

Интерфейс Windows API

МДК 01.01 «Системное программирование»

Понятие интерфейса Windows API

Интерфейс программирования (API - Application Program Interface) - средства для использования прикладными программами системных ресурсов ОС и реализуемых ею функций. Основное внимание - базовым системным службам, включая управление файловой системой, процессами и потоками, межпроцессное взаимодействие, сетевое программирование и синхронизацию.

Win32/Win64 API (или обобщенно Windows API) – набор функций и классов, которые используются для программирования приложений, работающих под управлением семейства 32- и 64-разрядных ОС компании Microsoft.

Вопросы перехода от платформы Win32 к развивающейся платформе Win64 обсуждаются по мере необходимости. Win64, поддерживаемый в качестве 64-разрядного интерфейса в некоторых версиях Windows Server и Windows XP, почти идентичен Win32.

Интерфейсы программирования (API) в различных ОС

Windows API является важнейшим фактором, который оказывает влияние на весь процесс разработки приложений, и во многих случаях вытесняет поддерживаемый операционными системами UNIX и Linux POSIX API, поскольку считается более предпочтительным или, по крайней мере, предоставляющим те же возможности для приложений, ориентированных на настольные и серверные системы. Поэтому многие опытные программисты заинтересованы в скорейшем изучении Windows API.

Платформенно независимый системный интерфейс для компьютерных сред - **POSIX** (Portable Operating System Interface for Computer Environments) – определяет минимальный набор системных вызовов для открытых ОС, базируясь на UNIX системах.

Интерфейс программирования приложений Windows API ориентирован язык программирования C и на процедурные языки программирования более высокого уровня (например, на C++) .

Сравнение POSIX и WinAPI

POSIX	назначение	Win32 API
fork	создать дочерний процесс, идентичный родительскому	CreateProcess (fork+execve)
waitpid	ожидать завершения дочернего проц.	WaitForSingleObject
execve	переместить образ памяти процесса	-
exit	завершить выполнение процесса	ExitProcess
open	открыть файл	CreateFile
close	закрыть файл	CloseHandle
read	чтение данных из файла в буфер	ReadFile
write	запись данных из буфера в файл	WriteFile
lseek	переместить указатель файла	SetFilePointer
stat	информация о состоянии файла	GetFileAttributesEx
mkdir	создать каталог	CreateDirectory
rmdir	удалить каталог	RemoveDirectory
link	создать новый элемент каталога, ссылающийся на другой	-
unlink	удалить элемент каталога	DeleteFile
mount	монтирование файловой системы	-
umount	демонтирование файловой системы	-
chdir	изменить рабочий каталог	SetCurrentDirectory
chmod	изменить биты защиты файла	-
kill	послать сигнал процессу	-
time	получить системное время	GetLocalTime

Принципы, лежащие в основе Win32 API

Win32 API – набор функций и классов, которые используются для программирования приложений, работающих под управлением ОС Windows.

Рассмотрим основные принципы Windows, которые определяют стиль и методику программирования.

- Системные ресурсы Windows представляются в виде **объектов ядра (kernel object)**, для идентификации и обращения к которым используются **дескрипторы (handles)**.
- Любые манипуляции с объектами ядра ОС осуществляются только с использованием WinAPI.
- К объектам относятся файлы, процессы, потоки, каналы межпроцессорного взаимодействия, события и др. Объекты имеют атрибуты защиты.

Принципы, лежащие в основе Win32 API

- Одни задачи могут решаться с помощью разных функций
- Имеются вспомогательные функции (convenience function), полученные объединением часто встречающихся последовательностей функциональных вызовов в одну функцию (например, функция *CopyFile*)
- Функции имеют множество параметров и флагов
- Имеются механизмы синхронизации и взаимодействия, обеспечивающие удовлетворение самых разнообразных запросов
- Базовой единицей выполнения является поток (thread).
- В одном процессе (process) могут выполняться один или несколько потоков

Принципы, лежащие в основе Win32 API

- Для функций используются длинные описательные имена:

`WaitForSingleObject`

`WaitForSingleObjectEx`

`WaitForMultipleObjects`

- Имена predetermined типов данных пишутся прописными буквами:

`BOOL`

`HANDLE`

`DWORD`

`LPTSTR`

`LPSECURITY_ATTRIBUTES`

- В именах переопределенных типов указателей операция * не используется (`LPTSTR = TCHAR *`)

Принципы, лежащие в основе Win32 API

- В имена некоторых символических констант входит компонент WIN32, хотя они используются также и в Win64 (н-р, WIN32_FIND_DATA)
- Имеется большое количество «устаревших» (16-разрядных) функций, которые могут быть ошибочно использованы (например, *OpenFile* - должна использоваться *CreateFile*)
- Ориентирован на язык программирования C, не использует классы, ссылки и механизм обработки исключений.

Категории Win32 API

Функционально Win32 API делятся на категории:

1. Base Services (базовые сервисы) – обеспечивают приложениям доступ к ресурсам компьютера.
2. Common Control Library (библиотека общих элементов управления) – содержит классы окон, которые часто используются в приложениях.
3. Graphics Device Interface (интерфейс графических устройств) – обеспечивает функции для вывода графики на экран, принтер и другие графические устройства.
4. Network Services (сетевые сервисы) – используются при работе компьютеров в сети.

Категории Win32 API

5. User Interface (интерфейс пользователя) – обеспечивает функции для взаимодействия пользователя с приложением, используя окна для ввода-вывода информации
6. Windows NT Access Control (управление доступом под Windows NT) – содержит функции, которые используются для защиты информации путем контроля и ограничения доступа к защищаемым объектам.
7. Windows Shell (оболочка Windows) – функции для работы с оболочкой ОС
8. Windows System Information (информация о системе Windows) – функции для конфигурации ОС.

Типы данных в Win32 API

CHAR	char
UCHAR	unsigned char
BYTE	unsigned char
SHORT	short
USHORT	unsigned short
WORD	unsigned short
INT	int
BOOL	int
UINT	unsigned int
LONG	long
ULONG	unsigned long
DWORD	unsigned long
FLOAT	float

FALSE
TRUE

Объекты и их дескрипторы

Объектом в Win32 API называется структура данных, которая представляет системный ресурс (н-р, файл, канал, графический рисунок)

Категории объектов:

- User (объекты интерфейса пользователя) (н-р, окна, курсоры и пр.)
- Graphics Device Interface (объекты интерфейса графических устройств) (н-р, кисти, перья)
- Kernel (объекты ядра ОС) (н-р, файлы и каналы)

Под **доступом к объекту** понимается возможность приложения выполнить над объектом некоторые функции.

Приложение не имеет прямого доступа к объектам, а обращается к ним косвенно – каждому объекту ставится в соответствие **дескриптор**.

Объекты и их дескрипторы

Дескриптор объекта (handle) представляет собой запись в таблице, которая поддерживается системой и содержит адрес объекта и средства для идентификации типа объекта.

- Дескриптор имеет тип HANDLE.
- Дескриптор создается операционной системой и возвращается функциям Win32 API, которые создают объекты (CreateObject)
- После завершения работы с объектом его дескриптор нужно закрыть функцией CloseHandle.

Вопросы для повторения:

1. Средства для использования прикладными программами системных ресурсов ОС и реализуемых ею функций. называется
2. Интерфес программирования – это
3. Интерфейсы Win32/Win64 API обобщенно называют
4. Набором функций и классов, которые используются для программирования приложений, работающих под управлением семейства 32- и 64-разрядных ОС компании Microsoft. Является
5. Платформенно независимый системный интерфейс для компьютерных сред называется
6. POSIX (Portable Operating System Interface for Computer Environments) определяет минимальный набор системных вызовов для открытых ОС, базируясь на системах
7. Системные ресурсы Windows представляются в виде
8. Для идентификации и обращения к системным ресурсам (объектам ядра) используются
9. Любые манипуляции с объектами ядра ОС осуществляются только с использованием
10. К объектам Windows относятся
11. Базовой единицей выполнения функций процесса программирования является
12. В одном процессе (process) могут выполняться
13. Для функций используются длинные описательные имена, например:
14. Имена предопределенных типов данных пишутся прописными буквами, например:
15. Имена символических констант иногда составляют с использованием компонента WIN32, например:
16. Объектом в Win32 API называется
17. Объекты в Win32 API подразделяют на следующие категории:
18. К объектам категории User (объекты интерфейса пользователя) относятся:
19. К объектам категории Graphics Device Interface (объекты интерфейса графических устройств) относятся:
20. К объектам категории Kernel (объекты ядра ОС) относятся:
21. Возможность приложения выполнить над объектом некоторые функции в Win32 API называется
22. Каждому объекту в Win32 API ставится в соответствие
23. Запись в таблице, которая поддерживается системой и содержит адрес объекта и средства для идентификации типа объекта называется

Спасибо за внимание!