

SDD

Wayne.Lib.Common

This document is the property of Dresser Wayne. It is not to be used or duplicated without the written permission of the owner, and is not to be used in any way inconsistent with purpose for which it was loaned.

Dresser Wayne shall not be liable for technical or editorial errors or omissions, which may appear in this document. It also retains the right to make changes to this document at any time, without notice.

Table of Contents

1	Document information	3
1.1	Revision history	3
1.2	Purpose and scope	3
1.3	Abbreviations and acronyms	3
1.4	References	3
2	Overview	4
2.1	Identifiable Entity	4
2.2	Connectable	4
2.3	Asynchronous methods support	5
2.4	Assemblies support	5
3	Namespace Wayne.Lib	6
3.1	Interfaces	6
3.1.1	Interface IConnectable	6
3.1.2	Interface IIdentifiableEntity	7
3.2	Classes	7
3.2.1	Class Assemblies	7
3.2.2	Class AsyncCompletedEventArgs	8
3.2.3	Class AsyncCompletedEventArgs`1	8
3.2.4	Class ConnectionChangedEventArgs	8
3.2.5	Class EventArgs`1	9
3.2.6	Class IdentifiableEntity	9
3.2.7	Class ObjectNotReservedException	10
3.2.8	Class ReserveCompletedEventArgs`1	10
3.2.9	Class UserTokenEventArgs	10
3.3	Enumerations	11
3.3.1	Enumeration DeviceConnectionState	11

1 Document information

File: SDD_Wayne.Lib.Common.doc

1.1 Revision history

Revision	Author	Date	Change description
1.0	Mattias Larsson	2006-11-29	Created

1.2 Purpose and scope

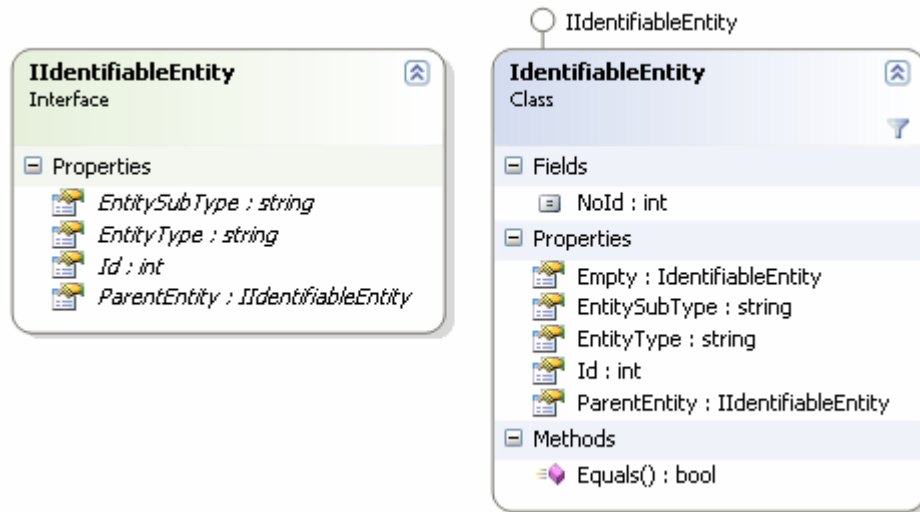
1.3 Abbreviations and acronyms

Abbreviation	Meaning

1.4 References

2 Overview

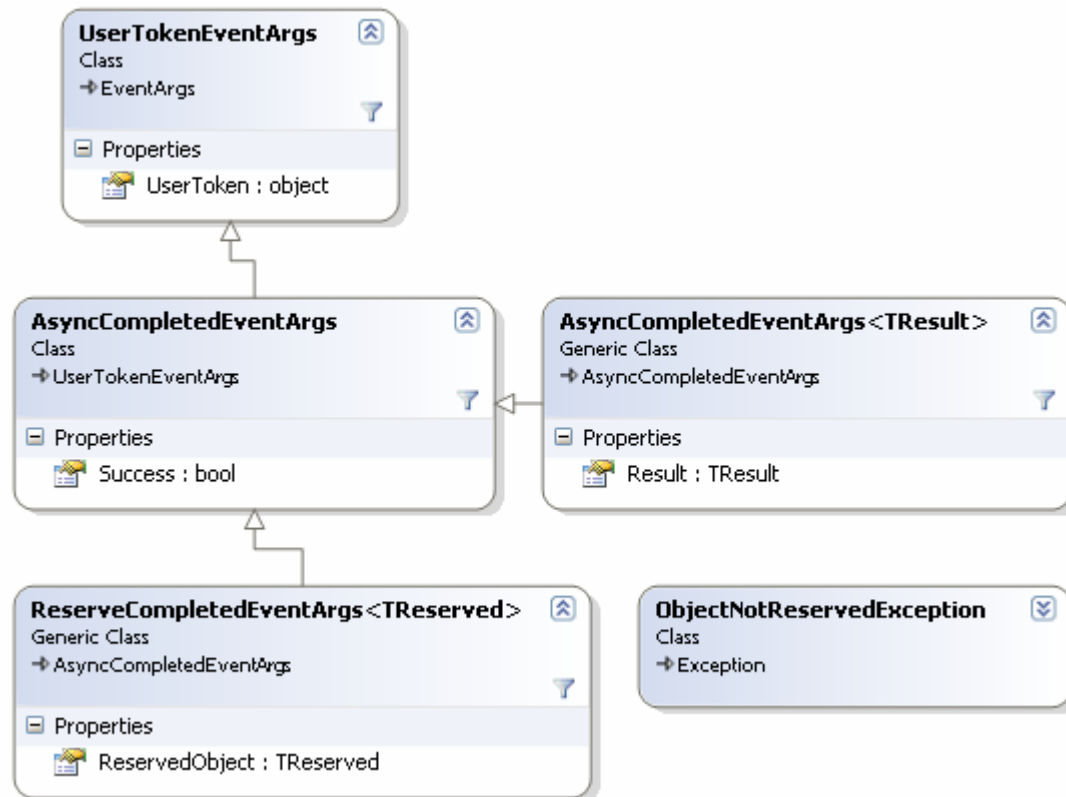
2.1 Identifiable Entity



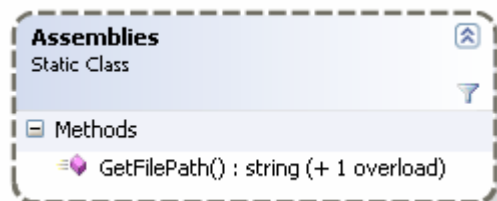
2.2 Connectable



2.3 Asynchronous methods support



2.4 Assemblies support



3 Namespace Wayne.Lib

Interfaces

IConnectable	The IConnectable defines a standard set of methods and properties for classes that represents devices that theoretically can have a connected/disconnected state.
IIdentifiableEntity	The IIdentifiableEntity represents an entity of some sort that has an integer ID and a possible parent.

Classes

Assemblies	Static help class for Assembly issues.
AsyncCompletedEventArgs	Generic event result of an asynchronous command.
AsyncCompletedEventArgs`1	Generic event result of an asynchronous command with an additional status property.
ConnectionChangedEventArgs	Event arguments for a connection changed event from a Wayne Device.
EventArgs`1	A generic event args holding an argument of some type.
IdentifiableEntity	A class implementing the IIdentifiableEntity interface. Also contains static properties and methods that handles IIdentifiableEntity.
ObjectNotReservedException	The ObjectNotReservedException is thrown when trying to access an object in a way that requires reservation, or if a Reserved interface is used after it has been unreserved.
ReserveCompletedEventArgs`1	The EventArgs is used when ...
UserTokenEventArgs	An EventArgs containing a UserToken.

Enumerations

DeviceConnectionState	The state of the connection to a device.
------------------------------	--

3.1 Interfaces

3.1.1 Interface IConnectable

```
public interface IConnectable
```

Summary

The IConnectable defines a standard set of methods and properties for classes that represents devices that theoretically can have a connected/disconnected state.

Properties

ConnectionState Lib.DeviceConnectionState	R	Indicates the state of the connection.
--	---	--

Methods

Connect <pre>public void Connect(string connectionString);</pre> Connect to the device.		
<i>connectionString</i>		A string that contains the connection parameters. The format is provided by the implementation.

Disconnect

```
public void Disconnect();
```

Disconnects from the device.

Events

OnConnectionStateChange

```
public EventHandler{Wayne.Lib.ConnectionChangedEventArgs}
OnConnectionStateChange;
```

An event that is fired when the ConnectionState changes for to device.

3.1.2 Interface IIdentifiableEntity

```
public interface IIdentifiableEntity
```

Summary

The IIdentifiableEntity represents an entity of some sort that has an integer ID and a possible parent.

Properties

EntitySubType string	R	A more refined type of the entity, e.g. a specific implementation or brand.
EntityType string	R	The main type of entity.
Id int	R	The ID of the entity.
ParentEntity Lib.IIdentifiableEntity	R	Reference to a possible parent device.

3.2 Classes

3.2.1 Class Assemblies

```
abstract public class Assemblies : Object
```

Summary

Static help class for Assembly issues.

Methods

GetFilePath

```
public string GetFilePath();
```

Get the calling assembly's file path.

GetFilePath

```
public string GetFilePath(Reflection.Assembly assembly);
```

Get the file path of the given assembly.

assembly

GetManifestResourceStreamWithPartialName

```
public IO.Stream GetManifestResourceStreamWithPartialName(string
knownResourceNameEnd, Reflection.Assembly assemblyWithResource);
```

Gets a manifest resource stream from an assembly, where only the file name is known.

knownResourceNameEnd The known file name, i.e. only the end of the Manifest resource name.

assemblyWithResource The assembly that is expected to contain the resource.

Return value	If the resource is found, the stream is returned, otherwise null.
--------------	---

3.2.2 Class AsyncCompletedEventArgs

```
public class AsyncCompletedEventArgs : UserTokenEventArgs
```

Summary

Generic event result of an asynchronous command.

Properties

Success bool	R	The status of the reservation.
-----------------	---	--------------------------------

Constructors

```
public AsyncCompletedEventArgs(bool success, object userToken);  
Construction.
```

<i>success</i>	The status of the reservation.
<i>userToken</i>	An optional user token.

Methods

3.2.3 Class AsyncCompletedEventArgs`1

```
public class AsyncCompletedEventArgs`1 : AsyncCompletedEventArgs
```

Summary

Generic event result of an asynchronous command with an additional status property.

Properties

Result	R	Refined status of the command.
--------	---	--------------------------------

Constructors

```
public AsyncCompletedEventArgs`1(bool success, result, object userToken);  
Construction.
```

<i>success</i>	The basic status of the command.
<i>result</i>	Refined status of the command.
<i>userToken</i>	An optional user token.

Methods

3.2.4 Class ConnectionChangedEventArgs

```
public class ConnectionChangedEventArgs : EventArgs
```

Summary

Event arguments for a connection changed event from a Wayne Device.

Properties

ConnectionState Lib.DeviceConnectionState	R	True if the device was connected.
--	---	-----------------------------------

Constructors

```
public ConnectionChangedEventArgs(Lib.DeviceConnectionState  
connectionState);  
Constructor
```

<i>connectionState</i>	The new state of the connection
------------------------	---------------------------------

Methods

3.2.5 Class EventArgs`1

```
public class EventArgs`1 : EventArgs
```

Summary

A generic event args holding an argument of some type.

Properties

Argument	R	The event argument.
----------	---	---------------------

Constructors

```
public EventArgs`1( argument);
Construction.
```

<i>argument</i>	The event argument.
-----------------	---------------------

Methods

3.2.6 Class IdentifiableEntity

```
public class IdentifiableEntity : Object
```

Summary

A class implementing the IIdentifiableEntity interface. Also contains static properties and methods that handles IIdentifiableEntity.

Fields

Nold int	A value representing a non-Id.
-------------	--------------------------------

Properties

Empty Lib.IdentifiableEntity	R	An empty IdentifiableEntity.
EntitySubType string	R	The main type of entity.
EntityType string	R	A more refined type of the entity, e.g. a specific implementation or brand.
Id int	R	The ID of the entity.
ParentEntity Lib.IIdentifiableEntity	R	Reference to a possible parent device.

Constructors

```
public IdentifiableEntity(int id, string entityType, string
entitySubType, Lib.IIdentifiableEntity parentEntity);
Construction
```

<i>id</i>	The ID of the entity.
<i>entityType</i>	The main type of entity.
<i>entitySubType</i>	A more refined type of the entity, e.g. a specific implementation or brand.
<i>parentEntity</i>	Reference to a possible parent device.

Methods

Equals

```
public bool Equals(Lib.IIdentifiableEntity entity1,
Lib.IIdentifiableEntity entity2);
```

Tests equality between two identifiable entities on the regards on the Id, EntityType, EntitySubtype, and the parent ancestry.

entity1

entity2

3.2.7 Class ObjectNotReservedException

```
public class ObjectNotReservedException : Exception
```

Summary

The ObjectNotReservedException is thrown when trying to access an object in a way that requires reservation, or if a Reserved interface is used after it has been unreserved.

Constructors

```
public ObjectNotReservedException();
```

Initializes a new instance of the class.

```
public ObjectNotReservedException(string message);
```

Initializes a new instance of the class.

message

```
public ObjectNotReservedException(string message, Exception inner);
```

Initializes a new instance of the class.

message

inner

3.2.8 Class ReserveCompletedEventArgs`1

```
public class ReserveCompletedEventArgs`1 : AsyncCompletedEventArgs
```

Summary

The EventArgs is used when ...

Properties

ReservedObject	R	The reserved object.
----------------	---	----------------------

Constructors

```
public ReserveCompletedEventArgs`1( reservedObject, object userToken);
```

Initializes a new instance of the class ReserveCompletedEventArgs where the reservation was successful.

<i>reservedObject</i>	The reserved object.
-----------------------	----------------------

<i>userToken</i>	An optional user token.
------------------	-------------------------

```
public ReserveCompletedEventArgs`1(object userToken);
```

Initializes a new instance of the class ReserveCompletedEventArgs where the reservation failed.

<i>userToken</i>	An optional user token.
------------------	-------------------------

3.2.9 Class UserTokenEventArgs

```
public class UserTokenEventArgs : EventArgs
```

Summary

An EventArgs containing a UserToken.

Properties

UserToken object	R/W	The optional user token.
---------------------	-----	--------------------------

Constructors

```
public UserTokenEventArgs(object userToken);  
Construction.
```

<i>userToken</i>	An optional user token.
------------------	-------------------------

3.3 Enumerations

3.3.1 Enumeration DeviceConnectionState

Summary

The state of the connection to a device.

Fields

Unknown	Unknown state of the connection.
Disconnected	Device is not connected.
Connecting	Trying to connect to device.
Connected	Connected to device.
Disconnecting	Disconnecting from device