Subject: C++ Server Pages (CSP)
Posted by kasome on Mon, 06 Apr 2009 00:39:18 GMT
View Forum Message <> Reply to Message

C++ Server Pages (CSP) is a Web Engine for advanced Web Application Development, that uses blended Markup Language / C++ scripts (such as HTML/C++, XML/C++, WML/C++ etc.)

Similar to ASP and JSP, it provides a great easiness in creating web pages with dynamic content, as well as complex business applications.

However, instead of Java, Javascript or VBscript, it uses C++.

Support Microsoft C++ Compiler, Borland C++ Compiler or GNU C++ Compiler

This brings some significant advantages:

* Incredibly high processing efficiency. Benchmarks have shown a range of 80 to 250 times higher processing speed than ASP.

* The use of pure C++ allows the use of tons of libraries that are currently available. It is important to notice that the libraries written in C++ are tens or hundreds of times more than in any other language.

* It is widely accepted that the most skilled programmers in the IT market are the C++ ones. However, CGI, ISAPI and other frameworks where C++ applies, do not provide the web developer with facilities for efficient application development. As a result, until now, Web Development could not take advantage of the best programmers.

* The processing efficiency of CSP allows the use of affordable systems even for complex Web Applications with heavy algorithms.

* The ability of making direct system calls, allows the development of advanced web applications that are impossible with ASP and JSP. For example, it is possible for a CSP page to use multiple threads and do blocking tasks (credit card check, database queries etc.) simultaneously and hence faster, whereas such an accomplishment is only a dream for other technologies.

http://www.micronovae.com/CSP.html

Some example of CSP Syntax:

EX1: Embedded C++ code

```
<P>This is HTML code.</P>
<%
// This is C++ code
int x = 0;
%>
```

EX2: Use C++ variable

```
<%! // global scope code
int iCount = 0;
%>

<% // body code
++iCount;
%>
<P>
Hits since script was first loaded:
<% // body code
Response.Write( iCount );
%>
</P>
```

EX3: Use C++ new/delete

```
<%! // global scope code
unsigned char* pMem;
%>

<%!onload: // initialization code
pMem = new unsigned char [1024];
%>

<%!onfree: // clean up code
delete[] pMem;
%>
```

EX4: Use third party libraries

* Header files.
  Copy your library's header files (.h, .hpp etc.) in the include directory (typically "C:\Program Files\Micronovae\CSP\Engine\include").

* Import library files
  Copy your import library files (.lib) in the lib directory (typically "C:\Program Files\Micronovae\CSP\Engine\lib").

* Executable files
  If your library is not static, copy your executable files (.dll) in the dlls directory (typically "C:\Program Files\Micronovae\CSP\Engine\dlls").

* Link import library
  In order for the library to be linked to a script, you have to use the C++ precompiler directive:
  #pragma comment(lib, "MyLibrary.lib")