

ColorBox2 Framework

Zadání

Cílem projektu *ColorBox2* je vytvoření kompletního programátorského frameworku pro snadný **vývoj webových klient-server aplikací**. Vysoká míra abstrakce, událostmi řízené programování, modulární a objektový návrh, snadno rozšiřitelná hierarchie tříd a další vlastnosti by měly minimalizovat programátorskou rutinu, neustále se opakující při vytváření webových aplikací běžnými prostředky, a oddělit logiku aplikace, její vzhled a implementační detaily potřebné k jejímu faktickému zprovoznění.

Motivací projektu je vytvoření programátorského prostředí, ve kterém bude v budoucnu napsán netriviální webový publikační a informační systém, který umožní nejen vystavování textových článků s obrázky a dalšími rozšiřujícími prvky, ale také bude obsahovat rozsáhlou sadu aplikací pro správu a komunikaci s koncovými uživateli a pokročilý content-management systém.



Známý způsob vývoje

Framework by měl poskytnout programátorům webových aplikací možnosti a prostředky, které jsou již delší dobu používány v objektových knihovnách MFC, VCL nebo WxWindows.

Na webové aplikace se bude tedy pohlížet jako na událostmi řízené formulářové aplikace, kdy některé události jsou řešeny přímo klientským browserem (rendering HTML/CSS, aktivní prvky jako jsou listboxy, tlačítka atd., a skriptování na straně klienta¹), zatímco zbylou část významných událostí ošetřuje přímo serverová aplikace (**Dialog API**).

Uživatелеm definovaný vzhled

Ty vrstvy frameworku, které pracují na generování samotného funkčního jádra budoucí aplikace (**Dialog API**), by měly být maximálně nezávislé na vrstvách, které se budou starat o vizuální prezentaci aplikace pomocí jazyka HTML/CSS (**Visual Template Layer**).

Optimálně by definice vizuálního vzhledu aplikací měla být definovatelná ne jen programátorskými prostředky, ale také snadno srozumitelnými textovými šablonami (**human readable visual templates**). To umožní i „laickou“ změnu vzhledu aplikací bez ovlivnění jejich funkčnosti.

Snadná implementace

Samotný princip fungování cílových aplikací bude před koncovými programátory skryt a abstrahován pomocí sady objek-

toových tříd. Zdrojový kód by měl být nápadně podobný analogické desktopové aplikaci navržené např. v již zmiňovaných knihovnách MFC, VCL nebo wxWindows.

Práce s daty

Framework by měl obsahovat sadu pomocných tříd pro snadnou práci s daty. Jednalo by se především o práci se zdroji dat (**data source, SQL**), práce se strukturovanými daty (**XML**) a práce s jednotlivými datovými položkami (**strings, regular expressions**).

Modulárnost a rozšiřitelnost

Framework bude sestávat z mnoha objektových tříd pro reprezentaci funkcí aplikací (**common dialogs & controls**). Maximální modularita jednotlivých tříd zajistí možnost, aby i koncový programátor vytvářel odvozené třídy již existujících objektů se specificky pozměněnými vlastnostmi.

Efektivita

Výsledné aplikace by měly být v rámci webového serveru prováděny v nativním kódu (což bude tedy spustitelný tvar výsledné aplikace, který bude komunikovat s klientem pomocí HTTP protokolu a jazyka HTML/CSS). Speciální modul pro webový server zajistí just-in-time kompilaci ze zdrojového tvaru programu v případě, že spustitelná podoba zatím neexistuje nebo je starší než zdrojový text (**just-in-time compile module**).

¹JavaScript