

ЛЕКЦІЯ 4

**Реєстр Windows та функції API для роботи з ним.
Засоби роботи з реєстром в сучасних системах
програмування.**

**Операційні системи
доц. Сінельнікова Т.Ф.**

РЕЄСТР WINDOWS

У Microsoft Windows настройка здійснюється через реєстр - механізм, який спрощує зберігання параметрів додатків і службових програм.

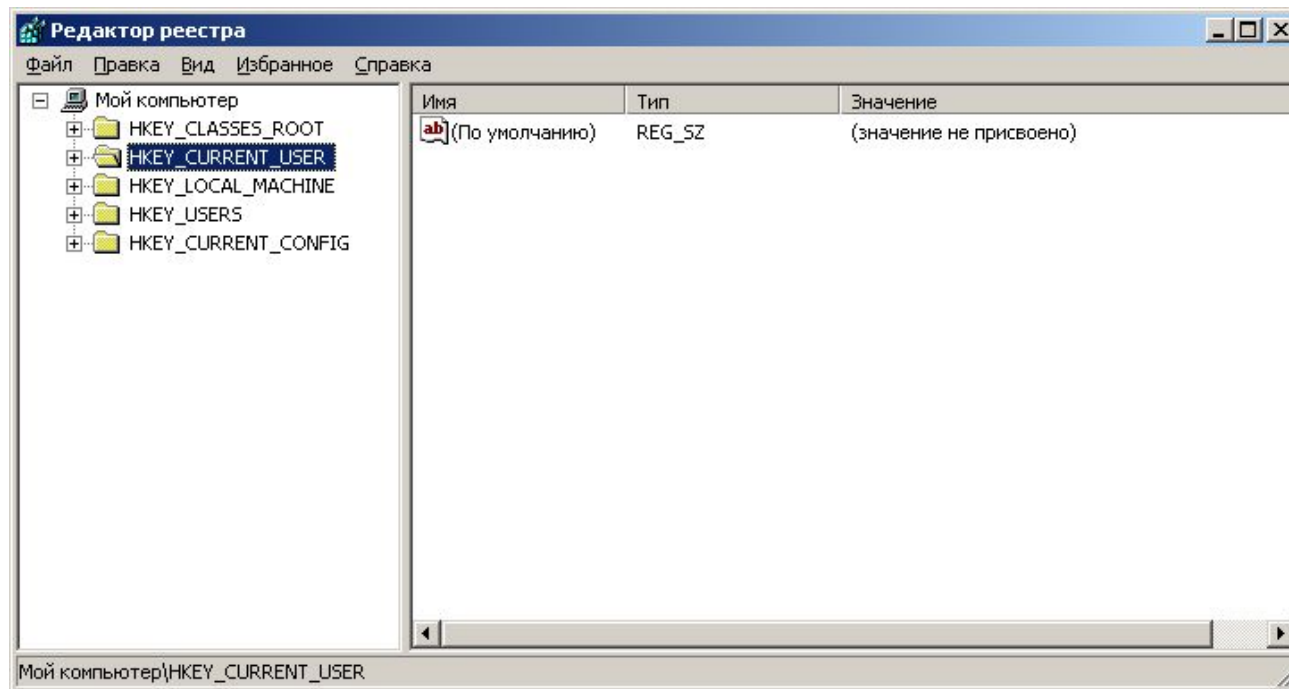
Реєстр - це централізована ієрархічна база даних, що складається з декількох файлів (для програмістів, що працюють з ним, він подається як єдина база даних) і володіє наступними можливостями:

- Ієрархічна організація даних програми дозволяє зберігати свої параметри довільним чином;
- Підтримує багатокористувацьку роботу, він розділений на дві частини, в одній з яких зберігаються параметри локального комп'ютера (HKEY_LOCAL_MACHINE), а в іншій - користувальницькі параметри (HKEY_USERS);
- Обмеження доступу до будь-яких розділів реєстру;
- Зберігання даних різних типів, у тому числі довічних DWORD, рядків і мультірядків.

Реєстр - обмежений спільний системний ресурс, у ризанні з яким потрібно дотримуватися певних правил етикету (або угод), багато з них накладаються самою системою.

РЕЄСТР WINDOWS

В основі системного реєстру лежить ієрархія розділів (ключів) і значень. Розділи (ключі реєстру) можуть складатися з довільного числа підрозділів (підключей) значень. У свою чергу підрозділи володіють такими ж правами і можуть включати в себе скільки завгодно значень і нових підрозділів. назви розділів одного рівня повинні розрізнятися, крім того, в них не повинна входити зворотна коса риска. Для перегляду реєстру можна використовувати стандартну програму regedit.exe.



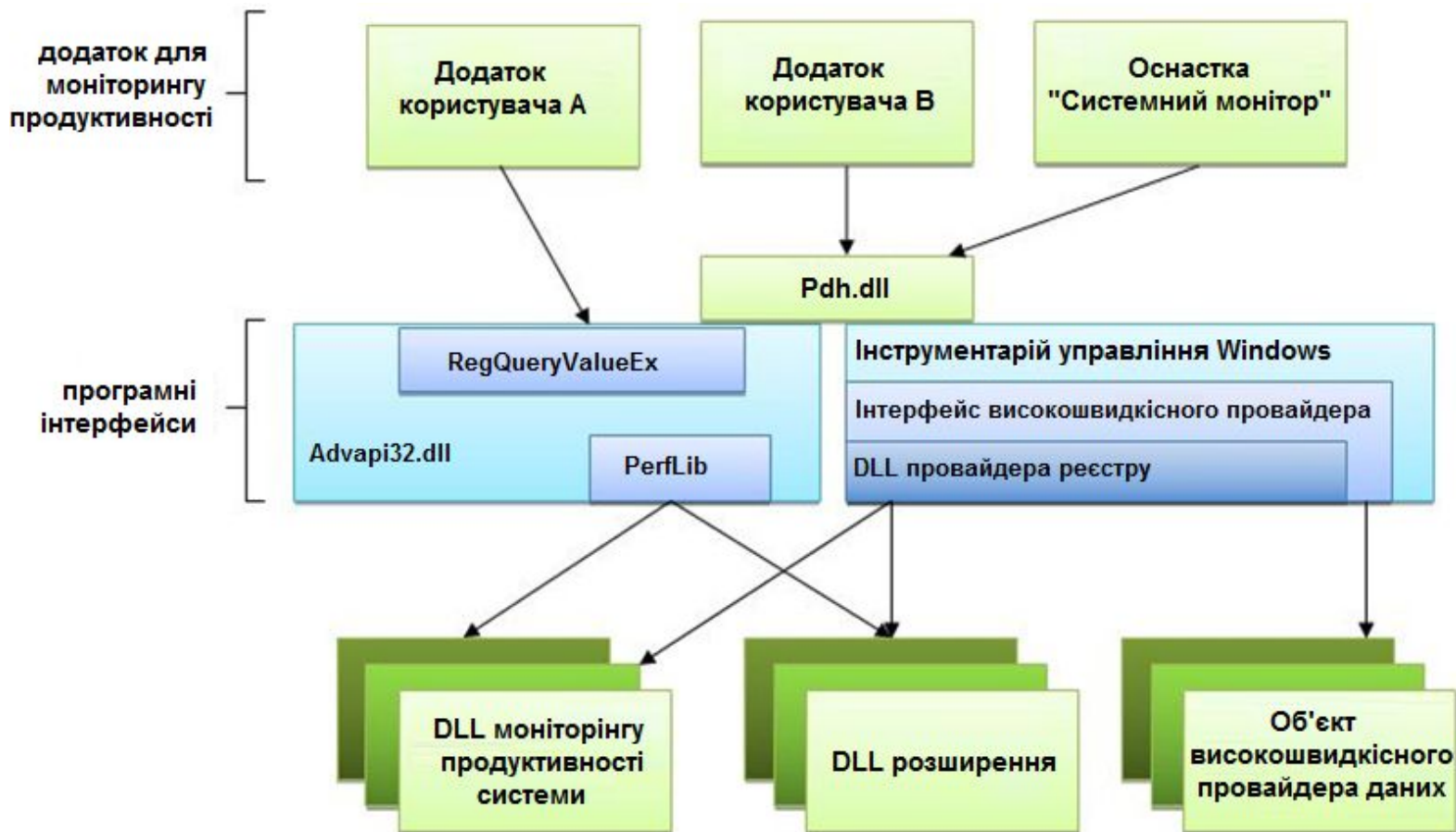
РОЗДІЛИ СИСТЕМНОГО РЕЄСТРУ WINDOWS 2000, ЩО ВИЗНАЧЕНІ ЗАЗДАЛЕГІДЬ

Назва розділу	Опис
HKEY_LOCAL_MACHINE	Містить інформацію конкретного комп'ютера, в тому числі про встановлене апаратне та програмне забезпечення, а також налаштування. Ці дані є загальними для всіх користувачів, що працюють за даними ПК.
HKEY_USERS	Набір даних, що застосовуються при роботі певного користувача. Цей розділ містить інформацію для користувача за замовчуванням і для користувача із завантаженим в даний момент профілем.
HKEY_CURRENT_USER	Визначена системою закладка, або псевдонім, для розділу HKEY_USERS, динамічно вказує на інформацію реєстру про користувача, пов'язаного з потоком, що визивається. Існують спеціальні правила, пов'язані з користувачами потоків, що викликаються, що впливають на роботу служб пов'язаних з цими завданнями.

РОЗДІЛИ СИСТЕМНОГО РЕЄСТРУ WINDOWS 2000, ЩО ВИЗНАЧЕНІ ЗАЗДАЛЕГІДЬ

Назва розділу	Опис
HKEY_CLASSES_ROOT	Включає всі типи відповідностей файлів, а також інформацію про OLE та ярлики і містить відомості про зареєстровані COM-компоненти і пов'язані з оболонкою (shell) асоціації.
HKEY_CURRENT_CONFIG	Визначена системою закладка, або псевдонім, для HKEY_LOCAL_MACHINE \ System \ CurrentControlSet \ Hardware Profiles \ Current назва якого відповідає імені користувача, що працює в даний момент. Тут зберігаються відомості про конфігурацію обладнання.
HKEY_PERFORMANCE_DATA	Містить дані про продуктивність, передані йому системою, службами та додатками.

АРХІТЕКТУРА ДОСТУПУ ДО ЛІЧИЛЬНИКІВ ПРОДУКТИВНОСТІ(РОЗДІЛ NKEY_PERFORMANCE_DATA)



ТИПИ ДАНИХ ЗНАЧЕНЬ РЕЄСТРУ(ПОВНИЙ СПИСОК ТИПІВ ДАНИХ, ЯКІ ПІДТРИМУЄ WINDOWS 2000)

Типи даних значень реєстру	Опис
REG_BINARY	Бінарні дані в будь-якій формі.
REG_BINARY	32-бітове число (часто використовується для зберігання байта, слова чи навіть біта).
REG_DWORD_LITTLE_ENDIAN	32-бітове число в форматі апаратних платформ Intel (з прямим порядком байтів починаючи з молодшого). Всі Windows-системи зберігають числа в цьому форматі.
REG_DWORD_BIG_ENDIAN	32-бітове число в форматі апаратних платформ Motorola (із зворотним порядком байтів починаючи зі старшого). Цей формат використовується для зберігання цілих чисел в деяких не-Windows системах, наприклад на деяких UNIX-комп'ютерах і в процесорах Motorola.
REG_QWORD	64-розрядне число.
REG_QWORD_LITTLE_ENDIAN	64-розрядне число з прямою послідовністю байтів, еквівалентно REG_QWORD.

ТИПИ ДАНИХ ЗНАЧЕНЬ РЕЄСТРУ(ПОВНИЙ СПИСОК ТИПІВ ДАНИХ, ЯКІ ПІДТРИМУЄ WINDOWS

Типи даних значень реєстру	Опис
REG_EXPAND_SZ	Рядок, що завершується нулем та містить посилання на змінні оточення (наприклад: "% PATH%"). Цей тип даних часто застосовується для зберігання шляхів до файлів, оскільки такі змінні, як «% SystemRoot%», дозволяють уникнути помилок, якщо користувач змінив шлях до різних системних файлів.
REG_LINK	Символічна ссылка в форматі UNICODE. Програми або серверний код не повинні звертатися або зберігати значення цього типу.
REG_MULTI_SZ	Масив рядків, кожен з яких завершується нулем. Наприкінці масиву розташовуються два нульових символа.
REG_NONE	Невизначений тип, функціонально співпадає з REG_BINARY.
REG_RESOURCE_LIST	Список ресурсів драйвера пристрою. Не застосовується в додатках режиму користувача.
REG_SZ	Рядок, що закінчується нулем.

УГОДА ПО РОБОТІ З РЕЄСТРОМ

Параметри реєстру, специфічні для комп'ютера. Існує угода, відповідно до якої програми та службові програми зберігають дані, специфічні для конфігурації виконуючого їх комп'ютера, в такій ієрархії розділів:

HKEY_LOCAL_MACHINE Software

Company Name

Your Product Name

Your Product Version (optional)

Key1

Value1

Value2

Key2

УГОДА ПО РОБОТІ З РЕЄСТРОМ

Параметри реєстру, специфічні для користувача. Підрозділи розділу `HKEY_USERS` містять відомості про параметри користувача. Система автоматично встановлює відповідність між `HKEY_CURRENT_USER` і підрозділом, що описує поточний, тобто асоційованого з поточним процесом, користувача. Якщо потік запозичує права іншого користувача, всі звернення цього потоку до `HKEY_CURRENT_USER` будуть ставитися до цього користувача. Якщо програма буде звертатися до специфічної для користувача інформації реєстру, вона має дотримуватися угоди, схожої на вживане в разі для розділу **`HKEY_LOCAL_MACHINE`**:

`HKEY_CURRENT_USER`
 `Software`
 `Your Company Name`
 `Your Product Name`
 `Your Product Version (optional)`
 `Key1`
 `Value1`
 `Value2`
 `Key2`

УГОДА ПО РОБОТІ З РЕЄСТРОМ

Дескриптор ключа (підрозділу) (key handle) зберігається у змінній типу HKEY. Щоб отримати дескриптор ключа HKEY, необхідно використовувати функції RegOpenKeyEx або RegCreateKeyEx. Перший аргумент цих функцій має тип HKEY. При виклику будь-якої з цих функцій в якості першого аргументу можна вказати константу, наприклад ім'я одного з кореневих ключів реєстру (наприклад, HKEY_CLASSES_ROOT). Windows дозволяє скористатися декількома константами типу HKEY, які за умовчанням є відкритими. Цими константами є:

HKEY_CLASSES_ROOT

HKEY_CURRENT_USER

HKEY_LOCAL_MACHINE

HKEY_USERS

HKEY_CURRENT_CONFIG

HKEY_DYN_DATA

Ці константи відповідають кореневим ключам реєстру.

ДЕЯКІ(ЧАСТО ВИКОРИСТОВУВАНІ) ФУНКЦІЇ АРІ ДЛЯ РОБОТИ З РЕЄСТРОМ

Функція	Опис
RegCloseKey	Закрити відкритий ключ реєстру
RegConnectRegistry	Підключитися до віддаленого реєстру
RegCreateKeyEx	Створити новий підключ
RegDeleteKey	Видалити ключ
RegDeleteValue	Видалити значення
RegEnumKeyEx	Перейти до наступного підключу (кожен раз повертає новий ключ)
RegEnumValue	Перейти до наступного значення (кожен раз повертає нове значення)
RegFlushKey	Внести значення ключа до реєстру (щоб переконатися в тому, що всі зміни збережені на диску)

ДЕЯКІ(ЧАСТО ВИКОРИСТОВУВАНІ) ФУНКЦІЇ АРІ ДЛЯ РОБОТИ З РЕЄСТРОМ

Функція	Опис
RegLoadKey	Завантажити ключ із спеціального файлу (див. також RegSaveKey)
RegOpenKeyEx	Відкрити ключ
RegQueryInfoKey	Запросити інформацію про ключ
RegQueryValueEx	Прочитати значення
RegReplaceKey	Замінити ключ після перезапуску системи
RegSaveKey	Записати ключ в файл
RegSetKeySecurity	Встановити дозвіл на доступ до ключа
RegSetValueEx	Встановити значення
RegUnloadKey	Вивантажити набір значень

ПРИКЛАД РОБОТИ З РЕЄСТРОМ

Приклад роботи з реєстром виконує перелік вмісту ключа реєстру HKEY_CURRENT_USER \ Software. Робота з реєстром реалізована з використанням виключно API функцій, але в цілому дану функцію можна буде застосувати тільки в додатках ATL або MFC, в зв'язку з використанням класу CString. Функція переліку вмісту ключа реєстру HKEY_CURRENT_USER \ Software

```
void RegReadSample (CString &strResult)
{
    TCHAR    szBuff[MAX_PATH];
    DWORD    dwBSize = MAX_PATH, dwIndex = 0;
    HKEY hKey = 0;
    LONG lResult = 0;
    FILETIME ft;
    lResult = ::RegOpenKeyEx(HKEY_CURRENT_USER, _T("software"), 0,
KEY_ENUMERATE_SUB_KEYS, &hKey);
    if(lResult==ERROR_SUCCESS)
    {
        while(lResult==ERROR_SUCCESS)
        {
            lResult::RegEnumKeyEx(hKey, dwIndex, szBuff, &dwBSize, 0, 0, &ft);
            if(lResult==ERROR_SUCCESS)
            {
                dwIndex++; dwBSize = MAX_PATH;
                strResult += (CString(szBuff)+_T("\r\n"));
            }
        }
    }
    UpdateData(0);
}
```

РОБОТА З РЕЄСТРОМ

Зауваження. Дуже важливо при роботі з реєстром пам'ятати про особливості функції `RegDeleteKey`. При виклику цієї функції необхідно обов'язково перевіряти параметр, що передається в цю функцію, на наявність в ньому інформації. Цей параметр не повинен бути порожнім, інакше операційна система може бути дуже серйозно пошкоджена, особливо, якщо це Windows 98.

У разі успішного виконання функцій `RegOpenKeyEx`, `RegEnumKeyEx`, повертається значення `ERROR_SUCCESS`. При виконанні функції `RegOpenKeyEx` використан параметр `KEY_ENUMERATE_SUB_KEYS`, який вказує тип маски доступу до ключів реєстру. Вказане значення не єдине. Всі можливі значення цього параметра наведені в таблиці.

РОБОТА З РЕЄСТРОМ

При роботі з реєстром необхідно пам'ятати, що доступ до реєстру в цілому і до його ключів зокрема регулюється правами користувача, профіль якого активний в даний момент на комп'ютері. Це стосується операційних систем сімейства Windows NT, а в ОС Windows 98/Me ці обмеження не діють. Облік прав користувача вкрай важливий, оскільки програми, що працюють з реєстром можуть не мати доступу до тих чи інших ключів в залежності від прав користувача. Повний доступ до реєстру мають програми, що запускаються в режимі адміністратора або користувачів з правами адміністратора.

ЗНАЧЕННЯ ТИПІВ ДОСТУПУ ДО КЛЮЧІВ РЕЄСТРУ

Значення параметра	Опис
KEY_ALL_ACCESS	Комбінація значень всіх масок безпеки для доступу: KEY_QUERY_VALUE, KEY_ENUMERATE_SUB_KEYS, KEY_NOTIFY, KEY_CREATE_SUB_KEY, KEY_CREATE_LINK, і KEY_SET_VALUE
KEY_CREATE_LINK	Дозвіл створення символічного посилання
KEY_CREATE_SUB_KEY	Дозвіл створення підключа
KEY_ENUMERATE_SUB_KEYS	Дозвіл переліку(перебору) підключей
KEY_EXECUTE	Дозвіл для читання

ЗНАЧЕННЯ ТИПІВ ДОСТУПУ ДО КЛЮЧІВ РЕЄСТРУ

Значення параметра	Опис
KEY_NOTIFY	Дозвіл для повідомлення про зміну
KEY_QUERY_VALUE	Дозвіл для запиту значення підключа
KEY_READ	Комбінація масок безпеки: KEY_QUERY_VALUE, KEY_ENUMERATE_SUB_KEYS и KEY_NOTIFY
KEY_SET_VALUE	Дозвіл встановлення значення підключа
KEY_WRITE	Комбінація масок доступу: KEY_SET_VALUE и KEY_CREATE_SUB_KEY

МОДИФІКАЦІЯ РЕЄСТРУ З ВИКОРИСТАННЯМ REG ФАЙЛА

Для модифікації (додавання) великої кількості даних в реєстр можна використовувати REG-файли. Це файли, про які згадувалося при описі функції LoadKey. **REG-файли** - це прості текстові ASCII-файли з інформацією про вміст реєстру. REG-файл можна створити з використанням програми REGEDIT за допомогою пункту меню Export цієї програми. Також REG-файл можна створити самостійно, за допомогою будь-якого текстового редактора. Приклад такого файлу наводиться нижче:

```
Windows Registry Editor Version 5.00
[HKEY_CURRENT_USER\Software\MyProgram\1.0]
[HKEY_CURRENT_USER\Software\MyProgram\1.0\Current]
"Parametr1"=dword:00000001
[HKEY_CURRENT_USER\Software\MyProgram\1.0\Current\Start]
"ProgramStart"=hex:00,00,00,00
"DefaultPath"="C:\\Program Files\\MyProgram"
@"C:\\Program Files\\MyProgram\\Database"
```

МОДИФІКАЦІЯ РЕЄСТРУ З ВИКОРИСТАННЯМ REG ФАЙЛА

Такий REG-файл можна використовувати для внесення інформації до реєстру за допомогою функції LoadKey або шляхом запуску цього файлу з програми Windows Explorer.

У першому рядку файлу міститься рядок, який вказує на версію програми REGEDIT, за допомогою якої був створений даний файл. При створенні власного REG-файлу за допомогою текстового редактора, необхідно помістити це ключове слово на початку файлу. Структура файлу наступна. Після першого рядка розміщується ім'я ключа в квадратних дужках, потім можна розмістити одну або кілька пар ім'я / значення. Знак @ позначає значення ключа за замовчуванням. Імена та значення повинні бути поміщені в лапки за винятком символу «?», Який відповідає значенню за замовчуванням. Замість символу «\» слід використовувати «\ \» (два поспіль символу). В кінці файлу слід додати порожній рядок, інакше REGEDIT може проігнорувати останній рядок файлу.