Structure of Data in the People of Northern England (1216-1286) database

Terms of use

Introduction

From the PoNE website front page:

> This database of the people in the counties of Northumberland, Cumberland and Westmorland is drawn from two types of material, one financial and one legal. The financial material is drawn from the pipe rolls from 1219 to 1286, and the legal material from the plea rolls from 1219 to Trinity Term 1275 (and sporadically to Easter Term 1277; see also Plea Roll Summaries, Michaelmas Term 1275-Trinity Term 1278). It is a unique database since nothing comparable has been constructed before for any English county.

The PoNE database was created as a part of the Breaking of Britain project. The work to create it was carried out 2011-13. The project team was (as listed on the website):

Project Co-ordination Team (with special responsibilities for PoNE)

Dr Sophie Ambler		Research Associate
Dr Amanda Beam-Frazier	University of Glasgow	Research Associate
Mr John Bradley	King's College London, Department of Digital Humanities	Co-Investigator
Professor Dauvit Broun	University of Glasgow	Principal Investigator
Professor David Carpenter	King's College London	Co-Investigator
Alejandro Giacometti	King's College London, Department of Digital Humanities	Technical Research Associate
Dr Beth Hartland	University of Glasgow	Research Associate
Dr Michele Pasin	King's College London, Department of Digital Humanities	Research Associate
Professor Keith J. Stringer	University of Lancaster	Co-Investigator

Support

David Little	King's College London, Department of Digital Humanities	Design
Ginestra Ferraro	King's College London, Department of Digital Humanities	Design
Christelle le Riguer	University of Glasgow	Administrative support/Finance

Knowledge Transfer Advisory Group (KTAG)

- Linda Bankier (Berwick Record Office)
- Dauvit Broun (University of Glasgow)
- David Carpenter (King's College London)

- Michael Geary (Northumberland Archives)
- Richard Hall (Cumbria Archive Centre, Kendal)
- Beth Hartland (University of Glasgow)
- Dawn Hurton (Cumbria Archive Centre, Carlisle)
- Jess Nelson (The National Archives)
- Carol Scott (Northumberland Archives)
- Keith Stringer (Lancaster University)
- Angus Winchester (Victoria County History, Lancaster University)

Acknowledgements:

On the PoNE website the project team thanked these scholars and resources for their help with the production of this database:

- Anglo-American Legal Tradition website: http://aalt.law.uh.edu/HenryIII.html
- Cumbrian Archive Services: Richard Hall, Anne Rowe
- Northumberland Archive Services: Linda Bankier, Michael Geary, Carol Scott
- Prof. Paul Brand (All Soul's College, Oxford)
- Dr Richard Cassidy (KCL)
- Dr David Crook
- Dr Jess Nelson (TNA)
- Dr Sarah Rose (Lancaster)
- Dr Henry Summerson (KCL)
- Christopher Whittick (East Sussex Record Office, Lewes)

About the historical sources for the data

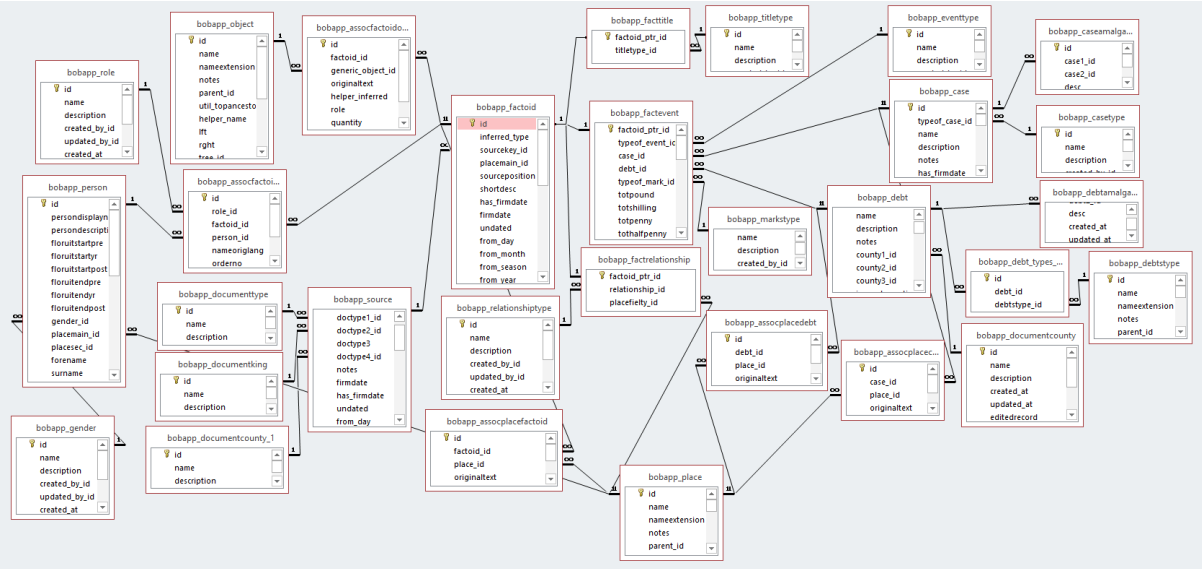
Historical background about the sources for the data can be found in the accompanying PDF file sources.pdf. The sections in that document describe the format and nature of the Plea Rolls and Pipe Rolls, describe how information was extracted from them, and provide summaries of each entry from these two types of sources that has been used for PoNE.

About the overall Data Structure

The PoNE project put almost all of its data into a MySQL relational database, and then built an online browser oriented web front end that gave access to it. This web front end was written in Django, and Django also stores materials it needs beyond the project's actual research data in the PoNE database. As a consequence, this archive contains a MySQL dump of *only* that part of the relational database that contained the research data. The other tables that Django created for its other purposes have been omitted.

The use of Django explains why the database's tables are named as they are: Django divides its operations into various "apps", and, since the PoNE work was carried out as a part of the Breaking of Britain (bob) project, all the PoNE research data was stored in the Django app named *bobapp*. Hence, all the tables belonging to the *bobapp* app have names that start with "bobapp_".

The structure of the database can be seen in the following entity-attribute-relationship diagram (which is slightly simplified from a full diagram in the interest of making the diagram as clear as possible):



(this same diagram is in the file “pone-EAR-diagram.png”)

PoNE’s database was designed according to the principles of factoid prosopography. A factoid is “a spot in a source where something is said about one or more persons”. See https://factoid-dighum.kcl.ac.uk/ for an introduction to its principles.

About the Django attributes

The way in which Django was used in the project caused a set of one or more of the following attributes to be inserted in almost all tables. They all relate to the editorial process that was involved in creating the database rather than being historical data drawn from the project’s medieval sources. Their contents are briefly given below:

- `created_at`: this is a timestamp recording when this particular record was created in the database.
- `updated_at`: this is a timestamp recording when this particular record was last changed in the database.
- `editedrecord`: this is a Boolean variable which could be used by the project team to indicate that a record had been edited, and was ready for publication. It was, in fact, rarely used at all.
- `review`: this is a Boolean variable which could be used by the project team to indicate that a record needed to be reviewed before publication. It was, in fact, rarely used at all.
- `internal_notes`: this was a text field in which the project team could record a note for internal use about the record.
- `created_by_id`: this is a numeric field which records who created the record. The table that maps this number to a particular person has been deliberately suppressed due to data protection rules.
- `updated_by_id`: this is a numeric field which records who last modified the record. The table that maps this number to a particular person has been deliberately suppressed due to data protection rules.

About PoNE’s Authority Lists

When PoNE was built, it was our practice to call entities that represented a set of names for something (e.g. titles, or types of relationships) *Authority Lists*. Since authority list tables are centred on providing names for things, all of them have “name” as an attribute, and several have

"description" as a place where a more detailed description of a particular item in the list, represented by that particular record, can be stored. In PoNE a description was only rarely provided.

Some Authority lists can be organised hierarchically. If so, then the entity will have a `parent_id` attribute as one way to express the hierarchy. It will also have `lft`, `rght`, `tree_id` and `level` attributes. These provide a "Nested Set" representation of the hierarchy. (see https://en.wikipedia.org/wiki/Nested_set_model)

Many of PoNE's tables are authority lists:

- `bobapp_role`: the list of possible roles that a person can play in factoids. See description of this in the section on factoids, below.
- `bobapp_titletype`: the list of possible titles or occupations that appeared for people in the PoNE data.
- `bobapp_eventtype`: the list of possible types of events that appear in the PoNE data.
- `bobapp_casetype`: the list of possible types of cases that appear in the PoNE data.
- `bobapp_debtstype`: a list of possible types of debts that appear in the PoNE data. This has an hierarchical organisation.
- `bobapp_markstype`: a list of the possible kinds of marks that scribes have added to entries in the rolls.
- `bobapp_relationshiptype`: a list of the types of personal relationships that appeared in the PoNE data. These are not only familial relationships.
- `bobapp_object`: a list of objects that can be associated with events related to cases or debts. It can be arranged hierarchically, although by the time PoNE data was finished only a few object types were needed, and the hierarchical structure is, thus, limited.
- `bobapp_gender`: there are, perhaps, surprisingly, four types of "genders" recorded. As well as the expected "Female" and "Male" values, there is an entry for "Institution" (for, say, abbeys or other similar institutions) and "Male/Female" for mixed sex institutions.
- `bobapp_documenttype`: a list of types of documents that appear as historical sources for the PoNE data.
- `bobapp_documentking`: a list of English Kings that ruled during the time period for the PoNE data. There are only two kings: Henry III and Edward I.
- `bobapp_documentcounty`: a list of Counties associated with roll entries for the PoNE data.
- `bobapp_place`: a list of places associated with the PoNE data (see the section on Place data below). This is an hierarchical list.
- `bobapp_object`: a list of objects associated with events in the PoNE data. This is an hierarchical list.

About PoNE and factoid prosopography

As mentioned above, PoNE's database is structured as a factoid prosopography. The principle tables involved in this aspect of the structuring are:

- `bobapp_factoid`: Records in this table (shown near the horizontal centre of the diagram) represent each individual factoid, and, as is described at the factoid prosopography website, can often usefully be viewed as the nexus of the data in that it connects the major entities, persons, sources, places, events, etc together. As much as possible, each factoid has an historical date associated with it, so many of the attributes are related to historical dating.

  PoNE has three kinds of factoids: events, relationships and titles. See their description below. The attribute `inferred_type` identifies which type each factoid is. All factoids are

taken from an historical source, which is identified by the sourcekey_id attribute, and which points in turn to the record for the source in the bobapp_source table. For event factoids, a place in the source is also identified, and stored as a text string in the sourceposition attribute. A short description of the factoid is given in shortdesc.

- bobapp_person: Records in this table (near the left edge of the diagram) represent the historical persons identified by PoNE. Attributes for this table provide names for the person, floruit or life dates, their gender, and a pointer to a geographical place (or two) associated with them. See the section on persons later in this document.
- bobapp_assocfactoidperson: Records in this table link persons to factoids that are about them. In the general factoid prosopography model this type of entity is now called a "Reference": a spot in a source (and in a factoid) where a person is referenced. This entity is needed because for many of the factoid more than one person must be associated. Thus, the *role* of the person (the list of roles used by PoNE is in the Authority List bobapp_role) is one of the attributes (role_id) of this entity. Also, this provides a place (attribute nameoriglang) where the form of the name, as it appears in the source document for this particular person, can be recorded. Finally, in some situations the order of the persons is specified in the historical document, and that order matters. In this situation, the orderno attribute is used to provide ordering information for the set of references attached to a particular factoid.
- bobapp_factevent: Records in this table represent data specifically for event-type factoids. The key for each record here is the same as the key used for the corresponding factoid data in bobapp_factoid, and a record here is thus joined as one-to-one with that table. Thus, a record from bobapp_factoid, combined with the corresponding record from bobapp_factevent makes up an event factoid. Event factoids are the most complex of the three types of factoids, and represent events that are found in the historical roll sources.

  Each event is connected to *either* a case (through the case_id attribute) or a debt (debt_id attribute) – but not both. Structure for Cases and Debts is described below. Furthermore, each event is typed according to what stage in the case or debt process is being represented (attribute typeof_event_id). Debt events usually involve money, so the amount involved is provided in a set of attributes related to money that can be found here. Finally, some entries in the rolls were marked by the scribe. If a particular entry for the event is so marked, this is recorded in the typeof_mark_id attribute which, in turn, links to an item in the bobapp_markstype Authority List.

- bobapp_factrelationship: Records in this table represent data specific to relationship factoids, which describe personal relationships between historical persons, as revealed in the roll sources. As for bobapp_factevent (just described), the key for each record here is the same as the key used for the corresponding factoid data in bobapp_factoid, and a record here is thus joined as one-to-one with that table. The full set of attributes for any relationship factoid thus comes from both tables.

  The particular relationship (having a son, having an attorney, etc) is specified by the relationship_id attribute, which points to the relationship in the bobapp_relationshiptype Authority List. Relationships involve at least two people, so there will be at least two bobapp_assocfactoidperson/reference records for each relationship factoid. The role (generally either "primary" or "secondary") specifies which

person is which. For the son-type, if person A has the "primary" role and B has the "secondary" role, then that is read that A has a son of B.

- bobapp_facttitle: Records in this table assert that a person held a particular title or is recorded has being involved in a particular occupation. As for bobapp_factevent, key for each record here is the same as the key used for the corresponding factoid data in bobapp_factoid, and a record here is thus joined as one-to-one with that table. The full set of attributes for any relationship factoid thus comes from both tables.

  The particular title is specified by the titletype_id attribute which points to a particular title/occupation in the bobapp_titletype Authority List table. For title factoids, generally only one person is linked, and the role for that Reference is "primary".

About PoNE's Person structure

Only two tables are involved in recording information about PoNE's historical persons. Note that persons in PoNE are not just human individuals, but were sometimes groups of people appearing in the documents as if they were legal person-like entities. The main table for persons, bobapp_person, is described in terms of its role in factoids above, as is the second linked table, the authority list bobapp_gender.

The bobapp_person table has a good number of attributes attached to it.

- *Naming*: a standard display name for the person can be found in persondisplayname. Sometimes a brief description to help with person identity is provided in persondescription. A person's display name is broken into component parts in attributes forename, surname, sonof, patronym, ofstring and placeandinst.

  Note that the display name for a person is not the only place where names for people appear in PoNE. The names as they are actually written in the sources can be found in the nameoriglanguage attribute in bobapp_assocfactoidperson records.

- *Dating*: Fields for Life dates for the person as *floruit* were provided but were in fact not used by PoNE.
- *Associated Places*: When the identity of a person is evidently connected to a place, this place can be identified through the placemain_id and placesec_id attributes.
- *Linked PoMS person*: PoNE was created as a part of the Breaking of Britain project which also added data to the then pre-existing PoMS (People of Medieval Scotland) database. When a person appeared in PoNE that was also in PoMS, the URL to the person record in PoMS is recorded in the pomslink field. The degree of certainty about this is recorded in pomslink_sureness field.

About Sources in PoNE

Each PoNE factoids is linked to the particular historical source (the rolls) from which it came. There are four tables associated with historical sources in PoNE:

- bobapp_source: This is the table which identifies the historical sources by naming them. There are, unfortunately, a set of obscurely named attributes in this table. A summary of all the attribute's functions are:
  - helper_doctype: actually provides a standardised name for the source.
  - reference: a short form of the shelfmark for the source
  - notes: a more detailed description of the source.

    - `dating`: a display form of the date for the roll. Years for the date range are also in from_year and to_year as numeric fields. When the `to_year` is missing, it can be assumed to be the same as the `from_year` value. In a few situations `datingnotes` has a comment in it. Other dating variables are not used in PoNE.
    - `doctype1_id`: links to the `bobapp_documenttype` authority list for the type of document.
    - `doctype2_id`: links to the king reigning at the time, and specified in `bobapp_documentking`
    - `doctype3_id`: apparently, is not a link at all, but another specification of the year associated with the document.
    - doctype4_id: the county associated with the document. Links to `bobapp_documentcounty`.
- `bobapp_documenttype`: a list of types of documents that appear as historical sources for the PoNE data.
- `bobapp_documentking`: a list of English Kings that ruled during the time period for the PoNE data. There are only two kings: Henry III and Edward I.
- `bobapp_documentcounty`: a list of Counties associated with roll entries for the PoNE data.

About PoNE's *Case* structures

All event factoids in PoNE are either events that arise from cases or debts. There are four tables associated with cases.

- `bobapp_case`: This is the main record for case data. It has fields for a case name, a description of the case, and notes. It also has a complex set of attributes for dating of the case but, in fact, these fields are not filled in except case with ID=7 (and this is perhaps only an accidental entry by the data entry folk). Dating information, for the various events associated with a case can be found in the event factoids attached to any case. Cases are linked by PoNE to associated counties. Up to five counties may be associated to the case (through attributes county1_id through county5_id).
- `bobapp_caseamalgamation`: From time to time cases would be combined together. This table provides the way to express this.
- `bobapp_casetype`: Cases came in a number of different types. This Authority List entity lists the types of cases that appear in PoNE.
- `bobapp_assocplacecase`: Cases can be connected to one or more geographic places. This intersection set table allows this connection to be specified, and also provides a place where the spelling of the name used to reference the place in the source can be provided.

About PoNE's *Debt* structures

There are five tables associated with debts.

- `bobapp_debt`: This is the main record for debt data. It has fields for debt name, a description of the debt, and notes. Like case, it has a complex set of attributes for dating, but these fields are not filled in. Instead, dating information for the various events associated with the debt can be found in the event factoids attached to any debt. Up to three counties may be associated to the debt (through attributes county1_id through county3_id).
- `bobapp_debtamalgamation`: As with cases, From time to time debts would be combined together. This table provides the way to express this.
- `bobapp_debttype`: Debts came in a number of different types. This Authority List entity lists the types of cases that appear in PoNE. Unlike cases, any debt may have more than one

type attached to it. Thus, an intersection set table bobapp_debt_types_of_debt enables a many-to-many linking of debts to debt types.

- bobapp_assocplacedebt: Debts can be connected to one or more geographic places. This intersection set table allows this connection to be specified, and also provides a place where the spelling of the name used to reference the place in the source can be provided.

About Objects in PoNE

Both events associated with cases and debts can have objects associated with them (often as renders). Two tables manage this data:

- bobapp_object: This is the main table for object data. In it the object is named in the name attribute. It can be arranged hierarchically, although by the time PoNE data was finished only a few object types were needed, and the hierarchical structure is, thus, limited.
- bobapp_assocfactoidobjects: Is an extended intersection set that connects factoids to objects. As well as having pointers to the appropriate factoid and object, it has an originaltext field to record how the object was described in the source, a role field to associate a role for the object in the factoid-event, and a quantity field when multiple instances of the object (e.g. horses) were involved.

About Places in PoNE

References to geographic places occur in several different contexts in PoNE. In the database most of them link to the bobsapp_place table which is an hierarchical Authority List of places, identified by their names. In two situations, however, the place, which will be specified as a county, comes from the bobapp_documentcounty table. The relationship diagram shows both situations.

Links to bobapp_place occur in several tables:

- bobapp_person: A person can be linked to a particular place through the placemain_id and placesec_id attributes.
- bobapp_factoidrelationship: A placefielty_id attribute allows fielty to a place to be specified. In fact, this is not used in PoNE. Check the bobapp_assocplacefactoid links for the factoid instead.
- bobapp_assocplacefactoid: This table acts as an intersection set table that allows any factoid to be connected to more than one place. Its originaltext attribute provides a place where the spelling of the name used to reference the place in the source can be provided.
- bobapp_assocplacecase: This table acts as an intersection set table that allows any case to be connected to more than one place. Its originaltext attribute provides a place where the spelling of the name used to reference the place in the source can be provided.
- bobapp_assocplacedebt: This table acts as an intersection set table that allows any debt to be connected to more than one place. Its originaltext attribute provides a place where the spelling of the name used to reference the place in the source can be provided.

-- John Bradley
15 March, 2021