# HealthVault iOS Library

## Class Index

**AuthenticationCheckState (The data required to perform the authentication flow)** ....................1
**Base64 (Provides Base64 encoding/decoding functionality)** ....................2
**DateTimeUtils (Provides date conversion utilities)** ....................3
**HealthVaultRecord (Stores the summary information associated with a record)** ....................4
**HealthVaultRecordTest (Implements tests for HealthVaultRecord class)** ....................5
**HealthVaultRequest (This class encapsulates the data that is contained in a request)** ....................6
**HealthVaultRequestTest (Implements tests for HealthVaultRequest class)** ....................8
**HealthVaultResponse (Implements HealthVault response )** ....................8
**HealthVaultResponseTest (Implements tests for HealthVaultResponse class)** ....................10
**HealthVaultService (A class used to communicate with the HealthVault web service)** ....................10
**HealthVaultServiceTest (Implements tests for HealthVaultService class)** ....................14
**HealthVaultSettings (Provides settings management functionality)** ....................14
**Logger (Implements logging related functionality)** ....................16
**MobilePlatform (Implements the platform-specific methods used by the HealthVaultService class)** 17
**MobilePlatformTest (Implements tests for the MobilePlatform class)** ....................18
**Provisioner (Implements the application authorization process)** ....................18
**ProvisionerTest (Implements tests for the Provisioner class)** ....................20
**WebResponse (Represents xml response from HealthVault server)** ....................20
**WebTransport (Class to simplify making POSTs and obtaining the responses)** ....................21
**WeightTest (Implements tests for Weight class)** ....................22
**XmlElement (Represents xml tree node)** ....................22
**XmlTextReader (Reads xml data and creates xml tree )** ....................24

# Class Documentation

## AuthenticationCheckState Class Reference

The data required to perform the authentication flow.

```
#import <AuthenticationCheckState.h>
```

### Public Member Functions

- (id) - initWithService:target:authCompletedCallBack:shellAuthRequiredCallBack:
  *Initializes a new instance of the AuthenticationCheckState class.*

### Properties

- HealthVaultService * service
  *Gets or sets the service.*
- NSObject * target
  *Gets or sets the callback handler.*

- SEL authCompletedCallBack
  *Gets or sets the authentication completed handler.*

## Detailed Description

The data required to perform the authentication flow.

## Member Function Documentation

### - (id) initWithService: (HealthVaultService *)*service* target: (NSObject *)*target* authCompletedCallBack: (SEL)*authCallBack* shellAuthRequiredCallBack: (SEL)*authRequiredCallBack*

Initializes a new instance of the AuthenticationCheckState class.

**Parameters:**

| | |
|---|---|
| *service* | - the HealthVaultService instance. |
| *target* | - callBack method owner. |
| *authCallBack* | - callback if shell authorization is completed. |
| *authRequiredCallBack* | - callBack if shell authorization is required. |

# Base64 Class Reference

Provides Base64 encoding/decoding functionality.

```
#import <Base64.h>
```

## Static Public Member Functions

- (NSString *) + encodeBase64WithData:
  *Encodes incoming data to base64 string.*
- (NSData *) + decodeBase64WithString:
  *Decodes base64 string.*

## Detailed Description

Provides Base64 encoding/decoding functionality.

## Member Function Documentation

### + (NSData *) decodeBase64WithString: (NSString *)*stringToDecode*

Decodes base64 string.

**Parameters:**

| | |
|---|---|
| *stringToDecode* | - string for decoding. |

**Returns:**

decoded data.

### + (NSString *) encodeBase64WithData: (NSData *)*dataToEncode*

Encodes incoming data to base64 string.

**Parameters:**

| | |
|---|---|
| *dataToEncode* | - data for encoding. |

**Returns:**

encoded base64 string.

# DateTimeUtils Class Reference

Provides date conversion utilities.

```
#import <DateTimeUtils.h>
```

## Static Public Member Functions

- (NSString *) + dateToUtcString:
  *Converts date to UTC formatted string.*
- (NSDate *) + UtcStringToDate:
  *Converts string with date in UTC format to date object.*

## Detailed Description

Provides date conversion utilities.

## Member Function Documentation

### + (NSString *) dateToUtcString: (NSDate *)*date*

Converts date to UTC formatted string.

**Parameters:**

| | |
|---|---|
| *date* | - date to be converted. |

**Returns:**

UTC formatted string.

### + (NSDate *) UtcStringToDate: (NSString *)*string*

Converts string with date in UTC format to date object.

**Parameters:**

| | |
|---|---|
| *string* | - string to be converted. |

**Returns:**

date in UTC format.

## HealthVaultRecord Class Reference

Stores the summary information associated with a record.

```
#import <HealthVaultRecord.h>
```

### Public Member Functions

- (id) - initWithXml:personId:personName:
  *Initializes a new instance of the HealthVaultRecord class.*
- (BOOL) - parseFromXml:
  *Initializes the fields using xml string provided.*

### Static Public Member Functions

- (id) + newFromXml:personId:personName:
  *Creates a new instance of the HealthVaultRecord class.*

### Properties

- NSString * xml
  *Gets or sets the full XML description of this record...*
- NSString * personId
  *Gets or sets the person id of the person who has access to this record.*
- NSString * personName
  *Gets or sets the name of the person who has access to this record.*
- NSString * recordId
  *Gets or sets the record id of this record.*
- NSString * recordName
  *Gets or sets the name of this record.*
- NSString * authStatus
  *Gets or sets the authorization status.*
- BOOL isValid
  *Gets whether or not record is valid.*

### Detailed Description

Stores the summary information associated with a record.

## Member Function Documentation

### - (id) initWithXml: (NSString *)*xml* personId: (NSString *)*personId* personName: (NSString *)*personName*

Initializes a new instance of the HealthVaultRecord class.

**Parameters:**

| | |
|---|---|
| *xml* | - the full XML describing this record. |
| *personId* | - the id of the person. |
| *personName* | - the name of the person. |

### + (id) newFromXml: (NSString *)*xml* personId: (NSString *)*personId* personName: (NSString *)*personName*

Creates a new instance of the HealthVaultRecord class.

**Parameters:**

| | |
|---|---|
| *xml* | - the full XML describing this record. |
| *personId* | - the id of the person. |
| *personName* | - the name of the person. |

**Returns:**

a HealthVaultRecord class instance or nil if xml is not well formed or record is invalid( see isValid property for more details)

### - (BOOL) parseFromXml: (NSString *)*xml*

Initializes the fields using xml string provided.

**Parameters:**

| | |
|---|---|
| *xml* | - the full XML describing this record. |

## Property Documentation

### - (BOOL) isValid `[read, assign]`

Gets whether or not record is valid.

Record could be invalid according to some parameters, like 'app-record-auth-action' != 'NoActionRequired'

# HealthVaultRecordTest Class Reference

Implements tests for HealthVaultRecord class.

```
#import <HealthVaultRecordTest.h>
```

## Detailed Description

Implements tests for HealthVaultRecord class.

Contains tests to check logic for parsing XML.

# HealthVaultRequest Class Reference

This class encapsulates the data that is contained in a request.

```
#import <HealthVaultRequest.h>
```

## Public Member Functions

- (id) - initWithMethodName:methodVersion:infoSection:target:callBack:
  *Initializes a new instance of the HealthVaultRequest class.*
- (NSString *) - toXml
  *Converts the request to xml representation ready to be submitted to HealthVault service.*

## Properties

- NSString * methodName
  *Gets or sets the name of the method to be called.*
- float methodVersion
  *Gets or sets the version of the method to be called.*
- NSString * infoXml
  *Gets or sets the request-specific information.*
- NSString * recordId
  *Gets or sets the record id that will be used to perform request.*
- NSString * personId
  *Gets or sets the person id that will be used to perform request.*
- NSString * authorizationSessionToken
  *Gets or sets the authorization token that is required to talk to the HealthVault web service.*
- NSString * appIdInstance
  *Gets or sets the application instance id.*
- NSString * sessionSharedSecret
  *Gets or sets the session shared secret.*
- NSString * country
  *Gets or sets the country that is used for responses.*
- NSDate * msgTime
  *Gets or sets request msg-time parameter.*
- int msgTTL
  *Gets or sets request msg-ttl parameter.*
- NSObject * userState
  *Gets or sets the user state.*
- NSObject * target
  *Gets or sets the callback handler.*

- SEL callBack
  *Gets or sets the callback that will be called when the request has completed.*

## Detailed Description

This class encapsulates the data that is contained in a request.

## Member Function Documentation

### - (id) initWithMethodName: (NSString *)*name* methodVersion: (float)*methodVersion* infoSection: (NSString *)*info* target: (NSObject *)*target* callBack: (SEL)*callBack*

Initializes a new instance of the HealthVaultRequest class.

**Parameters:**

| | |
|---|---|
| *name* | - the name of the method. |
| *methodVersion* | - the version of the method. |
| *infoSection* | - the request-specific xml to pass. |
| *target* | - callback method owner. |
| *callBack* | - the method to call when the request has completed. |

The method name and version must be one of the methods documented in the method reference at: http://developer.healthvault.com/pages/methods/methods.aspx

### - (NSString *) toXml

Converts the request to xml representation ready to be submitted to HealthVault service.

**Returns:**

xml representation of the request.

## Property Documentation

### - (NSString*) authorizationSessionToken `[read, write, retain]`

Gets or sets the authorization token that is required to talk to the HealthVault web service.

### - (NSString*) methodName `[read, write, retain]`

Gets or sets the name of the method to be called.

The method name and version must be one of the methods documented in the method reference at: http://developer.healthvault.com/pages/methods/methods.aspx

**- (float) methodVersion** `[read, write, assign]`

Gets or sets the version of the method to be called.

The method name and version must be one of the methods documented in the method reference at: http://developer.healthvault.com/pages/methods/methods.aspx

**- (NSObject*) userState** `[read, write, retain]`

Gets or sets the user state.

User state can be used by the caller to pass state to the handler.

## HealthVaultRequestTest Class Reference

Implements tests for HealthVaultRequest class.

`#import <HealthVaultRequestTest.h>`

### Detailed Description

Implements tests for HealthVaultRequest class.

Contains tests to check logic for composing request XML.

## HealthVaultResponse Class Reference

Implements HealthVault response.

`#import <HealthVaultResponse.h>`

### Public Member Functions

- (id) - initWithWebResponse:request:
  *Initializes a new instance of the HealthVaultResponse class.*
- (BOOL) - parseFromXml:
  *Initializes the fields using xml string provided.*
- (NSString *) - getInfoFromXml:
  *Retrieves info section from xml.*

### Properties

- int statusCode
  *Gets or sets numeric status code of the operation.*
- NSString * infoXml
  *Gets or sets the informational part of the response.*
- NSString * responseXml
  *Gets or sets the raw xml that was returned from the request.*

- NSString * errorText
  *Gets or sets the text of the error that occurred.*
- NSString * errorContextXml
  *Gets or sets a contextual xml description of the the error.*
- NSString * errorInfo
  *Gets or sets the informational part of the response.*
- HealthVaultRequest * request
  *Gets or sets the request that was sent.*
- BOOL hasError
  *Indicates whether the operation has failed.*

## Detailed Description

Implements HealthVault response.

## Member Function Documentation

### - (NSString *) getInfoFromXml: (NSString *)*xml*

Retrieves info section from xml.

Info section is represented by <wc:info> xml element.

**Parameters:**

| | |
|---|---|
| *xml* | - xml from which to retrieve info section. |

**Returns:**

info section in provided xml

### - (id) initWithWebResponse: (WebResponse *)*webResponse* request: (HealthVaultRequest *)*request*

Initializes a new instance of the HealthVaultResponse class.

**Parameters:**

| | |
|---|---|
| *webResponse* | - the web response from server side. - the original request. |

### - (BOOL) parseFromXml: (NSString *)*xml*

Initializes the fields using xml string provided.

**Parameters:**

| | |
|---|---|
| *xml* | - response xml representation. |

## Property Documentation

### - (NSString*) errorText `[read, write, retain]`

Gets or sets the text of the error that occurred.

If the error was returned from the HealthVault service, additional error information may be found in the ResponseXml.

## HealthVaultResponseTest Class Reference

Implements tests for HealthVaultResponse class.

```
#import <HealthVaultResponseTest.h>
```

### Detailed Description

Implements tests for HealthVaultResponse class.

Contains tests to check logic for parsing response XML.

## HealthVaultService Class Reference

A class used to communicate with the HealthVault web service.

```
#import <HealthVaultService.h>
```

### Public Member Functions

- (id) - initWithDefaultUrl:
  *Initializes a new instance of the HealthVaultService class using default platform and shell URLs.*
- (id) - initWithUrl:shellUrl:masterAppId:
  *Initializes a new instance of the HealthVaultService class.*
- (NSString *) - getApplicationCreationUrl
  *Returns the URL string to use to provision the application.*
- (NSString *) - getUserAuthorizationUrl
  *Returns the URL string to use to authorize additional records.*
- (NSString *) - getCastCallInfoSection
  *Returns info section for Cast call request.*
- (void) - saveCastCallResults:
  *Saves the results of a cast call into the session.*
- (void) - sendRequest:
  *Sends a request to the HealthVault web service.*
- (void) - authorizeRecords:authenticationCompleted:shellAuthRequired:
  *Authorizes more records.*
- (void) - performAuthenticationCheck:authenticationCompleted:shellAuthRequired:
  *Starts the authentication check.*
- (void) - saveSettings:
  *Saves the current configuration to isolated storage.*
- (void) - loadSettings:
  *Loads the last-saved configuration from isolated storage.*

- (void) - refreshSessionToken:
  *Refreshes the session token.*
- (void) - performAppCallBack:response:
  *Invokes the calling application's callback.*

## Properties

- NSString * healthServiceUrl
  *Gets or sets the URL that is used to talk to the HealthVault Web Service.*
- NSString * shellUrl
  *Gets or sets the URL that is used to talk to the HealthVault Shell.*
- NSString * authorizationSessionToken
  *Gets or sets the authorization token that is required to talk to the HealthVault.*
- NSString * sharedSecret
  *Gets or sets the application shared secret.*
- NSString * sessionSharedSecret
  *Gets or sets the session shared secret.*
- NSString * masterAppId
  *Gets or sets the master app id.*
- NSString * language
  *Gets or sets the language that is used for responses.*
- NSString * country
  *Gets or sets the country that is used for responses.*
- NSString * appIdInstance
  *Gets or sets the application instance id.*
- NSString * applicationCreationToken
  *Gets or sets the application creation token.*
- NSMutableArray * records
  *Gets the list of records that this application is authorized to use.*
- HealthVaultRecord * currentRecord
  *Gets or sets the person and record that will be used.*
- BOOL isApplicationCreated
  *Is YES if current application instance has already been created, otherwise FALSE.*

## Detailed Description

A class used to communicate with the HealthVault web service.

## Member Function Documentation

### - (void) authorizeRecords: (NSObject *)*target* authenticationCompleted: (SEL)*authCompleted* shellAuthRequired: (SEL)*shellAuthRequired*

Authorizes more records.

**Parameters:**

| | |
|---|---|
| *target* | - callback handler. |

| | |
|---|---|
| *authCompleted* | - method that is called when the authentication process is complete. |
| *shellAuthRequire* | - method that is called when the application needs to perform authorization. Applications should call SaveSettings in both of the handlers. |

### - (id) initWithDefaultUrl: (NSString *)*masterAppId*

Initializes a new instance of the HealthVaultService class using default platform and shell URLs.

The master application id must come from the HealthVault Application Configuration Center, and that application must set to be a SODA application. Before requests can be made, the method performAuthenticationCheck must be called.

**Parameters:**

| | |
|---|---|
| *masterAppId* | - the master application id for this application. |

### - (id) initWithUrl: (NSString *)*healthServiceUrl* shellUrl: (NSString *)*shellUrl* masterAppId: (NSString *)*masterAppId*

Initializes a new instance of the HealthVaultService class.

The master application id must come from the HealthVault Application Configuration Center, and that application must set to be a SODA application. Before requests can be made, the method performAuthenticationCheck must be called.

**Parameters:**

| | |
|---|---|
| *healthServiceUrl* | - the Url to use to talk to the HealthVault web service. |
| *shellUrl* | - the Url to use for the HealthVault Shell. |
| *masterAppId* | - the master application id for this application. |

### - (void) loadSettings: (NSString *)*name*

Loads the last-saved configuration from isolated storage.

**Parameters:**

| | |
|---|---|
| *name* | - the filename to use. |

### - (void) performAppCallBack: (HealthVaultRequest *)*request* response: (HealthVaultResponse *)*response*

Invokes the calling application's callback.

**Parameters:**

| | |
|---|---|
| *request* | - the request object. |
| *response* | - the response object. |

### - (void) performAuthenticationCheck: (NSObject *)*target* authenticationCompleted: (SEL)*authCompleted* shellAuthRequired: (SEL)*shellAuthRequired*

Starts the authentication check.

**Parameters:**

| | |
|---|---|
| *target* | - callback handler. |
| *authCompleted* | - method that is called when the authentication process is complete. |
| *shellAuthRequired* | - method that is called when the application needs to perform authorization. The authentication check is required to be called every time the application is initialized. It will perform the proper amount of initialization based on the information that was saved. Applications should call LoadsSettings method before calling this method, and should call SaveSettings in both of the handlers. |

### - (void) refreshSessionToken: (HealthVaultRequest *)*request*

Refreshes the session token.

Makes a CAST call to get a new session token, and then re-issues the original request.

**Parameters:**

| | |
|---|---|
| *request* | - the original request. |

### - (void) saveCastCallResults: (NSString *)*responseXml*

Saves the results of a cast call into the session.

**Parameters:**

| | |
|---|---|
| *responseXml* | - the response xml. |

### - (void) saveSettings: (NSString *)*name*

Saves the current configuration to isolated storage.

**Parameters:**

| | |
|---|---|
| *name* | - the filename to use. |

### - (void) sendRequest: (HealthVaultRequest *)*request*

Sends a request to the HealthVault web service.

This method returns immediately; the results and any error information will be passed to the completion method stored in the request. Please ensure that the method performAuthenticationCheck is called before making requests.

**Parameters:**

| | |
|---|---|
| *request* | - the request to send. |

## Property Documentation

### - (NSString*) applicationCreationToken `[read, write, retain]`

Gets or sets the application creation token.

The application token is returned from the CreateApplicationRequest method, and is only useful for passing to the HealthVault Shell to start the authorization and application creation process.

**- (NSString*) masterAppId `[read, write, retain]`**

Gets or sets the master app id.

The master application is predefined by the developer using the Application Configuration Center tool.

# HealthVaultServiceTest Class Reference

Implements tests for HealthVaultService class.

`#import <HealthVaultServiceTest.h>`

## Detailed Description

Implements tests for HealthVaultService class.

# HealthVaultSettings Class Reference

Provides settings management functionality.

`#import <HealthVaultSettings.h>`

## Public Member Functions

- (id) - initWithName:
  *Initializes settings with specific name.*
- (void) - save
  *Saves settings.*

## Static Public Member Functions

- (HealthVaultSettings *) + loadWithName:
  *Loads settings with specified name.*
- (NSString *) + makePrefixForName:
  *Makes special prefix for settings stored in user defaults on the device.*

## Properties

- NSString * name
  *Gets or sets the settings name.*
- NSString * version
  *Gets or sets application version.*
- NSString * applicationId
  *Gets or sets the application Id.*
- NSString * applicationCreationToken

*Gets or sets the application creation token.*

- NSString * authorizationSessionToken
  *Gets or sets the authorization session token.*
- NSString * sharedSecret
  *Gets or sets the shared secret.*
- NSString * country
  *Gets or sets country.*
- NSString * language
  *Gets or sets language.*
- NSString * sessionSharedSecret
  *Gets or sets session shared secret.*
- NSString * personId
  *Gets or sets person Id.*
- NSString * recordId
  *Gets or sets record Id.*

## Detailed Description

Provides settings management functionality.

## Member Function Documentation

### - (id) initWithName: (NSString *)*name*

Initializes settings with specific name.

**Parameters:**

| | |
|---|---|
| *name* | - settings file name. |

### + (HealthVaultSettings *) loadWithName: (NSString *)*name*

Loads settings with specified name.

**Parameters:**

| | |
|---|---|
| *name* | - settings file name. |

**Returns:**

settings instance loaded with specific name.

### + (NSString *) makePrefixForName: (NSString *)*name*

Makes special prefix for settings stored in user defaults on the device.

**Parameters:**

| | |
|---|---|
| *name* | - name for which to make a prefix. If nil then default name is used (DEFAULT_SETTINGS_NAME). |

**Returns:**

special prefix for settings.

## Logger Class Reference

Implements logging related functionality.

```
#import <Logger.h>
```

### Static Public Member Functions

- (void) + write:
  *Writes the message to a log.*
- (void) + setLoggingMethod
  *Implements logic to set up the correct logging mechanism: console, file, no logging.*
- (void) + redirectConsoleLogToDocumentFolder
  *Redirects stderr stream to log file in Document directory.*
- (NSString *) + createNameForLogWithName:
  *Creates a new name for the log file.*

### Detailed Description

Implements logging related functionality.

### Member Function Documentation

#### + (NSString*) createNameForLogWithName: (NSString *)*name*

Creates a new name for the log file.

File format is [appLog yyyy'-'MM'-'dd' 'HH'.'mm'.'ss'']

**Parameters:**

| | |
|---|---|
| *name* | - name of file. |

**Returns:**

path to log file.

#### + (void) write: (NSString *)*text*

Writes the message to a log.

**Parameters:**

| | |
|---|---|
| *text* | - text to write. |

# MobilePlatform Class Reference

Implements the platform-specific methods used by the HealthVaultService class.

```
#import <MobilePlatform.h>
```

## Static Public Member Functions

- (NSString *) + platformAbbreviationAndVersion
  *Gets a name and version for this platform/library combination.*
- (NSString *) + deviceName
  *Gets a name for the device.*
- (NSString *) + computeSha256Hash:
  *Computes a SHA 256 hash.*
- (NSString *) + computeSha256HashAndWrap:
  *Computes a SHA 256 hash and wraps the result in XML.*
- (NSString *) + computeSha256Hmac:key:
  *Computes a SHA 256 HMAC.*
- (NSString *) + computeSha256HmacAndWrap:key:
  *Computes a SHA 256 HMAC and wraps the result in XML.*

## Detailed Description

Implements the platform-specific methods used by the HealthVaultService class.

## Member Function Documentation

### + (NSString *) computeSha256Hash: (NSString *)*data*

Computes a SHA 256 hash.

**Parameters:**

| | |
|---|---|
| *data* | - the data to hash. |

**Returns:**

the hash as a base64-encoded string.

### + (NSString *) computeSha256HashAndWrap: (NSString *)*data*

Computes a SHA 256 hash and wraps the result in XML.

**Parameters:**

| | |
|---|---|
| *data* | - the data to hash. |

**Returns:**

the wrapped hash.

### + (NSString *) computeSha256Hmac: (NSData *) key: (NSString *)*data*

Computes a SHA 256 HMAC.

**Parameters:**

| | |
|---|---|
| *key* | - the key to use. |

**Parameters:**

| | |
|---|---|
| *data* | - the input data. |

**Returns:**

a base-64 encoded HMAC.

**+ (NSString *) computeSha256HmacAndWrap: (NSData *) key: (NSString *)*data***

Computes a SHA 256 HMAC and wraps the result in XML.

**Parameters:**

| | |
|---|---|
| *key* | - the key to use. |

**Parameters:**

| | |
|---|---|
| *data* | - the input data. |

**Returns:**

the wrapped result.

# MobilePlatformTest Class Reference

Implements tests for the MobilePlatform class.

```
#import <MobilePlatformTest.h>
```

## Detailed Description

Implements tests for the MobilePlatform class.

Contains tests to check hash (SHA256 and SHA256HMAC) logic.

# Provisioner Class Reference

Implements the application authorization process.

```
#import <Provisioner.h>
```

## Static Public Member Functions

- (void) + authorizeRecords:target:authenticationCompleted:shellAuthRequired:
  *Authorizes other records.*
- (void) + performAuthenticationCheck:target:authenticationCompleted:shellAuthRequired:
  *Checks that the application is authenticated.*
- (void) + performAuthenticationCheck:

*Checks that the application is authenticated.*

- (void) + castCall:
  *Makes the CreateAuthenticatedSessionToken call.*
- (void) + getAuthorizedPeople:
  *Gets the list of authorized people.*
- (void) + startNewApplicationCreationInfo:
  *Gets the new application info from the HealthVault platform.*

## Detailed Description

Implements the application authorization process.

## Member Function Documentation

### + (void) authorizeRecords: (HealthVaultService *)*service* target: (NSObject *)*target* authenticationCompleted: (SEL)*authCompleted* shellAuthRequired: (SEL)*shellAuthRequired*

Authorizes other records.

**Parameters:**

| | |
|---|---|
| *service* | - the HealthVaultService instance. |
| *target* | - callback method owner. |
| *authCompleted* | - method that is called when the authentication process is complete. |
| *shellAuthRequired* | - method that is called when the application needs to perform authorization. |

### + (void) castCall: (AuthenticationCheckState *)*state*

Makes the CreateAuthenticatedSessionToken call.

**Parameters:**

| | |
|---|---|
| *state* | - the state information. |

### + (void) getAuthorizedPeople: (AuthenticationCheckState *)*state*

Gets the list of authorized people.

**Parameters:**

| | |
|---|---|
| *state* | - the state information. |

### + (void) performAuthenticationCheck: (AuthenticationCheckState *)*state*

Checks that the application is authenticated.

**Parameters:**

| | |
|---|---|
| *state* | - the state information. |

**+ (void) performAuthenticationCheck: (HealthVaultService *)*service* target: (NSObject *)*target* authenticationCompleted: (SEL)*authCompleted* shellAuthRequired: (SEL)*shellAuthRequired***

Checks that the application is authenticated.

**Parameters:**

| | |
|---|---|
| *service* | - the HealthVaultService instance. |
| *target* | - callback method owner. |
| *authCompleted* | - method that is called when the authentication process is complete. |
| *shellAuthRequired* | - method that is called when the application needs to perform authorization. |

**+ (void) startNewApplicationCreationInfo: (AuthenticationCheckState *)*state***

Gets the new application info from the HealthVault platform.

**Parameters:**

| | |
|---|---|
| *state* | - the state information. |

# ProvisionerTest Class Reference

Implements tests for the Provisioner class.

```
#import <ProvisionerTest.h>
```

## Detailed Description

Implements tests for the Provisioner class.

# WebResponse Class Reference

Represents xml response from HealthVault server.

```
#import <WebResponse.h>
```

## Properties

- NSString * responseData
  *Gets or sets the response data.*
- NSString * errorText
  *Gets or sets the error text.*
- BOOL hasError
  *Gets error status for response. Returns YES if request has been failed.*

## Detailed Description

Represents xml response from HealthVault server.

The data related to a request.

# WebTransport Class Reference

Class to simplify making POSTs and obtaining the responses.

```
#import <WebTransport.h>
```

## Public Member Functions

- (void) - performCallBack:  
  *Performs callBack on target when response is received.*

## Static Public Member Functions

- (BOOL) + isRequestResponseLogEnabled  
  *Returns whether all requests and responses should be logged.*
- (void) + setRequestResponseLogEnabled:  
  *Sets current logging status.*
- (void) + sendRequestForURL:withData:context:target:callBack:  
  *Sends a post request to a specific URL.*
- (void) + addMessageToRequestResponseLog:  
  *Logs message, prints it to console.*

## Detailed Description

Class to simplify making POSTs and obtaining the responses.

## Member Function Documentation

### + (void) addMessageToRequestResponseLog: (NSString *)*message*

Logs message, prints it to console.

**Parameters:**

| | |
|---|---|
| *message* | - message to be written. |

### - (void) performCallBack: (WebResponse *)*response*

Performs callBack on target when response is received.

**Parameters:**

| | |
|---|---|
| *response* | - response to send. |

**+ (void) sendRequestForURL: (NSString *)*url* withData: (NSString *)*data* context: (NSObject *)*context* target: (NSObject *)*target* callBack: (SEL)*callBack***

Sends a post request to a specific URL.

**Parameters:**

| | |
|---|---|
| *url* | - string which contains server address. |
| *data* | - string will be sent in POST header. |
| *context* | - any object will be passed to callBack with response. |
| *target* | - callback method owner. |
| *callBack* | - the method to call when the request has completed. |

**+ (void) setRequestResponseLogEnabled: (BOOL)*enabled***

Sets current logging status.

**Parameters:**

| | |
|---|---|
| *enabled* | - If YES then both request and response will be logged. |

# WeightTest Class Reference

Implements tests for Weight class.

```
#import <WeightTest.h>
```

## Detailed Description

Implements tests for Weight class.

Contains tests to check logic for composing request XML and parsing response.

# XmlElement Class Reference

Represents xml tree node.

```
#import <XmlElement.h>
```

## Public Member Functions

- (NSArray *) - selectNodes:
  *Returns an array of children matching the given element name.*
- (XmlElement *) - selectSingleNode:
  *Returns the child node with the given name.*
- (XmlElement *) - selectSingleNode:at:
  *Returns the nth child with the given name.*
- (NSString *) - attrValue:

*Returns attribute value.*

## Protected Attributes

- NSString * _name
  *Element name.*
- NSMutableString * _text
  *Element inner text.*
- NSMutableDictionary * _attributes
  *Element attributes.*
- NSMutableDictionary * _children
  *Element children.*

## Properties

- NSString * name
  *Gets or sets element name.*
- NSMutableString * text
  *Gets or sets element inner text.*
- NSMutableDictionary * attributes
  *Gets or sets element attributes.*
- NSMutableDictionary * children
  *Gets or sets elements children.*

## Detailed Description

Represents xml tree node.

## Member Function Documentation

### - (NSString *) attrValue: (NSString *)*name*

Returns attribute value.

**Parameters:**

| | |
|---|---|
| *name* | - attribute name. |

**Returns:**

attribute value.

### - (NSArray *) selectNodes: (NSString *)*name*

Returns an array of children matching the given element name.

**Parameters:**

| | |
|---|---|
| *name* | - children name. |

**Returns:**

an array of children matching the given element name.

**- (XmlElement *) selectSingleNode: (NSString *)*name***

Returns the child node with the given name.

**Parameters:**

| | |
|---|---|
| *name* | - child name. |

**Returns:**

the first occurrence if there is more than one.

**- (XmlElement *) selectSingleNode: (NSString *)*name* at: (NSInteger)*position***

Returns the nth child with the given name.

**Parameters:**

| | |
|---|---|
| *name* | - child name. |
| *at* | - position index (starting from zero). |

**Returns:**

the nth child with the given name.

# XmlTextReader Class Reference

Reads xml data and creates xml tree.

```
#import <XmlTextReader.h>
```

## Public Member Functions

- (XmlElement *) - read:
  *Reads xml data, creates xml tree.*

## Protected Attributes

- XmlElement * _rootElement
  *Xml tree root element.*
- NSMutableArray * _elements
  *Stack used for creating xml tree.*

## Detailed Description

Reads xml data and creates xml tree.

## Member Function Documentation

**- (XmlElement *) read: (NSString *)*xml***

Reads xml data, creates xml tree.

**Parameters:**

| | |
|---|---|
| *xml* | - xml text to parse. |

**Returns:**

reference to tree root element.