

Zarządzanie pamięcią: Pliki mapowane

Przemysław Kuligowski
11.01.2019



Czym są pliki mapowane

Plik mapowany jest odzwierciedleniem pliku zawartym w pamięci wirtualnej. Dostęp do pliku mapowanego mają różne procesy poprzez odczyt/zapis pamięci.

System tworzy obiekt, dzięki któremu proces może odwoływać się do pliku.

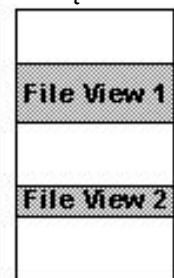
Zastosowania plików mapowanych

Dzięki mapowaniu plików, możliwy jest odczyt dużych plików po kawałku, bez konieczności wczytywania całej zawartości pliku do pamięci.

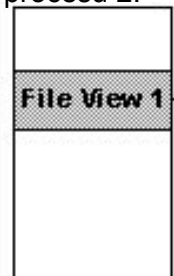
Możliwe jest także łatwe współdzielenie pamięci między procesami, gdyż jeden plik może być mapowany przez kilka procesów, które czytają i zapisują jego zawartość.

Współdzielenie plików mapowanych

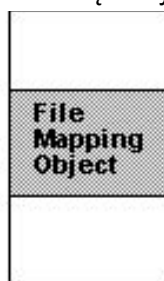
Pamięć wirtualna procesu 1.



Pamięć wirtualna procesu 2.



Pamięć fizyczna



Plik na dysku

File Mapping



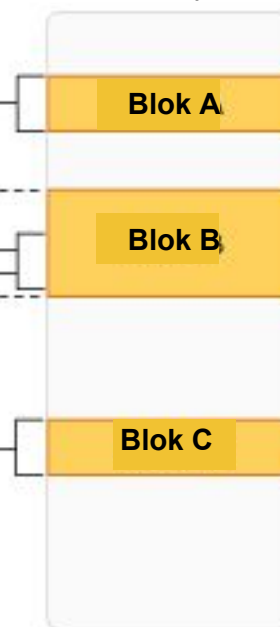
Proces A



Proces B



Plik mapowany w pamięci



Rys. 1, 2 Przykładowe widoki pliku mapowanego w pamięci

Widoki plików mapowanych

Dostęp do pliku jest możliwy poprzez widoki. Można utworzyć widok całości lub części pliku mapowanego oraz kilka widoków może odnosić się do tej samej części pliku.

Jeśli proces potrzebuje dostępu do innej części pliku, niż dostępna w obecnie utworzonym widoku, proces wymapowuje widok i mapuje nowy, odzwierciedlający żądaną zawartość.

Otwarcie pliku

```
HANDLE CreateFileA(  
    LPCSTR      lpFileName,  
    DWORD       dwDesiredAccess,  
    DWORD       dwShareMode,  
    LPSECURITY_ATTRIBUTES lpSecurityAttributes,  
    DWORD       dwCreationDisposition,  
    DWORD       dwFlagsAndAttributes,  
    HANDLE      hTemplateFile  
);
```

Tworzenie obiektu mapującego plik

```
HANDLE CreateFileMappingA(  
    HANDLE            hFile,  
    LPSECURITY_ATTRIBUTES lpFileMappingAttributes,  
    DWORD             flProtect,  
    DWORD             dwMaximumSizeHigh,  
    DWORD             dwMaximumSizeLow,  
    LPCSTR            lpName  
);
```

Otwieranie istniejącego obiektu mapującego

```
HANDLE OpenFileMappingA(  
    DWORD dwDesiredAccess,  
    BOOL bInheritHandle,  
    LPCSTR lpName  
);
```

Tworzenie widoku

```
LPVOID WINAPI MapViewOfFile(  
    _In_ HANDLE hFileMappingObject,  
    _In_ DWORD dwDesiredAccess,  
    _In_ DWORD dwFileOffsetHigh,  
    _In_ DWORD dwFileOffsetLow,  
    _In_ SIZE_T dwNumberOfBytesToMap  
);
```

MapViewOfFileEx zawiera opcjonalny lpBaseAddress

Używanie widoku

Z uchwytu widoku można czytać treść zamapowanego do pamięci pliku.

Zapisując zawartość do uchwytu, zapisujemy do widoku. Po wymapowaniu pliku lub po wymuszeniu zrzutu na dysk, zmieniona zawartość fizycznie trafia do pliku.

Zamykanie widoku i pliku

`UnmapViewOfFile` - wymapowanie widoku pliku

`FlushViewOfFile` - wymuszenie zapisu (domyślnie zapis dopiero gdy wszystkie aplikacje zakończą pracę z plikiem)

`CloseHandle` - usunięcie uchwytu do obiektu mapującego

Oficjalne źródło wiedzy:

Pliki mapowane WinAPI (dokumentacja w języku angielskim):

<https://docs.microsoft.com/pl-pl/windows/desktop/Memory/file-mapping>

Artykuł o zarządzaniu plikami mapowanymi (w języku angielskim):

[https://docs.microsoft.com/en-us/previous-versions/ms810613\(v=msdn.10\)](https://docs.microsoft.com/en-us/previous-versions/ms810613(v=msdn.10))

Prezentacja i mój kod są dostępne pod adresem:

<http://fizyka.umk.pl/~286289/winapi/>