Именованным каналом называется объект который обеспечивает обмен данными между процессами, выполняющимися на компьютерах в одной локальной сети. Процесс, создающий именованный канал, называется ***сервером именованного канала***. Процессы, которые связываются с именованным каналом, называются ***клиентами именованного канала***. Любой именованный канал идентифицируется своим именем, которое задается при создании канала.

Именованные каналы бывают: ***дуплексные*** (позволяющие передавать данные в обе стороны) и ***полудуплексны***е (позволяющие передавать данные только в одну сторону). Передача данных в именованном канале может осуществляться как потоком, так и сообщениями. Обмен данными в канале может быть ***синхронным*** и ***асинхронным***.

// **-- создать именованный канал**

// **Назначение:** функция предназначена для создания

// именованного канала

**HANDLE CreateNamedPipe**

**(**

**LPCTSTR pname,** // [in] символическое имя канала

**DWORD omode,** // [in] атрибуты канала

**DWORD pmode,** // [in] режимы передачи данных

**DWORD pimax,** // [in] макс. к-во экземпляров канала

**DWORD osize,** // [in] размер выходного буфера

**DWORD isize,** // [in] размер входного буфера

**DWORD timeo,** // [in] время ожидания связи с клиентом

**LPSECURITY\_ATTRIBUTES sattr** // [in] атрибуты безопасности

**);**

// **Код возврата:** в случае успешного завершения функция

// возвращает дескриптор именованного канала, иначе

// возможны следующие значения:

// **INVALID\_HANDLE\_VALUE** – неудачное завершение;

// **ERROR\_INVALID\_PARAMETER** - значение параметра **pimax**

// превосходит величину **PIPE\_UNLMITED\_INSTANCES**

// **Примечание: pname** - указывает на строку именем канала в

// локальном формате;

// **omode** - задает флаги направления передачи, например

// флаг **FILE\_ACCESS\_DUPLEX** разрешает чтение и запись в

// канал; помимо направления здесь могут быть заданы

// флаги асинхронной передачи, режимы буферизации и

// безопасности;

// **pmode** – задает флаги способов передачи данных,

// например, флаг **PIPE\_TYPE\_MESSAGE|PIPE\_WAIT** разрешает

// запись данных сообщениями в синхронном режиме, а флаг

//  **PIPE\_READTYPE\_MESSAGE|PIPE\_WAIT** разрешает чтение

// сообщений в синхронном режиме;

// **pimax –** максимальное количество экземпляров канала,

// значение должно находиться в пределах от 1 до

//  **PIPE\_UNLMITED\_INSTANCES**;

// **osize, isize –** значения рассматриваются Windows

// только как пожелания пользователя (рекомендуется **0**);

// **timeo** – параметр устанавливает время ожидания связи c

// c сервером в миллисекундах для функции **WаitNamedPipe**

// с параметром **NMWAIT\_USE\_DEFAULT\_WAIT;** может быть

// установлено значение **INFINITE** (ждать бесконечно);

// **sattr –** для установки атрибутов безопасности

// по умолчанию, следует установить значение **NULL**

**\\.\pipe\xxxxx**

где: **точка (.) -** обозначает локальный компьютер;

**pipe -** фиксированное слово;

**xxxxx -** имя канала

,...

// **-- соединить сервер с именованным каналом**

// **Назначение:** функция предназначена для

// подсоединения к экземпляру именованного канала

// клиента(который переводит наш сервер в состояние ожидание пока клиенты не подключатся)

**BOOL ConnectNamedPipe**

**(**

**HANDLE hP,** // [in] дескриптор именованного канала

**LPOVERLAPPED ol** // [in,out] используется для асинхр. связи

**);**

// **Код возврата:** в случае успешного завершения функция

// возвращает **TRUE**, иначе **FALSE**

// **Примечание:** параметр **ol** используется только в том случае,

// если используется асинхронная связь, в случае

// синхронной связи можно установить значение **NULL**

//..................................................................

**HANDLE hPipe;** // дескриптор канала

**try**

**{**

**if ((hPipe = CreateNamedPipe("\\\\.\\pipe\\ConsolePipe",**

**PIPE\_ACCESS\_DUPLEX,** //дуплексный канал(способ передачи(двухнаправленный))

**PIPE\_TYPE\_MESSAGE|PIPE\_WAIT**, // сообщения(говорим что будем обмениваться сообщениями)|чтение сообщеуний синхронный

**1, NULL, NULL** // максимум 1 экземпляр

**DWORD osize,** // [in] размер выходного буфера

**DWORD isize,** // [in] размер входного буфера

**DWORD timeo,** // [in] время ожидания связи с клиентом

**LPSECURITY\_ATTRIBUTES sattr** // [in] атрибуты безопасности

**INFINITE**,**NULL))** **== INVALID\_HANDLE\_VALUE)**

**throw SetPipeError("create:",GetLastError());**

**if (!ConnectNamedPipe(hPipe,NULL**)) // ожидать клиента

**throw SetPipeError("connect:",GetLastError());**

//..................................................................

**}**

**catch (string ErrorPipeText)**

**{cout << endl << ErrorPipeText;}**

//..................................................................

// **-- читать данные из канала**

// **Назначение:** функция предназначена чтения данных из

// именованного канала

**BOOL ReadFile**

**(**

**HANDLE hP,**  // [in] дескриптор канала

**LPVOID pb,** // [out] указатель на буфер ввода

**DWORD sb,** //[in] количество читаемых байт

**LPDWORD ps,** // [out] количество прочитанных байт

**LPOVERLAPPED ol** // [in,out] для асинхронной обработки

**);**

// **Код возврата:** в случае успешного завершения функция

// возвращает **TRUE**, иначе **FALSE**

// **Примечание** если не используется асинхронная обработка

// параметр **ol** рекомендуется установить в **NULL**

// **-- писать данные в канал**

// **Назначение:** функция предназначена записи данных в

// именованный канал

**BOOL WriteFile**

**(**

**HANDLE hP,**  // [in] дескриптор канала

**LPVOID pb,** // [in] указатель на буфер вывода

**DWORD sb,** //[in] количество записываемых байт

**LPDWORD ps,** // [out] количество записанных байт

**LPOVERLAPPED ol** // [in,out] для асинхронной обработки

**);**

// **Код возврата:** в случае успешного завершения функция

// возвращает **TRUE**, иначе **FALSE**

// **Примечание** если не используется асинхронная обработка

// параметр **ol** рекомендуется установить в **NULL**

|  |  |
| --- | --- |
| **DisconnectNamedPipe** | Закончить обмен данными |

Closehandle—zakryvaet diskriptor

// **-- открыть канал**

// **Назначение:** функция предназначена для подключения клиента

// к именованному каналу

**HANDLE CreateFile**

**(**

**LPCTSTR pname,** // [in] символическое имя канала

**DWORD accss,** // [in] чтение или запись в канал

**DWORD share,** // [in] режим совместного использования

**LPSECURITY\_ATTRIBUTES sattr** // [in] атрибуты безопасности

**DWORD oflag,** // [in] флаг открытия канала (открытие существующего канала)

**DWORD aflag,** // [in] флаги и атрибуты

**HANDLE exten,** // [in] дополнительные атрибуты

**);**

// **Код возврата:** в случае успешного завершения функция

// возвращает дескриптор именованного канала, иначе

// **INVALID\_HANDLE\_VALUE** – неудачное завершение

// **Примечание:-** параметр **pname** указывается в локальном или

// сетевом формате: в зависимости от способа применения

// - параметр **accss** может принимать значения **GENERIC\_READ**

// (чтение), **GENERIC\_WRITE** (запись) или

// **GENERIC\_READ|** **GENERIC\_WRITE** (запись,чтение)

// - параметр **share** можетпринимать значения

// **FILE\_SHARE\_READ** (совместное чтение),

// **FILE\_SHARE\_WRITE** (совместная запись),

// **FILE\_SHARE\_READ| FILE\_SHARE\_WRITE** (чтение и запись);

// - параметр **sattr** для установки атрибутов безопасности

// по умолчанию, следует установить значение **NULL;**

// - значение параметра **oflag** всегдаустанавливаетсяв

// **OPEN\_EXISTING**(открытие существующего канала);

// - значение параметров **aflag** и **exten** можно установить в // **NULL**, что соответствует значениям по умолчанию

// **-- изменить характеристики канала**

// **Назначение:** функция предназначена для изменения

// динамических характеристик именованного канала

**BOOL SetNamedPipeHandleState**

**(**

**HANDLE hP,**  // [in] дескриптор именованного канала

**LPDWORD pst,** // [in] указатель на новое состояние канала

**LPDWORD pcc,** //[in] указатель на макс. к-во байтов

**LPDWORD pto** //[in] указатель на интервал задержки

**);**

// **Код возврата:** в случае успешного завершения функция

// возвращает **TRUE**, иначе **FALSE**

функция которая объединяет операции чтения и записи в одну операцию. Такую объединенную операцию называют ***транзакцией***  именованного канала. Функция TansactNamedPipe может быть использована только в том случае, если сервер именованного канала установил флаг PIPE\_TYPE\_MESSAGE.

Применение TransactNamedPipe целесообразно, если другая сторона канала может обеспечить достаточно быструю реакцию и оправить ответ на пришедшее сообщение.

// **-- писать и читать данные канала**

// **Назначение:** функция предназначена для выполнения записи в

// канал и чтения из канала за одну операцию

**BOOL TransactNamedPipe**

**(**

**HANDLE hP,**  // [in] дескриптор канала

**LPVOID pw,** // [in] указатель на буфер для записи

**DWORD sw,** //[in] размер буфера для записи

**LPVOID pr,** // [out] указатель на буфер для чтения

**DWORD sr,** //[in] размер буфера для чтения

**LPDWORD pr,** // [out] количество прочитанных байт

**LPOVERLAPPED ol** // [in,out] для асинхронного доступа

**);**

// **Код возврата:** в случае успешного завершения функция

// возвращает **TRUE**, иначе **FALSE**

// Примечание: параметр **ol** используется для асинхронного

// доступа к каналу, если асинхронный доступ не

// не предполагается, то следует указать **NULL**

// **-- выполнить одну транзакцию**

// **Назначение:** функция предназначена для установки связи с

// именованным каналом, выполнения одной транзакции

// и разрыва связи

**BOOL CallNamedPipe (одиночная транзакция)(не надо функции creatFile использовать)(сама устанавливает связБсама отправляет сообщБ сама разрываеь связ)(так же не обязательно использовать функцию закритие дикриптора)**

**(**

**LPCTSTR nP,**  // [in] указатель на имя канала

**LPVOID pw,** // [in] указатель на буфер для записи

**DWORD sw,** //[in] размер буфера для записи

**LPVOID pr,** // [out] указатель на буфер для чтения

**DWORD sr,** //[in] размер буфера для чтения

**LPDWORD pr,** // [out] количество прочитанных байт

**DWORD to** // [in] интервал ожидания

**);**

// **Код возврата:** в случае успешного завершения функция

// возвращает **TRUE**, иначе **FALSE**

// **Примечание:** параметр **to** устанавливает интервал времени в

// в миллисекундах; кроме того, здесь могут быть

// установлены те же значения, что и в функции

// **WaitNamedPipe**