DATABASE PROCEDURES (VERSION 1.112)

DIGITAL ARHIVISTS
ARCHAEOLOGY DATA SERVICE
https://archaeologydataservice.ac.uk/

Created date:	26 January 2012
Last updated:	18 October 2019
Review Due:	31 March 2021
Authors:	Jen Mitcham, Stewart Waller, Michael Charno, Kieron Niven, Ray Moore, Jenny O'Brien, Teagan Zoldoske, Digital Archivists
Maintained by:	Digital Archivists
Required Action:	
Status:	Live
Location:	https://archaeologydataservice.ac.uk/advice/PolicyDocuments.xhtml

1. Purpose of this document

1.0.1 This documents current ADS procedures for production of dissemination and preservation copies of databases. It contains a list of current dissemination and preservation formats and how to migrate files to required formats. More information on this data type, can be found in the G2GP for Databases http://guides.archaeologydataservice.ac.uk/g2gp/DbSht_Toc.

2. Formats

Offered format	Accepted	Preservation	Presentation	Notes
Microsoft Access **.mdb**	YES	Comma Separated Value **.csv**	Comma Separated Value **.csv**	Proprietary Microsoft format. Not read by many database packages. We can accept Access 97 and above, but support for Access 97 will be short-lived. Ideally, versions earlier than 97 should be migrated prior to deposit. Where this is not possible, OpenOffice might offer a possible migration route.[1]
Microsoft Access 2007 onwards **.accdb**	YES	Comma Separated Value **.csv**	Comma Separated Value **.csv**	The .accdb format was first introduced in Access 2007 and continued with Access 2010 and was developed in order to include enhanced functionality over the previous .mdb format.[2]

[1] MS Access is the GUI, the actual database engine is MS Jet (msjet##.dll, where ## is the version number, plus the various msjt*.dll files) which is loaded by Access.Inevitably there have been updates to Jet (Access 2.0 uses version 2.5, Access 95 uses version 3.0, Access 97 uses version 3.5, Access 2000 – 2003 use version 4.0 etc). There have been format changes to the .mdb file, particularly between versions 2 and 3 of Jet and versions of Access using Jet versions below 3 are probably inaccessible.

[2] It is arguable that, from a robust database design standpoint, additional functions and enhancements such as Multivalued Fields and Attachments only increase the difficulty in preserving

Microsoft Excel **.xls**	YES	Comma Separated Value **.csv**	Comma Separated Value **.csv**	Although a proprietary Microsoft format, the Excel .xls format is widely used and can be imported by a number of third-party applications
Microsoft Excel 200 onwards **.xlsx**	YES	Comma Separated Value **.csv**	Comma Separated Value **.csv**	A relatively new format from Microsoft, released with Office 2007. They chose to develop their own specification (OOXML) rather than use the existing ODF international standard. The format consists of human readable XML files packed with other content within a single zipped file.
dBASE .**dbf**	YES	Comma Separated Value **.csv**	Comma Separated Value **.csv**	Ashton-Tate's dBASE format but they only registered the name, not the format.It is now a generic format referred to as xBase and is used by a number of databases and read by even more. The file structure is simple and stable and consequently all versions can be handled.
OpenDocument Database **.odb**	NO			
Paradox Database **.db**	NO			

such databases. It is also worth noting that, although the format is the default in Access 2007 and 2010, files created in Access 2010 may not be completely compatible with Access 2007. The .accdb format, as with previous .mdb files, continues to be based on the Jet Database Engine.

Delimited text (including **.txt, .csv, .tsv**	YES	Comma Separated Value **.csv**	Comma Separated Value **.csv**	
Exchange Formats				
JavaScript Object Notation JSON **.json**	YES	JavaScript Object Notation JSON **.json**	JavaScript Object Notation JSON **.json**	Human-readable text to transmit data objects consisting of attribute–value pairs and array data types. JSON files should be accompanied by 'JSON schema definition file' (?JSD or JSON).
XML **.xml**	YES	XML **.xml**	XML **.xml**	Should be accompanied with appropriate XSD file.
Resource Description Framework RDF **.rdf**	YES	Resource Description Framework RDF **.rdf**	Resource Description Framework RDF **.rdf**	RDF/XML is a syntax, defined by the W3C, to express (i.e. serialize) an RDF (Resource Description Framework) graph as an XML document

3. Documentation / Metadata

3.0.1 Alongside the standard metadata for files, the following additional documentation is required for any database. The current metadata template is available from the Guidelines for Depositors.[3]

Element	Description
Table Documentation	
Table Name	
Table Description	
Primary keys	

[3] https://archaeologydataservice.ac.uk/advice/guidelinesForDepositors.xhtml.

Foreign keys	
Row Count	
Field Documentation	
Field Name	
Field Description	
Field Data type	
Field length	
Supporting Documentation	
Entity relationship diagram	
Supporting documentation	codes used, units of measurement used in specific fields

3.0.2 This table is derived from the G2GP

http://guides.archaeologydataservice.ac.uk/g2gp/DbSht_2.

4. Accessioning checks

4.1 Checks

- Do we have necessary documentation?
- Scan for data consistency. For special collections (online search), these issues should be flagged up to the depositor at accession.
- Presence of forms/sql etc - we can't do anything with these
- Orphaned tables and records, or empty or unrelated tables.
- Check that referential integrity enforced for related tables. Run queries looking for duplicates and orphan records and highlight any issues with depositor. Where controlled vocabularies used to complete fields, make sure that the vocabularies controlled! We may want to create drop down lists of these terms in order to make the field searchable but this looks messy if there are consistency issues with data entry.
- If the database is supposed to link to a collection of images. Make sure there is a field in the database that holds the exact image name of the associated image file We don't want to have to do too much work at this end ensuring that images and database match up
- Check tables for duplicated rows, they are likely to result in incorrect or excessively duplicated records between linked tables. You can check by running a query such as SELECT field1, field2, … FROM table GROUP BY field1, field2, …

- Check text fields where the length of the data is the same as the field length – it may indicate truncated values. A macro for displaying this information is provided in the footnotes.

4.2 Significant properties

- The actual data within the database - including field headings and the values themselves. Associated with this is the use of special characters in the dataset, from ampersands to Greek characters (common in dating/scientific data) which must be identified and preserved.
- Relationships between tables. It is important that the relationships between tables and sheets are understood and documented.

4.3 File-naming

4.3.1 Where possible files should retain the same name as the original. On occasion (and normally for dissemination), it may be necessary to create different versions of the same file. In these cases a logical naming strategy should be used, and should be accompanied by explanation in the Processes section of the CMS.

4.3.2 Where multiple tables of a database are being converted, folder name should reflect the name of the original database and the filename the name of the table the data came from, for example:

- mydatabase-table_name.csv.

4.3.3 It may however be necessary to change the table names. For example an MS Access data table may have the name 'Catalogue Flaked Lithics, Sand test pits, early survey sites etc.' which isn't ideal and causes errors when you try and export it as a delimited text file. Alternatively, a table called ‘descriptions’ that is clearly a typo and will look bad when listed as a download on our web pages.

4.3.4 Any changes to database or table names within the preservation or dissemination versions should be recorded in the Process metadata.

4.3.5 Then placed in the appropriate location (see below).

5 How to convert files

Starting Format	Procedure	End Format	Checks
Microsoft Access **.mdb** Microsoft Access 2007 onwards **.accdb**	**ADS Toolkit** {internal access only} This has a very good function to export all Access tables as delimited text. It also removes carriage returns for you. This does not support Microsoft Access 2007 onward .accdb.	Comma Separated Value **.csv**	• Check all tables have been exported • Check row counts after export • Check text fields where the length of the data is the same as the field length – it may indicate truncated values. • Check for embedded new lines, tabs and quotes, these may corrupt exported delimited text files. VB code • Scan text fields for the presence of common delimiter characters ('(comma)','(pipe)', etc.). These will determine the need for text qualifiers • Check any special characters have been preserved. Scan text fields (VB code) for CP1252 characters (ASCII values between 128 and 161 - smart quotes, some accented characters, dashes etc. There is also a handy lists here and here). These are not preserved in CSVs that use ANSI encoding. If present you'll need to convert the file encoding to UNICODE UTF-8 (see below). If exporting from Access or OpenOffice software there should be options in the save/export facility to allow you to choose this setting.

			• Similarly, scan text fields (VB code) for characters beyond ASCII 167. If present then accented or other characters exist (as used in French, Gaelic, and Ancient Greek and so on) and the Code Page or language of the original data must be determined. ANSI files preserve these characters, but it is worth recording that they exist within the documentation. • **Whatever changes/replacements you do must be recorded in the Process record for the collection as this is editing the deposited data**
OpenOffice Database **.odb**	**OpenOffice/Microsoft Access** As with exporting from Access, export each table as a CSV ensuring field headings (and text qualifiers) are included.	Comma Separated Value **.csv**	As above
Dbase file **.dbf**	**Microsoft Access** Import into Access and use the 'Export' function in Access to export each table as a CSV ensuring field headings (and text qualifiers) are included. For some reason access will only recognise the file if the filename is 8 characters or less, otherwise it displays an error message. To get round this, using a working copy, rename the dbf file before importing into access.	Comma Separated Value **.csv**	As above
Various/ ArchSearch/ Special Collections (loaded into ArchSearch/Oracle)		Oracle tables (if requested also	• Carriage Returns. These are a pain, software can assume that they signify a new line of data. Carriage returns need to be removed from your file before loading. • Check ansi files do not preserve certain Windows-1252 characters. Files containing any troublesome characters should thus be

		Comma Separated Value **.csv**)	saved with UNICODE encoding. If exporting from Access or OpenOffice software there should be options in the save/export facility to allow you to choose this setting (see above). If the data is to be used in an online interface, you'll need to replace these with html equivalent.[4] • The decision as to whether to disseminate the data behind special collections is dependent on the depositor. If they wish for data to also be available in this way then follow the instructions for the above. • Datasets incorporated into Archsearch do not need to be made available for download (unless specifically requested, which is very rare). • **Whatever changes/replacements you do must be recorded in the Process record for the collection as this is editing the deposited data.**

[4] A good example of best practice can be seen in the Castelporziano archive. Here, the table (which also doubled as desciptive metadata for photos) was highly stylised, with characters such as •. Therefore the downloadable data was saved as UNICODE, and the same data imported into Oracle with • replaced by &#8226.

6 Storage

6.1 Storing data

6.1.1 Data should be stored in appropriately named folders, as described in the ADS Repository Operations manual.[5] Any directory structure from the SIP should be retained in the AIP. In some cases editing/restructuring may be necessary, but such restructuring should be recorded in the Processes section of the CMS.

6.1.2 Otherwise, store data in one of the following directory structure:

/preservation
    /{original_structure}
        mydatabase-table1.csv
        mydatabase-table2.csv
        mydatabase-table3.csv
        /documentation
            entity_relationship_diag.tif

/dissemination
    /{original_structure}
        mydatabase-table1.csv
        mydatabase-table2.csv
        mydatabase-table3.csv
        /documentation
            entity_relationship_diag.tif

6.1.3 Online database (Special Collections): The decision as to whether to disseminate the data behind special collections is dependent on the depositor. If they wish for data to also be available in this way then follow the instructions for the above.

6.1.4 ArchSearch: Datasets incorporated into Archsearch do not need to be made available for download

6.2 Storing metadata

6.2.1 File and metadata should be stored in an appropriate archival format with the preservation/dissemination files in a "documentation" folder within the requisite folder, for example:

/preservation
    /{original_structure}
        mydatabase-table1.csv

[5] https://archaeologydataservice.ac.uk/advice/PolicyDocuments.xhtml#RepOp.

mydatabase-table2.csv
mydatabase-table3.csv
/documentation
entity_relationship_diag.tif
abbreviations_codes_list.pdf

/dissemination
/{original_structure}
mydatabase-table1.csv
mydatabase-table2.csv
mydatabase-table3.csv
/documentation
entity_relationship_diag.tif
abbreviations_codes_list.pdf

7. Creating and linking objects in the OMS tables

7.0.1 See Match Objects Overview for general overview {internal access only}
see also CMS-OMS TableStructure for MOS data requirements {internal access only}

8. Tech watch / things to note

9. Archival notes

10. References