

Basic Archaeological Database Design

Alanna Ossa and Dr. Arleyn Simon
School of Human Evolution and Social Change, Arizona State University
Archaeological Research Institute
SHESC Graduate Workshop Series

Please do not cite without permission of the authors. Comments and suggestions very welcome. Email Alanna Ossa at aeossa@asu.edu for questions, comments, and suggestions.

Table of Contents

I. Introduction

II. What is a Database?

III. Database Creation as a Component of Research Design

IV. Basic Archaeological Database Architecture

V. Relational Database Design

VI. Archaeological Issues in Database Organization

VII. Guidelines for Data Fields

VIII. Guidelines for Analysis Forms

VIV. How to handle Database Changes

I. Introduction

The storage and retrieval of archaeological data within computer databases is a basic component of modern archaeological research. The introduction of relational database technology during the 1970's provided an important addition to the archaeologist's repertoire of analytical tools. However, it was the introduction of a variety of RAD (Rapid Application Design) products, such as Microsoft Access, FoxPro, DBASE, and FilemakerPro, each readily available for personal computers, that gave most archaeologists easy access to the creation and maintenance of electronic relational databases. The architecture of these database programs allowed access through a wide variety of 'front end' user interfaces, from off-the-shelf database applications to GIS programs to web browsers, increasing their utility to archaeological projects. As database technology evolved, archaeologists have been eager to adopt new methods in the interest of increasing the efficiency and flexibility of their datasets (Carroll 1988). However, archaeologists have been slow to take complete advantage of relational architecture, in spite of the fact that it often mirrors the way archaeological data are structured. As we explain below, this neglect is more often a function of how archaeological data "best practices" are handed-down rather than classroom taught, a situation which we propose to directly address.

A key aspect of modern archaeological research and design pedagogy should include the creation of electronic data both for current analyses and the creation of legacy data for future researchers. Even for those working outside of impending modern construction, ongoing looting, and natural resource mining, everyday destructive forces are always significant components of immediate research concerns. Therefore, lack of effective data management and research design has potentially dire consequences for the long-term use and availability of archaeological data. At ASU, Sylvia Gaines was one of the first faculty members to establish a course in data management as a basic part of research design in archaeology. We view this manual as a means of continuing her legacy in emphasizing effective electronic data management and the creation of analytically flexible electronic data for the long-term management and use of archaeological datasets.

Representing an early use of relational databases in archaeology, the Shoofly Village dataset derives from research conducted by the Arizona State University field school near Payson, Arizona in the 1970's – 1980's and related CRM work in nearby Star Valley. The Roosevelt Platform Mound Study dataset was the product of a major research project conducted through the auspices of Arizona State University. We use several legacy datasets, specifically, Shoofly Village and the Roosevelt Platform Mound Study, from Arizona State University's Archaeological Research Institute as our main sources of examples. In the following sections, we outline general guidelines to the successful design of a relational database applicable to many aspects of archaeological research.

II. What is a Database? (THIS SECTION IN PROGRESS...not used in Grad workshop)

A database is a prescribed storage area in the computer's file system managed by a database program. Most database programs have a graphical user interface much like word processing or spreadsheet processing programs and are relatively easy to use to start creating a database project.

Relational databases have multiple tables of data that are designed to interact with each other for efficiency of data organization and minimum of repetition of data entry and data error.

1. What is a table?

A table is a data structure that has columns and rows that describe relationships between sets of attributes (columns) and records (rows).

2. How do you create tables?

To create tables, you will need to know ahead of time the kinds, quantity, and relationships of information you want to manage (i.e. Strings, Numbers, Characters, etc.). First, give your table a name, (hopefully implying the entity it represents) and define its variable attributes (columns).

These relationships will guide you in structuring your tables and their relationships with each other.

3. What is a **primary key**?

A feature of relational databases is that all tables must have some kind of attribute (column) that links all of the attributes in the table to one another. Also, each value for that attribute (in each row) must be unique. This unique attribute is the index for that particular table. An example of an index that could be used in a table with individual employee data is a social security number. Each social security number is unique to each person, so it may be used as an identifier of that person in other tables.

4. How do tables link to each other?

The answer lies in **foreign keys**. Suppose you have a provenience table and a ceramics table, the primary key for the provenience table is a provenience key number, and the primary key for the ceramics is a ceramic number. How do you go about linking the two tables with one another? In order to link the tables, you could use the provenience key number in the ceramics table as the foreign key that links ceramics with their respective provenience locations.

III. Database Creation as a Component of Research Design

Archaeological field work is the culmination of considerable prior research and significant planning on the part of researchers. Research design has been an important component of the curriculum in academic programs for at least the last thirty years (Redman 1972). Additionally, due to increased governmental and agency oversight, most archaeologists, as contractors or grantees, have already been through a research review process before the first shovel ever hits the ground. Archaeological field methods based on specific research goals are decided upon prior to beginning fieldwork; however, the basics of electronic data recording and analyzing are not always a part of these field requirements. This may change because the national science foundation has added a requirement for research grants that includes a data access plan. In following the precept that publicly funded research projects should include data access and recovery that can be acquired by the scholarly community; we argue that database design should be a significant part of pre-fieldwork planning. An effective research design should include careful consideration of the kinds of variables that a researcher wants to record and analyze electronically.

A large part of designing an archaeological database revolves around deciding on the types of information the researcher wants to preserve and the attributes he or she has created for data collection and analysis and the *relationship* between these attributes. Obviously, basic decisions about which characteristics to record or not record for collected artifacts (such as ceramic type, dimensions of artifacts, and provenience) have long term implications for how a dataset may be used later. Less obvious is the way in which artifacts *should* be related to each other abstractly and how they should be recorded in relation to other archaeological analytical units like excavation units, features, and sites. As we will propose, decisions on defining these relationships are critical to subsequent data analysis and the creation of legacy data for curation purposes. Finally, database design should aim to minimize potential errors during data entry and subsequent data modification, and to maximize analytical flexibility—including the potential for other, later researchers to use the data. Because of these factors, planning the architecture of your database should be a significant aspect of the total research design and planning.

Important questions to ask before you collect data:

1. What attributes do I need to answer my research question/s?
2. How do the attributes of my data relate to one another?
3. What are my main organizing principles? Provenience?

Tasks to complete as a part of research design:

1. Pick a portable and flexible platform.
2. Design DB structure based on attribute relationships.

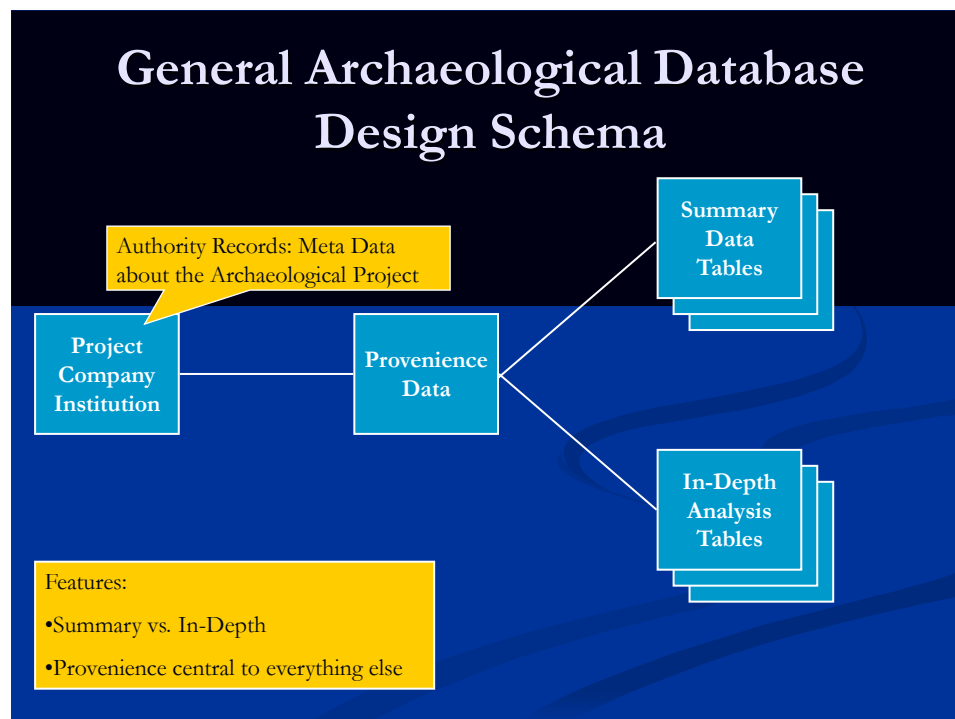
3. Design field work forms and analysis forms based on attribute retrieval.
4. Plan for archiving and managing electronic data.

IV. Basic Archaeological Database Architecture

Most archaeological databases are idiosyncratic structures, built from many different research designs and traditions with what seem like an insurmountable gulf of variation among them. However, at an abstract level, most archaeological databases share the same basic data structure. In describing this bedrock architecture, we will illustrate how each seemingly unique database can be considered comparable and how to maximize this comparability for maximum research flexibility and ease of inter-project research. When properly implemented, databases should provide for long-term use, future changes in research design, and cross project comparisons.

From our experience as researchers, we recognize that the core of archaeological databases is provenience. All of our field work and analyses are firmly rooted in the recording/collection of materials in association with provenience data. Typically, the analytical level at which location is recorded varies widely from project to project but the basic goals of relational comparison (whether they be intra-site or inter-site) are met. At the next level of data collection, researchers typically record summaries of artifact counts (ceramics, lithics, botanical remains, faunal remains, etc.) per provenience unit. Typically, more in-depth analyses of artifact sets or classes also take place and although each set of analyses is connected to the provenience data, the data is no longer summarized by unique provenience locales per row.

Perhaps the best way to illustrate the architecture is through the general schema of an archaeological database shown below (Figure 1. General Archaeological Database Architecture).



This schema is intended to provide a general understanding of how relational databases and research design intersect; it is not intended to be an actual representation of database tables and their connections with each other. However, a basic understanding of how archaeological datasets fit together will illustrate how database design enables research flexibility. As you can see from the schema, provenience is always connected to the materials collected or recorded, no matter the analytical level of the data (summary or in-depth). These two analytical concepts, summary and in-depth, are fundamental to the vast majority of archaeological analyses and are also the single largest cause of confusion and mistakes in the application of relational technology to archaeological databases.

By analytical flexibility we refer to the multiple ways in which archaeological data may be stored. Specifically, artifact analyses often produce both summary counts of artifact types per provenience and in-depth secondary stage analyses of the materials which, while connected to provenience data, are not summarized into rows of unique provenience units. Instead, an in-depth secondary stage analysis will be summarized into the most unique set of data that is being analyzed. These different levels of analyses do not mean that each data table cannot be reconciled with each other using relational technology. For example, a ceramic temper analysis of only diagnostic ceramics can easily be accommodated in the same database as summary counts of “types” by provenience with minimal hassle. This is a key point and under-emphasized in many relational approaches: *You do not need to just record summary data per provenience unit in order for your database to be “relational”*. Instead, whichever analytical level you require for different artifact analyses can be matched up with your provenience data with a few simple steps.

Analytical Flexibility

You do not need to record summary data per provenience unit in order for your data to be “relational”.

Summary Table

Ceramics	Lithics	Shell	Groundstone	Provenience Key
23	2	2	1	5
35	4	6	0	13
45	2	2	3	23

Provenience Data

In-Depth Table

Ceramic Type	Vessel Form	Vessel Part	Artifact_ID	Count	Provenience Key
Fine Orange	Bowl	Rim	1	23	5
Black on Red	Bowl	Rim	2	54	5
Buffware	Jar	Body	3	72	23

Provenience Data

This brings us to a vital point about relational technology and data analyses: analytical flexibility is not a drawback but rather a key feature of relational design.

Different research questions require flexible scales of data recording and analysis. Although archaeological research questions and methods fluctuate, database design can weather generational changes in fields of archaeological inquiry. As we hope to demonstrate in the following descriptions and examples, archaeological databases, if carefully designed, can allow for considerable variation among artifact analyses and enable researchers to change the scale at which they analyze their data.

V. Relational Database Design

In creating a database, many different problems can arise if the database structure is inefficient. These problems arise in the addition, change, or deletion of data from a flat file table in which related data must be repeated within rows. Flat files result in data redundancy, updating problems, and integrity constraints (the necessity of having a unique identifier). In order to combat these problems, relational databases have specific design considerations which involve breaking up data tables into a structure that better reflects their internal relations in a process called normalization. First, let's consider the following example of a flat file in order to demonstrate what these problems are and how relational technology helps solve them (Figure 3).

Figure 3.

Site Name	Unit #	Level	Artifact_ID	Artifact Name
Test Site	1	2	1	Red Ceramic Bowl Rim
Test Site	1	2	2	Black Ceramic Jar Body
Test Site	1	2	3	White Ceramic Plate Body

In the example ceramics table shown above, you can see that the provenience columns are repeated for each row. All of the relevant data about the artifacts are stored in one big table, often with information stored within the same cell. This sounds okay until you want to start changing information about your record entries. For example, if you were to change any one of the provenience variables, like site name for instance, you would have to change site name in every single row in which the faulty site name occurred. The potential data problems do not end there. Notice that several different attributes of each artifact are stored in the same column. This means that sorting the data by just bowls, for example, might prove more difficult given that this information is actually stored with other data within the same table cell. All of these data problems are simple examples of modification anomalies and can be handled by having a more efficient data structure that captures the optimal structure of the data. In relational database technology this is referred to as normalization. Most archaeologists utilize relational structures in the analyses of their data collections already when they evaluate the multiple spatial and temporal dimensions of their data. Creating and maintaining relational structures in archaeological databases should be a nice fit with the analyses that most researchers are interested in pursuing.

Thematic Tables

The basic idea behind relational database normalization is to limit each table to a single theme such as ceramics that can be related, through various internal table identifiers, back to other tables such as provenience. In theory, the relationships among tables in a relational database are logical models of the relationships between the variables that are defined for each table (Codd 1990). There is no definitive guide to the degree of normalization one *must* use. The degree to which data is broken down into individual themes is dependent upon one's data organization needs. With most modern programs designed for data entry, entry forms can be used to prevent inaccurate data or data repetitions through programming controls within the tables or forms themselves rather than relying on the database structure. For example, if you wanted to limit data entry within a table to a set list of values (often referred to as look-up tables) you could simply limit that field's entry to the matching list. As long as the values are restricted to a list, then table break-downs can be limited.

Data organization and analysis problems

One example of data organization causing analysis problems is the use of the type-variety system in ceramics classification schemes. In the classic type-variety application, vessel form is stored apart from surface treatment. In Mesoamerica, where often whole "types" of ceramics are often of one particular form (decorated types are typically all bowl forms for example) this need not be fatal. However, by storing the data separately as part of the classification scheme, one is prevented from future reanalysis or an in-depth study of ceramic production that would require one to know how many fine orange ceramics were bowls and how many were jars. Vessel form is not the only intra-type variation obscured by the specific type-variety application of ceramic typologies, often appliqué or intra-vessel variations are also subsumed under the major categories. Since ethnographic studies of ceramic production often show that variation within "types" often increases near the source of production, data storage techniques could be obscuring potentially valuable information for the archaeologist. This data problem has been recognized for the type-variety application (see Smith 1979); however, we want to emphasize it as primarily a data organization problem. Clearly, the ways in which data is stored during analysis itself is vital to the kinds of research we can conduct with our data. A clear demonstration of what normalization breakdown actually looks like will help in discussing the ways in which specific research projects can benefit.

Low Level Normalization Example

Before

Site Name	Unit	Level	Artifact_ID	Artifact Description
Test Site	1	2	1	Red Ceramic Bowl Rim
Test Site	1	2	2	Black Ceramic Jar Body
Test Site	1	2	3	White Ceramic Plate Base

After

Provenience Data

Provenience Key	Artifact ID	Count	Ceramic Type	Vessel Form	Vessel Part
2	1	23	1	1	1
2	2	35	2	2	2
3	3	52	3	3	3

Ceramic Type	Name
1	Red
2	Black
3	White

Vessel Form	Name
1	Bowl
2	Jar
3	Plate

Vessel Part	Name
1	Rim
2	Body
3	Base

As the figure demonstrates, a researcher could sort this particular data structure on different combinations of vessel part, form, and ceramic type with ease. This would not be possible in the “before” example as the field artifact description combines 3 different kinds of information into one field.

Examples of the Misuse of Normalization in Archaeological Databases

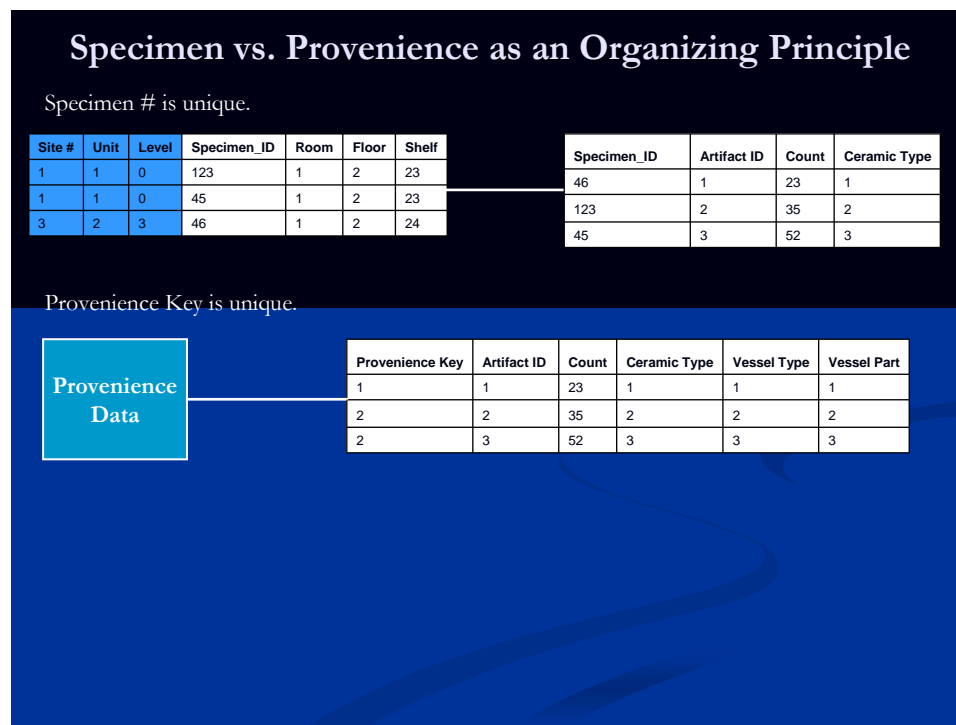
In legacy archaeological databases, researchers typically run into problems when data has been structured in ways that make it difficult to look up artifacts and collections using a different organizing principle than the original database creators had in mind. For example, if the most refined provenience connected to each artifact is a site identifier, than it would be impossible to organize artifacts along a different sub-site provenience variable, such as a house-lot. Establishing analytical flexibility means enabling the connection of artifacts to the most detailed level that researchers can practically imagine using within budgetary and time restrictions for data analysis. So in addition to considering your own needs, you also need to consider the long-term usability of your dataset.

Often, the enthusiastic application of relational technology results in the creation of separate tables of data results in nonsensical separations. Basic misunderstandings in the process of how and why variables are broken down into separate tables in relational database technology can lead to bad design or database structures whose related variables cannot be joined with other vital information. In archaeological databases, this disjoint often occurs between the provenience tables and the artifact analysis tables. Occasionally, mistakes in data structure will even lead to users being unable to join secondary data analyses to provenience tables. Separate tables should always have either a foreign key (key to a provenience table) or, in the case of many to

many relationships, an intersection table that includes both the analysis key and the appropriate provenience key.

VI. Archaeological Issues in Database Organization

Archaeological databases and data are often stored within electronic collections whose purpose is twofold: research data analysis and artifact storage location. This leads to a common problem for archaeological data analysis; the data is stored by what amounts to an artifact location (like a bag number) rather than via archaeological provenience. Unfortunately, this common state of affairs often leads to archaeological databases built upon a “specimen” number or “bag” number that cannot easily be abstracted to the original provenience of the artifact or artifacts in question. For example, this often means that what should be the provenience heart of the archaeological database is actually a specimen log table.



So, for example, a provenience/specimen table could look something like the following figure.

Specimen #	Site #	Unit	Level	Type
1	921	1	0	ceramics
2	921	1	0	lithics
3	921	1	1	ceramics

Figure 3: Specimen/Provenience Table

Notice that the unique provenience information, i.e., the Site #, Unit, and Level is repeated for the first two specimens. They are repeated because artifact or specimen numbers are often separated by artifact type. So, although the first two rows are from the same provenience, they are separated due to the fact that as separate artifact types, they are assigned to separate specimen numbers; typically this occurs because different artifact types are analyzed separately. This repetition means that errors in data entry are much more likely to occur. Also, changes and deletions made to provenience data will also result in repeated or inaccurate records within your specimen table. The best way to handle this problem is to make sure that your provenience data is entered only *once* and that it can be automatically updated within every data table in which it is used. Most major archaeological projects establish a method of entering in provenience data as part of their lab procedures. Small projects or individuals should establish a similar method in order to secure their data.

The following is an example of a work flow in excavation process designed for use in the RPMS (Roosevelt Platform Mound Study):

Field work involves the creation of a LEVEL FORM. The level form includes the basic minimum provenience information (the unique combination of excavation unit and level). When the artifacts are checked into the lab, the level form data is entered in *once* and the unique identifier for this information is attached to the specimen bags that are being checked in for analysis. In all future analysis of artifacts, a MPN (minimum provenience number) is always attached to the artifacts.

*For future reference, this minimum provenience unit can be arbitrary; often the analytical levels (such as soil levels, floors, etc.) are combinations of arbitrary levels. The important point is that one can always link the artifacts back to a recognized 3 dimensional (in the case of excavation) or 2 dimensional (in the case of survey) locations.

VII. Guidelines for Data Fields

Building a data dictionary or storing Meta data about your database is one of the basic components of database design and upkeep. A data dictionary is where a researcher documents for each variable defined its meaning, what kind of information it should contain, and the format that information should take (e.g., a real number, an ID consisting of four numerals a period and three numerals, open text, etc.). The researcher should create it in the initial stages of database design.

Another important aspect of any database design is providing consistent naming conventions for data fields and the information they are to contain. In other words, a data field variable like "type" or "locus" should refer to the same kind of information wherever it is used. Consistent naming avoids later confusion in the meaning of the information in variable fields, simplifies the creation of a data dictionary, and makes relating different kinds of information easier and more meaningful. Consistent naming

conventions and the careful maintenance of a data dictionary for the Shoofly dataset allowed for subsequent improvements to the structure of the database, providing more efficient storage and retrieval of data (Carroll 1988). For example, in the early data collection forms, codes were used to label different kinds of artifacts in order to conserve limited and costly disk space. Without such the proper documentation of a data dictionary, many of these codes could not have been converted to their full text labels in the modern formats.

During the course of data collection and analysis the meanings and format of variables may need to be changed and new variables created. Such changes should also be documented. In the Shoofly example, changes in collection strategies, data recording methods, recorded attributes, and analysis steps that altered the original database were better facilitated by careful documentation of the database's current variables at the time. Data dictionaries and a record of changes to data comprise an important component of what has come to be called metadata. The lack of such Metadata may result in confusion and even misinterpretation when a dataset is re-examined for later analysis. Thoughtful and well-documented design, however, will allow the researcher to make the most out of their data without sacrificing either research flexibility or future data accessibility

VIII. Guidelines for Analysis Forms

From our personal experiences, the single most important step in designing your database is when you design your artifact analysis forms. This ubiquitous archaeological activity will have the most far-reaching impact on the usability of your dataset both for your analyses and for future analyses. However, it has been our experience that many archaeological analysis forms are the result of legacies from prior research experiences of the form designer rather than based solely on the current research design of a project. This can lead to mismatches in data analyses and larger program research goals.

Legacy data forms, particularly those that may pre-date the more heavily electronic era, often include data storage techniques, such as cross tabs, in which summary counts with two related variables of different data attributes are recorded in one easy-to-read sheet.

Crosstabs and Summaries: Forms from a Pre-Electronic Era

Summary Form Example

Ceramics	Lithics	Shell	Groundstone	Provenience Data
23	2	2	1	5
35	4	6	0	13
45	2	2	3	23

Provenience Data

Crosstab Form Example

Provenience Data					
Ceramic Type	Bowl	Jar	Plate	Comal	Totals
Orange	3	2	0	2	7
Red	3	1	0	2	6
Cream	4	2	4	2	12

Each row ideally contains the most unique combination of desired information about ceramics.

Electronic Form Example

Provenience Key	Artifact ID	Vessel Form	Ceramic Type	Vessel Type	Count
1	1	1	1	1	23
2	2	2	2	2	35
2	3	3	3	3	52

Cross tab forms were a common technique prior to electronic databases partly because they enabled researchers to quickly look at data forms and get a lot of basic information immediately. By producing cross tab data forms researchers could perform basic Exploratory Data Analysis simply by looking at their data forms. However, storing data as cross tabs in an electronic format makes it impossible to sort the data in different ways. Analytical flexibility is low. Making your data more flexible means does not preclude being able to produce data cross tabs, it simply means that you can sort your data in more than one format.

We submit that a better form design approach is one that combines prior experience and knowledge with a thoughtful understanding of the goals of the current research. The most important thing to remember when designing an artifact analysis form is this: *am I recording all of the attributes that I need to answer my research questions?* Equally important is to consider whether the *relationship* between the attributes recorded in your data analysis form matches research design goals. Once you have designed a form that records the most unique combination of information you wish to store, you can use sub queries and group –by functions to restore your data into summary format for analyses.

Example:

Provenience Key	Artifact ID	Vessel Form	Ceramic Type	Vessel Type	Count
1	1	1	1	1	23
2	2	2	2	2	35
2	3	3	3	3	52

Group by ROW (Provenience Key), Group by COLUMN (Vessel Form), SUM VALUE (Count)

Produces this table:

Provenience Key	Bowls	Jars	Plates
1	23	0	0
2	35	52	0

You could also group columns together such as “VesselForm”&”CeramicType”&”VesselType” and perform Group by functions on these as well.

In summary, there is no limitation to the combinations of attributes you can perform, so long as you have recorded them separately. The key to having flexible data sets is to record your attributes (such as ceramic type, form, and part) into separate attribute fields. They can always be combined fairly easily in relational database management systems for whatever your research needs may be in the future.

Questions for workshop:

What is wrong with the crosstab form? How would you use it to compare to other data?

How can you turn your electronic form that stores your data in unique combinations of attributes into summary forms?

VIV. How to Handle Database Changes

Besides providing an effective means of maintaining data, database structure can also ensure the integrity of the dataset through efficient design. Many people see computer database construction as the last step in the data collection process, but modern relational database technology is designed specifically to combat the modification anomalies and redundancy that can occur with more traditional spreadsheet approaches to data entry. In this way, the creation of collection forms for fieldwork and analysis forms for artifact analyses are part and parcel of the successful creation and maintenance of an archaeological dataset. In both the Shoofly and RPMS projects, databases were made conformable to research goals as part of the research process at different stages in the life of the research project. The experiences of both projects can effectively demonstrate how database design, creation, and use are applied in active research programs and legacy data efforts. Although data fields will often have to change, following the course of changes in research design, these changes need not harm your dataset if simple rules and procedures are adopted.

The Inevitable Database Changes: How to Insulate Archaeological Data

A necessary feature of archaeological fieldwork is the need to add and accept changes in data analysis or even field forms. These changes have a cascading effect from field methods and crew re-training, down to lab procedures and data analysis. This is, obviously, a necessary, if occasionally headache inducing part of scientific archaeological analysis. Making changes to the way your data is collected and analyzed need not be fatal to your archaeological database. Establishing clear database protocols will minimize future problems. In certain kinds of analysis, in which the data is complex and multiple variables are recorded, minute alternations are often necessary. This is a common issue in complex datasets, such as faunal data. The danger in keeping electronic records and forms, even when paper forms are used, is that electronic data tables are changed with the stroke of a keyboard, without a paper trail necessarily being created. This means that an analyst making changes could theoretically add fields without future database users realizing that the analysis methods changed partway through the field project. The potential ramifications for future and current analyses are obvious, but well worth explaining with a common archaeological example.

Example from major CRM project:

In the course of the ceramic analysis for a project, refinements often take place in the ways in which analysts are able to recognize different types of ceramics. As most archaeologists can attest, ceramic typologies are a work in progress. Refinements, even in major projects, are quite typical as researchers learn more about the ceramics from continued study. In our example case, a refinement of a particular type of ceramics led to the creation of two different types, a prehistoric and a historic designation of a type (in this case, the type Red Indeterminate was subdivided into prehistoric Red and Historic Red). However, although the changes were known by the ceramic analysts, the categories were simply added to the ceramic analysis tables without creating a new table. Subsequently, a multivariate statistical analysis of the ceramic data showed some patterning based solely upon the *analysis changes* in the ceramic category rather than a *real pattern*. Luckily, because records were kept of when the analysis change occurred and the ceramic data was time-stamped, the error was caught and the data analysis re-run.

In conclusion, potential for analysis error occurs much more easily in the era of electronic data file management and analysis. Fortunately, records can be kept about changes in analyses and researchers can avoid the problems of seeing patterns based on analysis changes rather than real archaeological patterns. However, the issues of adding attribute fields and modifying attributes in artifact analyses means that a best practices method of re-analyses in any artifact category should probably result in an entirely new electronic analysis table.

Acknowledgements

The authors would like to acknowledge the Arizona State University field schools (1970s-1980s) under the direction of Charles Redman who excavated at Shoofly Village

with permission from the Tonto National Forest. We would also like to thank Mary Carroll for her help with the latest translation of the Shoofly Village dataset to modern database formats. We thank Steve Savage, Tony Stratton, and Michael Smith for their valuable advice and comments. However, any shortcomings remain our own.

Additional Technical Background

The lingua franca used to define and manipulate data within all database products (for relational databases) is SQL (Standard Query Language). SQL itself is not a complete language like C or COBOL; it does not tell the system how to get what you want, it merely asks for what you want. SQL is used to create tables (places to store information), enter, retrieve, and correct data within a database. A database (made up of component tables) is a prescribed storage area in the computer's file system managed by any RDBMS (Relational Database Management System) product (such as SQL server). You can use the power of SQL to create complex queries in easy to understand statements that can be used on any RDBMS. Although a complete understanding of SQL is not required to use and operate many of the relational database products now widely available for consumption, a general acquaintance with some of the basic principles of SQL will facilitate database design and querying datasets.

Bibliography

Carroll, Mary.

1988. *A dynamic Archaeological Database For the Microcomputer: The Shoofly Village Example*. MA Thesis, Arizona State University.

Codd, E. F.

1990. *The relational model for database management : version 2*
Reading, Mass. : Addison-Wesley.

Fife, Dennis W., W. Terry Hardgrave and Donald R. Deutsch.

1986. *Database concepts*. South-Western Pub. Co ,Cincinnati.

Redman, Chuck

1993. *People of the Tonto Rim*. Smithsonian Institution Press, Washington and London.

Smith, Michael E.

1979. A Further Criticism of the Type-Variety System: The Data can't be Used.
American Antiquity 44:4,pp 822-826.

*ADD citation of RPMS data flow model from manual (Kintigh, Spielmann, Simon, etc.)