

# 基于 GoAhead webserver 的应用开发<sup>①</sup>

王向群<sup>1</sup>, 徐沛平<sup>2</sup>, 渠毅<sup>1</sup>, 吴明<sup>1</sup>

<sup>1</sup>(国网电力科学研究院, 南京 210003)

<sup>2</sup>(徐州供电公司, 徐州 221000)

**摘要:** 为了实现嵌入式 web 服务器中的动态数据交互功能, 在简要介绍了 GoAhead webserver 的特点之后, 分析并比较了创建动态页面的两种方法。在此基础上, 阐述了使用 Ejscrip 和 GoForm 创建动态数据和进行功能配置的方法。并用实例详细说明了 GoAhead webserver 的开发步骤, 以及动态数据和表单处理的实现过程, 并灵活地与 javascript 结合起来, 有效地实现了 B/S 下的动态数据交互。

**关键词:** 嵌入式 web 服务器; GoAhead webserver; 动态; Ejscrip; GoForm

## Application Development of GoAhead Webserver

WANG Xiang-Qun<sup>1</sup>, XU Pei-Ping<sup>2</sup>, QU Yi<sup>1</sup>, WU Ming<sup>1</sup>

<sup>1</sup>(State Grid Electric Power Research Institute, Nanjing 210003, China)

<sup>2</sup>(Xuzhou Power Supply Company, Xuzhou 221000, China)

**Abstract:** In order to realize the interact function of dynamic data in an embedded webserver, this paper analyzes the dynamic web page creation after presenting the characteristics of GoAhead webserver. Then the methods of daynamic data creation and function setting using Ejscrip and GoForm are introduced. Besides, an example is discussed to explain the development steps of GoAhead webserver and the implementation process of dynamic data and form handling, also in the example, dynamic data interaction is realized effectively on B/S architecture combined with javascript flexibly.

**Key words:** embedded webserver; GoAhead webserver; dynamic; Ejscrip; GoForm

随着 IP 技术的日益普及与成熟, 越来越多的嵌入式设备添加了基于 web 的管理功能, 客户端可以通过 web 浏览器进行远程访问, 与嵌入式设备进行方便有效的交互。要实现上述 B/S 架构的管理, 需要在设备中植入嵌入式 webserver。常见的 webserver 有: goahead、boa、lighttpd、shttpd、thttpd、mathopd、minihttpd、appweb。下面介绍 goahead webserver 的特点及应用开发方法。

## 1 GoAhead webserver 的特点

GoAhead webserver 是一款开放源代码的嵌入式 WEB 服务器程序, 小巧精致, 功能较强。支持 ASP、嵌入式的 javascript、CGI 执行, 支持内存中的 CGI

处理 GoForm; 如果不包含 SSI, 仅要求 60K 的内存; web 页面可以存在 ROM 或文件系统中; 支持多种操作系统, 并且移植非常方便; 软件架构清晰明了, 只需稍作配置便可进行开发<sup>[1,2]</sup>。

## 2 GoAhead Webserver 开发技术

嵌入式 web 服务器的应用系统如下:

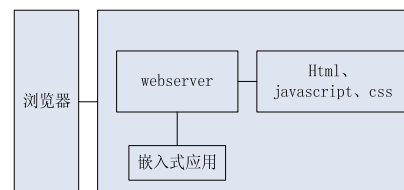


图 1 嵌入式 web 服务器的软件体系结构

① 收稿时间:2011-04-21;收到修改稿时间 2011-05-21

图1中, Web服务器通过http协议与浏览器进行通信, webservice在收到客户端的请求后, 与嵌入式应用模块交互, 实现对嵌入式设备的读取与设置, 同时动态创建页面, 实现设备状态及参数在浏览器端的显示<sup>[3]</sup>。

在嵌入式设备中, 大部分的web页面是动态生成的。由于GoAhead web server的主要功能是提供对设备的访问控制, 所以, 显示设备状态和提供操作反馈非常重要。创建动态页面主要有两种方法:

- ① 完全通过C程序生成HTML标签。
- ② HTML中通过扩展的标签插入动态数据。

前一种方式是在服务器端通过C代码构造HTML文档, 可以在服务器端一气呵成, 这种方式虽然比较灵活, 但是生成的页面必须经过编辑和编译运行服务器端程序才能看到最终的显示, 若想得到满意的页面需要反复得修改编译运行, 过程比较繁琐。第二种方式开发起来较快捷, 可以借助某些工具开发可视直观的页面, 但是服务器端脚本要放入HTML页面中正确的位置, 在执行时可以被动态数据替换。上述两种方法可以根据具体应用灵活运用。

为了方便的创建具有高度交互性的动态网页, GoAhead提供了嵌入式JavaScript和GoForms过程两种方法。它们实际上都与定义在服务器端的某个C函数绑定在一起, 嵌入式JavaScript用来在ASP中创建动态数据, 而GoForms过程则用来处理用户输入和修改设置, 它们一起构成了GoAhead webservice的核心。

### 2.1 嵌入式JavaScript

GoAhead开发了一种嵌入式的脚本语言Ejscrip。JavaScript是一种庞大的语言, 占用资源较多, 因此不适用于嵌入式设备中。GoAhead设计的嵌入式JavaScript—Ejscrip解决了这个问题。Ejscrip是JavaScript的一个严格子集, 实现了其基本语言要素。Ejscrip可以方便地应用在ASP中创建动态数据。

在使用Ejscrip创建ASP页面时, 脚本通过分隔符来界定, 基本格式为:

```
<% function(arguments, ...); %>
```

JavaScript程序通过websAspDefine API进行注册, 这样声明了C程序作为JavaScript的全局变量<sup>[1]</sup>。例如:

```
extern int outputMyIP(int ejid, webs_t wp, int argc, char_t **argv);
```

```
websAspDefine(“outputIP”, outputMyIP);
```

websAspDefine声明了JavaScript命令, 并把此命令和C程序“outputMyIP”关联起来, 当一个ASP页面被浏览器请求时, 此页面中所有使用“outputIP”脚本的地方都将调用“outputMyIP”函数。

### 2.2 GoForm

GoAhead通过叫做GoForm的内存表单处理来实现标准的通用网关接口(CGI)<sup>[4,5]</sup>。与传统的CGI不同, GoForm不需要对浏览器的每个连接请求都建立一个进程, 而是通过与GoAhead服务器共享地址空间, 来直接获取全部请求的上下文。GoForm可以自动解析所有的Post和Query数据, 并提供了一组API方便的访问CGI变量。

GoForm过程与ASP过程不同, 它主要用来处理用户更改配置或执行某种动作。GoForm程序通过websFormDefine API进行声明注册<sup>[1]</sup>, 例如:

```
extern int outputMyForm(int ejid, webs_t wp, int argc, char_t **argv);
websFormDefine(“outputForm”, outputMyForm);
void outputMyForm(webs_t wp, char_t *path, char_t *query)
{
websHeader(wp);
websWrite(wp, “IP:%s”, websGetVar(wp, “IP”, “”));
websWrite(wp, “Gateway: %s”, websGetVar(wp, “Gateway”, “”));
websFooter(wp);
websDone(wp, 200);
}
```

websFormDefine声明了表单函数, 并把此命令与C程序“outputMyForm”关联起来, 当提交表单时, 服务器端的C函数“outputMyForm”将被执行。GoForm函数用来写HTTP头和HTML文档内容, 并返回到浏览器端。其中, websHeader使用HTML的起始标签来创建标准的HTTP头, websFooter使用HTML的结束标签关闭此文档。在Goform程序中, websGetVar, websWrite, websRedirect等是经常用到的API。

## 3 实例说明

使用GoAhead webservice动态交互web页面的开

发流程如下:

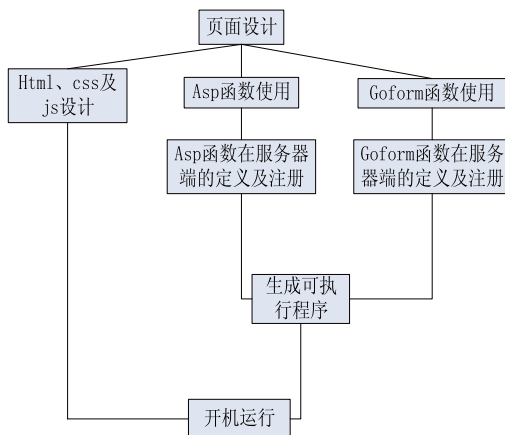


图 2 GoAhead webservice 开发流程

大致分为三部分:

- ① web 页面的设计, 在页面中加入 Ejscript 及表单函数。
- ② 在服务器端 C 文件中定义对应的 C 程序。
- ③ 在服务器端通过 websAspDefine 和 websFormDefine 进行注册。

下面举例说明 ASP 过程和 GoForm 过程

文件 Example.asp 中:

```

<form name="network_form" method="post" action
="/goform/NetworkSetting" target="mid" >
  <input type="text" name="ip_addr" size="30" ma-
xlength="30"
  value="<%get_ip_addr();%>"onChange="return
CheckIP(this)">
  <input type="text" name="sub_mask" size="30"
maxlength="30"
  value="<%get_sub_mask();%>"onChange="return
CheckSubnet(this)">

```

```

<input type="text" name="gate_way" size="30"
maxlength="30"
  value="<%get_gateway();%>"onChange="return
CheckIP(this)">
  <input name="button" type="submit" value= "App-
ly">
</form>

```

文件 Example.c 中:

```

websAspDefine( "get_ip_addr", get_ip_addr);
websAspDefine( "get_sub_mask", get_sub_mask);
websAspDefine( "get_gateway", get_gateway);
websFormDefine( "NetworkSetting", NetworkSetti-
ng);
int get_ip_addr (int ejid, webs_t wp, int argc, char_t
*argv){};
int get_sub_mask (int ejid, webs_t wp, int argc,
char_t *argv){};
int get_gateway (int ejid, webs_t wp, int argc, char_t
*argv){};
void NetworkSetting(webs_t wp, char_t *path,
char_t *query){};

```

在 Example.asp 中, 该页面可以显示当前设备的 IP 地址, 掩码及网关, 同时, 如果浏览器端如果进行修改后, 点击 Apply 按钮后, 设备可以进行相应的修改。页面中有三个输入框, 在该页面载入的时候, 要向服务器端请求三个动态数据填入。

在 Example.c 文件中, 定义了三个 ASP 函数及一个表单函数, 并通过 websAspDefine 和 websFormDefine 与服务器端 C 函数关联起来。

以上面 <%get\_ip\_addr()%> 为例介绍动态数据的获取, 此 Ejscript 函数目的是获取设备的 IP 地址, 并在第一个输入框中显示出来。当浏览器端请求 Example.asp 页面时, GoAhead 在将此页面发送到浏览器之前, 对该 ASP 页面进行解析, 当解析到 <%get\_ip\_addr()%> 时, GoAhead 将调用相应的 C 函数, 执行的结果格式化为 HTML 输出。后面掩码及网关的获取过程与 IP 地址相同。ASP 页面解析完成之后, 发送到浏览器即可显示这三个参数。

对于此三个动态数据的显示也可以采用另外一种方法, 取消这三个 ASP 函数, 可以在 HTML 中对三个参数赋初值。只定义一个 ASP 函数, 可以将该函数放

入页面 body 的 OnLoad 函数中,此函数在相关联的 C 程序中,要输出在浏览器端使用的 JavaScript 语句。格式化输出的内容应该为:

```
Document.network_form.ip_addr.value="10.144.66.86"
```

```
Document.network_form.sub_mask.value="255.255.255.0"
```

```
Document.network_form.gate_way.value="10.144.66.254"
```

这种方法是对本文第三章讲到的创建动态数据方法 B 的一种延伸,可以完全把动态数据和页面设计分开,可以充分地利用一些工具进行操作界面的设计。

当点击 Apply 按钮之后,GoAhead 解析 action="/goform/NetworkSetting" 将执行 goform/后面的表单函数 NetworkSetting。在该表单函数中,执行完功能配置后的返回页面有两种方法,一种是利用 websHeader(),websFooter(),websDone()和 websWrite()来构造 web 页面,另一种可以通过 websRedirect()函数载入页面<sup>[1]</sup>。

## 4 结语

GoAhead webserver 是一款小巧而功能丰富的嵌入式 web 服务器,可以利用它提供的 Ejscrip 和 GoForm 进行方便有效的动态数据交互。除此之外,GoAhead 还支持使用 SSL 进行数据加密和认证,还支持摘要认证机制,以及它的用户管理功能可以对不同的用户设置不同的访问权限。由于具有以上的种种优点,GoAhead webserver 的应用将日益广泛。

## 参考文献

- 1 <http://www.goahead.com>.GoAhead webserver.
- 2 房芳,马旭东.基于嵌入式 Web 技术的监控系统设计与实现.计算机工程与设计,2009,35(23):237-239.
- 3 潘琢金,王秋实.嵌入式 Web 服务器中动态 Web 技术的研究.计算机工程设计,2010,31(18):3975-3978.
- 4 谢仕义,徐兵.嵌入式 Web 服务器的设计及其 CGI 实现.计算机工程与设计,2007,28(7):1598-1600.
- 5 张曦煌,柴志雷.嵌入式 Web 服务器中 CGI 的特点及实现.小型微型计算机系统,2003,24(11):2046-2047.

(上接第 239 页)

④ 简单直观操纵。拖放模式生成图表。用户还可以通过直接操作图形界面控制模型的一些参数,并可以测试不同的方案来进行假设分析操作。

现如今,MyM 越来越受世界各国科研小组重视,利用其开发了很多大型动态模型,如荷兰环保局研发的大型动态模型 IVM 模型和 TARGETS 模型。MyM 也可以为已存在的模型构建图形化界面,如美国航天宇航局 NASA 也用其构建月球相关的数学模型。无论科研领域是环境科学、流行病学、机械工程或系统控制,所有可用方程系统描述的数学模型都可以在 MyM 中处理。MyM 的应用具有很好的推广前景。

## 参考文献

- 1 谭勇基,蔡志杰.数学模型.第 2 版.上海:复旦大学出版社,2011.2-10.
- 2 MyM 主页.<http://www.my-m.eu/>.
- 3 MyM 手册文档.MyM Language Reference Manual,2008.[http://www.my-m.eu/?page=support\\_manuals](http://www.my-m.eu/?page=support_manuals).
- 4 MyM 手册文档.MyM\_manual\_GettingStarted,2008.[http://www.my-m.eu/?page=support\\_manuals](http://www.my-m.eu/?page=support_manuals).
- 5 MyM 手册文档.MyM Language Tutorial,2008. [http://www.my-m.eu/?page=support\\_manuals](http://www.my-m.eu/?page=support_manuals).
- 6 孙俊逸,朱永松.计算方法.北京:机械工业出版社,2011.139-147.