



# Sametime Links 3.0 Toolkit

*Community Services API*

**Developer's Guide and Reference**

---

# Copyright and Trademark Information

## Disclaimer

THIS DOCUMENTATION IS PROVIDED FOR REFERENCE PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS DOCUMENTATION, THIS DOCUMENTATION IS PROVIDED "AS IS" WITHOUT ANY WARRANTY WHATSOEVER AND TO THE MAXIMUM EXTENT PERMITTED, IBM DISCLAIMS ALL IMPLIED WARRANTIES, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE SAME. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION, DIRECT, INDIRECT, CONSEQUENTIAL OR INCIDENTAL DAMAGES, ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS DOCUMENTATION OR ANY OTHER DOCUMENTATION. NOTWITHSTANDING ANYTHING TO THE CONTRARY, NOTHING CONTAINED IN THIS DOCUMENTATION OR ANY OTHER DOCUMENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF THE APPLICABLE LICENSE AGREEMENT GOVERNING THE USE OF THIS SOFTWARE.

## Copyright

Under the copyright laws, neither this documentation nor the software may be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form, in whole or in part, without the prior written consent of IBM Corporation, except in the manner described in the documentation or the applicable licensing agreement governing the use of the software.

© Copyright 2002 IBM Corporation

Lotus Software, IBM Software Group  
One Rogers Street  
Cambridge, MA 02142  
All Rights Reserved. Printed in the United States.

## Trademarks

AIX, DB2, Domino, IBM, Lotus, Lotus Notes, LotusScript, Notes, Sametime, TeamRoom, and WebSphere are trademarks or registered trademarks of Lotus Development Corporation and/or IBM Corporation in the United States, other countries, or both. Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. Microsoft, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation in the United States, other countries, or both. Other company, product, and service names may be trademarks or service marks of others.

---

---

# Table of Contents

<b>About This Guide .....</b>	<b>v</b>
<b>Chapter 1 Introducing Sametime Links .....</b>	<b>1</b>
<b>Overview .....</b>	<b>1</b>
<b>Sametime Links Functionality .....</b>	<b>1</b>
<b>Sametime Links Sample Pages .....</b>	<b>2</b>
<b>Chapter 2 Getting Started .....</b>	<b>3</b>
<b>Preparing the Page for the Addition of Sametime Links.....</b>	<b>3</b>
<b>Adding a Sametime Link .....</b>	<b>4</b>
<b>Using the Sample Page .....</b>	<b>5</b>
<b>Enabling the Sample .....</b>	<b>5</b>
<b>Error Messages.....</b>	<b>6</b>
<b>Using Sametime Links in Frames.....</b>	<b>6</b>
<b>Chapter 3 Other Sametime Links Elements.....</b>	<b>7</b>
<b>Online Awareness Icons .....</b>	<b>7</b>
<b>Change Status Links .....</b>	<b>7</b>
<b>Place-based Awareness .....</b>	<b>7</b>
<b>Preparing the Page for the Addition of Place-based Awareness .....</b>	<b>9</b>
<b>Adding a Place Counter .....</b>	<b>9</b>
<b>Adding a Place-based Awareness Link.....</b>	<b>10</b>
<b>Entering and Leaving a Place.....</b>	<b>10</b>

<b>Chapter 4</b>	<b>Advanced Features and Customization .....</b>	<b>11</b>
	<b>Advanced Sametime Links API Functions.....</b>	<b>11</b>
	<b>Customizing the Style of Sametime Links .....</b>	<b>12</b>
	<b>Copying Sametime Dialog Pages to a Different Location.....</b>	<b>12</b>
	<b>Customizing Sametime Dialogs.....</b>	<b>12</b>
	<b>Handling Events.....</b>	<b>13</b>
<b>Chapter 5</b>	<b>Enabling Your Domino Applications .....</b>	<b>14</b>
	<b>Using Tokens .....</b>	<b>14</b>
	<b>Locating the Sametime Server.....</b>	<b>14</b>
	<b>Using the Sample Domino Application .....</b>	<b>15</b>
	<b>Enabling Your Template.....</b>	<b>16</b>
	<b>Step 1: Prepare Your Template .....</b>	<b>16</b>
	<b>Step 2: Enable a Form .....</b>	<b>17</b>
	<b>Deploying your application on a Domino server .....</b>	<b>18</b>
<b>Chapter 6</b>	<b>Sametime Links JavaScript API Reference.....</b>	<b>20</b>
	<b>Functions .....</b>	<b>20</b>
	<b>openPlaceChat.....</b>	<b>22</b>
	<b>openPlaceWin .....</b>	<b>22</b>
	<b>openStatusWindow .....</b>	<b>22</b>
	<b>prepareSametimeLink .....</b>	<b>22</b>
	<b>setSTLinksURL .....</b>	<b>23</b>
	<b>STLinksAddToContactList.....</b>	<b>24</b>
	<b>STLinksAudioEnabled.....</b>	<b>24</b>
	<b>STLinksChangeMyStatus.....</b>	<b>25</b>
	<b>STLinksCreateMeeting .....</b>	<b>25</b>

<b>STLinksCreateIM.....</b>	<b>25</b>
<b>STLinksEnterPlace .....</b>	<b>26</b>
<b>STLinksGetPrivateGroups .....</b>	<b>26</b>
<b>STLinksLeavePlace .....</b>	<b>26</b>
<b>STLinksLogin .....</b>	<b>27</b>
<b>STLinksLogout.....</b>	<b>27</b>
<b>STLinksMeetingsEnabled .....</b>	<b>27</b>
<b>STLinksMyStatus .....</b>	<b>28</b>
<b>STLinksMyStatusMessage.....</b>	<b>28</b>
<b>STLinksResetList.....</b>	<b>28</b>
<b>STLinksVideoEnabled .....</b>	<b>28</b>
<b>STLinksWatchGroups .....</b>	<b>28</b>
<b>STLinksWatchUsers .....</b>	<b>29</b>
<b>writePlaceCounter .....</b>	<b>29</b>
<b>writeSametimeLink .....</b>	<b>29</b>
<b>writeSTLinksApplet .....</b>	<b>31</b>
<b>Events.....</b>	<b>33</b>
<b>STLinksAddToContactListFailed .....</b>	<b>33</b>
<b>STLinksAppletStarted .....</b>	<b>33</b>
<b>STLinkClicked .....</b>	<b>34</b>
<b>STLinksLoggedIn.....</b>	<b>34</b>
<b>STLinksLoggedOut.....</b>	<b>34</b>
<b>STLinksPrivateGroupsFailed.....</b>	<b>35</b>
<b>STLinksPrivateGroupsReceived .....</b>	<b>35</b>
<b>STLinksUserEnteredPlace .....</b>	<b>35</b>

<b>STLinksUserLeftPlace .....</b>	<b>36</b>
<b>STLinksUserStatusChanged .....</b>	<b>36</b>
<b>Appendix A Setting the HTTP Tunneling Address and Port.....</b>	<b>38</b>

---

---

# About This Guide

## Intended Audience

This guide is intended for Web authors who want to enable their Web pages and Web applications with Sametime® awareness and real-time collaboration. You should have basic familiarity with Web authoring to use this manual.

## Requirements

The IBM® Lotus® Sametime Links Toolkit 3.0 is targeted for use with the Sametime server 2.5 and above. Using the toolkit with a Sametime server 2.5 requires a server addon installation available from the Web ([www.lotus.com/sametimedevolvers](http://www.lotus.com/sametimedevolvers)). The toolkit is already installed on Sametime server 3.0.

## How to Use This Guide

This guide contains the following main sections.

Chapter	Title	Description
Chapter 1	Introducing Sametime Links	Provides an overview of Sametime links and the functionality they provide, and the installation and contents of the toolkit
Chapter 2	Getting Started	Describes basic Sametime links usage, including how to add Sametime Links to Web pages and how to use the sample page provided in the toolkit
Chapter 3	Other Sametime Links Elements	Covers Online Awareness icons, Change Status links, and place-based awareness
Chapter 4	Advanced Features and Customization	Describes advanced API functions, link style customization, working with Sametime dialogs, and event handling
Chapter 5	Enabling Your Domino™ Applications	Covers token authentication and locating servers, describes the sample Domino application provided in the toolkit, and shows how to enable a Domino template
Chapter 6	Sametime Links JavaScript API Reference	Provides a reference to the functions available in the Sametime Links API

## Guide Conventions

The following conventions are used in this guide:

- Sample code is in the `Courier New` font.
- Sample code that has been added to a previous sample step is in bold **Courier New**.

## Additional Information

Additional information can be found at the following Web sites:

<http://www.lotus.com/sametime>

<http://www.lotus.com/sametimedevolvers>

## Related Documents

Introducing the Sametime Links Toolkit (white paper)

---

---

# Chapter 1 Introducing Sametime Links

## Overview

Sametime Links is a toolkit that allows Web developers to Sametime-enable their Web pages and applications with "live names." A simple HTML / JavaScript API allows Web developers to turn existing names into Sametime links by simply adding a few lines of HTML code without affecting the layout of the page. While rich in functionality, the toolkit is light in size – using an embedded applet of only about 20K.

Additional information about Sametime is available at <http://www.lotus.com/sametime>.

## Sametime Links Functionality

The following table provides an overview of the features provided by the Sametime Links Toolkit.

Feature	Description
Simple API	Using any Web design tool, existing names in a Web page are turned into Sametime links simply by adding a few lines of HTML code. No programming or layout changes are required to enable a Web page.
Light weight	The total size of the applet embedded in the Web page/application is approximately 20K.
Rich feature set	Awareness with status icons, instant messages, and N-way chat, launch instant meetings, set status, chat rooms. See details below.
No installation	The integration with the site/application should be transparent to the end user. No installation or code that requires execution privileges (signed applet).
Extranet support	Uses HTTP to communicate with the Sametime server for connectivity (works through firewalls).
Authentication	Supports transparent authentication by access token. No need to challenge users that have already logged on to the Web application.
Encryption	The same level of chat encryption provided by other Sametime clients.
Supported Platforms	Windows® 95/98/NT/2000, with MSIE 5 and above (MSIE 6 with the Microsoft JVM), and Netscape 4.7x. On other browsers/platforms, the Sametime link will appear as regular HTML text.
Customizable	The user interface is HTML and is fully customizable.

Feature	Description
Localized UI	Localized for the full set of languages offered by Lotus Software in its other client applications.
Scalable	The Sametime server supports very large numbers of Sametime Links users.
Advanced JavaScript API.	The advanced Sametime JavaScript API allows you to access additional Sametime functionality, customize the behavior of Sametime Links, and customize the look and feel of the Sametime Links.

## Sametime Links Sample Pages

The Sametime Links 3.0 Toolkit includes sample pages enabled with Sametime links. To access the samples, open the toolkit home page in your browser. Select the Samples link to go to the Samples page. The toolkit home page is located at <http://<sametimeserver>/sametime/toolkits/st30linkstk/index.html>

If you are using Sametime server 3.0, you can go to the toolkit home page by selecting the Toolkit link in the Sametime server home page. In the toolkits page, select the Sametime Links Toolkit link.

**Note** The toolkit also includes a simpler sample page that is easier to use as a reference for enabling your pages; see the instructions for using the sample page in Chapter 2.

---

## Chapter 2 Getting Started

To add Sametime links anywhere in an HTML page, first prepare the page by including the necessary files and embedding the hidden applet.

### Preparing the Page for the Addition of Sametime Links

1. Include the necessary files by adding the following HTML code to the Head HTML section:

```
<LINK REL=STYLESHEET HREF="codebase/stlinks.css" TYPE="text/css">
<SCRIPT src="codebase/stlinks.js"></SCRIPT>
<SCRIPT>
setSTLinksURL("codebase");
</SCRIPT>
```

The codebase is the URL of the directory where the Sametime Links runtime package is installed (<http://<sametimeserver>/sametime/stlinks>, where <sametimeserver> is your Sametime server host name).

**Note** You can use Sametime links to enable Web pages or Web applications that reside on a separate server. For additional considerations, see the description of `setSTLinksURL` in Chapter 6.

**Note** If you are using a Sametime server version 2.5 that was configured to force HTTP tunneling on port 80, you may have to configure Sametime Links to work over port 80. See Appendix A for the details.

2. Add the code that puts the hidden Sametime Links Java™ applet on the page. This code can be put anywhere in the HTML code for your page because the applet itself is placed on a hidden, zero-size HTML layer.

```
<SCRIPT>
writeSTLinksApplet (loginName, key, isByToken);
</SCRIPT>
```

In the `writeSTLinksApplet` call

- The *loginName* argument is the login name of the user.
- The *key* argument is the password or the token. Sametime Links provides three methods for logging in to a Sametime community: log in by password, log in as anonymous and log in by token. See the description of `writeSTLinksApplet` in Chapter 6 for a detailed discussion of these methods.
- The *isByToken* argument is true if you use a token to authenticate and false if you use a password. The default is false.

## Adding a Sametime Link

Once you have prepared the page for adding Sametime Links, you add a Sametime link anywhere in the HTML text by typing the following:

```
<script>writeSametimeLink(userName, displayName, bResolve,  
                          options)</script>
```

In the `writeSametimeLink` call:

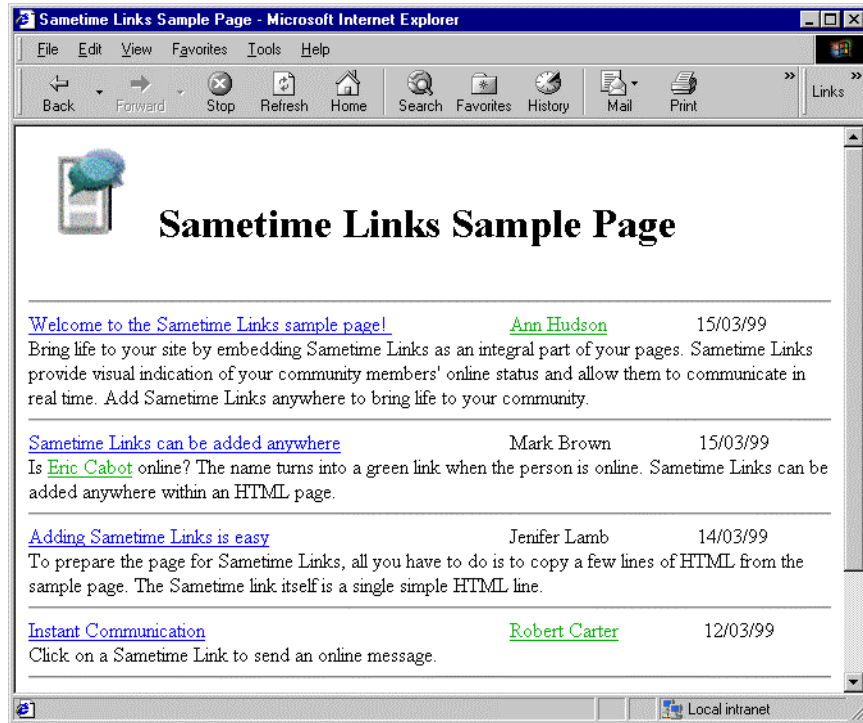
- The *userName* argument is the unique user name. To ensure uniqueness, use the canonical name, or the distinguished name if you are using an LDAP directory. However, note that Sametime server 2.5 cannot resolve distinguished names. Therefore, if you are using Sametime server 2.5 with LDAP, use the user's common name.
- The *displayName* argument is the display name of the user. This name is displayed as the text of the link.
- The *bResolve* argument is true if the Sametime server has to resolve the user name. False indicates that the user name is already resolved.
- The *options* argument is a string of semicolon-delimited display options. Each element in the list has the format "*option:value*." The options allow you to change the rendering and the behavior of the Sametime link. For example, by specifying "text:yes;icon:no" you create a Sametime link without a status icon. See the description of `writeSametimeLink` in Chapter 6 for the list of options.

To improve performance, use resolved names (`bResolve` set to false). By using the resolved name, you save the time it takes for the server to resolve the name, and the user status is updated faster when the Web page is displayed. For LDAP, request the server to resolve the name (`bResolve` set to true).

**Note** For Lotus Notes® directories, resolved names are names in the canonical form (for example, `CN=Reuben D. Smith/OU=Ottawa/O=Acme/C=CA`).

## Using the Sample Page

The HTML page “page.html” is provided as a sample. It is located in the subfolder “simple” under the folder “samples.” It demonstrates how Sametime Links can be embedded in a message board page. The sample page is shown below:



## Enabling the Sample

3. In page.html, replace the three appearances of `<your sametime server>` with the host name of the Sametime server. For example, if the host address of the Sametime server is `sametime.acme.com`, the head section of the page should be as follows:

```
<HEAD>
<TITLE>Sametime Links Sample Page</TITLE>
<LINK REL=STYLESHEET
  HREF="http://sametime.acme.com/sametime/stlinks/stlinks.css"
  TYPE="text/css">
<SCRIPT
  src="http://sametime.acme.com/sametime/stlinks/stlinks.js">
</SCRIPT>
<SCRIPT>
  setSTLinksURL("http://sametime.acme.com/sametime/stlinks");
</SCRIPT>
</HEAD>
```

4. In order for the Sametime links to show online users, change the Sametime link parameters “name1” through “name6” to the names of people that are registered in your directory. For example, replace:

```
writeSametimeLink("Name1")
```

with

```
writeSametimeLink("Ann Banks")
```

5. Replace the login name and password in the writeSTLinksApplet call:

```
writeSTLinksApplet("<login name>", "<password>", false)
```

with the login name and password of a registered user.

6. To see a Sametime link show an online user, log in to Sametime Connect using one of the names that you used for the Sametime links. The name will turn green. You can click the name to send an online message.

## Error Messages

To avoid interrupting visitors to the site, no error message is displayed if the Sametime login process fails for any reason. If an error occurs, the Sametime links are displayed as offline. In particular, no message is displayed if the Sametime link name is not a name of a registered user, or the name resolves to more than one name in the directory.

If the user is logged in successfully and later disconnected, no error message is displayed unless there is an open instant message or N-way chat window.

If you want to display a message when the user is logged out, you can handle the Logged Out event. See Chapter 4 for instructions on how to handle events.

## Using Sametime Links in Frames

You can use Sametime links in multiple frames on a Web page. Enable the page in each frame separately, following the usual process for enabling a page with Sametime links. Although each frame with Sametime links includes an instance of the Sametime Links applet, these instances communicate with each other and share communication with the Sametime server, so performance is not affected. Also, since the browser keeps the applet CAB or JAR file in the memory cache for the duration of the session, applet download is performed only once per session. Browsers also keep the applet in the disk cache, so that if you visit pages with Sametime links often, you may not need to download the applet from the server again.

---

## Chapter 3 Other Sametime Links Elements

### Online Awareness Icons

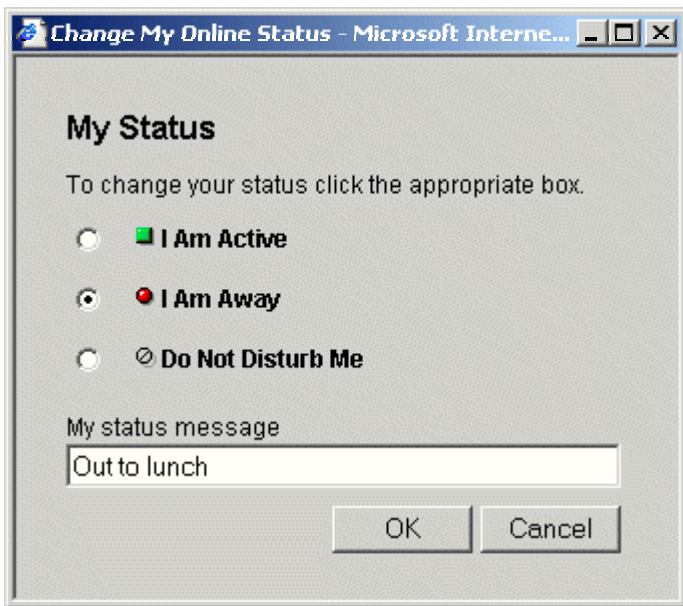
The Sametime Links API allows you to place clickable awareness icons on your page instead of turning the names themselves into Sametime links. Use these icons if you want the name to be a link with another action (for example, send mail). This option also allows you to separate the name from the icon. For example, you might want to display names and status icons in separate table columns.

By default, both the icon and the user name for a Sametime link are displayed and clickable. To display only the clickable icon, specify the option “text:no” in the writeSametimeLink call.

### Change Status Links

To add a link that opens the Change Status window, add a JavaScript link:

```
<a href="JavaScript:openStatusWindow()">...</a>
```



### Place-based Awareness

Place-based awareness in Sametime allows you to be aware of people who are in the same virtual place with you, and to collaborate in real time with people in the place. The place is identified by its name, and therefore the application gives the virtual place a meaning by assigning the place

name. For example, if you want to be aware of people that are in the same Web page, use the URL as the place name.

Some examples of virtual places are:

- A Web page
- A Web site
- A document
- A discussion forum

Sametime Links provides three ways to enrich your Web application with place-based awareness:

- A *place window* displays a list of people in the place.
- A *place counter* displays the current number of people in the place, formatted as an HTML link. The number is dynamically updated when people enter or leave the place. When a user clicks the link, a place window appears.
- A *place chat window* is a chat room associated with the place.



## Preparing the Page for the Addition of Place-based Awareness

To prepare the page for place-based awareness, follow the steps in "Preparing the Page for the Addition of Sametime Links," in Chapter 2.

## Adding a Place Counter

Add a place counter anywhere within the HTML text by typing the following:

```
<SCRIPT>writePlaceCounter('placeName', 'winTitle')</ SCRIPT>
```

In the writePlaceCounter call:

- The *placeName* argument is a unique string identifier of the place.
- The *winTitle* argument is the title of the place window that opens when the place counter is clicked.

## Adding a Place-based Awareness Link

Add a link that opens a place window or a place chat window by using a JavaScript link.

- To add a place window link:

```
<A HREF="JavaScript:openPlaceWin('placeName','winTitle')">...</A>
```

- To add a place chat window link:

```
<A HREF="JavaScript:openPlaceChat('placeName','winTitle')">...</A>
```

In the above calls:

- The *placeName* argument is a unique string identifier of the place.
- The *winTitle* argument is the title of the place window.

## Entering and Leaving a Place

If you added a place counter to the page, the user automatically enters the place when the page is loaded. However, if you added a link that launches a place window or a place chat window, the user only enters the place when the place window or the place chat window is opened.

If you add a link that launches a place window, it is recommended that you add code to the page that enters the user to the place when the page is loaded. This way, the place window will show all users that are currently viewing the page (or any other pages that are associated with this virtual place), and not just users that have the place window open.

To enter the place when the page is loaded, use the function `STLinksEnterPlace`. This function should be called after successful login. Add the following code to your page:

```
<SCRIPT>
function STLinksLoggedIn(myUserId, myUserName)
{
    STLinksEnterPlace('placeName');
}
</SCRIPT>
```

If you add a link that launches a place chat window, users should only enter the place when the place chat window is opened. Users that enter the place associated with the chat without opening the place chat window are listed as chat participants although they are actually not able to view the chat.

Sametime Links automatically takes the user out of the place when the user navigates out of the page.

---

## Chapter 4 Advanced Features and Customization

The Sametime Links Toolkit supports many more operations than the basic features described in the previous chapters. This chapter briefly describes advanced API function calls and discusses various Sametime Links customizations you can make.

### Advanced Sametime Links API Functions

Advanced Sametime Links API functions provide access to additional Sametime functionality. Lower-level API calls also allow you to customize the rendering and behavior of Sametime Links and place-based awareness.

Advanced API functions and events let you:

- Log in and log out - `STLinksLogin` and `STLinksLogout`
- Check and control the user's current status and view the status description- `STLinksChangeMyStatus`, `STLinksMyStatus`, and `STLinksMyStatusMessage`
- Monitor the online status of individual users and groups of users. - `STlinksWatchUsers`, `STLinksWatchGroups`, and `STLinksResetList`
- Handle user status change events - `STLinksUserStatusChanged` event
- Override the action taken when the user clicks a Sametime link - `STLinkClicked` event
- Create instant messages and instant meetings – `STLinksCreateIM` and `STLinksCreateMeeting`
- Add users to the Sametime Connect contact list - `STLinksAddToPrivateGroup`
- Enter and leave a place – `STLinksEnterPlace` and `STLinksLeavePlace`
- Handle place events – `STLinksUserEnteredPlace` and `STLinksUserLeftPlace`
- Check availability of multimedia services - `STLinksMeetingEnabled`, `STLinksAudioEnabled`, and `STLinksVideoEnabled`
- Generate HTML code that represents a Sametime link – `prepareSametimeLink`

For complete information about these functions, see their descriptions in Chapter 6, "Sametime Links JavaScript API Reference."

## Customizing the Style of Sametime Links

You can change the style used to display a Sametime link in online and offline mode. The styles are defined in the `stlinks.css` file. The three styles in this style sheet are:

- *online* is the style used to represent online names. By default, the online status is represented by bold green text.
- *offline* is the style used to represent offline names. The default offline style definition is empty, meaning that it inherits the style attributes of the parent HTML element.
- *iconSpace* is the style used for the space between the icon and the name.

You can also use other styles for offline and online modes by specifying the style names in the `options` parameter of the `writeSametimeLink` function. For example, if you add the option `"onlineStyle:MyOnline"` in the `writeSametimeLink` call, the Sametime link will be displayed with the `MyOnline` style when the person is online. By using this option, you can specify different styles for Sametime links on the same page.

## Copying Sametime Dialog Pages to a Different Location

The Sametime Links chat windows and other dialogs are all HTML pages. These HTML files are located in the `stlinks` directory on the Sametime server. If you would like to customize the dialogs, you can copy these files to another location or to another Web server. The Sametime Links CAB and JAR files must be loaded from the Sametime server to satisfy Java security requirements. All other Sametime Links files, including the CSS and JS files, can be copied to another Web server.

To create a copy of the Sametime Links files, create a new directory at the target location and copy the content of the `stlinks` directory (excluding the `toolkit` subdirectory, `hostInfo.js` and the CAB and JAR files) to the new directory.

To link your page to your copy of the Sametime Links pages, use the `docBase` parameter of the `setSTLinksURL` function:

```
setSTLinksURL (codeBase, language, docBase)
```

Pass the URL of the directory to which you copied the Sametime Links files as the `docBase` parameter.

## Customizing Sametime Dialogs

All Sametime Links dialogs are HTML pages. This allows you to easily customize the graphics; for example, you can change the background color or add your company's logo.

You can also add functionality to the dialogs or disable existing functionality. For example, you can add a link to the user profile of your IM partner in the IM window. In an extranet application, you might want to prevent users from browsing the directory or inviting others to the chat. To do this, simply remove the HTML code for the "Invite Others" button from the file `inputFrame.html`. Similarly, to prevent users from starting Sametime meetings, remove the code for the "Add Tools" button.

## Handling Events

The Sametime Links API provides a set of events. To handle one of the event types, add a function with the same name and parameters to your page. For example if you want to handle the `STLinksLoggedOut` event, you must add an `STLinksLoggedOut` function:

```
function STLinksLoggedOut(reason)
{
    alert(reason);
}
```

---

## Chapter 5 Enabling Your Domino Applications

This chapter describes the process for enabling a Domino database template with Sametime Links. The Sametime Links toolkit contains a sample Domino application, STLinksSample.nsf, that contains all the agents and script libraries you need to enable your template.

The agents do the following

1. Generate an access token for the user.
2. Locate the Sametime server.

### Using Tokens

Token authentication allows a user who has logged in to your application to log in to the Sametime server without being challenged to authenticate again. The application generates a token and passes it to Sametime Links.

The Sametime server 3.0 includes two separate security features capable of generating the authentication token used by Sametime:

- *Domino Single Sign-On (SSO) Authentication.* This authentication method uses Lightweight Third Party Authentication (LTPA) tokens. This authentication method is new in Sametime 3.0.
- *Secrets and Tokens authentication databases* – Sametime 2.5 uses only the Secrets and Tokens authentication databases to create authentication tokens. Sametime 3.0 supports Secrets and Tokens authentication in addition to LTPA tokens.

The sample application STLinksSample.nsf generates a token by first running the agent that looks for the LTPA token cookie. If the LTPA token cookie is not found, a Secrets and Tokens token is generated. This ensures that the application can successfully generate a token for both Sametime 3.0 and Sametime 2.5. This method also allows the application to authenticate with a Secrets and Tokens token if the Domino server uses a DSAPI filter for authentication instead of LTPA.

### Locating the Sametime Server

Sametime-enabled applications running on a non-Sametime server must locate a Sametime server, in order to link to the Sametime Links files on the Sametime server.

Instead of using a hard-coded address, you can create your application so that it finds the Sametime server address. To improve performance, the application should locate the user's home Sametime server or a geographically near Sametime server (a server in the user's LAN). Previous

versions of Sametime toolkits recommended that you use a LotusScript® agent to look in the Domino Directory for the user's person record and use the Sametime home server field in that record. This method currently works only with the Domino Directory. You cannot use it with LDAP directories or for anonymous access.

A more reliable, efficient, and flexible method is to deploy a special XML file, `hostAddress.xml`, on the application server. This file contains a near Sametime server location. This method requires some simple administrator intervention but uses less code, performs more efficient lookup, works with both Domino and LDAP directories, and works with both anonymous and authenticated access.

The format of this XML file is:

```
<?xml version="1.0" encoding="UTF-8" ?>
<sametime>
  <hostAddress>myhost.lotus.com</hostAddress>
  <httpPort>80</httpPort>
</sametime>
```

Note that the host address and the port are the address (DNS name) and port of the HTTP server associated with the Sametime server. These are the address and port that you use to access Web pages on the Sametime server. These are not the DNS name and port used for HTTP tunneling to the Sametime Community server. The HTTP tunneling address is provided automatically by Sametime Links, unless the Sametime server is a 2.5 Sametime server that was configured to allow HTTP tunneling on port 80. See Appendix A for more information.

The Sametime 3.0 server installation process generates this XML file, inserting the address of the Sametime server on which the toolkits are installed. It then stores the file in the "sametime" directory under the HTML root. If you are using Sametime server 2.5, you can create and store the file manually.

You must deploy this XML file to the same location (the sametime directory under the HTML root) on all your application servers. The application can then use HTTP to read the address of the Sametime server from this XML file.

The Sametime server whose address appears in the XML file is not necessarily the user's home server; however, it will redirect the user to the appropriate home server.

**Note** A Sametime 3.0 server updates the generated XML file automatically if the DNS name or the HTTP port of the server changes. If your application uses the file, the administrator must then re-deploy the updated file to your application servers.

The agent `SametimePopulateHostFields` in the `STLinksSample.nsf` sample application works with the XML file to locate the Sametime server. You can use this agent in your own applications. See "Using the Sample Domino Application," below.

## Using the Sample Domino Application

The sample application `STLinksSample.nsf` is located in the domino directory under the samples directory (...html\sametime\toolkits\st30linkstk\samples\domino).

To use this sample, copy it to the Domino Data directory. Open the database by navigating to the URL `http://<sametimeserver>/STLinksSample.nsf` in a browser. If login is successful, you should see your name turn green.

## Enabling Your Template

The following steps are required to enable an application or template with Sametime:

1. Prepare your template.
2. Enable a form.

### Step 1: Prepare Your Template

Copy the Sametime script libraries and agents from the `STLinksSample.nsf` sample database.

#### Copy the Sametime script libraries to your template

Copy the following script libraries from the `STLinksSample.nsf` database in the Script Libraries view and paste them into the same view in your template:

- `SametimeAgent`
- `SametimeStrings`

#### Copy the Sametime agents to your template

Copy the following agents from the `STLinksSample.nsf` database in the Agents view and paste them into the same view in your template:

- `SametimePopulateTokenLTPA`
- `SametimePrePopulateTokenST`
- `SametimePopulateTokenST`
- `SametimePopulateHostFields`

#### Sign the agents

Because of Sametime's security features, the agent must be signed by an authorized developer to run on a Sametime or Domino server. Your server administrator should grant you the required server access controls. Once you are granted proper access, sign the Sametime agents that you copied to your template.

1. Ask the server administrator to add your name to the "Run unrestricted LotusScript/Java Agents" list in the Agent Manager section of the server document for the Sametime or Domino server on which you want to put the application.

2. Choose agents in the enabled database's client menu. Double-click the agent that you want to sign.
3. Save the agent. The agent is now signed.

To verify that the agent was signed successfully:

1. Right-click the agent and choose Properties.
2. Click the Fields tab and check that the \$UpdatedBy field contains your name.

You might need to repeat the agent-signing procedure for each new server on which the template will be installed.

## Step 2: Enable a Form

To turn names in a form into Sametime Links, do the following:

1. Add three hidden computed-for-display text fields. Name them HTTP\_Cookie, Server\_name, and Server\_Port, after the CGI variables that are used by the agents. Specify an empty string as the value formula for each field. Domino will place the CGI variable values in the fields automatically.
2. Add the following lines to the WebQueryOpen event of the form:

```
@Command([ToolsRunMacro]; "SametimePopulateHostFields");
@Command([ToolsRunMacro]; "SametimePopulateTokenLTPA");
@if((SAMETIME_TOKEN=""));
    @Do(@Command([ToolsRunMacro]; "SametimePrePopulateTokenST");
        @Command([ToolsRunMacro]; "SametimePopulateTokenST"));
```

- Use Pass-Thru HTML to add the HTML lines required to prepare the page for Sametime Links. Use computed-for-display text fields for the Sametime server address, the token, and the login name. Mark the whole paragraph as Pass-Thru HTML.

```
<link rel=stylesheet
href="[SAMETIME_HOST_1]/sametime/stlinks/stlinks.css"
type="text/css">
<script
src="[SAMETIME_HOST_2]/sametime/stlinks/stlinks.js"></script>
<script>setSTLinksURL("[SAMETIME_HOST_3]/sametime/stlinks")
</script>
<script>writeSTLinksApplet("[loginName]", "[SAMETIME_TOKEN]");
</script>
```

In the HTML, SAMETIME\_HOST\_1, SAMETIME\_HOST\_2, SAMETIME\_HOST\_3, SAMETIME\_TOKEN and loginName are all computed-for-display text fields.

- Specify an empty string as the value formula for the fields SAMETIME\_HOST\_1, SAMETIME\_HOST\_2 and SAMETIME\_HOST\_3. The agent SametimePopulateHostFields places the address of the Sametime server in these fields.
- Specify an empty string as the value formula of the SAMETIME\_TOKEN field. The agents place the token value in this field.
- Specify the value formula “@UserName” for the loginName field.

**Tip** You can copy the subform SametimeSharedHeader from STLinksSample.nsf to your template and embed the subform in all the forms that you want to enable with Sametime links. This subform contains the three fields for the CGI variables and the HTML lines needed to prepare the page for Sametime links.

- To turn a name that appears in a name field into a Sametime link, wrap the field with a writeSametimeLink JavaScript call:

```
<script>writeSametimeLink("[THE FIELD]")</script>
```

Mark the text, starting with the <script> tag and ending with the </script> tag, as Pass-Thru HTML.

To ensure uniqueness, use the canonical format of the user name. To do this, use two fields, one with the canonical name and one with the display name:

```
<script>writeSametimeLink("[CANONICAL_NAME]",
"[DISPLAY_NAME]")</script>
```

## Deploying your application on a Domino server

To use your application on another Domino server, you need to deploy the file hostAddress.xml on that server. This XML file is used by the agent SametimePopulateHostFields to locate the

Sametime server. Note that you don't need to deploy this file, if the address of the Sametime server is hard coded in your application or your application uses another method for locating the Sametime server instead of using the agent SametimePopulateHostFields.

The agent reads the file from the URL `http://<server>/sametime/hostAddress.xml` (where `<server>` is the host name of the Domino server that runs the application). To deploy the file on the Domino server do the following:

1. Create a directory called "sametime" under the HTML root of the Domino server.
2. Put the file `hostAddress.xml` in the `sametime` directory:
  - If you are using a Sametime 3.0 server, copy the file `hostAddress.xml` from the Sametime server. The file is located in the `sametime` directory under the HTML root.
  - If you are using Sametime 2.5, create a file called `hostAddress.xml` in the `sametime` directory. The content of the file should be the following (replace `mysametime.mycompany.com` with the host name of your Sametime server):

```
<?xml version="1.0" encoding="UTF-8" ?>
<sametime>
    <hostAddress>mysametime.mycompany.com</hostAddress>
    <httpPort>80</httpPort>
</sametime>
```

---

## Chapter 6 Sametime Links JavaScript API Reference

This chapter describes the functions and events included in the Sametime Links JavaScript API. Each section provides an alphabetical list of components, followed by descriptions of every component.

### Functions

The SametimeLinks JavaScript API provides the following functions.

- `openPlaceChat`
- `openPlaceWin`
- `openStatusWindow`
- `prepareSametimeLink`
- `setSTLinksURL`
- `STLinksAddToContactList`
- `STLinksAudioEnabled`
- `STLinksChangeMyStatus`
- `STLinksCreateMeeting`
- `STLinksCreateIM`
- `STLinksEnterPlace`
- `STLinksGetPrivateGroups`
- `STLinksLeavePlace`
- `STLinksLogin`
- `STLinksLogout`
- `STLinksMeetingsEnabled`
- `STLinksMyStatus`

- STLinksMyStatusMessage
- STLinksResetList
- STLinksVideoEnabled
- STLinksWatchGroups
- STLinksWatchUsers
- writePlaceCounter
- writeSametimeLink
- writeSTLinksApplet

## openPlaceChat

```
openPlaceChat(placeName, winTitle)
```

### Description

This function launches a place chat window.

### Parameters

<i>placeName</i>	A string identifying the place.
<i>winTitle</i>	The title of the place chat window.

## openPlaceWin

```
openPlaceWin(placeName, winTitle)
```

### Description

This function launches a place window that shows the list of people in the place.

### Parameters

<i>placeName</i>	A string identifying the place.
<i>winTitle</i>	The title of the place window.

## openStatusWindow

```
openStatusWindow()
```

### Description

This function opens the Change Status window.

## prepareSametimeLink

```
prepareSametimeLink(userName, displayName, bResolve, options)
```

### Description

This function generates and returns the HTML code that represents a Sametime link. Unlike `writeSametimeLink`, this function does not write the HTML code to the document. You can use a `document.write` call to place this code on the page. Use this function if you are dynamically creating some HTML code that includes a Sametime link.

## Parameters

<i>userName</i>	The unique user name.
<i>displayName</i>	The display name of the user displayed as the text of the link. (Default is <i>userName</i> .)
<i>bResolve</i>	If the value is true, the name should be resolved. If the value is false, the name is already resolved. (Default is true.)
<i>Options</i>	A string of semicolon-delimited display options. See the list of options in the description of <code>writeSametimeLink</code> in this chapter.

## setSTLinksURL

```
setSTLinksURL (codeBase, language, docBase)
```

### Description

This function sets the base URLs for Sametime Links Toolkit files and the preferred language version. This function should be called before calling other Sametime Links API functions.

### Parameters

<i>codeBase</i>	<p>The URL of the directory where the Sametime Links package is installed. The Sametime Links applet CAB and JAR files will be loaded from this directory.</p> <p>You can use Sametime links to enable Web pages or Web applications that reside on a separate application server. The application server does not need to be in the same DNS domain as the Sametime server. There are two exceptions:</p> <ul style="list-style-type: none"><li>• The Sametime server and the application server must be in the same security zone defined in your Internet Explorer settings. This means that either both servers are in the intranet zone, or both are in the internet zone.</li><li>• Sametime links are not fully functional when you access an enabled page as a local file. You must put the enabled page on a Web server and use the URL, not the local file system address, to access the page. When an enabled page is accessed as a local file, the Sametime links are active, but a JavaScript error is displayed when you click a Sametime link.</li></ul>
-----------------	---

*language*

Indicates the user's preferred language version for ST Links user interface dialogs and user messages.

The language is indicated by an ISO 639 language code with an optional ISO 3166 country or region code to specify a national or regional variant. For example, "zh" indicates Simplified Chinese and "zh\_TW" indicates Traditional Chinese.

If you specify a country or region code and Sametime does not have a language version for the national or regional variant, the code is ignored. For example, "en", "en\_UK", and "en\_US" all have the same result, because Sametime has one common English version.

The language encoding is the same encoding used in the Accept-Language CGI variable. To ensure that users get their preferred language version, pass the content of the Accept-Language CGI variable as the language parameter. The sample Domino applications provided with the Sametime Links Toolkit, STLinksSample.nsf and STLinksSample25.nsf, show how to do this in a Domino application.

The default language encoding is "en" (English). If you specify a language that is not supported by Sametime, English will be used.

*docBase*

The URL of the base directory from which the Sametime Links HTML files will be loaded. The default is the same as the codeBase.

## STLinksAddToContactList

```
STLinksAddToContactList(userName, group)
```

### Description

This function adds a person to a private group in the user's contact list.

### Parameters

*userName*

The unique user name.

*group*

The name of a private group in the contact list. If the contact list does not have a private group with that name, a new group is created.

## STLinksAudioEnabled

```
STLinksAudioEnabled()
```

### Description

This function checks the availability of audio services.

## Return Value

This function returns true if audio is enabled on the server or false if audio is not enabled.

## STLinksChangeMyStatus

```
STLinksChangeMyStatus(status, statusMessage)
```

### Description

Changes the user's online status and the associated status message.

### Parameters

<i>status</i>	The new status, a numeric constant. See the list of statuses in the description of the STLinksUserStatusChanged event.
<i>statusMessage</i>	The message for the new status, such as "I am in a meeting."

## STLinksCreateMeeting

```
STLinksCreateMeeting(userNames, tools, topic, inviteText)
```

### Description

This function creates an instant meeting with one or more users.

### Parameters

<i>userNames</i>	The list of unique invitee names separated by semicolons.
<i>tools</i>	The list of tool names separated by semicolons. The available tools are chat, audio, video, share, and whiteboard.
<i>topic</i>	The topic of the meeting.
<i>inviteText</i>	The invitation message.

## STLinksCreateIM

```
STLinksCreateIM(partnerName)
```

### Description

Use this function to create an IM with a user in your watch list. The partner name parameter should be a name that was added to your watch list by a call to writeSametimeLink.

## Parameters

*partnerName*                      The unique name of the partner.

## STLinksEnterPlace

STLinksEnterPlace(placeName, bDelayLeave)

### Description

Use this function to enter the indicated place.

### Parameters

*placeName*                      A string identifying the place.

*bDelayLeave*                      If the value is false, the user is taken out of the place immediately when the user navigates out of the page. If the value is true, Sametime Links waits 30 seconds before taking the user out of the place. The purpose of this delay is to give the user time to enter the place again after the new page loads. This way, when a user navigates between pages that enter the same virtual place, the user will remain in the place during navigation. The default value is false.

## STLinksGetPrivateGroups

STLinksGetPrivateGroups()

### Description

This function requests the list of private groups in the user's Sametime Connect contact list. If the request is successful, the result is returned in the STLinksPrivateGroupsReceived event. If the request failed, a STLinksPrivateGroupsFailed event is received.

## STLinksLeavePlace

STLinksLeavePlace (placeName)

### Description

Use this function to leave the indicated place.

### Parameters

*placeName*                      A string identifying the place.

## STLinksLogin

```
STLinksLogin(userName, key, isByToken)
```

### Description

This function logs in to the Sametime server. This function can be used to log in to the Sametime server after the page has been loaded. For example, using `STLinksLogout` and `STLinksLogin` you can add links to your page to allow the user to log out and log back in to the Sametime server. If you pass the `loginName` and `key` arguments to the `writeSTLinksApplet` function, you do not need to use this function to log in when the page loads. If you want to call `STLinksLogin` immediately after the page is loaded, implement the event handler `STLinksAppletStarted` to make sure that you don't try to call `STLinksLogin` before the applet has finished loading.

To log in as anonymous, pass an empty string as the `key` argument.

### Parameters

<i>userName</i>	The login name
<i>key</i>	The password or token
<i>isByToken</i>	This value is true if a token will be used for authentication or false if a password will be used. The default is true.

### See Also

`writeSTLinksApplet`

## STLinksLogout

```
STLinksLogout()
```

### Description

This function logs off the Sametime server.

## STLinksMeetingsEnabled

```
STLinksMeetingsEnabled()
```

### Description

This function checks the availability of Meeting Services.

### Return Value

This function returns true if meetings are enabled on the server or false if meetings are not enabled.

## STLinksMyStatus

`STLinksMyStatus()`

### Description

Returns the user's online status as a numeric constant. For details, see the description of the `STLinksUserStatusChanged` event.

## STLinksMyStatusMessage

`STLinksMyStatusMessage()`

### Description

Returns the user's online status message.

## STLinksResetList

`STLinksResetList()`

### Description

This function removes all users and groups from the watch list.

## STLinksVideoEnabled

`STLinksVideoEnabled()`

### Description

This function checks the availability of video services.

### Return Value

This function returns true if video is enabled on the server or false if video is not enabled.

## STLinksWatchGroups

`STLinksWatchGroups(groups)`

### Description

This function adds one or more public groups to the watch list.

### Parameters

*groups*                      List of unique group names separated by semicolons.

## STLinksWatchUsers

```
STLinksWatchUsers(users, bResolve)
```

### Description

This function adds one or more users to the watch list.

### Parameters

<i>users</i>	List of unique user names separated by semicolons.
<i>bResolve</i>	This value is true if the names should be resolved or false if the names should not be resolved.

## writePlaceCounter

```
writePlaceCounter(placeName, winTitle, bDelayLeave)
```

### Description

This function embeds a place counter on the page.

### Parameters

<i>placeName</i>	A string identifying the place.
<i>winTitle</i>	The title of the place window that opens when the counter is clicked.
<i>bDelayLeave</i>	If the value is false, the user is taken out of the place immediately when the user navigates out of the page. If the value is true, Sametime Links waits 30 seconds before taking the user out of the place. The purpose of this delay is to give the user time to enter the place again after the new page loads. This way, when a user navigates between pages that enter the same virtual place, the user will remain in the place during navigation. The default value is false.

## writeSametimeLink

```
writeSametimeLink(userName, displayName, bResolve, options)
```

### Description

This function embeds the Sametime link on the page.

To improve performance, specify a resolved name for the `userName` argument and set `bResolve` to false. You save the time it takes for the server to resolve the name, and the user status will be updated faster when the Web page is displayed. For LDAP, request the server to resolve the name

(bResolve set to true). Specifying a resolved (distinguished) name may result in an invalid link, since the resolved name format differs from server to server.

**Note** For Notes directories, resolved names are names in the canonical form (for example, CN=Reuben D. Smith/OU=Ottawa/O=Acme/C=CA).

## Parameters

<i>userName</i>	The unique user name.
<i>displayName</i>	The display name of the user. This name is displayed as the text of the link. The default is <i>userName</i> .
<i>bResolve</i>	This value is true if the name should be resolved or false if the name is already resolved. The default is "true."
<i>options</i>	A string of semicolon-delimited display options. (See "Options" below for details.)

## Options

The options parameter is a string of semicolon-delimited display options. Each element in the list has this format:

*option:value*

If an option is not specified, the default value of the option is used. If the options parameter is omitted, the default values of all the options are used.

The following table lists and describes the available options.

<i>icon</i>	A value of "yes" indicates that the Sametime link includes an online status icon. A value of "no" indicates that the Sametime link does not include a status icon. The default value is "yes."
<i>text</i>	A value of "yes" indicates that the Sametime link includes a text link (the display name). A value of "no" indicates that the Sametime link does not include a text link. The default value is "yes."
<i>onlineStyle</i>	The name of the style sheet that will be used to represent the name when the person is online. The default is "online."
<i>offlineStyle</i>	The name of the style sheet that will be used to represent the name when the person is offline. The default is "offline."
<i>offlineLink</i>	A value of "yes" indicates that the Sametime link is also clickable when the user is offline. If you use this option, you should override the default click behavior by providing a definition of the STLinksClicked event on your page. The default value is "no," which means that the link is clickable only when the user is online.

*iconSpace*            The number of space symbols separating the status icon from the link text. The default is 1.

## writeSTLinksApplet

```
writeSTLinksApplet(loginName, key, isByToken)
```

### Description

This function embeds the hidden Sametime Links applet on the page and provides the applet with the user's login name and password or token. If you don't want the user to be logged in immediately when the page loads, you can call `writeSTLinksApplet` with no parameters and use `STLinksLogin` to log in to the server at a later stage.

### Parameters

*loginName*            The login name of the user.

*key*                    The password or token.

*isByToken*            This value is true if a token will be used for authentication or false if a password will be used. The default is true.

Sametime Links provides three methods for logging in to a Sametime community:

- *Log in by password* enables logging in using a login name and password. The drawback of this method is that it requires challenging the user to authenticate even if the user has already logged in to the application server. This method is therefore mainly used for testing and at the initial stages of application development.
- *Log in as anonymous* lets anonymous users log in to the community. Anonymous login to a Sametime Community must be enabled by the administrator. To log in as anonymous, pass an empty string as the key argument to `writeSTLinksApplet`. If you pass an empty string as the `loginName` argument, the server assigns a name of the form "UserN/Guest", where N is a number. If you pass a name as the `loginName` argument, the name that the server assigns is "name/Guest". The prefix "User" and the suffix "Guest" can be changed by the administrator.

**Note**    Additional permissions for anonymous users might be required. For example, to allow Sametime Links to resolve names, the administrator should turn on the anonymous access setting "Users can type names (resolve users and groups) to add them to an awareness list." This setting is not required if you provide Sametime Links with resolved names (`bResolve` set to false in the `writeSametimeLink` call). For

place-based awareness, the administrator should turn on the anonymous access setting "Users can participate in meetings or enter virtual places."

- *Log in by token* enables token authentication, which allows a user who has already logged in to Domino or WebSphere® to log in to the Sametime server without being challenged to authenticate again. See Chapter 5 in this guide to learn how your application can generate a token and use it to log in to Sametime Links.

## Events

The Sametime Links JavaScript API provides the following events.

- `STLinksAddToContactListFailed`
- `STLinksAppletStarted`
- `STLinkClicked`
- `STLinksLoggedIn`
- `STLinksLoggedOut`
- `STLinksPrivateGroupsFailed`
- `STLinksPrivateGroupsReceived`
- `STLinksUserEnteredPlace`
- `STLinksUserLeftPlace`
- `STLinksUserStatusChanged`

### **STLinksAddToContactListFailed**

`STLinksAddToContactListFailed(reason)`

#### **Description**

This event occurs when the request to add a user to the contact list fails.

#### **Parameters**

*reason*                      Identifies the reason the attempt failed.

### **STLinksAppletStarted**

`STLinksAppletStarted()`

#### **Description**

This event occurs when the Sametime Links applet has been loaded and started. This event is useful if you need to call `STLinksLogin` immediately after the page loads. After the `STLinksAppletStarted` event occurs, you can safely call `STLinksLogin`. Note that `STLinksAppletStarted` occurs before the user is logged in to the Sametime server. Wait for the `STLinksLoggedIn` event before calling API functions that require the user to be logged in.

## STLinkClicked

```
STLinkClicked(userName, displayName, status, evt)
```

### Description

This event occurs when a Sametime link or an online awareness icon is clicked. By default, the Sametime Links API opens an instant message window when a link is clicked. You can override this behavior by providing a definition of `STLinkClicked` on your page.

### Parameters

<i>userName</i>	The unique user name as provided in the <code>writeSametimeLink</code> call.
<i>displayName</i>	The display name of the user.
<i>status</i>	A numeric constant indicating the current status of the user. See the list of status values in the description of <code>STLinksUserStatusChanged</code> in this chapter.
<i>evt</i>	The JavaScript Event object. It provides information on the click event, such as the mouse position.

## STLinksLoggedIn

```
STLinksLoggedIn(myUserId, myUserName)
```

### Description

This event occurs when the login process from the Sametime community was successfully performed.

### Parameters

<i>myUserId</i>	The user's ID.
<i>myUserName</i>	The display name.

## STLinksLoggedOut

```
STLinksLoggedOut(reason)
```

### Description

This event occurs when the logoff process from the Sametime community was successfully performed or when a login request was refused. To avoid interrupting visitors to the site, no error message is displayed.

## Parameters

*reason* The reason that the login attempt failed or the reason that the user logged off after successfully logging in:

- 0x0 – Logged out by calling STLinksLogout
- 0xFFFFFFFF - Network error
- 0x80000211 - Incorrect login
- 0x80000229 - Disconnected because you logged in from another computer

## STLinksPrivateGroupsFailed

`STLinksPrivateGroupsFailed(reason)`

### Description

This event occurs if the request for a list of private groups fails.

### Parameters

*reason* The reason the attempt failed.

## STLinksPrivateGroupsReceived

`STLinksPrivateGroupsReceived(groups)`

### Description

This event occurs when the list of private groups in the user's contact list is received.

### Parameters

*groups* The list of the names of the private groups, separated by semicolons.

## STLinksUserEnteredPlace

`STLinksUserEnteredPlace(userId, displayName, placeName)`

### Description

This event occurs when a user enters a place. The event is received only for places that were entered by calling `STLinksEnterPlace` or `writePlaceCounter`.

## Parameters

<i>userId</i>	The user's ID.
<i>displayName</i>	The display name.
<i>placeName</i>	A string identifying the place.

## STLinksUserLeftPlace

`STLinksUserLeftPlace(userId, displayName, placeName)`

### Description

This event occurs when a user has left a place. It is received only for places that were entered by calling `STLinksEnterPlace` or `writePlaceCounter`.

### Parameters

<i>userId</i>	The user's ID.
<i>displayName</i>	The display name.
<i>placeName</i>	A string identifying the place.

## STLinksUserStatusChanged

`STLinksUserStatusChanged(userId, displayName, status, statusMessage, groupName)`

### Description

This event occurs when the status of a user in the watch list changes. It is received for users that were added to the watch list either by placing a Sametime link on the page or by calling `STLinksWatchUsers` or `STLinksWatchGroups`.

### Parameters

<i>userId</i>	The user's ID.
<i>displayName</i>	The display name.
<i>status</i>	The new status, a numeric constant: <ul style="list-style-type: none"><li>• 0 – Offline</li><li>• 32 – Active</li><li>• 64 – Not using the computer</li><li>• 96 – Away</li></ul>

- 128 – Do not disturb (DND)
- 544 – Mobile Active
- 608 – Mobile Away

*statusMessage* The message for the new status, such as “I Am Away.”

*groupName* The name of the public group that was added by an STLinksWatchGroups call. Otherwise, groupName contains an empty string.

---

## Appendix A      Setting the HTTP Tunneling Address and Port

To allow working through firewalls, Sametime Links uses HTTP to communicate with the Sametime server. Sametime Links connects to the network address and port used by the Sametime server for HTTP tunneled connections to the community server.

In Sametime server 3.0, Sametime Links automatically finds the HTTP tunneling address and port. However, if you are using Sametime Links with a 2.5 Sametime server, you may need to manually set the tunneling port.

The file `hostInfo.js`, located in the `stlinks` folder, contains the HTTP tunneling port. In Sametime server 3.0 it also contains the HTTP tunneling network address, because Sametime server 3.0 allows the administrator to change the network address used for HTTP tunneling. This file is automatically included by Sametime Links, so you do not need to add code to your enabled page to include the file.

In Sametime server 3.0, `hostInfo.js` is automatically updated by the server whenever the administrator changes the HTTP tunneling network address or port. In Sametime 2.5, `hostInfo.js` is updated only during the installation of Sametime Links. If the administrator configures the server to force tunneling on port 80 after the installation of Sametime Links, you must change the port in `hostInfo.js`.

The default tunneling port on a 2.5 Sametime server is 8082. The default value in `hostInfo.js` on a 2.5 server is therefore:

```
var ST25_TUNNELING_PORT=8082;
```

If the administrator configures the server to force tunneling on port 80, you must change the port in `hostInfo.js` to 80.

For more information on HTTP tunneling on port 80, see the Sametime 2.5 administrator's guide.