[1 Introduction 1](#_Toc34959674)

# Introduction

## DXS

The DXS system links into the Batch import facility described above. It has the added requirement that, after the customer has been matched, an XML file has to be sent back to DXS, with our CustomerId included, where it could be found.

To achieve this end, a special DXS form is accessible from the Maintenance menu. It has two functions.

The first function reads the DXS XML. It does this in two steps. First it assigns a unique id to it for identification purposes, and then it converts it to the ProposedCustomer format XML file.

You will notice that I use an intermediate core datatable to load the DXS stuff, and then I use a foreach loop to load that data into a in core table that is a mirror image of the MIMSDXS table in the database. You might ask why the MIMSDXS table is not an exact image of the original DXS file. My answer is that the original DXS file has its won peculiarities, and that I could not read that XML into an in core datatable that was not exactly in the same format than the XML data we receive from DXS. See the sub heading on how I created an in core table from the XML.

The second function reads the ProposedCustomer XML, once it has been matched, and updates the original data in the MIMSDXS table. From the MIMSDXS table, it generates XML, which can be sent back to DXS.

### How to create database tables from XML

From MIMSUserReport.xml, one can create MIMSUserReport.xsd.

This can be used, in turn, to create a dataset, called “newdataset”. I guess you do this via the designer?

Where is this dataset stored?