



Remote Server Management with Intel® Server Control V2.x and Wireless Application Protocol (WAP)



Revision 1.0

October 18, 2000

**Enterprise Platforms Group
Server Products Division**

Revision History

Date	Revision Number	Modifications
10/18/2000	0.5	Draft release.
10/18/2000	1.0	Released version

Disclaimers

Information in this document is provided in connection with Intel® products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

This document contains information on products in the design phase of development. Do not finalize a design with this information. Revised information will be published when the product is available. Verify with your local sales office that you have the latest datasheet before finalizing a design.

The Intel® Server Control product may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Copyright © Intel Corporation 2000. *Other brands and names are the property of their respective owners.

Table of Contents

1. Introduction	1
1.1 What Is WAP?	1
1.2 Why is WAP Important?	1
1.3 About This Whitepaper	2
2. Prepare the Server for WAP	4
2.1 Install SNMP Service	4
2.2 Configure the IP Address “127.0.0.1” to Access SNMP Information.....	5
2.3 Install Intel® Server Control	5
2.4 Use Platform Instrumentation Control to Confirm SDLINK.DLL.....	5
2.5 Test SDLINK from DOS	5
2.6 Add MIME Types.....	6
2.6.1 Adding MIME Types to the Default Web Site	6
3. Install and Configure the WAP 2.0 Toolkit	9
3.1 Install ASPEXec	10
3.2 Install WAP Scripts	11
3.2.1 Extract the “wml scrips.zip” File	11
3.2.2 Edit the index.wml File	12
4. WAP Gateway	13
5. Demonstration of WAP-Enabled Cell Phone with ISC	14

List of Figures

Figure 2-1. SNMP Service Properties Dialog for SNMP Configuration	4
Figure 2-2. MIME File Types for the Default Web-Site	7
Figure 2-3. Home Directory Dialog for the Default Web Site	8
Figure 2-4. Text of the sample test.wml Script	8
Figure 3-1. Browser Dialog	9
Figure 3-2. WAP 2.0 Simulator Results from test.wml	9
Figure 3-3. Dir.ASP File	10
Figure 3-4. Execution of Sample dir.asp Active Server Page	11
Figure 3-5. Default Web-Site in Microsoft's IIS 5.x Server	11
Figure 5-1. WAP Demo Main Menu	14
Figure 5-2. Scrolling to Fans	15
Figure 5-3. Fan Indicating 0 RPM	16

1. Introduction

Remote management of servers is no longer the luxury feature it once was. Due to the Internet and the subsequent e-business model spawned by the Internet, servers are now synonymous with doing business. Server transactions are prime revenue generators for companies around the world. When a server unexpectedly goes down, it often means direct loss of company revenue.

As servers continue to proliferate, so do the demands placed on IT staff and support personnel. Remote server management technology, which includes health monitoring, alerting, remote access, and emergency management and diagnostics, is quickly becoming the preferred means of server access. Remote server management has quickly progressed from modem-only server access to LAN access and now, to WAP access using a cellular phone.

1.1 What Is WAP?

WAP, or the Wireless Application Protocol, was developed by the WAP Forum (www.wapforum.org), a non-profit vendor consortium. WAP is not a single protocol. The current version, version 1.2, specifies a full application model, a set of service components, a suite of protocols, a mark-up syntax, a scripting language, and other interoperability specifications. Enterprises will increasingly run mobile applications over wireless transports, and WAP is an important interoperability framework for application developers and implementors.¹

WAP supports client/server communication sessions from portable devices over wireless access links. WAP standards enable wireless client software developers to write to a single set of programming interfaces, thereby abstracting applications from many of the air-link specific details associated with lower-level wireless protocols. The WAP suite of specifications receives considerable support among wireless handset vendors, wireless infrastructure vendors, and wireless application software providers.

Queries originating from a WAP-enabled wireless device are currently decoded by a WAP server located on the operator's network. The WAP server connects to the chosen Internet site, which must be coded in WML (wireless markup language). Finally, the WAP server conveys the WML-encoded data back to the handheld device, where a micro-browser displays the text information on the device's screen. From start to finish, this procedure may take only a few seconds. WAP is designed to make efficient use of today's small mobile device screens, from simple two-line text displays to large full graphics screens. WML also facilitates simple text navigation in an environment that lacks the flexibility of a mouse.²

1.2 Why is WAP Important?

As our lives become increasingly intertwined with the digital revolution that surrounds us, the computing world is shifting. What was once a computing lifestyle that forced us to adjust to the technology, is quickly becoming a lifestyle computing model in which systems and devices meld into our lives and become much less obvious by adapting to our needs. With this model, interactions with the digital environment become interwoven into every aspect of daily life.

¹ James Kobielus, [Network Strategy Overview: Wireless Application Protocol](#) (The Burton Group, April 7, 2000)

² Bryan Prohm and Paul Dittner, [Does WAP Hold the Keys to the Kingdom of the Wireless Internet?](#) (Dataquest, April 24, 2000)

Of the several key factors that characterize lifestyle computing: user centricity, simplicity, virtual identity, independence, diversity, and control of personal information, simplicity is one of the most important. Users will not accept technology that does not work the first time. Users also become increasingly intolerant of technology solutions that do not improve the quality of their lives but rather add yet another layer of complexity. WAP puts the world of the Internet at the touch of a cellular button.

It is quickly becoming clear that the companies that are able to make complex technology simple and reliable to use will win.³ Many companies have acknowledged this. Of the fifteen worldwide cellular phone manufacturers, thirteen have models that are WAP-enabled. While customers in the U.S. are still realizing the functional benefits WAP can provide, customers in Europe and parts of Asia have already embraced WAP-enabled technology to simplify and enhance their lives.⁴

Intel acknowledges this trend as well. By providing remote server management using Intel® Server Control and WAP technology, Intel is leading the server industry by providing more simple, more robust, and more readily accessible remote server management functionality.

1.3 About This Whitepaper

This whitepaper describes how to access Intel Server Control through a WAP-enabled cellular phone for remote management of Intel SPD servers. The specific implementation of WAP discussed in this whitepaper is based upon SNMP and is intended solely to provide insight into available remote server management capabilities utilizing WAP.

This document assumes that Microsoft Windows* 2000 is installed and that the server is an Intel server using Intel Server Control (ISC) and SNMP, such as the L440GX, C440GX, SC450NX, or SKA4 server systems. The functionality available using this SNMP implementation includes:

- Server health status
- Voltage monitoring
- Temperature monitoring
- Fan speed
- Server reset

The actual implementation of WAP in the Intel Server Control product will be based on CIM. Implementing with CIM provides product consistency in technology development standards as well as ultimately providing more feature-rich functionality to the user. Therefore, it is important to note that the scripts detailed later in this document will not work when ISC discontinues support of DMI and SNMP, as is currently planned for ISC version 5.x.

Functionality available using a CIM implementation will include:

- Server health status
- Voltage monitoring

³ Steve Kleynhans, [Lifestyle Computing: The User Takes Center Stage](#) (META Group, July 28, 2000)

⁴ Peter Richardson, [Wall-to-Wall WAP: Critical Success Factors and Trade Show Announcements, Spring 2000](#) (Dataquest, April 6, 2000)

- Temperature monitoring
- Fan speed
- Server reset
- Server power control
- Access to server error logs
- E-mail alerting
- Enhanced security

2. Prepare the Server for WAP

Before beginning, follow the Microsoft installation instructions to install Windows 2000 on the server. Ensure the default IIS and TCP/IP services are installed. Then use the following steps to guide you through preparing the server to be monitored through WAP.

2.1 Install SNMP Service

1. In the Control Panel , select “Administrative tools”.
2. Select “Services”.
3. Right click on the “SNMP” service
4. Add the community “isc” (lower case) with read only access. See Figure 2-1.
5. Use the “Traps” tab to specify the IP address, or hostname, of the server to which the traps should be sent. Leave all other folders with their default settings.
6. For SNMP to successfully send traps it may be necessary to select the “Accept SNMP packets from any host” button. See Figure 2-1. This will provide SNMP trap access from the entire network. The possibility exists for SNMP data to be misconstrued, but open access may be necessary for testing purposes.

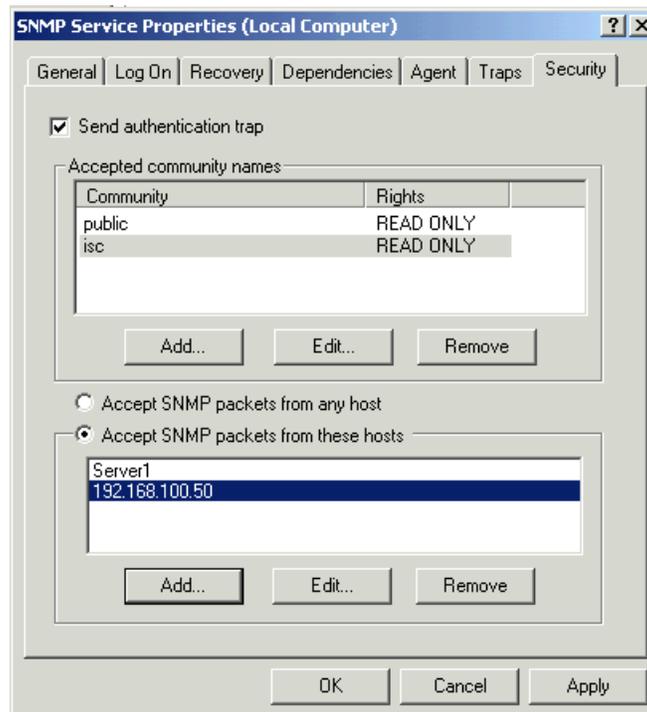


Figure 2-1. SNMP Service Properties Dialog for SNMP Configuration

2.2 Configure the IP Address “127.0.0.1” to Access SNMP Information

Copy “SNMPUTIL.EXE” from the Windows NT* resource kit to C:\ (or to the partition where Windows 2000 is installed). Note that by default setup.exe from the Windows 2000 resource kit installs “SNNMPUTIL.exe” to the “C:\Program Files\NT Resource Kit” directory. You will need to copy it to the root directory for the Active Server Pages (ASP) work correctly.

Test SNMP with the following DOS command at the run prompt:

```
C:\SNMPUTIL GETNEXT 127.0.0.1 isc .1.3
```

You should see the following result:

```
Variable = system.sysDescr.0  
Value    = String Hardware: x86 Family 6 Model 7 Stepping 2 AT/AT CO  
Software: windows 2000 Version 5.0 (Build 2195 Multiprocessor Free)
```

Note: Instead of the IP address, it may be necessary to type “SNMPUTIL GETNEXT servename isc .1.3” to achieve the desired result.

2.3 Install Intel® Server Control

Intel Server Control provides the monitoring capabilities. It is this component that provides access to the hardware information through the cell phone.

1. Insert the Country Kit CD with the latest ISC 2.3 software for your baseboard.
2. Run <CDROM drive>:\Software\ISC\setup.exe.
3. Upon completion of the installation, the server will reboot. Wait for the reboot to complete before proceeding.

Note: For further assistance with the ISC installation, see the “ISC Installation Guide” found in the same directory as the setup.exe file.

2.4 Use Platform Instrumentation Control to Confirm SDLINK.DLL

1. Launching Platform Instrumentation Control to start SDLINK.dll.
2. Confirm that SDLINK.Dll is running in the Windows “Task Manager” under the processes folder.

2.5 Test SDLINK from DOS

Type the following from a DOS prompt:

```
C:\SNMPUTIL GETNEXT 127.0.0.1 isc .1.3.6.1.4.1.343.2.6.2.2.1.4.1.3
```

You should see one of the following results:

```
Variable = .iso.org.dod.internet.private.enterprises.intel.products.server-prod
cts.platforms.basebrd.dmtfGroups.tSystemBios.eSystemBios.a4BiosVersion.7.1
value     = String S450NX0.86B.0073.P12.9910181550
```

```
Variable = .1.3.6.1.4.1.343.2.6.2.2.1.4.1.3.7.1
value     = String S450NX0.86B.0073.P12.9910181550
```

The “Value” is the Bios ID of the server, in the example above a SC450NX Bios Version P12 is “S450NX0.86B.0073.P12.9910181550”

2.6 Add MIME Types

Note: This section assumes that Internet Information Server (IIS) was configured when the server was installed. If this is not true, it is necessary to load it from the Windows 2000 Server Installation CD. Use the “Default Website” as defined the Administrative Tools’ Internet Services Manager, or Management Console. As an option, you can create a virtual directory that references c:\wap as its “Home Directory”. If you do not create the virtual directory, the ASP scripts and executables will be stored in the c:\ directory.

2.6.1 Adding MIME Types to the Default Web Site

1. Right click on the default web-sites to select “properties”
2. Select “HTTP Headers”
3. Select “MIME” file type

It is not necessary to remove MIME from the existing registered file types. However, “New Types” must be added for .wml, wbmp, and .wmlc extensions.

Document Type	MIME Type	Typical Extension
Plain WML documents	text/vnd.wap.wml	wml
Wireless Bitmap Images	image/vnd.wap.wbmp	Wbmp
Compiled WML documents	application/vnd.wap.wmlc	Wmlc
WMLScripts	text/vnd.wap.wmlscript	Wmls
Compiled WML Scripts	application/vnd.wap.wmlscriptc	wmlsc

4. On the server console, open the management console.
5. Right click on the directory or on the parent directory in which the MIME type should be added.
6. Select the HTTP headers tag.
7. Click the File Types button at the lower right.

- Click New Type and supply the extension and the content type as listed in the table above. See Figure 2-2.

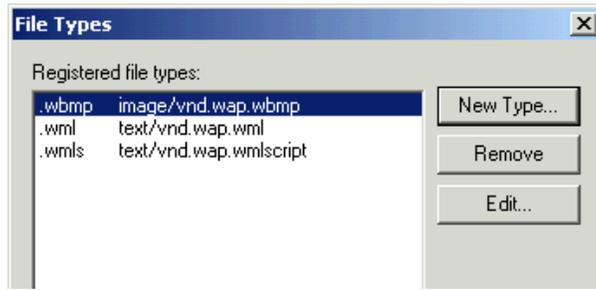
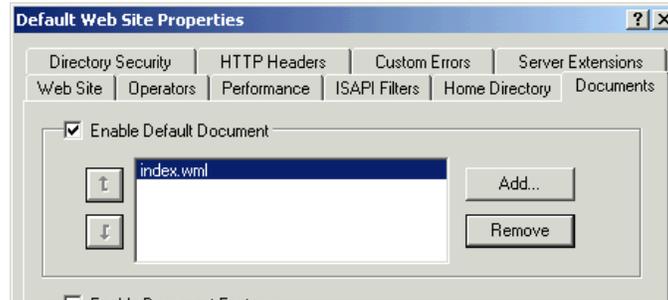


Figure 2-2. MIME File Types for the Default Web-Site

- Right click on the server to add a MIME type to it. This is located on the lower portion of the screen, which is not displayed in Figure 2–2.
- Click the File Types button.
- Click New Type and supply the extension and the content type as listed above.
- Set the default document to index.wml. See Figure 2-3. The default home directory for the default web server is "c:\inetpub\wwwroot".



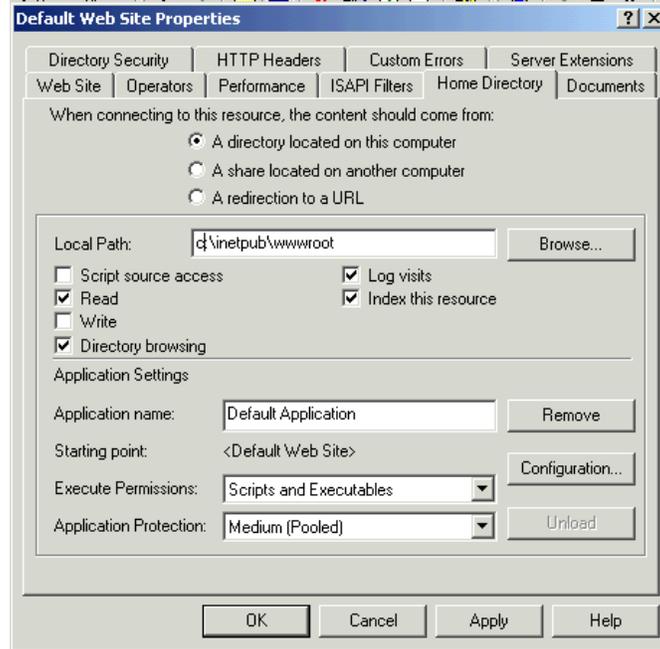


Figure 2-3. Home Directory Dialog for the Default Web Site

Use Notepad to create the “test.wml” file with the content displayed below. Save it to the “c:\inetpub\wwwroot”.directory.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<!-- Main menu for WML examples -->
<wml>

  <card id="index" >
    <p >
      WAP Test Page <br/>
    </p>
  </card>
</wml>
```

Figure 2-4. Text of the sample test.wml Script

3. Install and Configure the WAP 2.0 Toolkit

The WAP 2.0 toolkit is available from <http://www.nokia.com/corporate/wap/sdk.html>

1. Sign up for the Nokia forum to download their trial toolkit.
2. During the toolkit installation, launch only the setup.exe for the WAP 2.0 toolkit. This will automatically install the Java Runtime Environment 1.2.2.
3. Run the WAP toolkit on a remote system in the network.
4. Select the location of the server followed by the test.wml page.

Note: "wap-test" is the DNS hostname of your server.

5. Test they installation by typing <http://wap-test/test.wml> in the "Location" dialog of the WAP Toolkit browser, as displayed in Figure 3-1.



Figure 3-1. Browser Dialog

The script will generate a text message on the cellular phone, as indicated by Figure 3-2.



Figure 3-2. WAP 2.0 Simulator Results from test.wml

3.1 Install ASPEXec

1. Download ASPEXec from <http://www.serverobjects.com/comp/Aspexec.zip>
2. Unzip the ASPEXec.zip file. Assuming that c:\aspexec and its contents are unzipped to C:/, change to C:\ and run:

```
regsvr32 aspexec.dll
```

If the command is successful a dialog states "DLLRegisterServer in server aspexec.dll succeeded".

```
<html>
<head><title>ASPEXec Test (ExecutewinApp)</title><head>
<body bgcolor=white text=black>
<blockquote>
<H3>ASPEXec ExecutewinApp Test</H3>
<%
    Set Executor = Server.CreateObject("ASPEXec.Execute")
    Executor.Application = "cmd"
    Executor.Parameters = "/C dir "
    Response.Write "Attempting to execute " &
Executor.Application & "<br/>"
    strResult = Executor.ExecuteDosApp
    Response.Write "The result of this call was: <pre> " &
strResult & " <pre/>"
%>
<p>
</blockquote>
</body>
</html>
```

Figure 3-3. Dir.ASP File

3. Copy the Dir.asp file from C:\aspexec to the default web directory, c:\inetpub\wwwroot).
4. Test ASPEXec by running the dir.asp file from a remote machine, using a web browser as seen in the figure below.

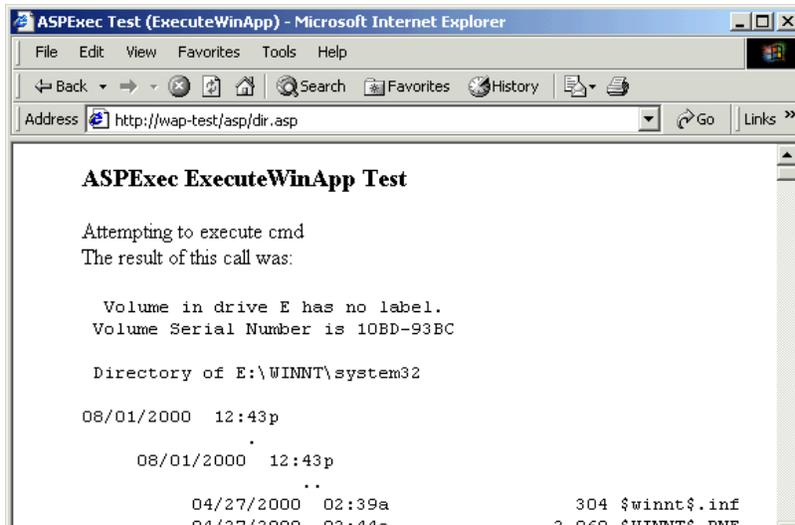


Figure 3-4. Execution of Sample dir.asp Active Server Page

3.2 Install WAP Scripts

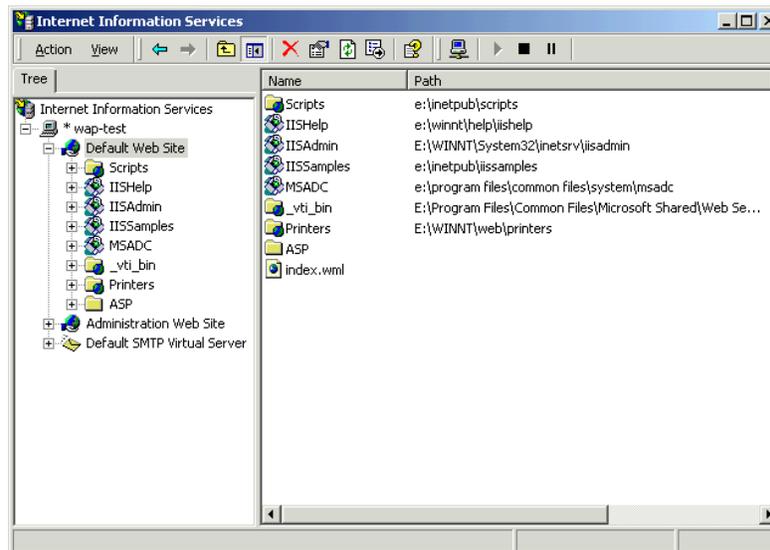


Figure 3-5. Default Web-Site in Microsoft's IIS 5.x Server

3.2.1 Extract the "wml scrips.zip" File

1. Copy the index.wml to the default web site ("c:\inetpub\wwwroot".)
2. Copy all the .asp files to a directory called asp in the default website ("c:\inetpub\wwwroot\asp".)

3.2.2 Edit the index.wml File

The index.wml file is displayed in Figure 3-.

1. Edit the select section of the file to add and remove servers in your network. Type in the IP address or DNS name followed by the name of the server, such as:

```
<option value="127.0.0.1">LocalHost </option>
```

2. Run the WAP toolkit on a remote machine in the network.
3. Verify that wap server has the correct IP address and hostname in the index.wml script.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<!-- Main menu for WML examples -->
<wml>
  <card id="index" >
    <p mode="nowrap"> WAP Management Demo <br/><br/>
    <select name="Agent">
      <option value="127.0.0.1">LocalHost </option>
      <option value="192.168.100.80">Wap-Test </option>
      <option value="192.168.100.90">WapGateway </option>
      <option value="111.112.113.12">Server3 </option>
      <option value="111.112.113.51">Server4 </option>
    </select>
    <anchor >HealthStatus <go href="ASP/WASPHHealth.ASP" method="post">
    <postfield name="Agent" value="\$(Agent)" /> </go> </anchor> <br/>
    <anchor >SecurityStatus <go href="ASP/WASPChassis.ASP" method="post">
    <postfield name="Agent" value="\$(Agent)" /> </go> </anchor> <br/>
    <anchor >Fans <go href="ASP/WASPFans.ASP" method="post">
    <postfield name="Agent" value="\$(Agent)" /> </go> </anchor> <br/>
    <anchor >Temperature <go href="ASP/WASPtemp.ASP" method="post">
    <postfield name="Agent" value="\$(Agent)" /> </go> </anchor> <br/>
    <anchor >Voltage <go href="ASP/WASPVolt.ASP" method="post">
    <postfield name="Agent" value="\$(Agent)" /> </go> </anchor> <br/>
    <anchor >BiosID <go href="ASP/WASPBios.ASP" method="post">
    <postfield name="Agent" value="\$(Agent)" /> </go> </anchor> <br/>
    <a href="ASP/WASPSShutdown.ASP">RestartServer</a><br/>
  </p>
</card>
</wml>
```

Figure 3-10. Index.wml File

4. WAP Gateway

The WAP gateway is the component that allows html code to be converted to XML. This makes WAP viewable on cell phones. WAP Gateways can be established with Windows 2000, Windows NT 4.0, or Linux* 6.x.

To use the WAP Gateway, connect to the Internet and use the mobile phone operator's WAP Gateway or install WAP Gateway software. For details of the WAP Gateway installation, refer to the Nokia WAP web-site at <http://www.nokia.com/networks/17/wdss.html>. The WAP Gateway documents are titled "Setting up Nokia WAP Gateway with Win NT 4.0 RAS" and "Setting up Nokia WAP Gateway with Win 2000 RAS."

5. Demonstration of WAP-Enabled Cell Phone with ISC

This demonstration is based on the Nokia WAP Toolkit 2.0. Screen shots are identical to content that would be displayed on a real cell phone.

1. Launch the "WAP" toolkit from the "Nokia WAP Toolkit 2.0" program option.
2. Type `http://wapserver/index.wml` in the browser dialog. The screen displayed in Figure 5-1 will launch, providing the options to check HealthStatus, SecurityStatus, Fans, Temperature, Voltage, and reset the server.

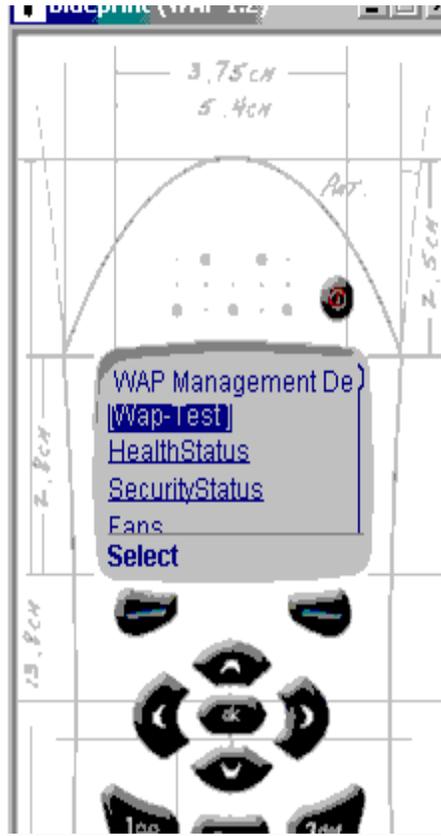


Figure 5-1. WAP Demo Main Menu

3. Scroll down and select the “Fans” option to read the speed of the fans. See Figure 5-2.

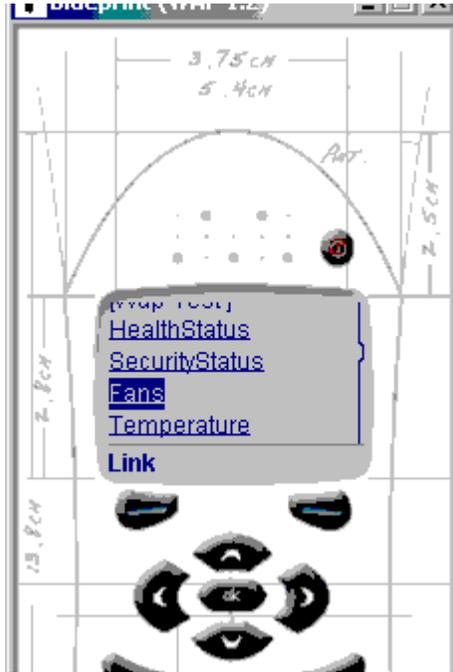


Figure 5-2. Scrolling to Fans

4. Select the middle button (“OK”) on the cell phone/cell phone simulator to launch the card with the fan speeds listed on the system server. As indicated by Figure 5-3, one fan is running at 0 RPM, suggesting a possible hardware failure or a disconnected fan.

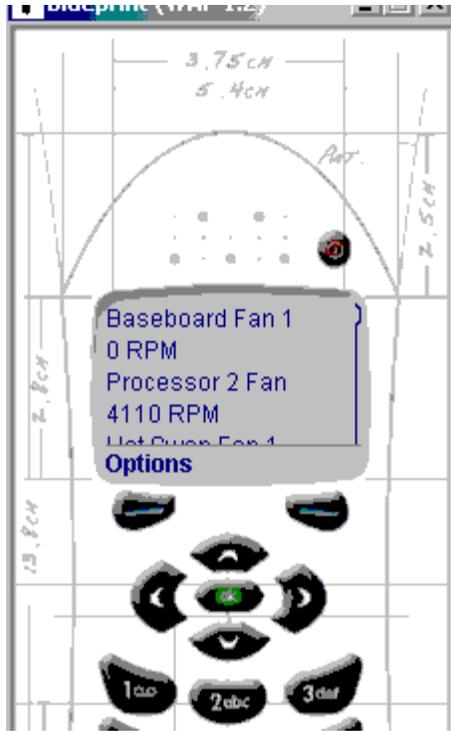


Figure 5-3. Fan Indicating 0 RPM

Return to the main menu by clicking on the Options button and selecting back. The screen portrayed by Figure 5-1 will be displayed again and you can proceed to view other sensor information or reboot the server.

Note: If multiple servers are defined in the index.wml file, the servers will be listed on the first screen displayed. From the list of displayed servers, you can select a specific server to view.