

Project Requirement: Java Object Persistence

3.4 Design Constraints, DC2

Native Serialization

- Using FileOutputStream to write and read to files
- Can be messy and we would have to ensure a reliable structure to each file
- Would also require us to write custom methods to write and then read the files in our given structure

XStream

- Saving objects as formatted XML
- Much easier to read and write to (contains methods that automatically create XML from a Java Object and then convert XML to a Java Object)
- Easier to debug with
- Does have advanced support for JSON if required
- Much nicer to read when it comes to debugging files
- Relatively quick, would be ideal for saving after each user's turn for example

GSON

- Uses JSON to save objects and data
- Provides simple functions to convert objects to and from JSON

XStream Testing

I have done some testing using XStream as it seems like a good library to use for storing Java objects easily. As seen below, it's super easy to convert an object to XML which we can then save into a file. It is also just as easy to convert back from XML to a Java object.

Additionally, it gives us great control over how we want the file to be structured as we can assign aliases to certain classes, making it much easier to read for debugging etc.

```
import com.thoughtworks.xstream.XStream;

public class Main {

    public static void main(String[] args) {

        // create a school
        University aber = new University( uniName: "A Cool School", uniPostcode: "SY233FL");

        // create a new instance of the Person class
        Person student = new Person( fName: "Ash", lName: "Bagnall", lAge: 21, aber);

        // convert to XML
        XStream xstream = new XStream();

        // give the class an alias
        xstream.alias( name: "person", Person.class);
        xstream.alias( name: "university", University.class);

        // convert to XML
        String xml = xstream.toXML(student);

        // print out XML
        System.out.println(xml);

    }
}
```

Figure 1 Java Code for converting an Object to XML using XStream

```
<person>
  <firstName>Ash</firstName>
  <lastName>Bagnall</lastName>
  <age>21</age>
  <uni>
    <name>A Cool School</name>
    <postcode>SY233FL</postcode>
  </uni>
</person>
```

Figure 2 Output from converting an Object to XML using XStream