常 [wxConnection](#) から新しい接続クラスを継承すべきである。

An application should normally derive a new connection class from [wxConnection](#), in order to override the communication event handlers to do something interesting.

plugin::pdf::PDFParser=HASH(0x534d58) 基本クラス

[wxConnectionBase](#)

wxObject

plugin::pdf::PDFParser=HASH(0x534d58) インクルードファイル

<wx/ipc.h>

plugin::pdf::PDFParser=HASH(0x534d58) 型

wxIPCFormat は以下のように定義されている。 :

wxIPCFormat is defined as follows:

```
enum wxIPCFormat
{
    wxIPC_INVALID =          0,
    wxIPC_TEXT =             1, /* CF_TEXT */
    wxIPC_BITMAP =          2, /* CF_BITMAP */
    wxIPC_METAFILE =        3, /* CF_METAFILEPICT */
    wxIPC_SYLK =             4,
    wxIPC_DIF =              5,
    wxIPC_TIFF =             6,
    wxIPC_OEMTEXT =          7, /* CF_OEMTEXT */
    wxIPC_DIB =              8, /* CF_DIB */
    wxIPC_PALETTE =          9,
```

```

wxIPC_PENDATA =      10,
wxIPC_RIFF =         11,
wxIPC_WAVE =         12,
wxIPC_UNICODETEXT =  13,
wxIPC_ENHMETAFILE =  14,
wxIPC_FILENAME =     15, /* CF_HDROP */
wxIPC_LOCALE =        16,
wxIPC_PRIVATE =       20
};

```

plugin::pdf::PDFParser=HASH(0x534d58) 参考

wxClient, wxServer, プロセス間通信の概要

plugin::pdf::PDFParser=HASH(0x534d58) メンバ

wxConnection::wxConnection

wxConnection::Advise

wxConnection::Execute

wxConnection::Disconnect

wxConnection::OnAdvise

wxConnection::OnDisconnect

wxConnection::OnExecute

wxConnection::OnPoke

wxConnection::OnRequest

wxConnection::OnStartAdvise

wxConnection::OnStopAdvise

wxConnection::Poke

wxConnection::Request

wxConnection::StartAdvise

wxConnection::StopAdvise

wxConnection::wxConnection

wxConnection()

wxConnection(char* buffer, int size)

コネクションオブジェクトを構築する。wxConnectionin を継承したユーザ定義のコネクションオブジェクトが無い場合、コンストラクタを直接呼ぶべきではない。なぜなら、デフォルトのコネクションオブジェクトは、コネクションの要求（または受け入れ）の準備をするからである。しかし、ユーザが自身の継承コネクションオブジェクトを定義する場合、wxServer::OnAcceptConnection および（または）wxClient::OnMakeConnection メンバは、新しいコネクションオブジェクトを構築するような関数に置き換えるべきである。

Constructs a connection object. If no user-defined connection object is to be derived from wxConnection, then the constructor should not be called directly, since the default connection object will be provided on requesting (or accepting) a connection. However, if the user defines his or her own derived connection object, the wxServer::OnAcceptConnection and/or wxClient::OnMakeConnection members should be replaced by functions which construct the new connection object.

wxConnection コンストラクタに引数が無い場合、wxConnection オブジェクトは、必要なメモリを確保して自身でコネクションバッファを管理する。プログラマが準備したバッファは必要に応じて増やされることは無いため、十分なバッファが無いと、プログラムはアサートする。プログラマが準備したバッファは、主に下位互換のために存在する。

If the arguments of the wxConnection constructor are void then the wxConnection object manages its own connection buffer, allocating memory as needed. A programmer-supplied buffer cannot be increased if necessary, and the program will assert if it is not large enough. The programmer-supplied buffer is included mainly for backwards compatibility.

wxConnection::Advise

bool Advise(const wxString& item, char* data, int size = -1, wxIPCFormat format = wxCF_TEXT)

渡された item に関連付けられたデータの変更をクライアントに通知するために、サーバアプリケーションにより呼ばれる。クライアントコネクションの wxConnection::OnAdvise メンバが呼ばれる。成功時には、true が返る。

Called by the server application to advise the client of a change in the data associated with the given item. Causes the client connection's wxConnection::OnAdvise member to be called. Returns true if successful.

wxConnection::Execute

bool Execute(char* data, int size = -1, wxIPCFormat format = wxCF_TEXT)

サーバ上のコマンドを実行するために、クライアントアプリケーションにより呼ばれる。サーバに対する任意のデータを転送することもできる（その点では wxConnection::Poke と同様である）。サーバコネクションの wxConnection::OnExecute が呼ばれる。成功時には true を返す。

Called by the client application to execute a command on the server. Can also be used to transfer arbitrary data to the server (similar to wxConnection::Poke in that respect). Causes the server connection's wxConnection::OnExecute member to be called. Returns true if successful.

wxConnection::Disconnect

bool Disconnect()

クライアントまたはサーバアプリケーションにより、他のプログラムから切断するために呼ばれる。wxConnection::OnDisconnect メッセージが他のプログラムの対応するコネクションオブジェクトに送られる。成功時、または既に切断されているときにはtrueを返す。Disconnectを呼んだアプリケーションは、その接続側で明示的に破棄されなければならない。

Called by the client or server application to disconnect from the other program; it causes the wxConnection::OnDisconnect message to be sent to the corresponding connection object in the other program. Returns true if successful or already disconnected. The application that calls Disconnect must explicitly delete its side of the connection.

wxConnection::OnAdvise

virtual bool OnAdvise(const wxString& topic, const wxString& item, char* data, int size, wxIPCFormat format)

Advise を使用した時に、渡された item に関連したデータの変更をサーバがクライアントに通知する際、クライアントアプリケーションに送られるメッセージ。

Message sent to the client application when the server notifies it of a change in the data associated with the given item, using Advise.

wxConnection::OnDisconnect

virtual bool OnDisconnect()

他のアプリケーションが接続の終了を通知する際に、クライアントまたはサーバアプリケーションに送られるメッセージ。デフォルトの動作は、コネクションオブジェクトを破棄し、true を返す。そのため、コネクションオブジェクトがもう使用できないことを知るために、通常、アプリケーションは OnDisconnect をオーバーライドする（最後に、継承メソッドを呼んだほうが良い）。

Message sent to the client or server application when the other application notifies it to end the connection. The default behaviour is to delete the connection object and return true, so applications should generally override OnDisconnect (finally calling the inherited method as well) so that they know the connection object is no longer available.

wxConnection::OnExecute

virtual bool OnExecute(const wxString& topic, char* data, int size, wxIPCFormat format)

Execute を使用した時、渡された data を実行するよう、クライアントがサーバに通知する際にサーバアプリケーションに送られるメッセージ。このメッセージに関連付けられる item が無いことに注意。

Message sent to the server application when the client notifies it to execute the given data, using Execute. Note that there is no item associated with this message.

wxConnection::OnPoke

virtual bool OnPoke(const wxString& topic, const wxString& item, char* data, int size, wxIPCFormat format)

渡された data を受け付けることをクライアントがサーバに通知する際、サーバアプリケーションに送られるメッセージ。

Message sent to the server application when the client notifies it to accept the given data.

wxConnection::OnRequest

virtual char* OnRequest(const wxString& topic, const wxString& item, int *size, wxIPCFormat format)

クライアントが `wxConnection::Request` を呼んだときに、サーバアプリケーションに送られるメッセージ。サーバの `OnRequest` メソッドは、文字列で応答するか、データが無いことを意味する `NULL` で応答し、`*size` を設定しなければならない。当然、その文字列は、関数から戻った後も継続しなければならない。

Message sent to the server application when the client calls `wxConnection::Request`. The server's `OnRequest` method should respond by returning a character string, or `NULL` to indicate no data, and setting `*size`. The character string must of course persist after the call returns.

`wxConnection::OnStartAdvise`

`virtual bool OnStartAdvise(const wxString& topic, const wxString& item)`

クライアントが、渡された `topic` と `item` に対する "通知ループ (advise loop)" の開始を希望するとき、クライアントからサーバアプリケーションに送られるメッセージ。サーバは、`false` を返すことで、拒否することができる。

Message sent to the server application by the client, when the client wishes to start an 'advise loop' for the given topic and item. The server can refuse to participate by returning `false`.

`wxConnection::OnStopAdvise`

`virtual bool OnStopAdvise(const wxString& topic, const wxString& item)`

クライアントが、渡された `topic` と `item` に対する "通知ループ (advise loop)" の停止を希望するとき、クライアントからサーバに送られるメッセージ。サーバは、`false` を返すことにより通知ループの停止を拒否することができるが、実際にはそれほど意味が無い。

Message sent to the server application by the client, when the client wishes to stop an 'advise loop' for the given topic and item. The server can refuse to stop the advise loop by returning `false`, although this doesn't have much meaning in practice.

`wxConnection::Poke`

`bool Poke(const wxString& item, char* data, int size = -1, wxIPCFormat format = wxCF_TEXT)`

サーバにデータを送信 (poke) するために、クライアントアプリケーションから呼ばれる。サーバに任意のデータを転送する際に使用できる。サーバコネクションの `wxConnection::OnPoke` メンバが呼ばれる。サイズが `-1` の場合、サイズは文字列長から計算される。

Called by the client application to poke data into the server. Can be used to transfer arbitrary data to the server. Causes the server connection's `wxConnection::OnPoke` member to be called. If size is `-1` the size is computed from the string length of data.

成功時には `true` を返す。

Returns `true` if successful.

`wxConnection::Request`

char* Request(const wxString& item, int *size, wxIPCFormat format = wxIPC_TEXT)

サーバからのデータを要求するために、クライアントアプリケーションから呼ばれる。サーバコネクションの wxConnection::OnRequest メンバが呼ばれる。size は、NULL または要求された item のサイズを受け取るための引数へのポインタである。

Called by the client application to request data from the server. Causes the server connection's wxConnection::OnRequest member to be called. Size may be NULL or a pointer to a variable to receive the size of the requested item.

成功時には文字列(実際には、コネクションのバッファへのポインタ)を返し、失敗時には NULL を返す。このバッファは削除されてはならない。

Returns a character string (actually a pointer to the connection's buffer) if successful, NULL otherwise. This buffer does not need to be deleted.

wxConnection::StartAdvise

bool StartAdvise(const wxString& item)

サーバで通知ループ (advise loop) が開始したかどうかを問い合わせるために、クライアントアプリケーションにより呼ばれる。サーバコネクションの wxConnection::OnStartAdvise メンバが呼ばれる。サーバが了承した場合には true を返し、そうでなければ false を返す。

Called by the client application to ask if an advise loop can be started with the server. Causes the server connection's wxConnection::OnStartAdvise member to be called. Returns true if the server okays it, false otherwise.

wxConnection::StopAdvise

bool StopAdvise(const wxString& item)

通知ループ (advise loop) が停止できるかどうか問い合わせるために、クライアントアプリケーションにより呼ばれる。サーバコネクションの wxConnection::OnStopAdvise メンバが呼ばれる。サーバが了承した場合は true を返し、そうでなければ false を返す。

Called by the client application to ask if an advise loop can be stopped. Causes the server connection's wxConnection::OnStopAdvise member to be called. Returns true if the server okays it, false otherwise.