

Registre

Registre sú mechanizmus umožňujúci aplikácii ľahšie spravovanie konfiguračných nastavení. V podstate sa jedná o centrálnu uloženú databázu s hierarchickou stromovou štruktúrou, ktorá ponúka nasledujúce možnosti:

- Napriek tomu, že registre sú zložené z niekoľkých fyzických súborov, zaobchádza sa s nimi ako s jednoduchou databázou na ukladanie a získavanie informácií.
- Nakoľko sa jedná o hierarchickú štruktúru, aplikácia môže nanútiť konfiguračným nastaveniam svoju vlastnú organizáciu.
- Poskytuje podporu pre viacerých užívateľov a sú rozdelené na dve časti: časť pre nastavenia lokálnej stanice (HKEY_LOCAL_MACHINE) a časť pre nastavenia užívateľa (HKEY_USERS).
- Na špecifické kľúče je možné aplikovať bezpečnostné povolenia a kontrolné nastavenia.
- Rôzne dátové typy: binary, DWORD, string, multistring, expandable string.

Štruktúra systémových registrov

Registre sú obmedzeným, zdieľaným systémovým zdrojom. Jedná sa o hierarchiu kľúčov a hodnôt. Kľúče môžu obsahovať ľubovoľný počet podkľúčov a hodnôt. Takisto každý podkľúč môže obsahovať ďalšie podkľúče a hodnoty. Mená kľúčov na tej istej úrovni musia byť unikátne a nesmú obsahovať znak *backslash* (\). Rovnako aj hodnoty pod konkrétnym podkľúčom musia mať unikátne mená – okrem hodnoty (*Default*), ktorá je nepomenovaná.

Štruktúra registrov môže byť logická alebo fyzická. Programátori sa bežne zaoberajú iba logickou štruktúrou. Registre logicky obsahujú 5 koreňov:

- **HKEY_LOCAL_MACHINE** – množina dát registru aplikovaná na lokálnu stanicu, nezávisle na prihlásenom užívateľovi.
- **HKEY_USERS** – množina dát registra aplikovaná na konkrétneho užívateľa. Obsahuje stromy pre defaultného užívateľa a pre aktuálne načítané užívateľské profily.
- **HKEY_CURRENT_USERS** – systémom definovaný alias pre podstrom stromu HKEY_USERS, ktorý dynamicky ukazuje na informácie registra pre užívateľa asociovaného s volajúcim vláknom.
- **HKEY_CLASSES_ROOT** – systémom definovaný alias pre HKEY_LOCAL_MACHINE\Software\Classes. Tento strom obsahuje konfiguračné informácie pre registrované COM komponenty.
- **HKEY_CURRENT_CONFIG** – systémom definovaný alias pre HKEY_LOCAL_MACHINE\System\CurrentControlSet\Hardware Profiles\Current. Tento strom registra uchováva konfiguračné informácie hardwaru.

Fyzicky sú registre udržiavané niekoľkými súbormi, tzv. *hives*, na pevnom disku užívateľa. Programátor musí myslieť iba na cestu ku kľúču v registri. Systém potom rozhodne, ktorý *hive* obsahuje tento kľúč a sprístupní náležitý súbor.

Zaužívané konvencie pri práci s registrami

Skôr, než začnete pracovať s registrami, musíte sa zamyslieť nad tým, či sa majú vaše nastavenia aplikovať na pracovnú stanicu (alebo pre všetkých užívateľov), alebo pre konkrétneho užívateľa. Typicky sa informácie v registroch pri inštalovaní aplikácie aplikujú na všetkých užívateľov systému. Na druhej strane, nastavenia užívateľa a zmeny v konfigurácii počas behu aplikácie sú bežne uložené pre špecifického užívateľa.

Po rozhodnutí, na koho sa majú aplikovať konfiguračné nastavenia, sa adekvátne k tomu zvolí koreň registra a vytvorí sa hierarchia pre váš produkt. Napríklad pri nastaveniach aplikujúcich sa na pracovnú stanicu by mal mať register nasledujúcu podobu:

```
HKEY_LOCAL_MACHINE
  Software
    Názov vašej spoločnosti
      Názov vášho produktu
        Verzia vášho produktu (voliteľné)
          Kľúč1
            Hodnota1
            Hodnota2
            :
            :
          Kľúč2
            :
            :
```

Práca s kľúčami registra

Existujú dva typy funkcií pre prácu s registrami – funkcie registrov (*Reg**) a funkcie shellu (*SH**). Nakoľko shell funkcie sú postavené na funkciách registrov, budeme sa venovať *Reg** funkciám.

Otváranie kľúčov registra

Skôr než začnete ukladať a získavať informácie z hodnôt registrov, musíte získať *handle* ku kľúču. *Handle* získate buď otvorením existujúceho kľúča, alebo vytvorením nového kľúča. Existujúci kľúč vytvoríte volaním funkcie *RegOpenKeyEx*:

```
LONG RegOpenKeyEx(
    HKEY hkeyRoot,
    PCTSTR subKey,
    DWORD options,
    REGSAM samDesired,
    PHKEY resultKey
);
```

Pre otvorenie kľúča odovzdajte funkcii *handle* na koreňový kľúč, ako *hkeyRoot* parameter. Táto *handle* by mala byť buď rodič podkľúča, ktorý chcete otvoriť, alebo jeden z jeho predchodcov.

Parameter *subKey* je textové meno kľúča, ktorý chcete otvoriť. Toto meno musí obsahovať všetky kľúče medzi požadovaným kľúčom a koreňom, oddelené dvojitém opačným lomítkom (\\). Ak bude tento parameter NULL alebo prázdny reťazec, systém vráti *handle* ku kľúču koreňa.

Parameter *options* je rezervovaný pre budúce využitie. Parameter *samDesired* popisuje, čo chce vaša aplikácia s kľúčom robiť. Ako hodnotu *resultKey* parametra by ste mali odovzdať adresu premennej typu HKEY. Ak funkcia uspeje, systém do tejto premennej uloží *handle* otvoreného kľúča.

Vytváranie kľúčov registra

Vytvárať kľúče registra vám umožní funkcia *RegCreateKeyEx*. Túto funkciu je možné použiť aj na otvorenie kľúča. Ak daný kľúč neexistuje, pokúsi sa ho vytvoriť. Funkcia vyzerať nasledovne:

```
LONG RegCreateKeyEx(  
    HKEY hkeyRoot,  
    PCTSTR subKey,  
    DWORD reserved,  
    PTSTR pClass,  
    DWORD options,  
    REGSAM samDesired,  
    PSECURITY_ATTRIBUTES psa,  
    PHKEY resultKey,  
    PDWORD disposition  
);
```

Parametre *hkeyRoot*, *subKey*, *samDesired* a *resultKey* majú význam ako pri funkcií *RegOpenKeyEx*. Parameter *reserved* je rezervovaný a mal by byť 0, parameter *pClass* by mal byť vždy NULL. Parameter *psa* ukazuje na štruktúru SECURITY_ATTRIBUTES, ktorá umožňuje definovať prístupové práva pre novovytvorený podkľúč. Ak tento podkľúč už existuje, parameter je ignorovaný.

Parameter *options* môže nadobudnúť jednu z nasledujúcich hodnôt:

- REG_OPTION_NON_VOLATILE – Kľúč je vytvorený v registroch a je trvale uložený na pevný disk.
- REG_OPTION_VOLATILE – Kľúč je vytvorený v pamäti a je odstránený pri reštartovaní systému.
- REG_OPTION_BACKUP_RESTORE – Kľúč je používaný zálohovacím softwarom na zálohovanie a obnovenie registrov.

Parameter *disposition* môže nadobúdať dve rôzne hodnoty: REG_CREATED_NEW_KEY a REG_OPENED_EXISTING_KEY. Skontrolovanie tejto premennej po skončení funkcie odhalí, či bol vytvorený nový kľúč, alebo bol otvorený už existujúci kľúč. Pokiaľ to nepotrebujete zisťovať, zvolte tento parameter NULL.

Či už obdržíte *handle* ku kľúču registra volaním funkcie *RegOpenKeyEx* alebo *RegCreateKeyEx*, po skončení jej používania by ste ju vždy mali zatvoriť volaním funkcie *RegCloseKey*:

```
LONG RegCloseKey(HKEY hkey);
```

Prechádzanie kľúčov registra

Mená kľúčov registra sú pri inštalácii alebo konfigurácii softwaru väčšinou známe. Môže však nastať prípad, kedy mená podkľúča v otvorenom kľúči nepoznáte. V takomto prípade je na získanie týchto mien vhodné použiť funkciu *RegEnumKeyEx*:

```
LONG RegEnumKeyEx(  
    HKEY hKey,  
    DWORD index,  
    PTSTR name,  
    LPDWORD nameSize,
```

```
PDWORD reserved,  
PTSTR classBuffer,  
PDWORD classBufferSize,  
PFILETIME lastWriteTime  
);
```

Buffer pre vrátené mená kľúčov a tried musia mať dostatočnú veľkosť, inak funkcia zlyhá. Preto by ste pre získanie dĺžky najdlhšieho mena podkľúča a triedy pod žiadaným kľúčom mali najskôr volať funkciu [RegQueryInfoKey](#). Váš kód by však aj potom mal zvládnuť situáciu, že sa vyskytne podkľúč s dlhším názvom, keďže môže byť dodatočne vytvorený.

Práca s hodnotami registra

V hodnotách registra sú uložené aktuálne konfiguračné dáta vašej aplikácie. V okamihu, keď máte *handle* k otvorenému kľúču registra, ste pripravení získať dáta obsiahnuté v tomto kľúči. K tomuto účelu existujú 3 funkcie: [RegQueryValueEx](#), [RegEnumValue](#) a [RegQueryMultipleValues](#).

Funkciu [RegEnumValue](#) by ste mali použiť, keď nepoznáte meno hodnoty, z ktorej chcete získať dáta, prípadne chcete systematicky získať dáta pre všetky hodnoty v danom kľúči. Funkcia [RegQueryMultipleValues](#) získa dáta zo všetkých hodnôt obsiahnutých v zozname, ktorý je jedným z parametrov tejto funkcie. Pre získanie dát z konkrétnej hodnoty kľúča, ktorej meno dopredu poznáte, je vhodné použiť funkciu [RegQueryValueEx](#).

Typicky pokiaľ aplikácia chce získať konkrétnu hodnotu kľúča registra, volá dvakrát funkciu [RegQueryValueEx](#). Iniciálne volanie tejto funkcie získa požadovanú veľkosť bufferu a po alokovaní samotného bufferu sa ďalším volaním získajú žiadané dáta.

Existuje však aj iný prístup, ktorý môže byť v závislosti na počte hodnôt efektívnejší: zavolajte funkciu [RegQueryInfoKey](#), ktorá vráti veľkosť najväčšej hodnoty obsiahnutej v danom kľúči. Potom už stačí alokovať jediný buffer a použiť ho pre viacero volaní funkcie [RegQueryValueEx](#).

Ukladanie dát do registra

Pre uloženie dát do registra volajte funkciu [RegSetValueEx](#):

```
LONG RegSetValueEx(  
    HKEY hKey,  
    PCTSTR valueName,  
    DWORD reserved,  
    DWORD valueType,  
    const BYTE *pData,  
    DWORD dataSize  
);
```

Parameter *hKey* odkazuje na kľúč, pod ktorým má byť hodnota s menom *valueName* uložená. Pokiaľ je hodnota *valueName* NULL alebo prázdny reťazec, systém uloží vaše dáta do hodnoty (*Default*). Parameter *valueType* určuje typ dát. Hodnoty, ktoré môže nadobudnúť, nájdete [tu](#). Parameter *pData* ukazuje na aktuálne dáta, ktoré chcete do registra uložiť a parameter *dataSize* určuje veľkosť dát v bajtoch. Ak funkcia uspeje, návratová hodnota je `ERROR_SUCCESS`.

Pokiaľ potrebujete, aby vaša aplikácia sledovala nejakú zmenu v registroch, je vhodné použiť funkciu [RegNotifyChangeKeyValue](#):

```
LONG RegNotifyChangeKeyValue(  
    HKEY hKey,  
    BOOL watchSubtree,  
    DWORD notifyFilter,  
    HANDLE hEvent,  
    BOOL asynchronous  
);
```

Po zavolaní tejto funkcie sledujete *hKey* kľúč. Pokiaľ v ňom nastane nejaká zmena, bude signalizovaná udalosť špecifikovaná pomocou *hEvent handle*.

Použitá literatúra:

RICHTER, Jeffrey. CLARK, Jason D. *Programming Server-Side Applications for Microsoft Windows 2000*. 1. vyd. 2000. ISBN 0-7356-0753-2

Microsoft Developer Network, domovská www stránka, dostupná na URL <http://msdn.microsoft.com>