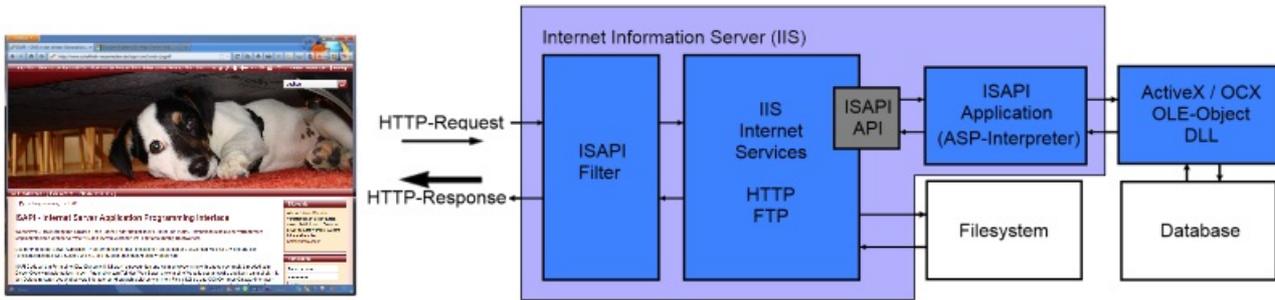


ISAPI – Internet Server Application Programming Interface

Webserver - Erweiterung mit Lazarus (Free Pascal) oder Delphi. ISAPI - Filter und ISAPI - Extensions sind die performanteste Möglichkeit einen Webserver wie z.B. den IIS von Microsoft im Funktionsumfang zu erweitern.



Das ISAPI (Internet Server Application Programming Interface) bietet einen eleganten und effizienten Weg zur Erweiterung der Funktionalität eines Webserver wie z.B. des IIS oder auch des Apache Webserver. Da ISAPI-Anwendungen kompiliert sind, sind sie viel schneller als ASP-Dateien oder Dateien, die COM+-Komponenten verwenden.

ISAPI-Code wird in Form einer DLL (Dynamic-link library) eingebunden und kann entweder single-threaded oder multi-threaded sein. Dieser Code wird automatisch in dem Augenblick zum Teil des Webserver, wenn eine Website gestartet wird und kann dann direkt mit den Datenstrukturen des Webserver interagieren. Hierdurch bietet es weit mehr Flexibilität als das CGI (Common Gateway Interface) welches einen separaten Prozess nutzt und hat auch nicht dessen Leistungsgrenzen. Da der einmal geladene Code im Server integriert ist und nicht immer neu geladen werden muss, ist er zudem noch sehr performant.

Haben Sie sich für ISAPI entschieden, müssen Sie entscheiden, ob Sie einen ISAPI-Filter oder eine ISAPI-Extension erstellen möchten. Filter reagieren auf Events, die im Server während jedem HTTP-Request ausgelöst werden, Extensions hingegen generieren nur Response auf einen HTTP-Request auf die entsprechende URL der ISAPI-DLL hin.

Unter Umständen stehen zwei Methoden zum Verarbeiten Ihrer ISAPI-DLLs zur Verfügung. Out-of-Prozess bedeutet, dass ein separater ISAPI - Runner den Code verarbeitet. In-Prozess bedeutet, dass der Code in den Haupt-Webserver-Prozess geladen und verarbeitet wird, was potenziell gefährlich ist, aber auch schneller! Wenn der Code einen Fehler enthält oder nicht "thread-safe" ist, kann es zum Absturz des gesamten Servers kommen.

Die Realisierung

Sie benötigen Lazarus oder Delphi, alle erforderlichen Komponenten werden im Rahmen dieses Artikels entwickelt / besprochen und stehen als Download auf der Homepage zur Verfügung. Unter Delphi existieren zwar schon Komponenten zur Erstellung von ISAPI-Extensions diese benutze ich aufgrund Ihrer Komplexität sehr ungern.

Um eine möglichst universelle, aber auch nicht zu komplexe Lösung zu erstellen, hat es sich bei meiner Tätigkeit als Entwickler bewährt, ein Konzept bezüglich einer Schnittstelle in drei Ebenen aufzuteilen:

1. einer Interface-Datei, die lediglich die genaue Beschreibung der Schnittstelle mit deren Funktionen und Übergabeparametern enthält (das entspricht der C Header-Datei in diesem Fall `httpext.h` und `httpfilt.h`).
2. einer Klassen-Datei, die dem Entwickler ein einfaches Zugreifen auf alle Eigenschaften und

Funktionen der Interface-Datei ermöglicht (in diesem Fall werden es zwei Klassen-Dateien, eine mit den Klassen des Filters und die andere mit den Klassen der Extension).

3. einer Projekt-Datei mit der abgeleiteten Klasse, aus der Klassen-Datei und der Implementierung des Beispiels (in diesem Fall natürlich auch ein Beispiel zu jeder Klassen-Datei).

Autor: [Udo Schmal](#), veröffentlicht: 31.12.2011, letzte Änderung: 29.08.2024

[© Copyright 2024 Udo Schmal](#)