Web Services-Integration with WebServicesCore on MacOS-X

This documentation was written by Andrew Lindesay (http://www.lindesay.co.nz) in 2006 as part of supported code in the LEWOStuff framework, but this material has been transcribed here. It was written around the time of WebObjects 5.2 and 5.3 on the 1.4 JVM. WebServicesCore is the framework used in MacOS-X for communications with SOAP services vended on the internet. This page describes some problems that exist at the time of writing with getting WebServicesCore to communicate properly with WebObjects' WebServices framework.

Multirefs

WebServicesCore does not appear to be able to process the 'multiref' values which are returned from the AXIS engine. To get around this, turn off multiref support in the WebObjects application. To achieve this, locate the `server.wsdd` file and the element "`parameter`" with the name "`sendMultiRefs`". Modify this element to have the value "`false`".

Nil Serialisation

This is a very simple deserialiser that can be used to deserialise to `nil`. Although this may seem less than useful, it proves very useful in the situation where you are using WebServicesCore which is a part of MacOS-X.

The WebServicesCore framework tends to serialise a `nil` argument as `<xyz xsi:type="nil"/>`. This is unfortunately missing type and so the SOAP engine is not able to instantiate the correct deserialiser for the type. The correct output would be `<xyz xsi:type="xsd:timeInstant" xsi:nil="true"/>`, but it appears that it may be difficult to get WebServicesCore to do this.

Regardless of the inbound type however, this serialiser will be able to convert the poorly encoded `nil` to `null` in the java environment.

You can use the `registerWebServicesCoreNilDeserialiserFactoryForKnownTypes` method on the class `LEWOWebServicesCoreNilDeserializerFactory` to setup easily for common known classes.

```
package nz.co.lindesay.common.webobjects;

/*
------------------------------------------------------------
LICENSE
------------------------------------------------------------

Copyright (c) 2006, Andrew Lindesay
All rights reserved.

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the
following conditions are met:

* Redistributions of source code must retain the above
  copyright notice, this list of conditions and the
  following disclaimer.

* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the
  following disclaimer in the documentation and/or other
  materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

------------------------------------------------------------
```

```
*/

/**
 * <P>This is a <B>very</B> simple deserialiser that can be used
 * to deserialise to <TT>nil</TT>.  Although this may seem less
 * than useful, it proves very useful in the situation where you
 * are using WebServicesCore which is a part of MacOS-X.</P>
 *
 * <P>The WebServicesCore framework tends to serialise a <TT>nil</TT>
 * argument as <TT><xyz xsi:type="nil"/></TT>.  This is
 * unfortunately missing type and so the SOAP engine is not able
 * to instantiate the correct deserialiser for the type.  The
 * correct output would be <TT><xyz xsi:type="xsd:timeInstant" xsi:nil="true"/></TT>,
 * but it appears that it may be difficult to get WebServicesCore
 * to do this.</P>
 *
 * <P>Regardless of the inbound type however, this serialiser will
 * be able to convert the poorly encoded <TT>nil</TT> to <TT>null</TT>
 * in the java environment.</P>
 *
 * <P>You can use the <TT>registerWebServicesCoreNilDeserialiserFactoryForKnownTypes</TT>
 * method on the class <TT>LEWOWebServicesCoreNilDeserializerFactory</TT>
 * to setup easily for common known classes.</P>
 */

public class LEWOWebServicesCoreNilDeserializer extends org.apache.axis.encoding.DeserializerImpl
{

// -------------------------------------------------

        public LEWOWebServicesCoreNilDeserializer() { super(); }

// -------------------------------------------------

    public Object getValue() { return null; }

// -------------------------------------------------

}
```

```
package nz.co.lindesay.common.webobjects;

/*
------------------------------------------------------------
LICENSE
------------------------------------------------------------

Copyright (c) 2006, Andrew Lindesay
All rights reserved.

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the
following conditions are met:

* Redistributions of source code must retain the above
  copyright notice, this list of conditions and the
  following disclaimer.

* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the
  following disclaimer in the documentation and/or other
  materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
```

```
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

-------------------------------------------------------------
*/

import com.webobjects.foundation.*;
import com.webobjects.appserver.*;
import com.webobjects.eocontrol.*;

import javax.xml.namespace.*;

import java.util.*;

/**
 * <P>See the class <TT>LEWOWebServicesCoreNilDeserializer</TT>
 * for more details about this class.</P>
 */

public class LEWOWebServicesCoreNilDeserializerFactory
        implements org.apache.axis.encoding.DeserializerFactory
{
        public final static String NAMESPACEURI_ANON = "http://anonuri/";

        private Set mechanisms = null;

// -------------------------------------------------

/**
 * <P>This method will setup the deserialiser for
 * classes commonly encountered in development
 * of web services.  This is a quick-setup that
 * might be invoked from the application setup.</P>
 */

        public static void registerWebServicesCoreNilDeserialiserFactoryForCommonClasses(String namespaceURI)
        {
                Set classesS = new HashSet();

                classesS.add(NSTimestamp.class);
                classesS.add(NSData.class);
                classesS.add(EOGlobalID.class);
                classesS.add(String.class);
                classesS.add(Number.class);
                classesS.add(Integer.class);
                classesS.add(Float.class);
                classesS.add(java.math.BigDecimal.class);
                classesS.add(java.util.Date.class);

                registerWebServicesCoreNilDeserialiserFactoryForClasses(classesS.iterator(),namespaceURI);
        }

// -------------------------------------------------

/**
 * <P>This method will setup the deserialiser for the
 * classes supplied.  This is a quick-setup that might
 * be invoked from the application startup.</P>
 */

        public static void registerWebServicesCoreNilDeserialiserFactoryForClasses(Iterator classesI, String
namespaceURI)
        {
                if(null==namespaceURI)
```

```
                        namespaceURI = NAMESPACEURI_ANON;

                QName qn = new QName("http://anonuri/","nil");
                LEWOWebServicesCoreNilDeserializerFactory factory = new
LEWOWebServicesCoreNilDeserializerFactory();

                while(classesI.hasNext())
                        WOWebServiceRegistrar.registerFactoriesForClassWithQName(
                                null,
                                factory,
                                (Class) classesI.next(),
                                qn);
        }

// -------------------------------------------------

    public LEWOWebServicesCoreNilDeserializerFactory() { super(); }

// -------------------------------------------------

    public javax.xml.rpc.encoding.Deserializer getDeserializerAs(String mechanismType)
        { return new LEWOWebServicesCoreNilDeserializer(); }

// -------------------------------------------------

    public Iterator getSupportedMechanismTypes()
        {
                if(null==mechanisms)
                {
                        mechanisms = new HashSet();
                        mechanisms.add(org.apache.axis.Constants.AXIS_SAX);
                }

                return mechanisms.iterator();
    }

// -------------------------------------------------

}
```