

原文は[こちら](#)をご覧ください。

## wxServer

wxServer オブジェクトは、クライアント - サーバ間の DDE ( Dynamic Data Exchange ) ライクな通信のサーバ部分を表現する。wxDDEServer 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 wxTCPServer 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 wxServer object represents the server part of a client-server DDE-like (Dynamic Data Exchange) conversation. The actual DDE-based implementation using wxDDEServer is available on Windows only, but a platform-independent, socket-based version of this API is available using wxTCPServer, which has the same API.

適合するクライアントとの通信を行うサーバを作成するためには、wxConnection を継承したクラスと、wxServer を継承したクラスを準備する必要がある。カスタマイズした wxConnection クラスはクライアントとの通信を受信する。カスタマイズした wxServer では、コネクションを確立するときに、ユーザによりオーバーライドされた wxServer::OnAcceptConnection メンバが要求されたクラスの wxConnection を返すことができるよう要求される。IPC サンプルと[プロセス間通信の概要](#)を参照のこと。

To create a server which can communicate with a suitable client, you need to derive a class from wxConnection and another from wxServer. The custom wxConnection class will intercept communications in a 'conversation' with a client, and the custom wxServer is required so that a user-overridden wxServer::OnAcceptConnection member can return a wxConnection of the required class, when a connection is made. Look at the IPC sample and the [Interprocess communications overview](#) for an example of how to do this.

plugin::pdf::PDFParser=HASH(0x53e184) 基本クラス

### wxServerBase

plugin::pdf::PDFParser=HASH(0x53e184) インクルードファイル

<wx/ipc.h>

plugin::pdf::PDFParser=HASH(0x53e184) 参考

wxClient, wxConnection, [プロセス間通信の概要](#)

plugin::pdf::PDFParser=HASH(0x53e184) メンバ

wxServer::wxServer

wxServer::Create

wxServer::OnAcceptConnection

wxServer::wxServer

## wxServer()

サーバオブジェクトを構築する。

Constructs a server object.

## wxServer::Create

bool Create(const wxString& service)

渡された service 名を使用してサーバを登録する。Unix では、サーバ名は、インターネットドメインソケットが通信に使用される場合にはポート番号を、Unix ドメインソケットを使用する場合には適当なファイル名(存在しないファイル。後で削除される。)のどちらかを指定する。呼び出しに失敗すると(例えば、ポート番号が既に使用されている) false が返る。

Registers the server using the given service name. Under Unix, the service name may be either an integer port identifier in which case an Internet domain socket will be used for the communications, or a valid file name (which shouldn't exist and will be deleted afterwards) in which case a Unix domain socket is created. false is returned if the call failed (for example, the port number is already in use).

## wxServer::OnAcceptConnection

virtual wxConnectionBase \* OnAcceptConnection(const wxString& topic)

クライアントが MakeConnection を呼ぶと、サーバはメッセージを受け取り、このメンバが呼ばれる。アプリケーションは、このメッセージを受信するためにメンバを継承し、標準の wxConnection 型のコネクションオブジェクト、またはユーザが継承した型を返さなければならない。

When a client calls MakeConnection, the server receives the message and this member is called. The application should derive a member to intercept this message and return a connection object of either the standard wxConnection type, or (more likely) of a user-derived type.

topic が STDIO の場合、アプリケーションはコネクションの拒否を望むかもしれない。UNIX では、サーバが作成されたとき、標準入出力の度に OnAcceptConnection を受け取る。しかし、DDE メッセージに関しては、意味をなさない。

If the topic is STDIO, the application may wish to refuse the connection. Under UNIX, when a server is created the OnAcceptConnection message is always sent for standard input and output, but in the context of DDE messages it doesn't make a lot of sense.