

# BoXCRIp PROJECT REPORT – CASE STUDY

## 1. SUMMARY

The Bolton XCRI project (BoXCRIp) set out to develop an XCRI compliant course advertising feed of all course promoted by the University of Bolton including undergraduate, postgraduate, short and CPD courses. The BoXCRIp is a sub project of a wider University course information project that aims to redevelop and refresh the course catalogue and to link the collection of course advertising information closer to the course development and maintenance cycle.

## 2. BACKGROUND

The University of Bolton is undertaking strategic repositioning, building on its curriculum and research strengths to develop into a new model called the Professional University. As well as refining undergraduate and postgraduate programmes around the new and “para” professions, the University is expanding continuous professional development (CPD) provision to support the regional and national skills agenda.

The University needed a new course catalogue which was capable of giving rich information to learners about learning opportunities and which could represent the full range of learning opportunities from programme to a single module of CPD. The previous web based course catalogue, was based on *flat html files* produced by hand and suffering from all the familiar problems of lack of timeliness, accuracy and consistency.

The University has a number of disconnected databases which contain information about course programmes: course and module validation database, SITS student record system, short course database, prospectus files, web course catalogue, course handbooks, module timetable. As these have been created and are maintained by different business functions to support their own activities, our current processes involve the passing of course information from one area to another by document. Information is re-keyed, is not adequately linked and cross verified. There was no common course reference system is not in place.

To address this, a Course database project was started to consolidate information for advertising both validated and proposed courses. This project was extended with the BoXCRIp project, in order to create an XCRI conforming data feed for course advertising.

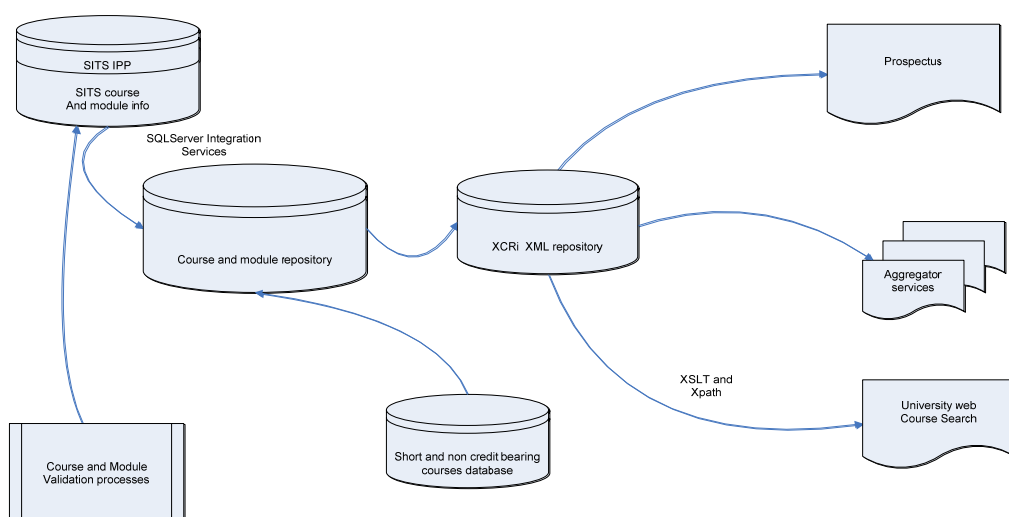
### 3. OBJECTIVES

The objectives of the BoXCRIP project were

- To establish an easily searchable and comprehensive, web-based, University catalogue for the purposes of marketing courses to potential students, by June 2007.
- To test and use the XCRI schema to present the course catalogue in a way that can be consumed by external services including UCAS, mash-ups and aggregators.
- To eliminate the need to duplicate data entry relating to course advertising information
- To identify and take the opportunities to improve the automated collection of this information by describing the existing processes through which course promotional information is built up, proposing improvements and creating the collection mechanisms.
- To create a single repository of information relating to the advertising and promotion of courses, including undergraduate, postgraduate, CPD modules, and short courses

### 4. TECHNICAL APPROACH

At the inception of the course database project, it was proposed to use the SITS module, Institutional Published Programmes (IPP) as the means of generating the course catalogue. The advantage of this was seen to be the preservation of SITS as the authoritative course database containing current programmes and modules. The initial position was to increase the range of course types and course status within SITS to include course proposals and short courses which are currently managed separately. Normally this course information would be accessed by users via the SITS e:Vision portal. As SITS Vision does not currently support extended XML feeds, the bid proposal included using Microsoft SQL Server Integration services to feed a repository that included non-validated courses such as short course offerings e.g. AutoCAD for Architects. The XCRI CAP feed would then be generated from this.



#### 4.1 Change of Design

Early on in the project planning and design process, it became clear that there were a number of problems for the University of Bolton in using IPP as a central and authoritative course catalogue which related to business processes, user preference/prejudice and maintenance of the integrity of the SITS data. The issues that arose were:

1. A large number of courses and modules that are proposed and begin validation process do not progress beyond ideas. The Registry who administer the student records system were keen to maintain the integrity and purpose SITS course database and did not want redundant data in the system.
2. The lead officer in Admissions and Recruitment was strongly opposed to using SITS for course advertising based on her previous experiences of SITS and what she felt to be the inadequacy of the user interface and 'usability' of SITS.
3. The primary client group Admissions and Recruitment, felt strongly that the system had to render primary (core details about the course) and secondary information (student experience, additional information etc) relating to the course/module. There was concern that IPP could not adequately represent all the primary data required and link with the secondary data adequately.
4. SITS Vision and IPP does not yet support extended XML and so was unsuitable as the XCRI feed. Also there is little normalisation in the SITS data structure, and Design Team university felt uneasy in interrogating the existing course data with confidence.
5. Course lifecycles are such that courses and modules flagged with a status of "current" may not be recruiting students and those

So whilst modifications could be made to the configuration of SITS Vision to address some of these issues, there was strong resistance from the primary administrative groups to any changes that would increase the administrative overhead across admin functions e.g. new status codes, additional housekeeping to strip out redundant data.

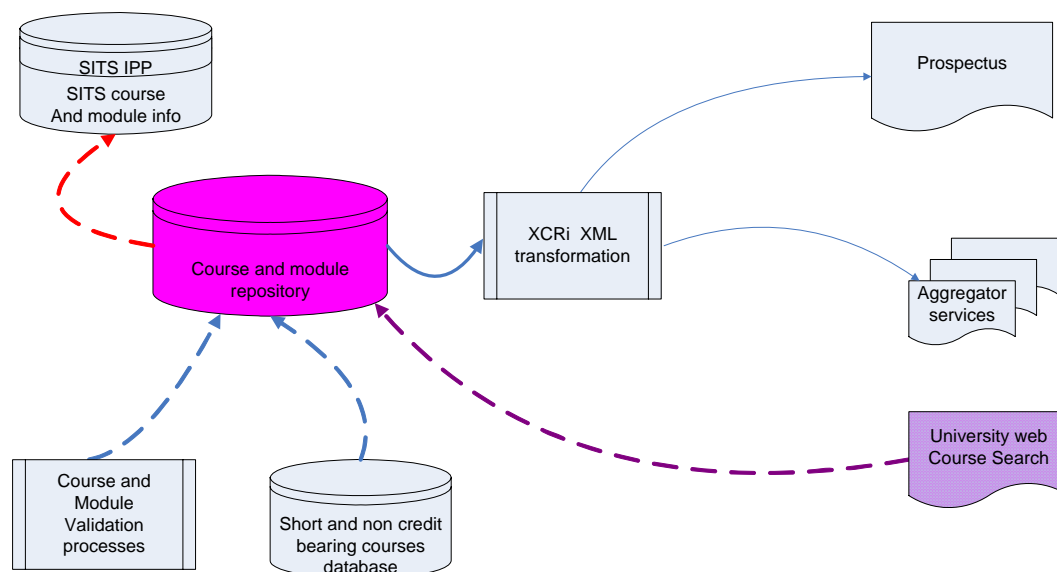
The Design Team therefore decided to create a repository of currently recruiting and pre-production courses separate from SITS IPP from which to generate the XCRI compliant feed. The derived brief was for a database, written in MS SQL Server 2005, which had to support processes and data around course advertising including:

- acquiring course and module advertising data during from the course validation process
- drill down from Programme and Course to Module
- Ability to represent all forms of course, credit and non-credit bearing, short courses, CPD etc
- Course search based on module (i.e. this module can be studied as part of these courses)

- Rendering of secondary course information

The Design Team also set design constraints of developing the system using reusable classes and objects so that elements of the system can be repurposed as the full design encompassing course validation etc evolves.

The resulting design is show below



## 5. RESULTS

In creating the Course database schema, the Design Team spent considerable time

- Discovering what course data was held by the University of Bolton.
- Investigating the structure of programmes, course, modules, and short courses
- Reviewed the course data requests from outside bodies including GMSA, Hot Courses, UCAS and Learndirect
- Looking at the business processes by which course data is acquired/built up
- Identifying the attributes associates with each stage of course development

It became clear that course structures were complex to represent due to a number of factors

1. Modularisation creates complexity - a substantial proportion of courses have variable module diets consisting of core and elective modules.
2. A large number of programmes present with interim awards e.g. HND + 1 Year
3. The same course may run in a variety of modes (e.g. full time, part time) and in a number of locations (e.g. on campus, at partner colleges, overseas) and each of these has to be represented
4. The same programme/course may need to be rendered in different ways e.g. Undergraduate Degrees as CPD, for different target markets
5. Some courses/programme/module names have been changed without being revalidated so the approved course name may not be the advertised name.

The process of proposing and developing a course requires additional information that serves no purpose beyond the validation stage e.g. market research on the need for the course, the cost and revenue model for the course. Whilst this is not required for XCRI CAP, the supporting course database has been designed to allow this data to be attached.

The Design Team has produced

1. A full database schema describing the make up of programmes, course, modules etc including the attributes required to produce an XCRI feed
2. SQL database, classes and stored procedures relating to a comprehensive course database
3. A web based .NET application (C#) in for entering and maintaining course data in central location.
4. Mapped and implemented an xml feed of the xcri course data and successfully submitted this to the XCRI test aggregator

## 5.1 Reflections on XCRI-CAP 1.0 and suggestions for XCRI-CAP 1.1

XCRI CAP 1.0 has been matched against the schema and the course advertising requirements of the University. The following recommendations for XCRI CAP 1.1 are made based on the identification of our own and partner requirements.

### ***Related Courses***

The University of Bolton would value a way of representing related courses through the XCRI CAP i.e. courses that are close or in the family of courses e.g. BSc (Hons) Sound Engineering & Design, BSc (Hons) Music Technology

### ***Keywords***

We think Keywords are important to improve the course search. Often the course title may not be enough to identify the relevance of the course for the searcher e.g. Bsc (Hons) Mechatronics, control systems, engineering etc. If the XCRI feed is transformed into a search page, keywords would be a massive advantage.

### ***Employment Sector***

The University would welcome Employment Sector being added to the XCRI CAP as a way of tailoring course searches for sector skill council purposes etc. The problem may be that organisations such as Learndirect also want to classify courses by sector according to their own criteria.

## **5.2 WORK TO BE COMPLETED**

At the end of the BoXCRI project the remaining objectives are still in progress

1. Transferring all course data to the new database to generate a comprehensive XCRI feed (programmed to complete by 20<sup>th</sup> May 2008)
2. Developing a course search tool (by May 2008)
3. Go live with the new course database and XCRI feed (25<sup>th</sup> May)

## **6. CONCLUSION**

Through the BoXCRI project we quickly established that not much of the existing course catalogue data at Bolton could be easily harvested and represented through XCRI. The course descriptions produced for different stages of the course production cycle do not necessarily lend themselves to course advertising. There was a the legacy of poorly maintained course advertising information: where elements of course specifications had changed; where additional information was sent out by departments; where the communication on which courses were and weren't running had dried up.

The structure of running courses is represented well on the SITS student record system but this does not contain any detailed description of the course suitable for advertising. When we investigated the origins of the course specification, we found that often the original description of the course established in the course validation process, is often incomplete or not updated. There are multiple points at which information on a course might be change but not be consistently recorded.

We therefore set about establishing a central course repository, capable of supporting the data requirements of courses through every stage of the production cycle upto the point of enrolment. This has been tested against a sample of different courses and an XCRI feed has been generated from this. In our wider project plan we understood the time, effort and difficulty there would be in obtaining and verifying the course advertising data and we are only now getting close to re-entering the bulk of course data

## **APPENDIX A.** ASP CODE USED TO PRODUCE SAMPLE XCRI-CAP 1.0 FEED

```
<?xml version="1.0" encoding="UTF-8" ?>
<%
Dim dMonth, dDay, dHour, dMinute, dSecond

If datepart("m", Now) < 10 Then
    dMonth = "0" & datepart("m", Now)
Else
    dMonth = datepart("m", Now)
End If

If datepart("d", Now) < 10 Then
    dDay = "0" & datepart("d", Now)
Else
    dDay = datepart("d", Now)
End If

If datepart("h", Now) < 10 Then
    dHour = "0" & datepart("h", Now)
Else
    dHour = datepart("h", Now)
End If

If datepart("n", Now) < 10 Then
    dMinute = "0" & datepart("n", Now)
Else
    dMinute = datepart("n", Now)
End If

If datepart("s", Now) < 10 Then
    dSecond = "0" & datepart("s", Now)
Else
    dSecond = datepart("s", Now)
End If
%>
<catalog generated="<% Response.Write datepart("yyyy", Date) & "-" & dMonth & "-" & dDay
& "T" & dHour & ":" & dMinute & ":" & dSecond %>" xmlns="http://xcri.org/profiles/catalog"
xmlns:xcri="http://xcri.org/profiles/catalog" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:ukrlp="http://www.ukrlp.co.uk/"
xsi:schemaLocation="http://xcri.org/profiles/catalog cap.xsd">
<provider>
    <identifier>Bolton.ac.uk</identifier>
    <identifier xsi:type="ukrlp:UKPRN">10006841</identifier>
    <name>University of Bolton</name>
```

<description>At the University of Bolton we are a community of professionals committed to developing our students for their professional lives.

With a student body of around 9,000 you will get the best of all worlds at Bolton learning at a university where no one feels anonymous, in a thriving town close to the big cities of Manchester and Liverpool.

Our teaching quality has consistently won the highest ratings possible from the Government's quality control agency and we have professional accreditation for more than 30 of our teaching programmes.

</description>

<street>Deane Road</street>

<town>Bolton</town>

<postcode>BL3 5AB</postcode>

<url><http://www.bolton.ac.uk></url>

<image

src="http://www.bolton.ac.uk/SharedContent/SharedImages/Logos/universityofbolton.gif"

alt="Logo" title="Welcome to the University of Bolton" />

<!--#include file="conn/dbconn.asp"-->

<%

Response.Buffer = true

Response.ContentType = "text/xml"

Function ApplyXMLFormatting(strInput)

    If IsNull(strInput) Or strInput = "" Then

        Else

            strInput = Replace(strInput,"&", "&amp;")

            strInput = Replace(strInput,"'", "&apos;")

            strInput = Replace(strInput,"\"", "&quot;")

            strInput = Replace(strInput, ">", "&gt;")

            strInput = Replace(strInput,"<","&lt;")

            ApplyXMLFormatting = strInput

        End If

    End Function

    Dim query, query2, query3, query4, RS, RS2, RS3, RS4, CourseId, StartDate, EndDate, dStartDate, dEndDate

    query = "SELECT tbl\_Programme.ProgrammId, tbl\_Course.CourseTitle, tbl\_Course.CourseOverview, tbl\_Award.AwardDesc, tbl\_CourseType.CourseType, "

    query = query & "tbl\_Course.CourseId "

    query = query & "FROM tbl\_Programme INNER JOIN "

    query = query & "tbl\_Course ON tbl\_Programme.CourseId = tbl\_Course.CourseId INNER JOIN "

    query = query & "tbl\_CourseType ON tbl\_Course.CourseTypeId = tbl\_CourseType.CourseTypeId INNER JOIN "

```
query = query & "tbl_ProgrammeDate ON tbl_Programme.Programmeld =
tbl_ProgrammeDate.Programmeld INNER JOIN "
query = query & "tbl_Status ON tbl_Programme.StatusId = tbl_Status.StatusId INNER
JOIN "
query = query & "lkp_AwardProgramme ON tbl_Programme.Programmeld =
lkp_AwardProgramme.Programmeld INNER JOIN "
query = query & "tbl_Award ON lkp_AwardProgramme.AwardId =
tbl_Award.AwardId "
query = query & "WHERE (tbl_Programme.RetiredDate IS NULL) AND
(tbl_Status.StatusDesc = 'Active') AND (tbl_ProgrammeDate.ProgrammeStartDate >
GETDATE()) AND "
query = query & "(lkp_AwardProgramme.RetiredDate IS NULL) "
```

```
Set RS = Connect.Execute(query)
```

```
If (RS.BOF AND RS.EOF) Then
```

```
Else
```

```
Do While not RS.EOF
```

```
%>
```

```
<course>
```

```
<identifier><% =ApplyXMLFormatting(RS("ProgrammId")) %></identifier>
```

```
<title xml:lang="en"><% =ApplyXMLFormatting(RS("CourseTitle")) %></title>
```

```
<description><% =ApplyXMLFormatting(RS("CourseOverview"))
```

```
%></description>
```

```
<image
```

```
src="http://www.bolton.ac.uk/SharedContent/SharedImages/Logos/universityofbolton.gif"
```

```
alt="Logo" title="<% =ApplyXMLFormatting(RS("CourseTitle")) %>" />
```

```
<qualification>
```

```
<title><% =ApplyXMLFormatting(RS("AwardDesc")) %></title>
```

```
<level><% =ApplyXMLFormatting(RS("CourseType")) %></level>
```

```
</qualification>
```

```
<%
```

```
query3 = "SELECT tbl_Programme.ProgrammeCode,
tbl_Duration.DurationDesc, tbl_Programme.ProgrammeTitle,
tbl_Programme.ProgrammeDesc, "
```

```
query3 = query3 & "tbl_Attendance.AttendanceDesc, tbl_Location.LocationId,
tbl_Location.LocationTitle, tbl_Location.LocationDesc, tbl_Location.LocationStreet, "
```

```
query3 = query3 & "tbl_Location.LocationTown, tbl_Location.LocationPostcode,
convert(varchar, tbl_ProgrammeDate.ProgrammeStartDate, 103) AS [ProgrammeStartDate],
convert(varchar, tbl_ProgrammeDate.ProgrammeEndDate, 103) AS [ProgrammeEndDate] "
```

```
query3 = query3 & "FROM tbl_Programme INNER JOIN "
```

```
query3 = query3 & "lkp_DurationProgramme ON tbl_Programme.Programmeld =  
lkp_DurationProgramme.Programmeld INNER JOIN "  
query3 = query3 & "tbl_Duration ON lkp_DurationProgramme.DurationId =  
tbl_Duration.DurationId INNER JOIN "  
query3 = query3 & "tbl_Attendance ON tbl_Programme.Attendanceld =  
tbl_Attendance.Attendanceld INNER JOIN "  
query3 = query3 & "lkp_LocationProgramme ON tbl_Programme.Programmeld =  
lkp_LocationProgramme.Programmeld INNER JOIN "  
query3 = query3 & "tbl_Location ON lkp_LocationProgramme.LocationId =  
tbl_Location.LocationId INNER JOIN "  
query3 = query3 & "tbl_ProgrammeDate ON tbl_Programme.Programmeld =  
tbl_ProgrammeDate.Programmeld "  
query3 = query3 & "WHERE tbl_Programme.Programmeld = " & RS("Programmeld") & ""
```

```
Set RS3 = Connect.Execute(query3)  
If (RS3.BOF AND RS3.EOF) Then  
Else  
Do While not RS3.EOF  
  
If IsNull(RS3("ProgrammeStartDate")) Or RS3("ProgrammeStartDate") =  
"" Then  
dStartDate = ""  
Else  
StartDate = Split(RS3("ProgrammeStartDate"), "/")  
dStartDate = StartDate(2) & "-" & StartDate(1) & "-" &  
StartDate(0)  
End If  
  
If IsNull(RS3("ProgrammeEndDate")) Or RS3("ProgrammeEndDate") = ""  
Then  
dEndDate = ""  
Else  
EndDate = Split(RS3("ProgrammeEndDate"), "/")  
dEndDate = EndDate(2) & "-" & EndDate(1) & "-" & EndDate(0)  
End If  
  
%>  
<presentation>  
<identifier><% =ApplyXMLFormatting(RS3("ProgrammeCode"))  
%></identifier>  
<description><% =ApplyXMLFormatting(RS3("ProgrammeDesc"))  
%></description>  
  
<start><% =dStartDate %></start>  
<end><% =dEndDate %></end>
```

```

        <duration><%           =ApplyXMLFormatting(RS3("DurationDesc"))
%></duration>
        <studyMode><%         =ApplyXMLFormatting(RS3("AttendanceDesc"))
%></studyMode>
        <venue>
            <identifier><% =RS3("LocationId") %></identifier>
            <name><%           =ApplyXMLFormatting(RS3("LocationTitle"))
%></name>
            <description><%   =ApplyXMLFormatting(RS3("LocationDesc"))
%></description>
            <street><%         =ApplyXMLFormatting(RS3("LocationStreet"))
%></street>
            <town><%          =ApplyXMLFormatting(RS3("LocationTown"))
%></town>
            <postcode><%
=ApplyXMLFormatting(RS3("LocationPostcode")) %></postcode>
        </venue>

        <%

            query4 = "SELECT EntryRequirementsDesc "
            query4 = query4 & "FROM tbl_EntryRequirements "
            query4      =      query4      &      "WHERE
tbl_EntryRequirements.Programmeld = " & RS("Programmeld") & ""

            Set RS4 = Connect.Execute(query4)
            If (RS4.BOF AND RS4.EOF) Then
            Else
                Do While not RS4.EOF
                %>
                <entryRequirements><%
=ApplyXMLFormatting(RS4("EntryRequirementsDesc")) %></entryRequirements>
                <%
                RS4.Movenext
                loop

            End If
            RS4.Close

        %>
    </presentation>
    <%
    RS3.Movenext
    loop

```

```
End If
RS3.Close

query2 = "SELECT tbl_SubjectArea.SubjectAreaDesc "
query2 = query2 & "FROM lkp_SubjectAreaCourse "
query2 = query2 & "INNER JOIN tbl_SubjectArea ON
lkp_SubjectAreaCourse.SubjectAreaId = tbl_SubjectArea.SubjectAreaId "
query2 = query2 & "WHERE (lkp_SubjectAreaCourse.CourseId = " &
RS("CourseId") & ")"

Set RS2 = Connect.Execute(query2)
If (RS2.BOF AND RS2.EOF) Then
%>
    There are no records to display.
<%
Else
    Do While not RS2.EOF
%>
<dc:subject><%
    =ApplyXMLFormatting(RS2("SubjectAreaDesc"))
%></dc:subject>
<%
    RS2.Movenext
    loop
End If
RS2.Close
%>
</course>
<%
    RS.Movenext
    loop
End If
RS.Close
Connect.Close
Set Connect = Nothing

%>
</provider>
</catalog>
```