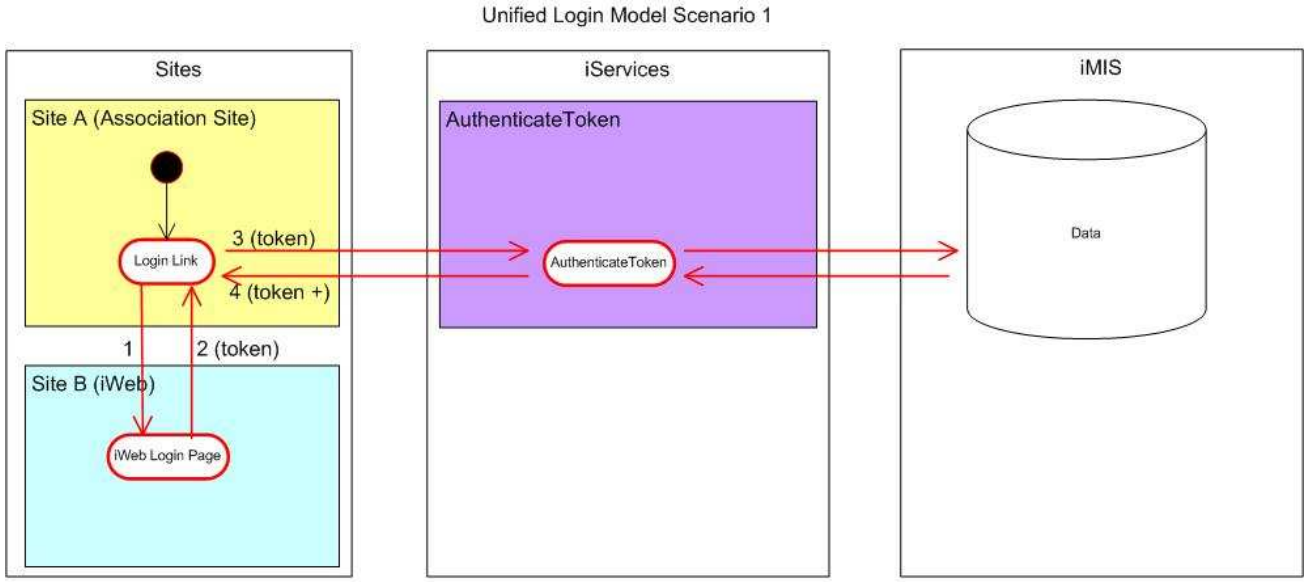
ISGweb for *iMIS* - Unified Login Model

As ISGweb is going to be embedded with the associations' websites, a Unified Login model is going to be implemented in order to let the associations share ISGweb's authentication and authorization schema. ISGweb's unified login process is carried out using web services. This process works regardless of the association's website technology. The website itself does not need to be able to consume web services, but only a token.

When an association invokes the ISGweb modules passing a token, ISGweb confirms that the user has an open session in ISGweb. If so, the user will access the invoked module according to its permissions; otherwise, the login window will appear to the user.

Unified Login Scenario 1: The following scenario applies where the authentication is performed by the ISGweb Login Page.



Scenario 1 - Site B (iWeb site) logs the user in and directs the user to Site A (association site) to then authenticate a token to create a unified session.

1.) The user goes to Site A (association site) and clicks the "login" link which opens a new browser window for Site B (iWeb site) displaying the login page. The "login" link on Site A (association site) will include a "ReturnPage" URL variable so that Site B (iWeb site) knows where to redirect the user and pass a unique session token after the login is successful.

2.) The user logs into Site B (iWeb) and a message displays to the user displaying they have successfully logged into the site. The user clicks the "Ok" button that closes Site B (iWeb site) window and refreshes the URL in the main browser window (originally Site A) to the URL passed in the "ReturnPage" URL variable with the Token as a URL Variable.

3.) Site A (association site) will need to track the Token URL variable. If there is not already a session and the Token URL variable exists, Site A (association site) will call the AuthenticateToken method in the Autentication Web Service and pass the unique Token.

4.) The AuthenticateToken method will validate the Token and return the results of a stored procedure XML format to the calling page in Site A.

5.) Site A (association site) can then maintain the values in Site A's user session.

* After sucessful login, all links from Site A (association site) that point to Site B (iWeb Site) will include the Token as a URL Variable.

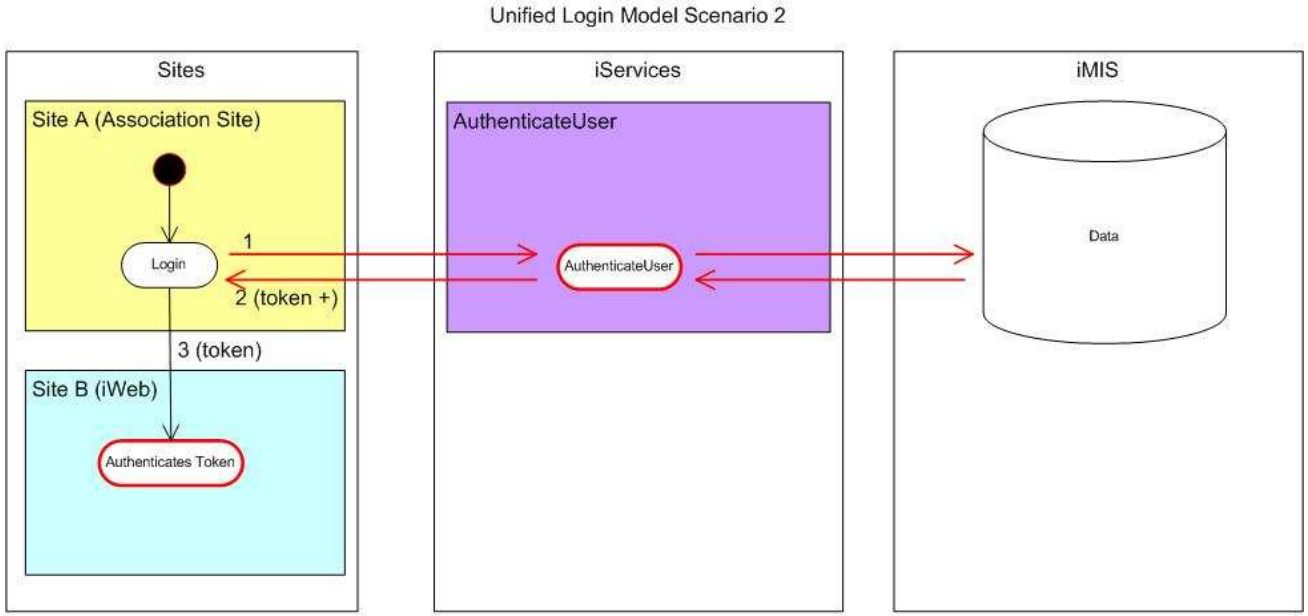
Authentication Web Service

Method	Inputs	Outputs	Description
1.) AuthenticateToken	Security Password, Token	Returns a string value with an XML packet	Verifies the existence of a user session.
2.) DeleteUserSession	Security Password, Token	Returns the deleted token	Deletes the user session.

Note: A security password is required when calling the Authentication Web Service.

Unified Login Scenario 2: The following scenario applies where the authentication is performed by the association site

The following string of XML is the return result from calling either the AuthenticateUser method or the AuthenticateToken method in the Authentication web service for either scenario 1 or 2:



Scenario 2 - Site A (association site) logs the user in and passes a token to Site B (iWeb site) to then authenticate the token to create a unified session.

1.) The user goes to Site A (association site) and submits the login. Site A (association site) passes the login information to the AutenticateUser method in the Authentication Web Service. The user is authenticated and a session is created generating a unique session token which is returned to the calling page. A set of predefined fields that relate to the logged in user can be returned in XML format.

2.) Site A (association site) will need to receive the Token and additional fields if defined to store as a session on the Site A (association site).

3.) When the user clicks a link on Site A (association site) to go to an iWeb page on Site B, the link will contain the Token as a URL variable. iWeb will validate the token and provide the user with access to that page.

* After sucessful login, all links from Site A (association site) that point to Site B (iWeb Site) will include the Token as a URL Variable.

Authentication Web Service

Method	Inputs	Outputs	Description
1.) AuthenticateUser	Security Password, Username, Password	Returns a string value with an XML packet	Creates a user session.
2.) DeleteUserSession	Security Password, Token	Returns the deleted token	Deletes the user session.

Note: A security password is required when calling the Authentication Web Service.

<iBridge xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"><User ID="9487"
TOKEN="7F1DEF6D-03C8-4387-A800-963ED61A8179" LAST_FIRST="SMITH,
JOHN" CO_ID="4627" MEMBER_TYPE="M"
MEMBER_TYPE_DESCRIPTION="Member"
EMAIL="jsmith@abc.org" SECURITY_GROUP="5"/>
</iBridge>

Authentication Web Service
The **Authentication** web service provides operations for authenticating a user in the ***iMIS*** database and implements the "Unified Login" architecture.

1.1 Structures
The **Authentication** web service does not require any structure.

1.2 Operations
The **Authentication** web service provides the following operations:

- **AuthenticateUser**
- **AuthenticateToken**
- **DeleteUserSession**

1.2.1 AuthenticateUser operation
The **AuthenticateUser** operation is used to log a user into the system using the "Unified Login" architecture. To authenticate a user, this method accomplishes the following steps:

- Verify whether the username and password passed as parameters exist in the ***iMIS*** database.
- Generate a new user session (TOKEN) for a specific ***iMIS*** account.
- Retrieve user information using the created user session (TOKEN).

1.2.1.1 Parameters
The **AuthenticateUser** operation requires three parameters in the following order:

Parameter name	Description	Type	Max. Length
Security Password	Use this parameter to specify the security password to consume this web service.	Alphanumeric	36
Username	Use this parameter to specify the user name.	Alphanumeric	60
Password	Use this parameter to specify the password.	Alphanumeric	60

1.2.1.2 Return values

In case of success, the **AuthenticateUser** operation returns a string value that includes a XML package with the following format:

```
<?xml version="1.0" encoding="UTF-16"?>
<iBridge xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        < Here will appear the result of the execution of the stored procedure associated to this method for retrieve user information after the creation of the user session>
</iBridge>
```

In case of an unsuccessful, it returns a string value that includes an error number and description as XML package with the following format:

```
<?xml version="1.0" encoding="UTF-16"?>
<iBridge xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Errors>
<Error Code="< Error Code >" Description="< Error Description >" />
</Errors>
</iBridge>
```

1.2.1.3 Usage

To authenticate a user using the "Unified Login" architecture, the consumer shall simply instantiate the **Authentication** web service and invokes its **AuthenticateUser** operation.

1.2.2 AuthenticateToken operation

The **AuthenticateToken** operation is used to verify the existence of a user session (TOKEN) passed as parameter.

1.2.2.1 Parameters

The **AuthenticateToken** operation requires two parameters in the following order:

Parameter name	Description	Type	Max. Length
Security Password	Use this parameter to specify the security password to consume this web service.	Alphanumeric	36
TOKEN	Use this parameter to specify the TOKEN to be authenticated.	Alphanumeric	36

1.2.2.2 Return values

In case of success, the **AuthenticateToken** operation returns a string value that includes a XML package with the following format:

```
<?xml version="1.0" encoding="UTF-16"?>
<iBridge xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        < Here will appear the result of the execution of the stored procedure associated to this method for retrieve user information>
</iBridge>
```

In case of an unsuccessful, it returns a string value that includes an error message and description as XML package with the following format:

```
<?xml version="1.0" encoding="UTF-16"?>
<iBridge xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Errors>
<Error Code="< Error Code >" Description="< Error Description >" />
</Errors>
</iBridge>
```

1.2.2.3 Usage

To authenticate a user session (TOKEN) using the "Unified Login" architecture, the consumer shall simply instantiate the **Authentication** web service and invokes its **AuthenticateToken** operation.

1.2.3 DeleteUserSession operation

The **DeleteUserSession** operation is used for deleting a user session that has been created using the AuthenticateUser method.

1.2.3.1 Parameters

The **DeleteUserSession** operation requires two parameters in the following order:

Parameter name	Description	Type	Max. Length
Security Password	Use this parameter to specify the security password to consume this web service.	Alphanumeric	36
TOKEN	Use this parameter to specify the TOKEN to be deleted.	Alphanumeric	36

1.2.3.2 Return values

In case of success, the **DeleteUserSession** operation returns the deleted TOKEN. Otherwise it returns a string with an error code and an explanation of the error. The format of the error message is: "*Err Num: #### - error description*".

1.2.3.3 Usage

To delete a user session (TOKEN) the consumer shall simply instantiate the **Authentication** web service and invokes its **DeleteUserSession** operation.