

Opis architektury systemu

Źródło obrazów

Źródłem obrazów mogą być dowolne urządzenia dostarczające gigapikselowych zdjęć. Dopuszczalne są wszystkie graficzne formaty plików, które umożliwiają konwersję do formatu TIFF, JPEG lub JPEG2000. Wymienione trzy formaty są bezpośrednio wspierane przez system podczas importu. Konwersja z innych formatów pozostaje w gestii operatora systemu.

Proces importu polega na przekształceniu pliku graficznego do formatu TIFF o tzw. piramidalnej strukturze. Organizacja takiego pliku przypomina zbiór przekrojów poprzecznych piramidy – zawiera on oryginalny obraz, obraz 2x mniejszy, obraz 4x mniejszy itd. aż do obrazu o minimalnej rozdzielczości. Każdy z obrazów (zwanymi również poziomami) składa się z odpowiedniej liczby fragmentów o stałych rozmiarach – w budowanym systemie jest to 256 x 256 pikseli.

Serwer plików

Architektura systemu przewiduje wydzielony serwer lub serwery plików, których zadaniem jest przechowywanie obrazów pozyskanych z urządzeń. W rzeczywistej realizacji byłaby to maszyna o dużej przestrzeni dyskowej zorganizowanej w macierz. Sugerowana jest organizacja RAID 1+0 lub inna zapewniająca dużą wydajność i redundantność. Dedykowany serwer plików jest opcjonalnym elementem architektury i w wersji demonstracyjnej systemu zostanie pominięty.

Serwer WWW

Serwer WWW jest jednym z najważniejszych elementów systemu. Działa na nim aplikacja wykonana w technologii ASP.NET, której zadaniem jest:

- a) generowanie dynamicznych stron WWW mechanizmu nawigacji po strukturze katalogów,
- b) dostarczanie statycznych stron oraz strony z aplikacją JavaScript do przeglądania obrazów,
- c) obsługa interfejsu Web Service do wczytywania i zapisywania adnotacji do obrazów,
- d) dostarczanie fragmentów obrazów (ang. tiles) do kontrolki Google Maps.

Ostatnie z wymienionych zadań realizowane jest w oparciu o bibliotekę LibTIFF, która umożliwia odczytywanie wybranych fragmentów obrazu z odpowiedniego poziomu bez wczytywania nadmiarowych informacji. Ponieważ biblioteka LibTIFF udostępnia interfejs w języku C, a pozostała część aplikacji wykonana jest w technologii .NET, konieczna jest warstwa pośrednicząca w korzystaniu z biblioteki. Jej rolę pełni kod napisany w języku C++/CLI. Technologia ta umożliwia pisanie klas widocznych w .NET, których implementacja bezpośrednio wywołuje funkcje z plików nagłówkowych C. Jest to najefektywniejsze rozwiązanie tego typu problemu, znacznie wygodniejsze od mechanizmu P/Invoke lub JNI w technologii Java.

Aplikacja kliencka

Klient wchodzi w interakcję z systemem przy pomocy dowolnej współczesnej przeglądarki internetowej. Korzystając ze stron generowanych przez serwer WWW może nawigować po drzewie katalogów serwera plików i wybrać obraz, który chciałbym obejrzeć. Strona dotycząca danego obrazu zawiera podstawowe informacje o pacjencie i badaniu oraz kontrolkę Google Maps (lub analogiczną), która pozwala przeglądać obraz i jego adnotacje. Adnotacje dodawane są do obrazu graficznie przy pomocy kontrolki, natomiast za ich ładowanie i wczytywanie odpowiada kod JavaScript komunikujący się z Web Service na serwerze.