

# From Spreadsheet to Schema

---

*This article was originally published in the November 2006 edition of Government Computing*

## **Abstract**

All parts of the public sector are putting their services online and increasing the amount of information transferred to and from external bodies. In doing this, they are using e-GIF compliant XML schemas to define the messages that flow between systems. However, there is still frequently a disconnect between the business people defining the information required and the technical people who then have to create the detailed specifications, including XML schemas.

One of the benefits of XML is the ease with which it can be manipulated. This article describes how the Higher Education Statistics Agency and Boynings Consulting used this flexibility to allow business people to work with spreadsheets, with which they were already familiar, and turn these into schemas and web and print versions of specifications for implementation of the HESA Student Record. This in turn built on past work that Boynings Consulting had done to create XML test cases for taxation systems from spreadsheets, creating hundreds of test cases in a matter of minutes.

## **Text**

There are many occasions when one needs to create XML, but those who are specifying the requirements are not familiar with the technology. Although XML developers can work from a written specification, changes become tedious and error-prone manual work. It is clearly preferable if the XML can be generated automatically from a specification created using a familiar technology such as a spreadsheet.

HM Revenue and Customs has been doing this for several years to create test cases for the self-assessment online tax return. Each year, around 200 test cases are specified, and Boynings Consulting uses in-house tools to convert these to XML that is consistent with the tax return schemas for that year. Those providing the specifications do not even need to know the XML element names used - they work from the box numbers familiar from the paper forms and the tools provide the conversion required in a matter of minutes.

Although it is very useful to be able to generate a complete set of test cases in a matter of minutes, this is a fairly specialised requirement. Far more common is the need to generate XML schemas compliant with the e-GIF. This was the requirement at the Higher Education Statistics Agency. HESA collects student records for statistical purposes from Higher Education institutions throughout the UK. Currently, these are collected in a flat file format, but there is a need to move to the more hierarchical structure of XML.

Specifications must be provided for this so that the HE institutions sending the data know the format required, and there was a desire to provide these specifications both in a format suitable for printing and as an interactive online application. Validation must also be provided for several types of rules, and as much of the data as possible should be capable of being validated by the sender of the information.

To specify all the information required for the documentation and schemas, two spreadsheets were developed. The first contains the basic information required for the schemas, including any references to existing e-GIF schemas. The second contains lists of valid entries required for certain data items. For example, a list of ethnicity codes.

The first spreadsheet is based around a data model, so each item is either a field (something that contains data) or an entity (which contains either other entities or fields or both). Each of these has a name and a reference to the entity of which it forms a part. There is also a long name for the documentation. To create the schema, each field has associated information such as a base data type (which might be, for example, a string, date or an e-GIF data type such as an address), a maximum length and the minimum and maximum number of times that the field can be included within the entity. An entity has a subset of this information. Both also have metadata such as related fields, change management information and the countries within the UK to which the field applies (any or all of England, Scotland, Wales and Northern Ireland). Supplementary files can be used to provide structured information such as a description and examples. This supplementary information can contain formatting such as italic text, tables and lists. Business rule information can also be added, for processing automatically into a rules-based language that can be used directly for testing of a complete student record.

The second spreadsheet is simpler, and just has a list of valid entries where a field contains data from a list. Each entry has an associated label and the field or fields to which it applies. Some entries can apply to more than one field. For example, the code for Germany might apply to both domicile and nationality, while that for England can only represent a domicile and that for United Kingdom only a nationality.

Once the spreadsheets have been created, conversion to the schemas and web-based documentation is a simple two-stage process. The first stage converts the spreadsheets to a simple XML format. There are a few ways of doing this, and which is most appropriate depends on the circumstances of the individual project. The process can be as simple as clicking on a button in the spreadsheet.

The second stage is again a simple click of the mouse. This will run the Boynings Consulting tools to create XML schemas, business rules coded in a language for automated processing and the interactive web-based documentation. Another simple operation using standard software creates the print documentation.

**HESA Student Record Specification**  
**Fields for Welsh institutions only**  
**Bilingual ITT marker**

[return to index](#)

Short Name	BITTM
Type	field
Description	This field is a one digit field giving details about whether the ITT course is bilingual.
Applicable to	NI Scotland Wales
Coverage	
Base Data Type	BilingualITTCodeType
Field Length	1
Part Of	<b>Course</b>
Minimum Occurrences 0	0
Maximum Occurrences 0	1
Related Fields	
Reason Required	To indicate whether ITT is bilingual.
Examples	
Notes	In this context: <ul style="list-style-type: none"> <li>◆ Bilingual means English/Welsh for institutions in Wales.</li> <li>◆ Bilingual means English/Gaelic for institutions in Scotland.</li> <li>◆ Bilingual means English/Gaelic for institutions in Northern Ireland.</li> </ul> For further guidance on the completion of this field please refer to HEFCW/SE/DELNI.
Owner	
Version	
Based On	
Schema Components	Element: <b>BITTM</b> Data type: <b>BilingualITTCodeContentType</b>
Change Management Notes	
Valid Entries and Labels	0 Course does not lead to a formal certificate of bilingual education nor is it designed to enable students to teach bilingually. 1 Course does not lead to a formal certificate in bilingual education but is designed to enable students to teach bilingually. 2 Course leads to a formal certificate of bilingual education.

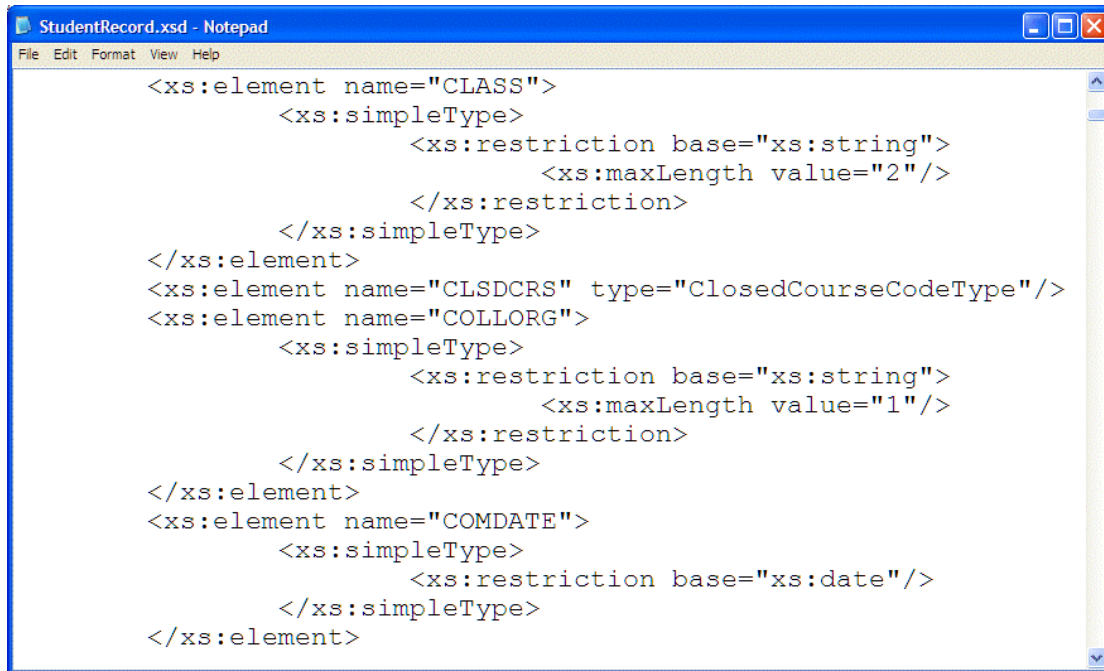
[return to index](#)

Find:  Find Next Find Previous Highlight all Match case

Done

Figure 1 The web-based documentation

In HESA's case, the intermediate XML format works well with their metadata processes and version control systems, and they have decided to maintain this rather than the spreadsheets. They will write a simple forms-based front-end to achieve this. This flexibility always exists - either the spreadsheet can be used as a master document, or it can be used to get things going quickly, then dedicated tools developed for future maintenance.

The image shows a Notepad window titled "StudentRecord.xsd - Notepad". The window contains XML schema code. The code defines several elements: "CLASS" with a simple type restriction to a string of length 2; "CLSDCRS" with a type of "ClosedCourseCodeType"; "COLLORG" with a simple type restriction to a string of length 1; and "COMDATE" with a simple type restriction to a date. The code is as follows:

```
<xs:element name="CLASS">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="CLSDCRS" type="ClosedCourseCodeType"/>
<xs:element name="COLLORG">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="COMDATE">
  <xs:simpleType>
    <xs:restriction base="xs:date"/>
  </xs:simpleType>
</xs:element>
```

Figure 2 An extract from the schema

The benefits of this system to HESA are:

- hierarchical data can now be sent
- two formats of specification and the schemas are all generated from a single source
- senders of the data can validate it before transmission using simple, free tools
- e-GIF schemas are seamlessly integrated into the application

So there it is. The system users or analysts write a specification as a spreadsheet, with supplementary documentation in simple files. A few minutes later, you have XML schemas, coded business rules and full web-based and print documentation. What could be easier?

*Paul Spencer is a Director of Boynings Consulting Ltd and a Consultant on the use of XML in Government. He is certified as an Expert Practitioner by the e-GIF Accreditation Authority. Paul developed the tools and carried out both projects mentioned in this article.*

5 October 2006

[paul.spencer@boynings.co.uk](mailto:paul.spencer@boynings.co.uk)