Abstract

The Great Britain Historical GIS has been rebuilt around a single central table holding all statistics in one column, currently containing 14,541,491 data values. This architecture enables extremely flexible data handling, but requires that the context of each data value be captured entirely as metadata. Statistical reporting areas are defined via an ontology of administrative units, in which hierarchical relationships are compulsory while boundary polygons are optional. What a number measures is recorded via a relational implementation of the Data Documentation Initiative standard, locating each value within an n-dimensional matrix, or nCube, whose dimensions are variables such as age, gender and occupation. The data library can be extended to additional countries or more statistical topics without adding any database tables.

Keywords: statistical databases; historical GIS; Data Documentation Initiative

Rebuilding the Great Britain Historical GIS:

(1) Building an indefinitely scalable statistical database

Introduction

It is by now just about the conventional wisdom that any researcher with a large body of *aggregate* sub-national historical statistics should build a historical Geographical Information System (GIS); and most historical GIS projects began because researchers had assembled a large collection of census or vital registration data, generally computerized from published reports, at county, district, municipality or parish level. This certainly describes the origins of the Great Britain Historical (GBH) GIS founded by the author, of the US National Historical GIS (NHGIS), and of a series of national projects in continental Europe including systems for Belgium, the Netherlands, Norway and Sweden.

However, while the construction of these systems has enabled academic researchers to map and otherwise analyze geographical cross-sections, has historical GIS fulfilled its broader promise to provide a better understanding of change over time, and of “geographical change”? By 2001, the GBH GIS had spent seven years, and by the standards of historical research much money, mapping Britain’s changing administrative geography down to parish-level, constructing a system using ArcGIS software containing tens of thousands of

boundary polygons (Gregory and Southall, 1998). However, this structure had only a semi-detached relationship with what had already existed in 1994: a database of historical statistics, held using Oracle software in thousands of columns grouped into hundreds of tables, each organized much like the historical sources from which they had been transcribed. Three particular problems will be highlighted.

Firstly, ArcGIS could list the columns within a table that could be used to construct a statistical map, but the user had to know which table; and there were literally hundreds. It was hard for even an expert user to identify the most appropriate divisor to compute a rate. Earlier historical GISs going back to the Great American History Machine (Miller and Modell, 1988) had shared this problem: bad choices of divisors led to nonsensical maps, as would mapping most frequency counts without a divisor. Further, in 2001 we were awarded a large grant by the UK National Lottery to put our data on-line for the general public, but this meant we could not assume our users would be able to choose appropriate combinations of variables to create mappable rates. We therefore needed to find a way to embed knowledge about relationships between our variables within the statistical database, and to make that knowledge accessible to the mapping system.

Secondly, the geographical units in our statistical sources were identified not by reliable unique identifiers but by names, and while individual censuses were internally fairly consistent in their use of names, there were great variations between censuses: “Hautbois Magna” in Norfolk in 1851 becomes “Great Hautbois” in 1911, and English-speaking census officials had immense problems with Welsh place-names. As a result, rather than doing simple joins between tables in the statistical database and the GIS system, we had to do three-way joins using “gazetteer” tables (a misnomer) which “standardized” the names in the data tables to those used in the GIS; but the latter had not been chosen to meet or form any kind of standard, and a worrying number were simply mis-transcriptions of names listed by the 1911 census. Because some names were duplicated within counties, additional information had been added in brackets to some names in the GIS, selected arbitrarily, to disambiguate them. In general, tactics which were at least pragmatically defensible when working with six hundred or so Registration Districts were creating an unmanageable mess when applied to *circa* 15,000 parishes, and a more systematic approach to identifying reporting units was essential.

Thirdly and most fundamentally, our new lottery funding was to create a resource not for academic demographers but for “life-long learners” interested in *local* history, and for this audience it was more important that we be able to present our

data as local time series than as maps. This apparently modest ability is in practice beyond the capabilities of almost all historical GIS systems and other databases built from census data, because they organize their data into *datasets*, and however exactly this is done each dataset contains data from only one census. This is strikingly true of the data access system developed by the NHGIS: the very first decision a user must make is to choose from a list of “datasets”, each of which is a subset of statistics from a single census. We could, of course, have reorganized the data from single census datasets to time series datasets each covering a single reporting unit, but that would have enabled us to create time-series graphs but removed the equally essential ability to create maps. Our solution was to banish the whole notion of datasets and build a database of individual statistical data values, from which our software could equally easily select out time-series or cross-sections.

The architecture of the original GBH GIS was not "wrong", and in some senses we were the victims of our own success: an innovative and effective architecture for working with district-level geographies containing a thousand or so administrative areas, and a relatively narrow range of statistical data, enabled us to obtain funding for a much larger parish-level GIS, and to acquire from collaborators statistics on almost every topic imaginable. This paper describes how we built a more scalable version of our statistical database, a system able to

hold not just any amount of British demographic data but data on any topic and for any geography. Crucially, this new architecture is indefinitely extensible without adding *any* additional tables to the underlying database. This means not just that we could add data from the 2011 British census, but that we could add data on some quite different set of topics for some quite different part of the world without adding tables; the only caveat is that some small and obvious extensions will be needed if statistical topics are to be documented in multiple languages.

There are of course still technical limits to how large the database can grow, but the Postgres software we currently use sets no limit on the number of rows in a table. Postgres does currently limit a table to 32 terabytes of filestore, but our current data table, holding 14.5m data values, occupies about 15 Gb including indices so we could expand by four orders of magnitude before this became an issue; and by then this limit is likely to have risen. We do in fact have several terabytes of filestore on our server, but this was in order to hold a collection of scanned images of historical maps. By comparison with such digital image collections, even the largest social science datasets are not large; and by comparison with the costs of gathering the data, the cost of the necessary hard disks is now trivial. Unfortunately, our working practices and ways of thinking about data often date back thirty or forty years, to when our data archives were being established and storage hardware was vastly more expensive.

This paper covers all facets of how we hold statistics, including how we record time and geography, but the main emphasis is on how we record the meaning and provenance of statistics. The account given here of how we record geography would probably be sufficient if we were concerned with a single simple and fairly stable hierarchy of units, such as US counties, states and national totals, but a second article will describe the complex *Administrative Unit Ontology* (AUO) we developed to capture the peculiarities of European administrative and statistical geographies; an overview is provided by Von Luenen *et al* (2009). This system currently defines 79,166 units and 199 different *kinds* of unit, covering Britain, Ireland, Estonia and Sweden in considerable detail, although current statistical content is limited to Britain and Ireland. We refer to them as *administrative* not reporting units as the vast majority had administrative functions and our AUO has other functions besides being a framework for statistics.

This paper does not describe in detail how we reached these solutions, so we should state here our large debt to other projects and researchers, especially the Swedish National Topographic Database, the Alexandria Digital Library's gazetteer work (Hill, Frew and Zheng, 1999) and the US National Historical GIS (Minnesota Population Center, 2004). Our original contribution is to synthesize their work into a single well-integrated and computationally efficient data model. The focus on statistics means this paper does not describe the GBH GIS as a

whole: the same data structure includes the largest digital collection anywhere of British historical travel writing, all geo-referenced, and an extensive library of scanned historical maps; but nothing more will be said about these here.

Overall structure

[Figure 1 near here]

Figure 1 summarizes the overall architecture. All statistical data values are held in the data column of the data table, which is inevitably large: currently 14,541,491 rows; for example, 2,542,866 of these hold the number zero. While the individual books in a library or paintings in an art gallery have many unique characteristics, and much of the information held in the relevant collection's catalogue is just a summary of those characteristics, all those zeroes are absolutely identical and they can only be distinguished by the *metadata*, meaning data about data, that we hold about them in the other columns of that big data table. One aspect of our reliance on metadata is that the system is completely uninterested in the ordering of data values within the data table.

Such a table could be held in any number of relational databases, or even in a spreadsheet, but our need to hold spatial data not in some quite separate GIS but in close association with the statistics limits us to a much more limited set of *object*-relational database packages. Adding object extensions to a relational

database means it is no longer limited to a fixed set of data types and enables the inclusion of spatial data structures and tools for manipulating them: the database itself can acquire GIS functionality. We first implemented the approach described here using Oracle and Oracle Spatial in 2001-4, but in 2006-7 we re-implemented it using Postgres/PostGIS; PostGIS began as a separate project to provide Postgres with geospatial functionality but now forms part of the standard distribution.

This switch was made for three reasons. Firstly, our new funding from the European Union more or less mandated the use of open source software and, although spatial extensions do now exist for MySQL, Postgres/PostGIS were much more sophisticated, mature and widely used. Secondly, while Oracle and Oracle Spatial were available to us at no extra cost under a university site license, their very high cost to anyone without an existing license enormously constrained discussion of long-term hosting. Thirdly, it was felt that Oracle was over-complex and unnecessarily computationally demanding for a system in which, for example, fine-grained access controls are irrelevant. This all said, the object extensions matter mainly to our handling of geography and, in the AUO, dates. They are therefore largely irrelevant to this first paper.

Table 1 lists the actual columns within our current data table within Postgres, excluding a few columns relevant only to the construction process.

The first two columns are simply a unique sequence number identifying the data value, in effect an accession number for our collection, and the number itself. Although the majority of our data values are frequency counts, of people, of deaths, of sheep, the system can and does hold floating point values. The next three columns hold slightly *ad hoc* codes which affect how our web site presents data.

[Table 1 near here]

The most important is “precision”, which links to a small look-up table defining the possible values. Most census data are simply *exact* counts, based on the entire population; there are of course many reasons why they may still not be accurate. Some data, however, come from one per cent or ten per cent samples; these counts are scaled up in our database to give estimated values for the whole population. Some values are presented as counts, but have had to be computed from data given only as percentages, so substantial rounding is involved. Lastly, some values, especially our historical data for modern districts, are estimates computed through geographical redistricting applied to published data for historical units. Here the “precision” code indicates whether the redistricting was done by us using a vector overlay methodology, or by the *Linking Censuses through Time* project aggregating ward-level small area statistics (Martin, Dorling and Mitchell, 2002).

We are now arguably over-using this single attribute to track a number of different characteristics of certain parts of our data, although most values drawn directly from historical sources are identified simply as exact counts. Even calling this attribute “precision” is no longer appropriate, and this column will be re-thought in the next version. Another medium term plan is to move the calculation of redistricted data into the system itself, so making it self-documenting as discussed below.

Next, data “status” is used to control whether and how data are presented on our public web site. Depending on the value of the flag, data may be completely excluded from the site, or used only in graphs or maps, not listed as a number which users can copy. Various copyright or confidentiality issues with our most recent data made this mechanism necessary.

The “dominant” flag is a recent addition to the system and is currently used only in presenting listings of election results for individual constituencies: it identifies which candidates were elected, as historically there were many multi-member seats in which not just the candidate with the most votes who was elected.

The remaining columns in the data table mostly hold identifiers linking to major sub-systems, and locate each data value on four major dimensions: *where*,

meaning the geographical area covered, and *when*; *what* is being measured; and the *source* the data come from.

Where and When

The notion of space and time as dimensions is fairly obvious but there are some subtleties here.

Firstly, social and demographic statistics almost always relate to an area, not to a point, so there is no question of holding a simple coordinate here. It will also be obvious that, even though our database software is capable of directly holding a polygon within each row of the main data table, this would be very wasteful of storage space as very many data values relate to exactly the same area. However, we do not even hold a pointer to a polygon here, but rather a unit ID, which uniquely identifies a reporting unit: a township, a county, a state, an electoral district, etc.

We require that every data value be associated with a unit, but we do not require that every unit have an associated polygon. Although we obviously cannot map data for units without polygons, there are three benefits to this approach. Firstly, as we go further back in time, we increasingly encounter units whose boundaries are completely un-documented so polygon construction is impossible. Our system in fact also holds point coordinates which can often be inferred for such units, but

it cannot currently use these to create statistical maps. Secondly, even when there are historical sources recording boundary lines, constructing digital boundary polygons from these is expensive and may not be possible. For example, late nineteenth century British censuses reported some limited data firstly for Registration sub-Districts and secondly for Sanitary Districts. Polygons have yet to be constructed for these but our system does hold these data, and can tabulate and graph them. Thirdly, even when funding is available, boundary mapping is time consuming and it has helped greatly in scheduling large and complex projects that we can start loading statistical data for units well before boundary mapping is complete. Indeed, the AUO has provided scaffolding within which the textual information needed for boundary construction can be assembled.

Every unit belongs to a *type*, which it cannot change. We try to define types so that selecting all the units of a given type that existed at a particular date yields a complete map of the relevant country, with no holes and no overlaps. This may sound a trivial goal but the complexities of some of our geographies means it cannot always be achieved, even though we also use a more detailed concept of unit *status* to record some of the distinctions between kinds of unit. This is explained in the second paper.

Turning to dates, recording “when” is simpler than “where” or “what”, and does not involve a separate sub-system. However, historical information often contains

imprecise dates, and data like the 1861-70 mortality data cover periods, not time-points. The AUO uses a system of *object dates*, user-defined compound data types which can hold anything from a precise calendar date to “around the reign of Edward II”. Even the earliest statistical data we might want to hold is dated more precisely than that: the Domesday survey was in 1086. However, early surveys were gathered over periods of months while we usually date censuses simply by a year despite knowing the precise date. To allow for all possibilities, the data table uses five columns to store dates: period start and end dates which can each be either a year or a calendar date, plus a “decimal date” which is used for graphing. For now these have to be separate conventional database columns rather than a single user-defined type because Postgres cannot include components of a user-defined type within an index, although Oracle can do this. If the data are for a timepoint, not a period, we store it in the end date.

What: The Data Documentation System

Documenting the where and when of statistical data values is essentially about recording information already in our sources. Documenting the what means creating something new, and this is especially true if we are to construct time series from data gathered by a series of different censuses or surveys. Further, while the terms used in a specific census table may be very well defined, grouping together more loosely related numbers means defining some quite abstract

concepts. For example, we place the Registrar General's five-part classification of Social Class from 1921, the later seventeen-way Socio-Economic Groups and our own simple four-way grouping of 1831 occupational data within our theme "Social Structure".

Our approach is inspired by the work of the Data Documentation Initiative, and more specifically by the extension for aggregate and tabular data, developed by and for the US National Historical GIS project, which forms part of version 2 of the DDI standard. However, while we have written software that generates XML and checked that it meets the standard, our Data Documentation System (DDS) is not a collection of external text files but a set of tables within the same relational database that holds the data themselves. This is because our first priority in designing the system was performance: as discussed below, the metadata in the DDS directly controls how our statistical data are graphed, mapped and tabulated in response to user requests.

Because our funding came from digital library programs, it was very important that we be able to demonstrate to funders that all our metadata followed open published standards. However, in practice we have never had to supply DDI metadata to anyone. For example, while the UK Data Archive catalogue follows the DDI standard for study- and collection-level metadata, they do not support data item-level metadata. This is one of the reasons why we have not revised our

system to support version 3 of the DDI standard, but there are two other major reasons for this. Firstly, version 3 is very much larger than version 2, and we have never had the time or the funding to implement it. Secondly, and perhaps because of its complexity, implementation elsewhere of version 3 has been limited, and at least within Europe the rival SDMX standard has more momentum. One reason for this is that European governments are required to provide the European Union with statistics for their country for consolidation into EU-wide databases, and the EU statistical agency increasingly requires that these data be provided with SDMX metadata. That said, the DDI standard is better suited to documenting social survey data, such as census statistics, than at least the early versions of SDMX which were originally designed to document economic statistics.

Our system can and does hold just such economic data, such as wage rates for towns, but our approach is best understood in terms of data where the figures for geographical areas are aggregates constructed from microdata, which might be information on a census schedule about a person, or in a parish register about an individual death. The microdata include a number of different items about the individual, some numeric like *age* but mostly text, like *gender*, *occupation* or *cause of death*. However, published aggregate statistics tend to be based on simplifying the data by applying classifications, and even with a very simple underlying variable like age there is an almost infinite number of classifications.

Many reports use five year age bands, but even then they often give additional details for the under-5s and there is variation in what top band is used: “all over 75”, “all over 95”, “all over 100” and so on. In DDI terminology, each of these classifications is a different *variable*, defined by its set of *categories*, although we can bring together variables with similar scope into *variable groups*.

Although some published tabulations are concerned with just one variable, many census tables are based on cross-tabulating multiple variables, so occupation listings very commonly also include gender, while the large collection of data computerized by Robert Woods from the Registrar General’s *Decennial Supplements* 1851 to 1901 are often cross-tabulations of cause of death by age group by gender. For example, the report for 1861-70 cross-tabulates twenty-five cause of death categories, twelve age groups and two genders. In the DDI aggregate data extension, these combinations of variables are called *nCubes*, i.e. n-dimensional hypercubes, and even the simplest data are defined by their location within a one-dimensional nCube based on a variable with just one category.

An nCube is not defined solely by its component variables: two nCubes might both be based on age group and gender, but still be very different because one was census data while the other recorded mortality. The other part of an nCube definition is its *universe*, the total it adds up to: all persons, all deaths, all persons

aged 14 to 65, or all sheep. Each universe has a *measurement unit*: most universes for census data are counts of people, but land use statistics are counted in hectares or acres, and economic statistics in dollars or pounds.

One key attribute of an nCube is whether it is *additive*. For example, we hold hourly wage rates for different occupations in a single one-dimensional nCube, with occupations as categories. Adding together the rates for all available occupations at a given date in a given town does not produce a useful number: this is a non-additive nCube, and the rates should be graphed independently. Numbers of persons in different industrial sectors should, however, add up to total employment, and are better graphed over time as a stacked chart: this is an additive nCube. So long as they share a measurement unit, almost any collection of numbers can be defined as a non-additive nCube, but the approach is far more powerful if the time is taken to identify additive structures.

[Figure 2 near here]

Figure 2 shows the main entities in our DDI implementation and how they are related. Note that we do not use the DDI notions of *studies* and *collections*, because these are what data archives use to organize their holdings of datasets while our aim is to merge all datasets into a single collection, and we track the

projects that computerized the data in another part of the system, discussed below.

Instead, we have added a high-level set of organizing *themes*.

[Figure 3 near here]

Although figure 2 shows our entities, the actual structure of the database subsystem that holds this information is shown in figure 3. All the entities discussed so far, i.e. everything in figure 2 apart from the location map and the data table itself, are defined in one single table of DDS entities, currently with 6,598 rows. A second table holds 21,124 many-to-many relationships between these entities. Two further tables define what kinds of entity can exist within this framework, and which combinations of entity types can be directly related. One virtue of this implementation is that quite major changes can be made to the system by changing the contents of these tables, i.e. without changing either the table definitions or the software which accesses them. Another virtue is that only two of the tables in the DDS hold text for humans to read, the table of allowed entity types and, mainly, the table of entities, which will simplify making the system multi-lingual.

The Data Documentation System in Use

It would be possible to write articles about how particular large bodies of data have been documented using this system, such as the Agricultural Censuses or the 1841

Occupation Report, but a much simpler example will help explain the basic concepts.

[Table 2 near here]

As well as data on numbers in individual occupations, British twentieth century censuses usually listed numbers economically active and inactive, the former including those unemployed and looking for work as well as those in work. These data are usually cross-classified by gender. Everything human-readable goes into the table of DDS entities, and Table 2 lists all rows relevant to our example. It includes for each the entity ID, the associated label and selected attributes; the initial letter or letters of each ID indicates the entity type.

Firstly, we need to define “Themes” as high-level organizing concepts, and two of the ten we currently use are needed here, “Population” and “Work and Poverty”.

We also need to define the Universe covered by our particular data, i.e. “All Workers”, which in twentieth century data usually means all those above the relevant age of compulsory education and below retirement age. These entities are of course used very widely across our database, and what Table 2 does not show is the extensive introductory text held for each.

[Table 3 near here]

Next we need to define two variables. The first, “Whether economically active”, is fairly specific to our particular data while “Sex” is of course very widely used. Simply defining entities would be pointless unless they were connected up, and Table 3 shows the associated entries in the table of DDS relationships, beginning with the assignment of the variables to appropriate themes. We then define the categories within each variable, and link them to the variable.

nCubes are especially important as they “contain” the actual statistics, but they are defined mainly through their relationships with other entities, so we link N_CENSUS_ACTIVE_GEN to U_WORKERS and to our “Work and Poverty” theme, and link the two variables just defined to it. Once these definitions and relationships have been created, the other two tables in the DDS are populated automatically, as listed in Tables 4 and 5; these together comprise the “Location Map” and are central to how a fixed set of relational tables can document cross-tabulations with any number of dimensions. The identifiers actually held in the main data table are the “cell references” or “cellRefs” listed in Table 4, and it will be seen that the first part of each cellRef is essentially the ID of the nCube of which it forms part while the remainder is assembled from the “value” attributes of the relevant categories within the associated variables. It helps us load statistics into the data table that cellRefs makes some sense to humans, but as far as the system is concerned they are simply unique but otherwise arbitrary identifiers,

linking the data table to category definitions via the location map. The structure of the location map makes it inevitably verbose, so our 6,598 current DDS entities expand to 21,241 rows in the data map table and 58,279 rows in the coordinates table, but this is really just one of a number of ways we copy data around the system to improve performance.

Because our system is designed not simply to store data but to enable it to be presented to a non-expert audience as maps and graphs, we need to define which combinations of data values can sensibly be presented in time series graphs and choropleth maps, and to associate explanatory text with each such “Rate”. Our example here is the female activity rate, defined as the number of economically active women as a percentage of all women of working age. However, the actual definition of the rate does not appear in table 2 as it consists of three additional attributes, used only with rates: a numerator, a denominator and a multiplier. The first two hold cell references while, for example, a multiplier of 100 creates a percentage and a multiplier of one yields a simple decimal fraction. In this example, the denominator can be computed from the nCube containing the numerator but with, say, the infant mortality rate the numerator is the number of deaths of persons aged under one year while the denominator is the number of live births; and only by explicitly storing a multiplier can we record that, for

example, unemployment rates are usually reported per hundred while infant mortality is reported per thousand.

Finally, another key set of relationships shown in Figure 2 is that one variable can belong to another, and we then require that each of its categories be assigned to one of the categories of the other variable. Such mappings enable us to relate a multiplicity of historical classifications, of age or occupations or causes of death, to a single more generalized classification which can be employed over time.

Once we have created the mappings, we then generate *all* the derived values they imply, so the mappings themselves provide an audit trail for the creation of derived values. We will internalize the creation of redistricted statistics, discussed above, in a similar way: we have already built Geography Conversion Tables (Simpson, 2002) which define, for a specific date, what proportion of a frequency count for unit A should be assigned to unit B, where unit A is some particular historical unit while unit B is some more familiar modern unit; we will hold the individual weights from the Conversion Tables as relationships between units within the AUO, and then use these to generate all implied redistricted values.

This system will sound complex, and certainly holds some complex information.

As every DDS entity can be annotated the system is already, for example, a substantial knowledge-base on historical occupational classifications; and where else would you discover that a “wether” is a castrated male sheep? However, this

data model makes very good use of the underlying capabilities of relational databases and, for example, the generation of whole sequences of population pyramids for a given town over time, based on a 2 by 19 nCube, is almost instantaneous. Performance is partly the result of extensive de-normalizing, so for example each nCube is not just linked to a universe, it holds the universe ID as an attribute. The use of this structure to directly drive data visualization is further discussed in Southall (2008), while the end-user experience is described in Southall (2006). Despite its highly abstract foundations, this structure supports a very popular web site for local historians, receiving over 100,000 unique users per month:

www.VisionOfBritain.org.uk

Sources and acknowledgments

Our Data Documentation System is concerned only with what numbers mean, not with where they came from. That is recorded via two and a half further sub-systems. This may seem excessive, but with historical sources we always need to identify the original historical sources the data were transcribed from, and we often need to also identify and acknowledge the researchers or projects who did the transcriptions and provided us with the data.

Most sources are identified by a relatively simple system of authority identifiers which link to a central table of “authorities” designed around the Dublin Core standard and, for books, identifying title, author, publisher and so on. This table of authorities is used very extensively in other parts of the system, holding details of the books of travel writing, the sources for the AUO and so on. A second column in the data table, the “authority note”, usually holds uncontrolled text providing details of location within the authority, such as a page number.

However, we hold much more precise information on where a number comes from if the source is documented by our Source Documentation System (SDS), our third major sub-system. This is currently limited to British census reports, and the only likely extension is to Irish census reports. The SDS holds a complete listing of the censuses carried out in the United Kingdom since 1801, including exact dates; a complete listing of all reports published from those censuses up to 1966; and a complete listing of the 5,132 numbered tables within those reports. It defines a system of unique identifiers for those tables.

Where a number comes from such a table, the authority identifier is simply “SRC” and the authority note must then hold a source table identifier. Additional columns in the data table are then used to record *exactly* where in the table a number comes from. In the simplest case, where the source table has one row per reporting unit, the value in *src_tab_seq* puts the units in the right order and then a

column number orders the values for each unit. However, some more complex census tables have more than one row per reporting unit, and a few extremely complex tables involve more than one set of rows and columns, so we can define them as consisting of multiple “panels”.

This structure does not simply record where numbers come from, it enables a complete re-construction of a source table from the values held in the data table. *A Vision of Britain through Time* includes a whole area about the history of the census, including the full text of the most important reports and the entire contents of selected tables. To permit this, we do allow data values to be held in the data table without cell references linking to the DDS but with links to the SDS. Such numbers cannot appear in maps and graphs, only in source table reconstructions; but because all data values are linked to a unit, it is possible to jump straight into these reconstructions for a particular district or parish, as well as drill down into them starting with national totals.

Most statistics in the system were computerized by us, or by contractors, and have no associated “credits”. However, donations from other researchers have been very important and sometimes complex. One example is that parts of our transcription of the district-level age structure table in both the 1881 and 1891 censuses come partly from two different other researchers, one of whom did all the data for all the London districts, the other doing the individual years under 5

for the whole country; we did the rest. The other is that in several cases unpublished data was hand transcribed from an archival document by one researcher, and that transcription was then computerized by a second. There are also situations where somebody other than the transcriber holds some legal rights in a data set.

To ensure that this is all fully documented, our “acknowledgments” sub-system begins by defining and holding contact details for “contributors”. It then defines a set of “roles” for contributors, based on the Library of Congress’s MARC Code List for Relators (<http://id.loc.gov/vocabulary/relators>), and a set of “conditions” which a contributor may impose on use of their work. The individual definitions of contributors, roles and conditions hold a significant amount of information but only a few of each are needed for the whole database. We then use these to define “acknowledgments”, each identifying a contributor, their role and two distinct conditions, one for use of an individual data item, the other covering use of their contribution as a whole. An acknowledgment is also linked to a “credit”, whose definition is just an ID and a note: these credit IDs appear in the main data table and in the acknowledgments, so a whole series of acknowledgments can be associated with an individual data value, and possibly with no other data value. This system would be impossibly complex if we actually had to create different

sets of credits for even a small fraction of the database, but in practice only a handful of credits have been defined.

Conclusion

The construction of the GB Historical GIS, and the earlier Labor Markets Database, goes back over twenty years, and when we started a system like that described here would have been both unnecessary and impossible: unnecessary, because a more conventional solution worked fine for a few dozen statistical tables linked to district-level geographies containing under a thousand polygons; impossible because this solution can be implemented only in *object*-relational databases which did not exist twenty years ago.

However, while the *conventional* solution is also the *simpler* solution for a small-scale project, we have learned very painfully that as the resource grows, that solution becomes more and more complicated, as the number of data tables and GIS coverages grows. The immense virtue of the system described here is that the content can grow indefinitely using an absolutely fixed set of database tables, and consequently without requiring any changes to the software that drives the system. There may eventually be some fundamental limits to database size, but commercial users of the enterprise-class software we are using have built far larger non-historical databases than anything we are currently contemplating.

[Figure 4 near here]

The proof of the pudding is in the eating. Until recently, our own public web site was designed for a very general audience, and academic researchers wanting access to our statistical data obtained it via the UK Data Archive, where it is organized conventionally as a collection of datasets, and the census data are all in files each of which covers just one year. Our data structure consequently had no obvious advantage over the US National Historical GIS extraction interface shown in Figure 4: users were forced to select a year, and then “datasets” for that year, so if they wanted time series they had to build them.

[Figure 5 near here]

Figure 5 shows the equivalent first stage of our own new data extraction system, which we believe to be unique, exploiting the underlying data architecture already described: it lets users specify where, when and what. It does not require any knowledge of past reporting geographies, “where” being specified by simply centering the map on the location of interest. For now, “what” is specified simply by selecting one of our main statistical themes. In step 2, the system lists the reporting units whose boundary polygons covered the location, and the relevant nCubes available for each. The user can then select for download either whole units with all associated nCubes, or pick and choose from the available nCubes.

Newcomers to the field probably assume that an on-line historical GIS will enable them to find relevant statistical data for analysis, given that such systems usually contain both historical census data and the boundaries of reporting units. Our work shows that you also need the right information architecture, one in which the statistics rather than the polygons are at the center.

References

- Gregory, I. N., and H. R. Southall. 1998. Putting the Past in Its Place: the Great Britain Historical GIS. In *Innovations in GIS 5*, edited by S. Carver, 210-21. London: Taylor and Francis.
- Martin, D., D. Dorling and R. Mitchell. 2002. Linking censuses through time: problems and solutions. *Area* 34:82-91.
- Miller, D. W., and J. Modell. 1988. Teaching United States history with the Great American History Machine. *Historical Methods* 21:121-34.
- Minnesota Population Center. 2004 *National Historical Geographic Information System: Pre-release Version 0.1*. Minneapolis, MN: University of Minnesota.
- Simpson, L. 2002. Geography conversion tables: a framework for conversion of data between geographical units. *International Journal of Population Geography* 8: 69-82.

Southall, H.R. 2006. A Vision of Britain through Time: making sense of 200 years of census reports. *Local Population Studies* 76:76-84.

Southall, H.R. 2008. Visualization, data sharing and metadata. Geographical Visualization: Concepts, Tools and Applications, edited by M. Dodge, M. McDerby and M. Turner, M., 259-75. Chichester: Wiley.

Von Luenen, A., P. Aucott and H. Southall. 2009. Exposing the history of Europe: The creation of a structure to enable time-spatial searching of historical resources within a European Framework. *OCLC Systems and Services: International Digital Library Perspectives Journal* 25:270-286.

Table 1: Columns in the data table.

Topic	Column Name	Data type
Row ID	g_seq	Integer ID; effectively an accession number.
Data value	g_data	Numeric; the actual number all the other columns contextualize.
Usage	g_data_precision	Code identifying data values as exact counts, sample based or more complex estimates.
	g_data_status	Code recording how far data can be displayed in open access system.
	dominant	Hint for web interface identifying values to highlight, such as election winners.
Where?	g_unit	Integer ID for a reporting area, as defined in the Administrative Unit Ontology.
	g_unit_type	Type of the unit, as defined in AUO; “map layer”.
	g_rel_to	Integer ID for second reporting area, as defined in AUO; used only with interaction data.
What?	cellref	Character ID locating data value within an nCube as defined in data documentation sub-system.
When?	start_year	Calendar year. Used only if data cover a period.
	start_date	Database date value. Used only for periods.
	end_year	Calendar year.
	end_date	Database date value.
	end_date_decimal	End date in years, including fractional years.
Source?	g_authority	Authority ID as defined in authorities table.
	g_auth_note	Usually a text string with details of the source, such as page number; but if authority is “SRC”, contains census table ID as defined in source documentation sub-system.
	src_tab_seq	Integer giving order within source table.
	src_tab_col	Integer giving column within source table.
	src_tab_row	Integer giving row within source table, if multiple rows per unit.
	src_tab_panel	Integer giving “panel” within especially complex source tables.
Thanks	g_data_credit	Character ID linking to bundle of “credits” in acknowledgment sub-system.

Table 2: Selected DDS Entities.

Type	Entity ID	Label	Value	Add ?	Cont ?
Themes	T_POP	Population			
	T_WK	Work & Poverty			
Universes	U_WORKERS	All Workers			
Variable with two categories defining whether economically active:	V_CENSUS_ACTIVE	Whether Economically Active			
	C_CENSUS_ACTIVE_1	Economically active	act		
	C_CENSUS_ACTIVE_2	Economically inactive	inact		
Variable with two categories defining gender	V_SEX	Sex			
	C_SEX_1	Male	male		
	C_SEX_2	Female	female		
nCube holding cross-tabulation of activity by gender	N_CENSUS_ACTIVE_GEN	Economically Active by Sex		Yes	
Female Activity Rate	R_CENSUS_FEM_ACTIVE	Female Activity Rate			Yes

Table 3: Selected DDS Relationships.

Purpose	Entity ID	Related ID
Assigning Variables and Rates to Themes	V_CENSUS_ACTIVE	T_WK
	V_SEX	T_POP
	R_CENSUS_FEM_ACTIVE	T_WK
Assigning Categories to Variables	C_CENSUS_ACTIVE_1	V_CENSUS_ACTIVE
	C_CENSUS_ACTIVE_2	V_CENSUS_ACTIVE
	C_SEX_1	V_SEX
	C_SEX_2	V_SEX
Assigning nCube to Theme and Universe	N_CENSUS_ACTIVE_GEN	T_WK
	N_CENSUS_ACTIVE_GEN	U_WORKERS
Assigning Variables to nCube	V_CENSUS_ACTIVE	N_CENSUS_ACTIVE_GEN
	V_SEX	N_CENSUS_ACTIVE_GEN

Table 4: Data Map for N_CENSUS_ACTIV_GEN.

Data Item ID	Cell Reference
N_CENSUS_ACTIVE_GEN_1_1	CENSUS_ACTIVE_GEN:male/act
N_CENSUS_ACTIVE_GEN_1_2	CENSUS_ACTIVE_GEN:male/inact
N_CENSUS_ACTIVE_GEN_2_1	CENSUS_ACTIVE_GEN:female/act
N_CENSUS_ACTIVE_GEN_2_2	CENSUS_ACTIVE_GEN:female/inact

Table 5: Data Map Coordinates for N_CENSUS_ACTIV_GEN.

Data Item ID	Coord. No	Coord. Val.	Category ID
N_CENSUS_ACTIVE_GEN_1_1	1	1	C_SEX_1
N_CENSUS_ACTIVE_GEN_1_1	2	1	C_CENSUS_ACTIVE_1
N_CENSUS_ACTIVE_GEN_1_2	1	1	C_SEX_1
N_CENSUS_ACTIVE_GEN_1_2	2	2	C_CENSUS_ACTIVE_2
N_CENSUS_ACTIVE_GEN_2_1	1	2	C_SEX_2
N_CENSUS_ACTIVE_GEN_2_1	2	1	C_CENSUS_ACTIVE_1
N_CENSUS_ACTIVE_GEN_2_2	1	2	C_SEX_2
N_CENSUS_ACTIVE_GEN_2_2	2	2	C_CENSUS_ACTIVE_2

Figure 1: Architectural Overview

Figure 2: GBH Data Documentation
System Entities and Relationships

Figure 3: GBH DDS table structure

Figure 4: NHGIS Data Access Step 1

NHGIS Data Finder

[NHGIS Home](#) | [Data Finder](#) | [Shape Finder](#) | [Extract History](#) | [Contact Us](#) | [Log Out](#)

Step One - Select A Dataset

Select a dataset from one of the following groups. [HELP](#)

Census of Population and Housing (1970-2000)

Historical Census (1790-1960)

☐ 1850-1959 - Farms - State, and County

☐ 1850 - US, State, and County

☐ 1860 - US, State, and County

☐ 1870 - US, State, and County

☐ 1880 - US, State, and County

☐ 1890 - US, State, and County

☐ 1900 - US, State, and County

☐ 1910 - US, State, County, Tract, and Enum Dist

☐ 1920-1936 - Banking Establishments - State, and County

☐ 1920 - US, State, County, Tract, and Washington D.C. Enum Dist

☐ 1930 - US, State, County, Tract, Ward, and Place

☐ 1940 - US, State, County, and Tract

County Business Patterns

Special and Supplemental Reports

Select And Continue

Figure 5: GBH GIS Data Access Step 1