cansas1d binding Java

From canSAS

Documentation for the JAXB binding is spotty at this time. You can check it out with subversion:

```
svn checkout http://svn.smallangles.net/svn/canSAS/1dwg/trunk cansas-1d-standard
```

(where cansas-1d-standard is a local directory name).

Contents

JAXB

Question
: What is JAXB?

Answer
: Java Architecture for XML Binding (JAXB): http://java.sun.com/developer/technicalArticles/WebServices/jaxb/

Wow!
: Is it available for other languages?

Answer
: Ask Google. JAXB is for Java. One good advisory page: http://www.devx.com/ibm/Article/20261

Question
: How do I pull out the *I(Q)* data?

Answer
: see fragment below (gets data for desmearing)

JAXB_cansas1d_reader.java: example usage in JAVA

Here is a Java class that shows how to use the JAXB binding. Use this with any of the test data supplied with the cansas-1d-standard directory (above). By default, it shows what is in the *1998spheres.xml* example file: two samples. (You'll have to get the directory paths right until this documentation improves.)

```
/**
 *
 */
package jlake;

import java.io.File;
import java.util.List;

import javax.xml.bind.JAXBContext;
import javax.xml.bind.JAXBElement;
import javax.xml.bind.JAXBException;
import javax.xml.bind.Unmarshaller;
import cansas1d.SASdataType;
import cansas1d.SASentryType;
import cansas1d.SASrootType;
import cansas1d.SASentryType.Run;

/**
 * @author Pete Jemian
 *
 */
public class JAXB_cansas1d_reader {

	/**
	 * @param args
	 */
	@SuppressWarnings("unchecked")
	public static void main(String[] args) {
		JAXBContext jc;
		String xmlFile;
//		xmlFile = "cs_af1410.xml";
		xmlFile = "1998spheres.xml";
		try {
			// use the cansas1d/1.0 schema that is bound to a Java structure
			jc = JAXBContext.newInstance("cansas1d");
			Unmarshaller unmarshaller = jc.createUnmarshaller();
			// open the XML into a Java data structure
			JAXBElement<SASrootType> xmlJavaData = (JAXBElement<SASrootType>) unmarshaller
					.unmarshal(new File(xmlFile));
			// canSAS XML file is now loaded in memory
			SASrootType sasRootType = xmlJavaData.getValue();
			int numEntries = sasRootType.getSASentry().size();
			System.out.println("SASentry elements: " + numEntries);
			for( int i = 0; i < numEntries; i++ ) {
				System.out.println("SASentry");
				SASentryType entry = sasRootType.getSASentry().get(i);
				System.out.printf("Title:\t%s\n", entry.getTitle());
				List<SASentryType.Run> runs = entry.getRun();
				System.out.printf("#Runs:\t%d\n", runs.size());
				for( int j = 0; j < runs.size(); j++ ) {
					Run run = (Run) runs.get(j);
					System.out.printf("Run@name:\t%s\n", run.getName());
					System.out.printf("Run:\t%s\n", run.getValue());
				}
				List<SASdataType> datasets = entry.getSASdata();
				System.out.printf("#SASdata:\t%d\n", entry.getSASdata().size());
				for( int j = 0; j < runs.size(); j++ ) {
					SASdataType sdt = (SASdataType) datasets.get(j);
					System.out.printf("SASdata@name:\t%s\n", sdt.getName());
					System.out.printf("#points:\t%d\n", sdt.getIdata().size());
				}
				System.out.println();
			}

			System.out.println("the end.");

		} catch (JAXBException e) {
			// TODO Auto-generated catch block
			e.printStackTrace();
			System.out.printf("Could not open (unmarshall) XML file: %s\n", xmlFile);
		}
	}

}
```

example: how to retrieve I(Q)

Look for the line that has **Qsas[i] = sdt.getIdata().get(i).getQ().getValue();** to see the operations to unwind the data into usable double[] vectors. Pretty straightforward.

sdt
    SASdataType object
getIdata()
    /SASdata/Idata
get(i)
    /SASdata/Idata[i]
getQ()
    /SASdata/Idata/Q
getValue()
    /SASdata/Idata/Q (value, not the "unit")

GetSASdata.java

```
/**
 *
 */
package jlake;

import java.io.File;
import java.util.List;

import javax.xml.bind.JAXBContext;
import javax.xml.bind.JAXBElement;
import javax.xml.bind.JAXBException;
import javax.xml.bind.Unmarshaller;

import cansas1d.SASdataType;
import cansas1d.SASdetectorType;
import cansas1d.SASentryType;
import cansas1d.SASinstrumentType;
import cansas1d.SASrootType;
import cansas1d.SASentryType.Run;


/**
 * @author Pete Jemian
 *
 */
public class GetSASdata {


	private static SASrootType sasRoot;		// SAS data (from cansas1d/1.0 XML file)
	private static double[] Qsas;			// input Q
	private static double[] Isas;			// input I (slit-smeared)
	private static double[] Idev;			// input Idev (slit-smeared)
	private static double[] Ismr;			// calculated I slit-smeared
	private static double[] Idsm;			// calculated I desmeared
	private static double[] IdsmDev;		// calculated Idev desmeared
	private static double   slit_length;


	/**
	 * @param xmlPropertyFile
	 */
	public GetSASdata(String xmlDataFile)
	{
		// load SAS data into memory
		try {
			sasRoot = (SASrootType) loadXML("cansas1d", xmlDataFile);
		} catch (JAXBException e) {
			e.printStackTrace();
			System.out.println("ERROR: Cannot find or interpret SAS XML data file:\t" + xmlDataF
			return;
		}

		// SAS data are loaded
		// grab the SAS data to be desmeared
		int entryIndex = 0;	// /SASentry[1] : unit base in XML, 0 base in Java
		int dataIndex = 0;	// SASdata[1]
		int detectorIndex = 0;  // SASdetector[1]
		SASentryType entry = (SASentryType) sasRoot.getSASentry().get(entryIndex);
		SASdataType sdt = (SASdataType) entry.getSASdata().get(dataIndex);
		if (sdt.getName().trim().compareTo("slit-smeared") != 0) {
			System.out.println("selected SASdata element must start: <SASdata name=\"slit-smeare
			// throw something (an exception) here?
			return;
		}
		int numPoints = sdt.getIdata().size();
		Qsas = new double[numPoints];   // input Q
		Isas = new double[numPoints];   // input I (slit-smeared)
		Idev = new double[numPoints];   // input Idev (slit-smeared)
		for (int i = 0; i < numPoints; i++) {
			Qsas[i] = sdt.getIdata().get(i).getQ().getValue();
			Isas[i] = sdt.getIdata().get(i).getI().getValue();
			Idev[i] = sdt.getIdata().get(i).getIdev().getValue();
		}
		Ismr = new double[numPoints];		// calculated I slit-smeared
		Idsm = new double[numPoints];		// calculated I desmeared
		IdsmDev = new double[numPoints];	// calculated Idev desmeared
		SASinstrumentType instrument = (SASinstrumentType) entry.getSASinstrument();
		SASdetectorType detector = (SASdetectorType) instrument.getSASdetector().get(detectorIndex);
		slit_length = detector.getSlitLength().getValue();
```

test.xml

Ok, better to use SVN/TRAC for these files. This example will be improved but it proves the point.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="example.xsl" ?>
<SASroot version="1.0"
                xmlns="cansas1d/1.0"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                xsi:schemaLocation="cansas1d/1.0 http://svn.smallangles.net/svn/canSAS/1dwg/trunk/cansas1d.x
                >
 <SASentry>
   <Title>standard test dataset for Lake desmearing routine</Title>
   <Run>Run</Run>
   <SASdata name="slit-smeared">
     <Idata><Q unit="1/A">0.000371484</Q><I unit="1/cm">211554</I><Idev unit="1/cm">1874.86</Idev></Idata>
     <Idata><Q unit="1/A">0.000386255</Q><I unit="1/cm">201603</I><Idev unit="1/cm">1721.35</Idev></Idata>
     <Idata><Q unit="1/A">0.000392446</Q><I unit="1/cm">193423</I><Idev unit="1/cm">4250.66</Idev></Idata>
     <Idata><Q unit="1/A">0.000400937</Q><I unit="1/cm">205280</I><Idev unit="1/cm">1563.25</Idev></Idata>
     <Idata><Q unit="1/A">0.000415708</Q><I unit="1/cm">198569</I><Idev unit="1/cm">1446.58</Idev></Idata>
     <Idata><Q unit="1/A">0.000430391</Q><I unit="1/cm">198201</I><Idev unit="1/cm">1334.48</Idev></Idata>
     <Idata><Q unit="1/A">0.000445162</Q><I unit="1/cm">191430</I><Idev unit="1/cm">624.224</Idev></Idata>
     <Idata><Q unit="1/A">0.000451353</Q><I unit="1/cm">188171</I><Idev unit="1/cm">605.955</Idev></Idata>
     <Idata><Q unit="1/A">0.000459932</Q><I unit="1/cm">192450</I><Idev unit="1/cm">592.662</Idev></Idata>
     <Idata><Q unit="1/A">0.000474703</Q><I unit="1/cm">186589</I><Idev unit="1/cm">566.562</Idev></Idata>
     <Idata><Q unit="1/A">0.000489386</Q><I unit="1/cm">184247</I><Idev unit="1/cm">541.442</Idev></Idata>
     <Idata><Q unit="1/A">0.000504157</Q><I unit="1/cm">179316</I><Idev unit="1/cm">519.719</Idev></Idata>
     <Idata><Q unit="1/A">0.000510348</Q><I unit="1/cm">172441</I><Idev unit="1/cm">505.368</Idev></Idata>
     <Idata><Q unit="1/A">0.000518928</Q><I unit="1/cm">175473</I><Idev unit="1/cm">497.727</Idev></Idata>
     <Idata><Q unit="1/A">0.000533699</Q><I unit="1/cm">171012</I><Idev unit="1/cm">479.382</Idev></Idata>
     <Idata><Q unit="1/A">0.000548381</Q><I unit="1/cm">167081</I><Idev unit="1/cm">461.221</Idev></Idata>
     <Idata><Q unit="1/A">0.000563063</Q><I unit="1/cm">162303</I><Idev unit="1/cm">446.693</Idev></Idata>
     <Idata><Q unit="1/A">0.000569255</Q><I unit="1/cm">158623</I><Idev unit="1/cm">439.165</Idev></Idata>
     <Idata><Q unit="1/A">0.000577834</Q><I unit="1/cm">160015</I><Idev unit="1/cm">434.267</Idev></Idata>
     <Idata><Q unit="1/A">0.000592605</Q><I unit="1/cm">155494</I><Idev unit="1/cm">421.757</Idev></Idata>
     <Idata><Q unit="1/A">0.000607376</Q><I unit="1/cm">151073</I><Idev unit="1/cm">407.866</Idev></Idata>
     <Idata><Q unit="1/A">0.000622059</Q><I unit="1/cm">146555</I><Idev unit="1/cm">396.055</Idev></Idata>
     <Idata><Q unit="1/A">0.00062825</Q><I unit="1/cm">143885</I><Idev unit="1/cm">390.636</Idev></Idata>
     <Idata><Q unit="1/A">0.00063683</Q><I unit="1/cm">143034</I><Idev unit="1/cm">384.251</Idev></Idata>
     <Idata><Q unit="1/A">0.0006516</Q><I unit="1/cm">139041</I><Idev unit="1/cm">373.826</Idev></Idata>
     <Idata><Q unit="1/A">0.000666371</Q><I unit="1/cm">136947</I><Idev unit="1/cm">365.092</Idev></Idata>
     <Idata><Q unit="1/A">0.000681054</Q><I unit="1/cm">134324</I><Idev unit="1/cm">357.809</Idev></Idata>
     <Idata><Q unit="1/A">0.000687245</Q><I unit="1/cm">131392</I><Idev unit="1/cm">352.914</Idev></Idata>
     <Idata><Q unit="1/A">0.000695825</Q><I unit="1/cm">131867</I><Idev unit="1/cm">350.159</Idev></Idata>
     <Idata><Q unit="1/A">0.000710507</Q><I unit="1/cm">128250</I><Idev unit="1/cm">342.395</Idev></Idata>
     <Idata><Q unit="1/A">0.000725278</Q><I unit="1/cm">125404</I><Idev unit="1/cm">334.673</Idev></Idata>
     <Idata><Q unit="1/A">0.000739961</Q><I unit="1/cm">122355</I><Idev unit="1/cm">327.753</Idev></Idata>
     <Idata><Q unit="1/A">0.000746152</Q><I unit="1/cm">119544</I><Idev unit="1/cm">323.132</Idev></Idata>
     <Idata><Q unit="1/A">0.000754732</Q><I unit="1/cm">118748</I><Idev unit="1/cm">319.076</Idev></Idata>
     <Idata><Q unit="1/A">0.000769502</Q><I unit="1/cm">115545</I><Idev unit="1/cm">311.966</Idev></Idata>
     <Idata><Q unit="1/A">0.000784273</Q><I unit="1/cm">113124</I><Idev unit="1/cm">305.553</Idev></Idata>
     <Idata><Q unit="1/A">0.000798956</Q><I unit="1/cm">110665</I><Idev unit="1/cm">299.932</Idev></Idata>
     <Idata><Q unit="1/A">0.000805147</Q><I unit="1/cm">109629</I><Idev unit="1/cm">298.178</Idev></Idata>
     <Idata><Q unit="1/A">0.000813727</Q><I unit="1/cm">108497</I><Idev unit="1/cm">294.437</Idev></Idata>
     <Idata><Q unit="1/A">0.000828498</Q><I unit="1/cm">106067</I><Idev unit="1/cm">289.296</Idev></Idata>
     <Idata><Q unit="1/A">0.000843269</Q><I unit="1/cm">103730</I><Idev unit="1/cm">283.712</Idev></Idata>
     <Idata><Q unit="1/A">0.000857951</Q><I unit="1/cm">101055</I><Idev unit="1/cm">278.13</Idev></Idata>
     <Idata><Q unit="1/A">0.000864142</Q><I unit="1/cm">100263</I><Idev unit="1/cm">276.837</Idev></Idata>
     <Idata><Q unit="1/A">0.000872633</Q><I unit="1/cm">98975.8</I><Idev unit="1/cm">272.922</Idev></Idata>
     <Idata><Q unit="1/A">0.000887404</Q><I unit="1/cm">96617</I><Idev unit="1/cm">267.848</Idev></Idata>
     <Idata><Q unit="1/A">0.000902175</Q><I unit="1/cm">94721.9</I><Idev unit="1/cm">263.148</Idev></Idata>
     <Idata><Q unit="1/A">0.000916946</Q><I unit="1/cm">92784.7</I><Idev unit="1/cm">258.983</Idev></Idata>
     <Idata><Q unit="1/A">0.000923138</Q><I unit="1/cm">92237.6</I><Idev unit="1/cm">258.352</Idev></Idata>
     <Idata><Q unit="1/A">0.000931629</Q><I unit="1/cm">91165.3</I><Idev unit="1/cm">255.102</Idev></Idata>
     <Idata><Q unit="1/A">0.0009464</Q><I unit="1/cm">89261.8</I><Idev unit="1/cm">251.208</Idev></Idata>
     <Idata><Q unit="1/A">0.00096117</Q><I unit="1/cm">87432.7</I><Idev unit="1/cm">247.126</Idev></Idata>
     <Idata><Q unit="1/A">0.000975941</Q><I unit="1/cm">85430.8</I><Idev unit="1/cm">243.068</Idev></Idata>
     <Idata><Q unit="1/A">0.000982133</Q><I unit="1/cm">84676.8</I><Idev unit="1/cm">242.072</Idev></Idata>
     <Idata><Q unit="1/A">0.000990624</Q><I unit="1/cm">83582.8</I><Idev unit="1/cm">238.943</Idev></Idata>
     <Idata><Q unit="1/A">0.00100539</Q><I unit="1/cm">81637.9</I><Idev unit="1/cm">234.966</Idev></Idata>
     <Idata><Q unit="1/A">0.00102008</Q><I unit="1/cm">80007.9</I><Idev unit="1/cm">231.251</Idev></Idata>
     <Idata><Q unit="1/A">0.00103485</Q><I unit="1/cm">78480.9</I><Idev unit="1/cm">228.051</Idev></Idata>
     <Idata><Q unit="1/A">0.00104104</Q><I unit="1/cm">78020.3</I><Idev unit="1/cm">227.629</Idev></Idata>
     <Idata><Q unit="1/A">0.00104953</Q><I unit="1/cm">76980.6</I><Idev unit="1/cm">225.019</Idev></Idata>
     <Idata><Q unit="1/A">0.0010643</Q><I unit="1/cm">75445.5</I><Idev unit="1/cm">221.885</Idev></Idata>
     <Idata><Q unit="1/A">0.00107907</Q><I unit="1/cm">73855.7</I><Idev unit="1/cm">218.547</Idev></Idata>
     <Idata><Q unit="1/A">0.00109384</Q><I unit="1/cm">72137.1</I><Idev unit="1/cm">215.031</Idev></Idata>
     <Idata><Q unit="1/A">0.00110003</Q><I unit="1/cm">71587.7</I><Idev unit="1/cm">214.305</Idev></Idata>
     <Idata><Q unit="1/A">0.00110853</Q><I unit="1/cm">70608.1</I><Idev unit="1/cm">211.782</Idev></Idata>
     <Idata><Q unit="1/A">0.0011233</Q><I unit="1/cm">69061.3</I><Idev unit="1/cm">208.548</Idev></Idata>
     <Idata><Q unit="1/A">0.00113807</Q><I unit="1/cm">67611.7</I><Idev unit="1/cm">205.543</Idev></Idata>
     <Idata><Q unit="1/A">0.00115284</Q><I unit="1/cm">66343.4</I><Idev unit="1/cm">202.81</Idev></Idata>
     <Idata><Q unit="1/A">0.00115903</Q><I unit="1/cm">65983.2</I><Idev unit="1/cm">202.392</Idev></Idata>
```

- This page was last modified 15:13, 16 May 2008.