

SmileTiger WebPresentation™ Integration Guide



SmileTiger Software

SmileTiger Software Corporation

11615 Sir Francis Drake Drive

Charlotte, NC 28277 USA

Tel: + 1 704 321 9068

Fax: +1 704 321 5129

<http://www.smiletiger.com>

Revision 5.4, Oct. 2003

Table of Contents

1	BASIC SYSTEM CONFIGURATION	3
1.1	LINK YOUR WEBSITE TO A WEBPRESENTATION SERVER	3
1.2	CUSTOMIZE THE CONFERENCE LAUNCH PAGE	3
1.3	CHANGE THE CONFERENCING SERVICE PORT	3
1.4	RUN WEB SERVER AND WEBPRESENTATION SERVER ON THE SAME PORT	4
1.5	BLOCK OUTSIDE ACCESS TO THE CONFERENCE ROOMS.....	4
1.6	ENABLE CUSTOM SERVER MANAGEMENT	4
1.7	DISABLE ADMINISTRATION ACCESS FROM CONFERENCING PORT	5
2	USER ACCOUNT MANAGEMENT	6
2.1	INTEGRATE WITH AN USER ACCOUNT MANAGEMENT SYSTEM	6
3	VIRTUAL CONFERENCE ROOM MANAGEMENT	7
3.1	CREATE A ROOM	7
3.2	DELETE A ROOM.....	7
3.3	UPDATE A ROOM'S PROPERTIES	7
3.4	HANDLE RECURRING MEETINGS	8
3.5	A SAMPLE SCENARIO OF CUSTOM ROOM MANAGEMENT	8
4	SYSTEM REPORT	9
4.1	GET SYSTEM STATUS.....	9
4.2	GET USAGE REPORT	9
5	REDUNDANCY AND LOAD-BALANCING	10
6	APPENDIX	12
6.1	A PHP EXAMPLE OF THE CONFERENCE LAUNCHING MECHANISM	12

1 Basic System Configuration

1.1 Link your website to a WebPresentation server

You can use a HTTP GET or POST request in your webpage to launch a SmileTiger WebPresentation conference. To use the POST method to launch conferences, insert the following HTML code into one of you the web pages in your website:

```
<FORM method="post" action="http://[your server]:7770/wplogin.jsp">
  <table>
    <tr>
      <td><FONT face="Verdana" size="-1">Conference Name:</FONT></td>
      <td><input name="conference" value=""></td>
    </tr>
    <tr>
      <td><FONT face="Verdana" size="-1">Password:</FONT></td>
      <td><input type="password" name="conf-password" value=""></td>
    </tr>
    <tr>
      <td><FONT face="Verdana" size="-1">My name:</FONT></td>
      <td><input name="user-name" value=""></td>
    </tr>
    <tr>
      <td colspan=2>
        <center><br> <input type="submit" name="Submit" value="Enter"></center>
      </td>
    </tr>
  </table>
</FORM>
```

You need to replace **[your server]** with your SmileTiger WebPresentation Server machine's domain name or IP address. The port number **7770** may also need to change if you configure your SmileTiger WebPresentation Server to run on a different port, such as port 80.

1.2 Customize the conference launch page

Edit or replace file `<STWPS Home>/web/wplogin.html` to customize the conference launching interface.

1.3 Change the conferencing service port

The default conferencing service port is **7770**. You may want to change the port to match your network environment. For example, if some of your users behind a firewall that blocks port **7770**, you can use port **80** to provide the conferencing service, so that they can access the conference server without any configuration in their firewall. To change the port:

- 1) Edit file `<STWPS HOME>/wvhome/default/sysmgr.ini`
- 2) Change '7770' inside `<conferencePort>` tag to your desired port.

- 3) Restart your server after the change

1.4 Run web server and WebPresentation server on the same port

It is possible to run both your web server and the SmileTiger WebPresentation server on the same port of the same machine. You need multiple IP addresses in the server machine to do that.

- 1) Configure your Apache Server to listen on one IP address
- 2) Edit file <STWPS HOME>/wvhome/local/system.ini
- 2) Change 'localhost' inside <host> tag to an IP address of your server.
- 3) Restart your apache server and the ST WebPresentation server after the change

1.5 Block outside access to the conference rooms

You can block the conference room access from all websites except your front end website.

To block the access from all other websites:

- 1) Edit file <STWPS HOME>/wvhome/default/sysmgr.ini
- 2) Change 'ALL' inside <allowedDomain> tag to your website's domain name
- 3) Restart your server after the change

1.6 Enable custom server management

You can enable/disable the management of a SmileTiger WebPresentation server and conference rooms from a different computer using port 7771 as described in this document.

To enable the custom management:

- 1) Edit file <STWPS HOME>/wvhome/default/sysmgr.ini
- 2) Change 'disable' inside <customServerMgr> tag to 'enable'
- 3) Restart your server after the change

1.7 Disable administration access from conferencing port

You can enable/disable the administration access of a SmileTiger WebPresentation server from the conferencing port.

To disable the custom management:

- 1) Edit file <STWPS HOME>/wvhome/default/sysmgr.ini
- 2) Change 'disable' inside <adminInConferencePort> tag to 'disable'
- 3) Restart your server after the change

2 User Account Management

To avoid double authentications, SmileTiger WebPresentation Server does not authenticate users and manage user accounts. However, SmileTiger WebPresentation Server can integrate an existed user account management system seamlessly.

2.1 Integrate with an user account management system

SmileTiger WebPresentation can tightly integrate with your existed account management subsystem. When your account management subsystem is used, there are 3 components in the system: Your front-end website, account management subsystem, and SmileTiger WebPresentation server.

Your front-end website provides the user interface for signing up and signing in users. It communicates with the account subsystem to authenticate users and get authorization information. The authorization information includes the SmileTiger WebPresentation conference rooms that a user is allowed to enter. After a user signs in, the front-end website returns a webpage containing conference links to him. The user then clicks on a conference link to enter a SmileTiger WebPresentation conference room.

The above design requires you to block the conference access from all websites except your front-end website. The conference links in your website can be implemented by using HTTP GET or POST method. Please see section 1 for detailed instructions.

3 Virtual Conference Room Management

SmileTiger WebPresentation Server includes a default conference room management system. However, a custom conferencing system may prefer to offer its own conference management logic and custom look and feel. To create the custom management logic, developers should create several management web pages in the front-end website, and store the conference room data into user accounts. To integrate the management system with SmileTiger WebPresentation server, the front-end website posts HTTP requests to WebPresentation Server to add/remove conference rooms.

3.1 Create a room

Submit a HTTP POST request as the following form to create a room:

```
<form method="POST" action="http://[your server]:7771/createRoom.jsp">
  <input name="name" value="">
  <input name="description" value="">
  <input name="password" value="">
  <input name="moderatorPassword" value="">
  <input name="homePage" value="">
  <input name="confType" value="">
  <input name="mod" value="">
  <input name="seat" value="">
  <input name="audioBuffer" value="">
  <input name="pushSpeed" value="">
  <input name="shareSpeed" value="">
  <input name="sharedFolderSize" value="">
</form>
```

3.2 Delete a room

Submit a HTTP POST request as the following form to delete a room:

```
<form method="POST" action="http://[your server]:7771/deleteRoom.jsp">
  <input name="name" value="">
</form>
```

3.3 Update a room's properties

Submit a HTTP POST request as the following form to update a room:

```
<form method="POST" action="http://[your server]:7771/updateRoom.jsp">
  <input name="name" value="">
  <input name="description" value="">
  <input name="password" value="">
  <input name="moderatorPassword" value="">
  <input name="homePage" value="">
  <input name="confType" value="">
  <input name="mod" value="">
  <input name="seat" value="">
  <input name="audioBuffer" value="">
  <input name="pushSpeed" value="">
</form>
```

```
<input name="shareSpeed" value="">
<input name="sharedFolderSize" value="">
</form>
```

3.4 Handle recurring meetings

Some meetings recurs every day or week. The front-end website should handle the scheduling. It stores the scheduling information into the user account data store. When a user tries to enter a conference, the front-end website compare the current time with the valid conference time slots, and then return a PASS or DENY page to the user.

3.5 A sample scenario of custom room management

When a user signs in to his account, the front-end website returns him a web page which enables him to schedule conferences. The user then tries to create a conference room by submitting a conference setup request. The front-end website creates a unique number as the name (key) for the conference room, stores the conference information to the user account data store, and sends a request to SmileTiger WebPresentation Server to setup the room.

4 System Report

4.1 *Get system status*

The system status can be fetched from SmileTiger WebPresentation Server by using a HTTP GET or POST request. The general request is as following:

http://ST_SERVER_ADDR:7771/report/status.jsp?action=action-para¶m2=content2

Your firewall should block the access to port 7771 from all outside machines. The following action parameters are available:

user_total: The total number of the online users

active_rooms: List all those conference rooms that have users.

4.2 *Get usage report*

The usage report is dumped into the <STWPS HOME>/wpslog directory. Each log file in the directory contains information of a single day usage. The log files can be fetched by using a HTTP GET or POST method:

http://ST_SERVER_ADDR:7771/report/usage.jsp?start=FromDate&end=EndDate

The FromDate and EndDate need to be in the following format: YYYYMMDD

5 Redundancy and load-balancing

You can use multiple servers in one local network to achieve redundancy and load balancing. You need to create a shared network directory for this purpose.

To add the redundancy and load balancing feature:

1. Change the <dataDir> tag in file <STWPS HOME>/local/system.ini to point to the shared network directory.
2. Change the <host> tag in file <STWPS HOME>/local/system.ini to point to the IP address of the local computer.
3. Copy file <STWPS HOME>/default/sysmgr.ini to the shared directory

In your front-end website, you can use the logic shown below to create a conference launching mechanism, or use the PHP sample in the Appendix to do that:

1. Create a web page that includes a similar form like the one below for entering a conference room. The form action points to a web page that you will create follow the second step:

```
<FORM method="post" action="http://FRONT-END-WEBSITE/conf-login.php">
  <table>
    <tr>
      <td><FONT face="Verdana" size="-1">Conference Name:</FONT></td>
      <td><input name="conference" value=""></td>
    </tr>
    <tr>
      <td><FONT face="Verdana" size="-1">Password:</FONT></td>
      <td><input type="password" name="conf_password" value=""></td>
    </tr>
    <tr>
      <td><FONT face="Verdana" size="-1">My name:</FONT></td>
      <td><input name="username" value=""></td>
    </tr>
    <tr>
      <td colspan=2>
        <center><br> <input type="submit" name="Submit" value="Enter"></center>
      </td>
    </tr>
  </table>
</FORM>
```

2. Create a page as the action destination of the above login form. The page includes the logic below:

```
/* Get the server that is assigned to handle the submitted conference room */
String $ips[n] = {serverIP1, serverIP2, ... , serverIPn};
for (i=1; i<n; i++){
  send HTTP request http://\$ips\[i\]:7771/getRunTimeServer.jsp?room=\$conference ;
  if(timeout) continue;
  else{
    String $assignedIP = HTTP response;
    break;
  }
}
```

```
/*Request the conference launching code from the assigned server*/  
Send HTTP request http://\$ips\[il:7770\]/wplogin.jsp?conference=\$conference&user-  
name=\$username&conf-password=\$conf\_password ;
```

```
Return the HTTP response to client;
```

6 Appendix

6.1 A PHP example of the conference launching

The sample PHP page below is for launching a SmileTiger WebPresentation client when redundancy and load-balancing mechanism is in use.

```
<html>
<head><title> Launching SmileTiger WebPresentation ... </title></head>
<body>
Launching SmileTiger WebPresentation ...
<?php
    error_reporting(0);

    $conference="Demo"; //accept from a login page or link
    $conferenceSID="pass"; //accept from a login page or link
    $language=""; //accept from a login page or link
    $userName="steve"; //accept from a login page or link

    $adminPort=7771;
    $servicePort=7770;
    $ips = array("192.168.7.5", "192.168.7.8"); //IP address of the server machines

    for($i=0;$i<2;$i++){
        $fp = fopen ("http://" . $ips[$i] . ":" . $adminPort .
            "/getRunTimeServer.jsp?room=" . urlencode($conference) , "r");
        if($fp){
            $assignedIP=trim(fgets($fp, 256));
            if($assignedIP == "") continue;
        }else{
            continue;
        }
    }

    $fp = fopen ("http://" . $assignedIP . ":" . $adminPort . "/getSessionID.jsp" ,
        "r");
    if($fp)
        $session=trim(fgets($fp, 256));
    else
        $session="";

    $assignedAddr = $assignedIP . ":" . $servicePort;
?>

<OBJECT id="WebPresentation" height="1" width="1"
    codeBase="http://<? print $assignedAddr ?>/wp/wpax.cab#version=5,3,0,0"
    classid="CLSID:7783A65D-275D-42FB-8CBF-722E42003FB6">
    <PARAM NAME="appServer" VALUE="<? print $assignedAddr ?>">
    <PARAM NAME="confServer" VALUE="<? print $assignedAddr ?>">
    <PARAM NAME="conference" VALUE="<? print $conference ?>">
    <PARAM NAME="conferenceSID" VALUE="<? print $conferenceSID ?>">
    <PARAM NAME="language" VALUE="<? print $language ?>">
    <PARAM NAME="userName" VALUE="<? print $userName ?>">
    <PARAM NAME="style" VALUE="default">
    <PARAM NAME="csprotocol" VALUE="auto">
    <PARAM NAME="session" VALUE="<? print $session ?>">
    <PARAM NAME="AddShortcuts" VALUE="false">
</OBJECT>

</body>
</html>
```