

The  
**China Biographical Database**  
**USER'S GUIDE**

Michael A. Fuller

Revised Version 2.0  
February 28, 2015



# Preface to the User's Guide

Peter K. Bol

The China Biographical Database, as a relational database, can generate biographical data in response to simple queries (who came from a certain place?) and to far more complex queries (what were the social and kinship connections among all those who entered government through the civil service examination from a certain place within a certain span of years?). Users can query CBDB through an online database (follow the links on the CBDB website, <http://isites.harvard.edu/icb/icb.do?keyword=k16229>). Users also can download the entire database, together with query forms and utilities for exporting data for network and spatial analysis, from the CBDB website and explore the database on any computer with Microsoft Access. This User's Guide explains the structure and application for the downloadable, stand-alone database.

CBDB is a relational database. It categorizes and codes many different aspects of the life histories of men and women in China's past. In using it, there are several considerations that one should bear in mind and that I wish to stress before the Users' Guide begins its presentation of the specific details of the database, its design, and its use.

A way of thinking about people in context. CBDB is a way of modeling life histories; it is also a way of thinking about how to organize information. The subject of the database is people in society, but we treat people as entities that have relationships to their kin and their social associations, to places where they resided and worked, to times when they lived and moments when they acted, to names they were given and adopted, to books they wrote, to ways in which they entered government or other institutions, and to the modes in which they distinguished themselves from others. In contrast to the narrative of a life, CBDB sees people as entities defined by webs of relationships that can be quantified and analyzed.

Temporal scope. Over ninety percent of CBDB data pertains to the period from the Tang dynasty (618-907) into the early 20<sup>th</sup> c. As of January 1, 2011 it had about 100,000 figures with about 70,000 more in the preparation; further data on figures already in the data are frequently added. The systematic incorporation of figures from earlier periods will take place with the preparation of tables with place names and official titles.

Factoids versus facts. Like prosopographical databases for other parts of the world, CBDB for the most part deals in "factoids," the assertions of a fact (such as "Su Shi was a person from Meishan") found in the historical sources it references. It relates these assertions, including contradictory assertions when they appear, rather than judging their reliability. However, it does not treat all sources as equal.

Principal sources. CBDB began with research conducted by the late Robert Hartwell focused on the middle period of China's history. Since then, it has been comprehensively incorporating data from published indices, such as Wang Deyi's revised *Index to Biographical Sources for Song Figures* and similar works; from online databases, such as the Name Authority Database of the Ming Qing Archive at Academia Sinica and the Tang Knowledge Base at Kyoto University; from studies of text sources such as collections of grave biographies (墓誌銘); and from biographies in formal dynastic sources. Although CBDB editors at Harvard and Peking University are experimenting with mining data from other sources, it will take some time before the principal sources are exhausted.

Text-mining. The most efficient way to populate CBDB is through the use of computational text-mining techniques to cull factoids from searchable digital texts that have been provided by the Institute of History and Philology at Academia Sinica. This is being done with computer scientists on an US National Endowment for the Humanities grant. The Harvard editorial team works with computer scientists to develop appropriate "regular expressions," the Peking University editorial team review the marked-up text, and the managers then oversee the final coding of the data for inclusion in CBDB. This process does not guarantee that all possible factoids are discovered, simply that those included will accurately reflect the sources being mined.

Margin of error. Machines are more reliable than humans in sifting through large quantities of data but are incapable of interpretation and scholarly judgment. Errors can enter the database. The historical sources themselves can be incorrect. Editors may miss mistakes in tagging. Encoders may fail to properly disambiguate two entities with the same name. A user must always ask if the query to the database produces enough examples to ensure that the margin of error will not undermine confidence in the conclusions that are drawn. The discrepancies between the sources and the original CBDB data were significant, and considerable time was spent correcting the received data; with the adoption of computational techniques the discrepancies appear to be less than one percent. To put this in perspective: an argument based on 1000 examples of which ten are faulty is better than a finding based on ten examples of which one is erroneous.

A database is not a dictionary. CBDB can be used as a guide to biographical factoids about an individual, and it can provide more data about some aspects of a person's connections than would be found in a biographical dictionary. However, the standard for a dictionary is complete accuracy in all aspects, whereas the expectation for a database is that the cases discovered will be useful because they are extensive in range and number.

CBDB is a joint project of the Center for Research on Ancient Chinese History at Peking University, the Institute of History and Philology at Academia Sinica, and the

Fairbank Center for Chinese Studies at Harvard University. Collaborating databases include the Name Authority Database of the Ming Qing Archives at Academia Sinica and the National Palace Museum, the Ming Qing Women's Writings database at McGill University, and the Tang Knowledge Database at Kyoto University. Michael A. Fuller designed the current database.

This User's Guide explains the logic of CBDB as a relational database, the structure of its contents, and the primary query interfaces for getting data from the database.

## ***Table of Contents***

<i>i</i>	PREFACE
1	INTRODUCTION
3	CHAPTER 1: RELATIONAL DATABASES
	A. Relational Database and the Organization of Complex Data
	B. Rules for Structuring Data in a Relational Database
	C. Relational Databases and the Interactions of Complex Data
9	CHAPTER 2: THE STRUCTURE OF CBDB
	A. An Overview of the Entities in the Database
	B. Details of Entities
	Summary of Tables in CBDB
28	CHAPTER 3: CBDB TOOLS FOR ANALYSIS
	A. LookAtEntry
	B. LookAtAssociations
	C. LookAtOffice
	D. LookAtKinship
	E. LookAtNetworks
	F. LookAtAssociationPairs
73	CHAPTER 4: ADVANCED QUERY TECHNIQUES
	A. Kinship Networks for <i>Jinshi</i> -degree Holders in Putian
89	APPENDIX: INSTALLING CBDB FILE
	A. Installing the MS Access Files
	B. Installing the SQLite CBDB Database on a Mac

# Introduction

The China Biographical Database (CBDB) is a relational database of biographical information for China before the early twentieth century. Through the wide range of data it collects, CBDB offers many ways to examine the lives of past individuals and groups. While CBDB provides detailed information about people and can serve as a biographical dictionary, its more powerful use is as a tool for *prosopography*, the study of the lives of groups of people:

'Prosopography' is the investigation of the common background characteristics of a group of actors in history by means of a collective study of their lives. The method employed is to establish a universe to be studied, and then to ask a set of uniform questions – about birth and death, marriage and family, social origins and inherited economic position, place of residence, education, amount and source of personal wealth, occupation, religion, experience of office and so on. The various types of information about the individuals in the universe are then juxtaposed and combined, and are examined for significant variables. They are tested both for internal correlations and for correlations with other forms of behaviour or action. (L. Stone, 'Prosopography', in F. Gilbert and S. Graubard eds., *Historical Studies Today* (New York, 1972)

CBDB also supports a second approach to analyzing the lives of large numbers of people that has begun to emerge in recent years. Social network analysis (SNA) has been a tool for studying group structure in the social sciences for many decades. Recently, however, scholars have applied its techniques to data derived from historical documents. Charles Wetherell describes the project of *historical social network analysis* (HSNA):

Conceptualizing community as collections of personal relationships ... provides historians with a blueprint for evaluating when, how and why people in the past used kin and non-kin in the course of their lives. The findings of social network analysts that people need and seek emotional and economic support of different kinds, from different kinds of people, suggest new analytical imperatives. It is not enough now to look solely at how people used kin in times of crisis. Rather, historians need to pursue how people in the past used the kin and friends they had, for different things, throughout the life course, and in the context of the opportunities they enjoyed and the constraints they faced courtesy of demography and culture. Other approaches might be applied to the problem, but HSNA contains the essential perspectives that cannot only advance the debate, but also help historians to meet Tilly's challenge to connect the lives of ordinary people to large-scale change in meaningful ways. (Charles Wetherell, "Historical Social Network Analysis," *International Review of Social History* 43 (1998), Supplement)

In large measure, historians have used SNA approaches on small sample populations where the relations among all the member of the group are known, but CBDB hopes to provide data on relations among individuals in very large populations where the density of relationship data is adequate to produce statistically meaningful results about patterns in the social world of China's past.

Because CBDB records information about where people lived, where they studied, where they served in office, what offices they held, who their parents were, who they married, and who they knew, all these aspects of life can be correlated for very large groups of people. We can ask if local marriage alliance were typical during a particular period or in a particular region, or for a particular level of office-holder or occupation. We can ask about kinship patterns within occupations for any slice of time and/or any region of China. We can look at regional patterns of sponsorship or partisan opposition. We can look at social, kinship and regional factors in promotions within the Buddhist monastic orders. We can ask who associated themselves with certain ideologues and teachers and where they lived. There is almost no limit to the types of questions that can be asked about the people in the database.

The challenge is how to phrase the questions in ways to which CBDB can respond. The goal of this User's Guide is to provide you with enough information about CBDB, first, to use its interface for common types of queries and then to use other tools for more advanced queries of the dataset. Information about CBDB divides into three parts: general information about relational databases, the structure of CBDB in particular (the types of data it contains), and the interface for looking at the data in CBDB.

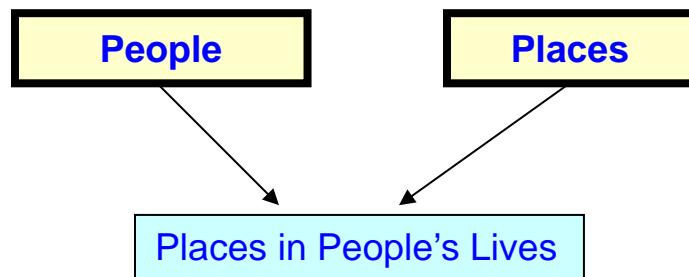
# Chapter 1. Relational Databases

## A. Relational Database and the Organization of Complex Data

The social historian Robert Hartwell, who was concerned with the kinship and social networks of Song Dynasty officials, first conceived of using a relational database to study collective biographies, and CBDB evolved out of his initial model.

Hartwell's important step was to see that he needed a powerful organizing tool to meet the challenges of the project he proposed. He wanted to look at relations between people, their kinship groups, their social networks, the offices they held, and the places with which they were associated. This is a long list, and the interactions between all of these elements grow complex and difficult to track. Hartwell realized that he could think of the interactions he saw in biographical data as relations between (1) people, (2) places, (3) a bureaucratic system, (4) kinship structures and (5) contemporary modes of social association. He built a relational database precisely to capture biographical data as the relations between these five “things.”

This structuring of relationships between entities is what a relational database does: it allows one to capture multiform relations between complex objects in the world that interact with one another. That is, “place” is a category of “things” in the world, but under this category we can list any and all places about which we have information and in which we are interested. Similarly, under “people” as yet another category of “thing” in the world, we list all the people about whom we have biographical information. Then we can list all the interactions we care to record between people and places: where they were born, where they moved, where they were buried, and so on. We have the abstract model of relations between entities:



This abstract model, when transformed into a relational database, becomes a series of tables filled with data:

PEOPLE		
ID	Name	Dates
1	Lü Benzong 呂本中	1084-1145
2	An Dun 安惇	1042-1101
3	Chao Buzhi 晁補之	1053-1110
4	Chen Jian(5) 陳薦	fl. 1069

PEOPLE-PLACES		
Person ID	Place ID	Relation Type ID
1	1	1
1	3	2
1	2	3

PLACES	
ID	Placee Name
1	Jinhua 金華
2	Shouzhou 壽州
3	Kaifeng 開封

### PEOPLE-PLACE TYPES

Relation Type ID	Relation Type
1	Basic Affiliation
2	Moved to
3	Ancestral addr

Note that with this arrangement of tables, there is no limit to the number of people, the number of places, or the number of types of relations between people and places.

From this example of how people and place relate to one another, we see that in relational databases there are three basic types of tables:

- Tables that describe the basic “entities.”** (The yellow tables “People” and “Places” above) In CBDB, these include people, places, kinship term, bureaucratic structure, and so on. The fields in these tables capture the attributes of these entities that we want to know about. For people, this would include their names, birth and death dates, gender, and the like. For places (“addresses” in CBDB parlance) it would include the administrative level of a place, its superior or subordinate units, and the period of validity. For offices this would include where the office fit in the administrative hierarchy during a particular dynastic period.
- Tables that describe relations between basic entities.** (The blue “People-Places” table) In CBDB, these translate the relations between people and their social, physical, and cultural environment into a structured format. The fields in these tables capture the features of the relations that are considered important in describing the relationship. For instance, when a person receives a posting to serve in a bureaucratic office, in addition to the basic information of who the person was and what the office was, we also would like to know (1) where the post was, (2) if the person served, and (3) when he served.
- Tables that describe the types of relations between entities.** (The pink “People-Place types” table.) Sometimes, there can be many ways for two “things” to interact in the world, and we need to be able to be more specific in recording the details of the interaction. In the example above, people can have many different ways of being related to a place: it might be the place at which they were formally registered, the place at which they actually lived, or the place where they were buried. We can group these relations into categories to give them structure.

## B. Rules for Structuring Data in a Relational Database

In databases, we try to record any particular datum only once. In the example above, the name Sima Guang 司馬光 appears in only one record in CBDB, in his basic entry in the table for PEOPLE entities (the table is called BIOG\_MAIN). All other records that record information about Sima Guang refer to him by his ID number. Thus, if, for example, I mistakenly entered the name Hong Shi for 洪适 (properly romanized as Hong Kuo) because I thought that the second character was the simplified form of *shi* 適, I would need to fix the mistake in only one place. This principle of “one datum, one place” is called *normalization*. There are occasions where CBDB violates this rule in order to speed processing, but if you wish to add additional tables to your own version of CBDB, we strongly recommend that you pay attention to the goal of a normalized database.

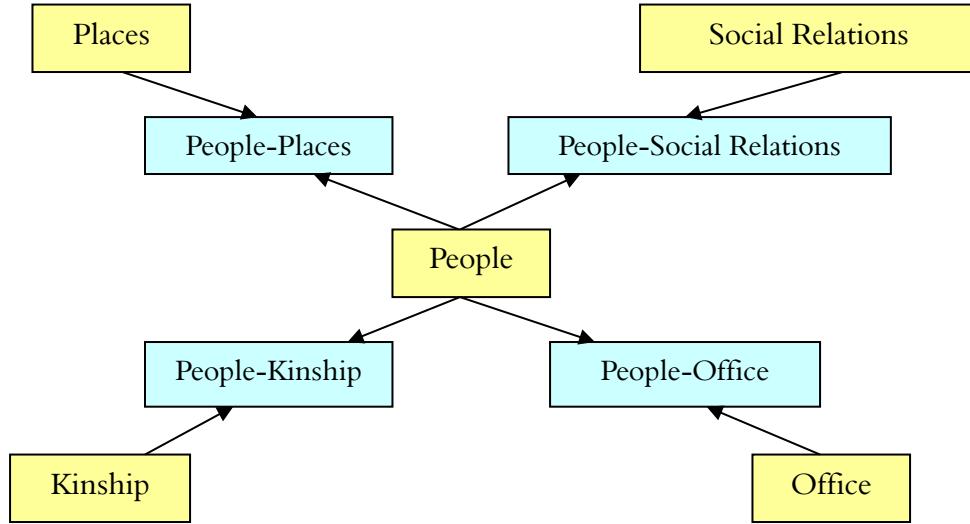
In the example of a person’s relationship to places discussed above, we encounter the fact that a person can move to many different places. This is called a “one-to-many” relationship. If one were to try to represent this relationship through a simple table with rows and columns, we either could create a number of columns in the basic biographical table (“Moved to 1”, “Moved to 2”, and so on), or we could add all entries into a single cell. If we create several columns for “Moved to,” we cannot be sure that we will not encounter an individual who moved so many times that it exceeds the number of columns we created. Moreover, every single record in the biographical table would have all of the “Moved to” cells, which would remain empty for most people. If one were to create just one column for “Moved to” information, searching through the entries in the cell for each individual would make retrieving the data very difficult. The disadvantages of these two approaches to keeping the “Moved to” data in the main table leads to the general rule: whenever we find this sort of one-to-many relationship between basic entities (here, PEOPLE and PLACES), we need a separate entity like PEOPLE-PLACES (and a table to represent that entity) to allow us to capture the interaction.

We encounter a different type of problem when we encode a book like *Record of Things at Hand*, which was edited by Zhu Xi and Lü Ziqian. Writings have a so-called “many-to-many” relationship: one book may have many authors or editors, and each of those writers may have written many books. In CBDB, as in many databases, we treat this situation as a pair of one-to-many relations between PEOPLE and WRITINGS and introduce a new entity, PEOPLE-WRITINGS, to capture the data.

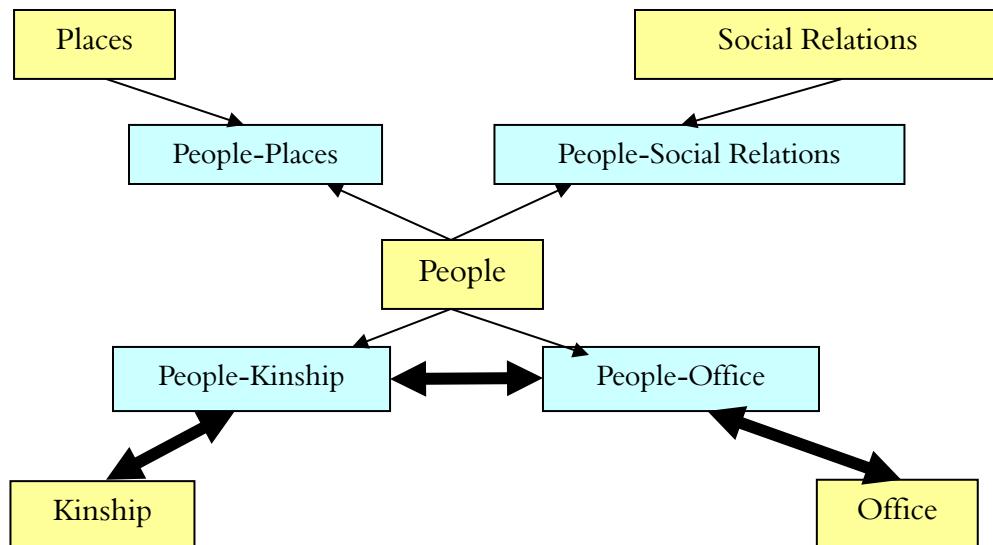
These three rules—normalize data, create new tables for one-to-many relations, and treat many-to-many like one-to-many—are important if you wish to add new data types to CBDB.

### C. Relational Databases and the Interactions of Complex Data

Consider the following set of entities and their relations with the basic entity PEOPLE:

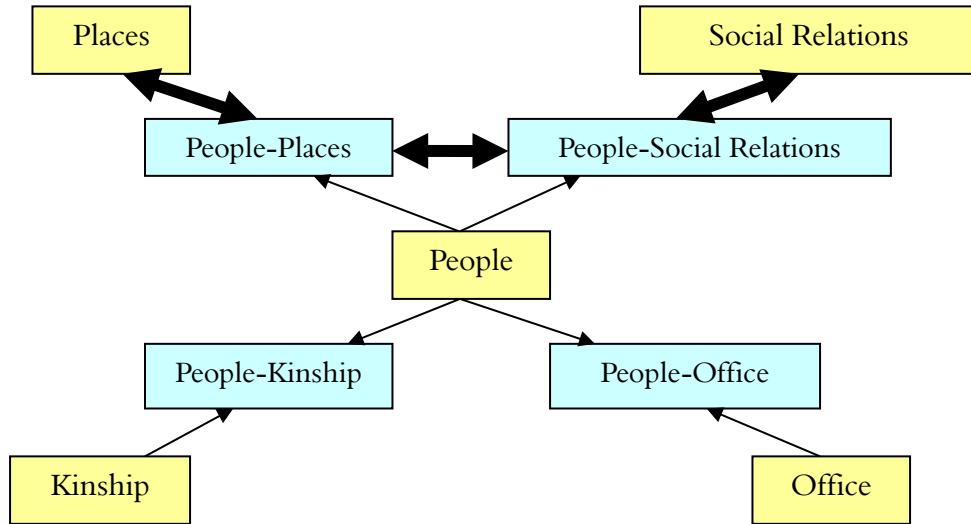


Although there is no direct link between KINSHIP and OFFICE, we still can explore the relation between them through the data we have accumulated about people. We can ask questions like "Was the role of medical officer hereditary, that is, were medical officers the sons or nephews of medical officers, and did the families of medical officers marry the children to one another?" What about men who held mid-level military ranks: were those who moved into civil posts likely to marry daughters of men who held civil posts?



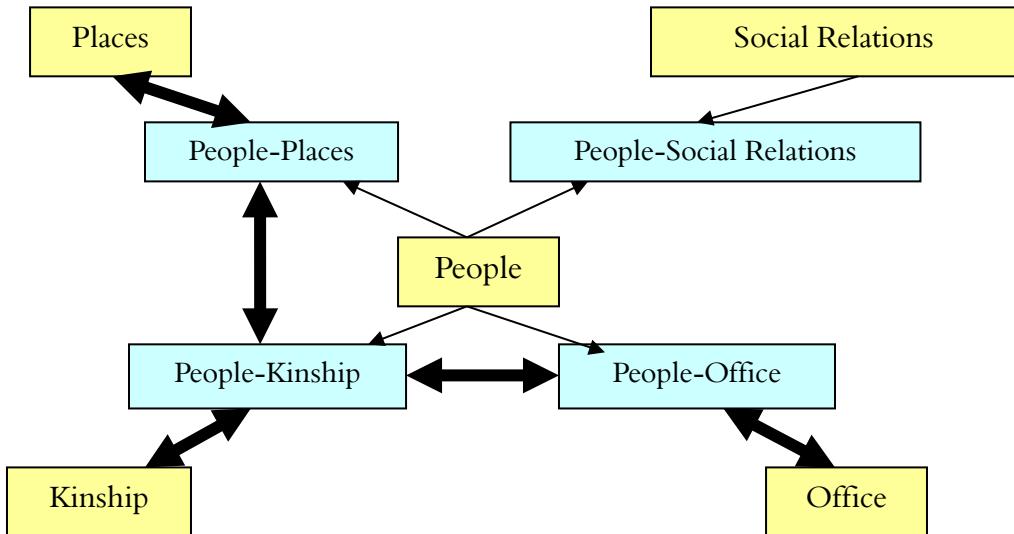
Querying the Relationship between OFFICE and KINSHIP

We can ask many, many questions about the relation of OFFICE and KINSHIP. Were there different patterns of marriage within rank for high civil officials and lower-ranking officials? Did these group form marriage alliances that created different strata? Did these patterns change over time? We can ask similar sorts of questions about PLACE and SOCIAL RELATIONS. Were people from Sichuan, for example, forming local connections, or did they establish empire-wide networks. Did these patterns change from the early to late Northern Song and then again from the late Northern Song to the late Southern Song?



### Querying the Relationship of PLACE and SOCIAL RELATIONS

Finally, we can look at the interaction of multiple factors like the role of PLACE in the relationship between KINSHIP and OFFICE:



### Querying the Role of PLACE in KINSHIP-OFFICE Relations

Were officials from Fujian more likely to develop local kinship networks that were official from Zhejiang? Did patterns differ depending on the rank, and did the patterns change over time?

In a relational database, the only real constraint on asking questions about the interactions of the entities in CBDB is how well one understands the database and the structure of the data in it.

## Chapter 2. The Structure of CBDB

### A. An Overview of the Entities in the Database

Database design uses tables to give concrete form to more abstract objects which we simply call “entities.” Since the goal of a database is to capture the relational information about entities, it remains useful to keep the abstract objects separate from the tables that represent their relations. That way, one can more easily ask the question of how the tables need to change to better stand in for the entities they represent.

The central entity that defines biography in the database is, of course:

#### 1. People

But since a *relational database* tracks the ways in which people form relations with other people, with their society (their political, social, economic and cultural institutions), and with the physical world, we also need entities with which **people** interact. First, relationships with people (these entities will be discussed in greater detail later):

#### 2. Kinship

#### 3. Social (Non-kin) Associations

Next, with political and socio-cultural institutions and activities:

4. **Status** (modes of social distinction such fame for calligraphy or serving as a monk)
5. **Modes of Entry** into Government or other careers (e.g., passing the civil-service examinations, nepotism or the *yin* protection privilege)
6. **Postings** to office (e.g., a magistrate or general)
7. **Events** of significance in which a person participates
8. **Possessions**
9. **Social Institution** in which people collectively participated (from Buddhist temples and Confucian academies to the repair of city walls and bridges)

There also are texts that people produced and through which we know about people:

10. **Texts** (including primary texts, secondary texts, and paleographic data)
11. **Data Sources** from which CBDB draws its information (this includes primary sources, secondary scholarly compilations, and digital resources).

Then, there are structured aspects of the world with which people interacted that must be included in CBDB. The three aspects on which we have focused are administrative geography, physical location, and bureaucratic structure:

12. **Geographic Administrative Hierarchy** (defined in political terms as administrative units)
13. **Longitude and Latitude** (fixed locations in space required for historical comparisons)
14. **Bureaucratic Organization** (the changes in bureaucracy and reporting responsibilities over time)

## ***B. Details of Entities***

NOTE: The database allows one to record the **Source** of information and to add additional **Notes** as seems appropriate. Every item in the database that records information on an individual has the attributes of **Source**, **Pages**, and **Notes**. Therefore I will not note these in the discussions below.

### **1. People**

#### **a. Basic Data: name, male or female, date of birth, and date of death.**

Precise dates of birth and death often are not available, and all we have is a period of **years of activity** (“floruit” dates). Sometimes, not even that is available: we simply know the **reign period** (*nianhao*) or **dynasty**. In order to capture the level of precision in the data, the database allows the use of reign period information for all dates. One can give a specific year within the reign period, but one also can simply indicate “beginning,” “middle”, “end”, or “unspecified.” For analytic purposes, the database will algorithmically produce Western dates from the reign period information for birth, death, years of activity, and any other date given in the traditional Chinese *nianhao* designation, but it will preserve the vagueness in the **nianhao** coding.

#### **b. Ethnicity and Tribe Affiliation**

CDBD tracks ethnicity, like Han, Uighur, Tibetan, etc. We have over 465 codes at present. These codes are in the table ETHICITY\_TRIBE\_CODES, which organizes ethnicity and tribe designations by group and subgroup and includes variant forms for ethnicity names.

#### **c. Choronym**

From the Six Dynasties into the Tang, membership in a clan was of central importance in defining one’s social status. From the Song Dynasty onward, the claims people made about their ancestral membership in a particular clan from a particular place (like the Cui clan of Boling) were difficult to verify. The combination of place name and clan name defined a *choronym*. The codes for these choronyms are in the table CHORONYN\_CODES.

### c. Index Year

For computational purposes, CBDB needs a single year value to locate a person in time. The *index year* (the “sixtieth year of age”) is an artificial value used in analyses. The rules for calculating the value are complex:

#### *Rules Based on a Person’s Birth/Death Dates*

- Rule 1: If ego’s death year < (ego’s birth year + 60 - 1) , then: ego’s index year = ego’s death year; otherwise: ego’s index year = (ego’s birth year + 60 - 1)
- Rule 2: If we only know ego’s death year, then: ego’s index year = ego’s death year
- Rule 3: If we only know ego’s birth year, then: ego’s index year = ego’s birth year + 60 - 1
- Rule 4W: Ego’s index year = (husband’s birth year +3)+60-1 = husband’s birth year + 62 (Note: If the woman was a concubine/second wife, then rule 9W precedes rule 4W.)

#### *Rules Based on Degree Dates*

- Rule 5: Ego’s index year = the year he obtained the Jinshi (進士) + (60-30) = his Jinshi year + 30
- Rule 5W: Ego’s index year = the year her husband obtained the Jinshi (進士) +30+3 = husband’s Jinshi year + 33
- Rule 6: Ego’s index year = the year he obtained the Juren (舉人) + (60-27) = his Juren year +33
- Rule 6W: Ego’s index year = the year her husband obtained the Juren (舉人) +33+3 = husband’s Juren year + 36
- Rule 7: Ego’s index year = the year he obtained the Xuicai (秀才/生員) + (60-21) = his Xuicai year + 39
- Rule 7W: Ego’s index year = the year her husband obtained the Xuicai (秀才/生員年) +39+3 = husband’s Xuicai year + 42

#### *Rules based on Birth Years of Kin*

- Rule 8: If we know the birth year of ego’s **father**, then ego’s index year is decided per assumption A3: ego’s birth year was 30 years later than father’s birth (Ego’s index year = (father’s birth year + 30) + 60 - 1 = father’s birth year + 89)
- Rule 9: If we know the birth year of a **male’s oldest child**, then ego’s index year is decided per A3: ego’s birth year was 30 years earlier than the birth year (Male’s index year = (oldest child’s birth year - 30) + 60 - 1 = oldest child’s birth year + 29)
- Rule 9W: If we know the birth year of a **female’s oldest child**, then ego’s index year is decided per A3: ego’s birth year was 27 years earlier than the birth year of her oldest child. (Female’s index year = (oldest child’s birth year - 27) + 60 - 1 = oldest child’s birth year + 32)

- Rule 10: If we know the birth year of ego's **older brother**, then ego's index year is decided per A4: ego's birth year was 2 years later than older brother's birth year. (Ego's index year = (older brother's birth year + 2) + 60 - 1 = older brother's birth year + 61)
- Rule 11: If we know the birth year of ego's **younger brother**, then ego's index year is decided per A4: ego's birth year was 2 years earlier than younger brother's birth year. (Ego's index year = (younger brother's birth year - 2) + 60 - 1 = younger brother's birth year + 57)
- Rule 12: If we know the birth year of a **male's oldest son-in-law**, then ego's index year is decided per A3 & A4: ego's birth year was 30 years earlier than the birth year of his oldest daughter, and his oldest son-in-law was 3 years older than oldest daughter. (Male's index year = (birth year of oldest son-in-law + 3 - 30) + 60 - 1 = birth year of oldest son-in-law + 32)
- Rule 12W: If we know the birth year of a **female's oldest son-in-law**, then ego's index year is decided per A3 & A4: female's birth year was 27 years earlier than the birth year of her oldest daughter, and her son-in-law was 3 years older than her oldest daughter. (Female's index year = (birth year of oldest son-in-law + 3 - 27) + 60 - 1 = birth year of oldest son-in-law + 35)
- Rule 13: If we know the birth year of ego's **grandfather**, then ego's index year is decided per assumption A3: ego's birth year was 60 years later than grandfather's birth year. (Ego's index year = (grandfather's birth year + 60) + 60 - 1 = grandfather's birth year + 119)

#### *Rules Based on the Index Years of Kin*

- Rule 14: If we know the index year of ego's **father**, then we use father's index year to decide ego's index year per A3. (Ego's index year = father's index year + 30)
- Rule 15: If we know the index year of a **male's oldest child**, then we use that year to decide ego's index year per A3. (Ego's index year = index year of oldest child - 30)
- Rule 15W: If we know the index year of a **female's oldest child**, then we use that year to decide ego's index year per A3. (Female's index year = index year of oldest child - 27)
- Rule 16: If we know the index year of ego's **older brother**, then we use that year to decide ego's index year per A4. (Ego's index year = index year of older brother + 2)
- Rule 17: If we know the index year of ego's **younger brother**, then we use that year to decide ego's index year per A4. (Ego's index year = index year of younger brother - 2)
- Rule 18: If we know the index year of a **male's oldest son-in-law**, then we use that year to decide ego's index year per A3 & A4: ego's birth year was 30 years earlier than the birth year of his oldest daughter, and his oldest son-in-law was 3 years older than oldest daughter. (Ego's index year = index year of oldest son-in-law + 3 - 30 = index year of oldest son-in-law - 27)

Rule 18W: If we know the index year of a **female's oldest son-in-law**, then we use that year to decide her index year per A3 & A4: female's birth year was 27 years earlier than the birth year of her oldest daughter, and her son-in-law was 3 years older than her oldest daughter. (Ego's index year = index year of oldest son-in-law + 3 - 27 = index year of oldest son-in-law - 24)

Rule 19: If we know the index year of ego's **grandfather**, then we use grandfather's index year to decide ego's index year per A3. (Ego's index year = grandfather's index year + 60)

The CBDB table that records this basic biographical information is BIOG\_MAIN. BIOG\_MAIN assigns each person a unique ID.

#### d. **Floruit years**

CBDB gives two years: the earliest and the latest. Often when there is no data for index year or for birth and death dates, texts nonetheless provide datable references to individuals. CBDB gives the earliest and the latest known dates given in the textual sources we have examined so far.

## 2. **Kinship**

An instance of the **Kinship** relationship for an individual has three components (plus the source information):

person  
kin  
kinship relation

This relationship is structured as: "Person A has Person B (the kin) as his/her Kinship Relation." E.g. {Wang Anshi, Wang Anli, B-} means Wang Anshi has Wang Anli as a younger brother.

The building-block relations for **Kinship** are the 9 basic categories:

e	Ego (the person whose kinship is being explored)
F	Father
M	Mother
B	Brother
Z	Sister
S	Son
D	Daughter
H	Husband
W	Wife
C	Concubine

There are also variations on the nature of the relationship, as well as additional types of notation to represent types of kinship relations beyond the nuclear family:

+	Older (e.g. older brother B+, 兄)
-	Younger (e.g. younger sister Z-, 妹)
★	Adopted heir (as in S★, adopted son)
○	Adopted
!	Bastard
^	Step- (as in S^ step-son)
½	half- (as in Z½, half-sister)
~	Nominal (as in M~, legitimate wife as nominal mother to children of concubine)
%	Promised husband or wife (marriage not completed at time of record)
y	Youngest (e.g., Sy is the youngest known son)
1, 2, 3...	Numbers distinguish sequence (e.g., S1, S2 for first and second sons; W1, W2 for the first and the successor wives)
n	precise generation unknown
G-#, G+#+	lineal ancestor (-) or descendant (+) of # generation
G-n, G+n, Gn	lineal kin of an unknown earlier generation (G-n), or unknown later generation (G+n), or unknown generation (Gn)
G-#B, BG+#+	a brother of a lineal ancestor of # generation; a brother's lineal descendant of # generation
K, K-#, K+#, Kn	Lineage kin, of the same, earlier (-), later (+) or unknown (n) generation. CBDB uses "lineage kin" for cases where kinship is attested but the exact relationship is not known. Lineage kin are presumably not <i>lineal</i> (direct descent) kin.
K-, K+	Lineage kin of the same generation, younger (-) or elder (+).
P, P-#, P+#+, Pn	Kin related via father's sisters or mother's siblings, of the same, earlier (-), later (+) or unknown (n) generation. Signified by the term <i>biao</i> (表) in Chinese. (CBDB uses these codes only when the exact relationship is not known).
P-, P+	Kin related via father's sisters or mother's siblings, of the same generation, younger (-) or elder (+).
A	Affine/Affinal kin, kin by marriage

The codes for the types of relationships are in the table KIN\_CODES.

The kinship data is in the table KIN\_DATA.

### 3. Non-kinship Associations

#### a. Simple Non-kinship Associations

These have a three-part structure: person + association + associate. The major challenge

in recording the non-kinship **Associations** that individuals formed over their lives is to control the proliferation of categories.

Because associations are between pairs of people, there must be symmetrical types of associations. That is, if  $\{A\}$  “is the student of”  $\{B\}$  is in the database, then  $\{B\}$  “is the teacher of”  $\{A\}$  also should be so. In fact, the current version of the program automatically generates this second entry. Thus Associations as an entity has an internal structure:

- Association type
- Paired Association type
- Association Categories/subcategories (3 levels at present)

When one creates a new category of **Association**, one must also create its converse. Mutual associations, of course, are their own converse:  $\{A\}$  “is friend of”  $\{B\}$  is the same as  $\{B\}$  “is friend of”  $\{A\}$ . In most associations, however, the two people play distinct roles, and CBDB needs the converse category to capture the roles of the two people from their different perspectives: to record for  $A$  that  $\{A\}$  “followed”  $\{B\}$  also means that for  $B$ ,  $\{B\}$  “was followed by”  $\{A\}$ . Some associations are not dyadic because the relation is not to a person but to a more abstract or general object. The most important type of association of this type is the faction. Thus we have  $\{A\}$  “is member of the purged Yuanyou group”  $\emptyset$  ( $\emptyset$  here is the *Null* (Empty) element.) A few rare relations are **triadic**:  $\{A\}$  “ordered”  $\{B\}$  “to execute”  $\{C\}$ . CBDB has added a third field to capture such situations but does not yet process such triadic relations in its analytic routines.

### b. Mediated Associations

In some important cases, associations form through the mediation of institutions or people. CBDB captures these types of relations by adding additional data to associations. For example, we might know of a relation between X and Y because X asked Y to write a biography for his mother’s tomb.

### c. Structure of an Association Record

Because associations in pre-modern Chinese society often are complex, the table tracking associations in CBDB uses a correspondingly large number of fields:

- Basic Information
  - 1. Person
  - 2. Associated person
  - 3. The association
  - 4. The number of objects or events establishing the association

Information about Kinship and Other Relations that played a role in the Association

5. The kinship relation, if the association was established through a relative of the person
6. The person whose kinship relation established the association
7. The kinship relation, if the association was established through a relative of the *associated* person
8. The kin of the associate through whom the association was established
9. The name of the person who claimed the existence of the association: for example, a son claiming it for his father

#### Time and Place of the Association

10. The place of the association
11. The sequence of an association, if one does not know the actual date
12. The date of the association (year, month, and day, if known)

#### Contextual Information

13. The social institution at or through which the association was established
14. The occasion on which the association was established
15. The genre of the writing that establishes the association, if relevant
16. The title of the work that established the association, if relevant
17. The scholarly topic around which the association was formed

#### Source and Notes

18. Source
19. Note

### d. Future Developments (in planning)

#### 1. Mourning Associations

In China, a person can choose to participate in mourning for a teacher or a person connected through some other form of non-kinship association. Mourning has five aspects that are represented in the database:

Mourner  
Mourned  
Length of Mourning  
Color of Mourning Robe  
Date

It may prove that this category of association is most important in early China, but it should be expected that as Chinese culture transformed over time, sources for biography stressed different forms of activities and relationships. The database needs to be able to accommodate these historical shifts. At present CBDB has no way to record the details of mourning association: once we have more data from earlier individuals, we will assess the best way to proceed.

## 2. Gift Giving

Another type of association is created through the giving of gifts. This practice also has five aspects:

Giver  
Recipient  
Gift  
Value (or quantity) of gift  
Date

At present, CBDB does not record this information, which is perhaps more important in early biographical texts. As with mourning, we will assess the situation once we have more information.

## 4. *Status*

CBDB has a separate table to take note of a person's "social distinctiveness," that for which they are known in society. Since the dating often is uncertain, however, the table has a field to record **sequence** if known. Some forms of social distinctiveness may combine roles (a Buddhist monk known for his calligraphy, or a literatus who runs a printing firm). At present, CBDB records the different aspects of status under distinct categories. This is a question awaiting future research: it may be better to aggregate the two social roles, but this is largely an empirical question of how often such merged roles appear and whether they seem to have been viewed as a single "status" rather than two. The structure of a Status datum for a person is:

Person  
Status code  
Status sequence  
Date  
Source information and notes

**Status** as a category of social experience (as opposed to any particular person's status within the structure of social distinction) is a simple entity:

Status code  
Status description  
Status category and subcategory 1  
Status category and subcategory 2

Since social distinctions change over time, CBDB will need to add to its current list as it draws upon sources for earlier and later periods.

## 5. *Modes of Entry*

**Entry** itself is a simple entity, just a name, a type, and a subtype. At present it largely describes entry into government, but CBDB also has begun to track categories like monks' ordinations. Because different routes of entry entail different types of information, the instance of an **Entry** event for an individual is more complex. If a person enters government through the examination system, for example, we would like to know the type of examination and the date of the degree. (CBDB also tracks failed examinations.) If, in contrast, one enters government through the merit of someone else, the person, and the relationship to the person should also be recorded, if known. Thus if Zhang Weisan entered office through *yin* protection privilege deriving from his uncle Zhang Jingyi, the entry would be:

Person: [ID of] Zhang Weisan  
 Entry type: [code for] *yin*  
 Entry relation type: [code for] Uncle  
 Entry relation: [ID of] Zhang Jingyi

Since it is also possible that one can enter office through the *yin* privilege of a non-kin associate, the “entry event” will need to have a way to record the non-kinship relation. In the end, then, the **Entry** event has many attributes, only some of which are relevant to any particular instance:

Person ID  
 Entry type code  
 Entry relation type code (for kin)  
 Entry associate type code (for non-kin)  
 Entry associate ID (used for both kin and non-kin)  
 Entry test date (both Western and *nianhao* + year (if known))  
 Entry test ranking

## 6. *Offices and Postings*

CBDB currently lists over 24,000 office titles and—at least for the Song—their place in the government bureaucracy. **Postings** are entities at the intersection of people, the bureaucracy, and—since most will be away from the capital—places. A person serves in an office at a given rank in particular place at a specified time. However, there are instances when a posting includes jurisdiction over more than one administrative unit, and there are times when a single posting entails more than one official position. Following the rule that one-to-many relations (i.e., one posting <> many addresses and one posting <> many office titles) require separate tables, information about postings requires *three* entities: a basic **postings** table, a **posted-to-office** table, and a **posted-to-office-address** table. Because the administrative unit is tied specifically to the *office* rather than the posting more generally, the address table records the relation of posting-office information and place:

**Posting-Data**

Posting ID (this is a unique number)

Person ID

Source and Notes

**Posted-to-Office**

Posting ID

Office ID

Office Type

Sequence (since often only the order of office is known with no further information about the years for any of the postings)

Year (both Western and *nianhao* + year: a person may have duties added while still serving in a post)

Sources and Notes

**Posted-to-Address**

Posting ID

Office ID

Address ID

**Considerations for Future Development**

## 1. Buddhism and Daoism

Buddhist and Daoist bureaucratic positions eventually will be added to the **Office Name/Office Function/Postings** entities. This, however, also entails significant research to clarify the historical changes in the structure of the Buddhist and Daoist bureaucracies.

## 2. Tracking Historical Change in Bureaucratic structure

One of the design issues that need to be considered again is how much of the complexity of the Chinese imperial bureaucratic system should be captured in the database. In the Chinese system from the Han through the Qing, the duties of a position may change even though the title of the office remains constant, or the duties may remain constant although the title changes. Scholars have objected that Charles Hucker's *Dictionary of Official Titles* tries to force a continuity of function onto office names when it would have been more useful to simply acknowledge the drifts. Hucker's translated titles are indexed, however, and provide those who do not read Chinese with an easy means of further investigation. CBDB is planning to create tables that will capture the historical changes in the functions designated by any particular office title. (**Office Name** would become one entity and **Office Function** would be another.) Most of the actual duties of an office at any particular time are not relevant to the CBDB because these details contribute little to the analytic power of the database; the attributes of an office that do matter are (1) office as an indication of salary/rank or actual function, (2) the other office to which it reports, and (3) the type of

the office (i.e. central military, prefectoral civil, etc.) At present, CBDB has captured some of this information, but clarifying the changes in office title is in itself a major research project.

## 7. *Places*

CBDB uses a strategy for coding places that derives from the CHGIS project and relies on two types of spatial entities: **Addresses** and **Places** (longitude and latitude).

To begin with, there are **Addresses**: these are specifically historical “instances” of place designation that refer to an administrative jurisdiction bounded in space with a particular name. They have a location defined by a point using x and y coordinates. If either the boundaries or the name changes, a new **address** must be created. These historical instances, however, are part of administrative hierarchies: this information is preserved in a “belongs-to” table that serves the same function as the “part-of” table in CHGIS. Since an address ID changes only when the unit changes shape or name, it *does not* change ID simply when it becomes part of a different higher level administrative unit. Thus there are two tables:

### **Address Code**

- Address code
- Address name
- Administrative type
- X coordinate
- Y coordinate
- Address first year
- Address last year

### **Belongs to**

- Address code
- Belongs-to Address code
- Belongs-to first year
- Belongs-to last year

From these two tables CBDB generates a convenient **Addresses** table that is used in the online database and can be consulted in the stand-alone version to provide information about the role of administrative units in the bureaucratic structure. Its structure is:

- Address code
- Address name
- Address first year (that the address belongs to the superior place)*
- Address last year (that the address belongs to the superior place)*
- Administrative type
- X coordinate

Y coordinate

*belongs1 (the parent: the larger administrative unit it reports to)*

*belongs2 (the parent of the parent)*

*belongs3 (etc.)*

*belongs4*

*belongs5*

To allow the examination of trends across dynamic boundaries, the database needs a way to examine what happens in a particular location over long periods of time. For this, CBDB relies on data about physical location, the x-y coordinates on the map. Robert Hartwell in his initial version of the database had co-located historical districts (or counties, *xian*) with the district boundaries in 1990. However, rather than using misleading boundaries, CBDB now uses centroids, the point equidistant from most borders, as a point location with x-y coordinates.

In sum, there are two tables that are part of the basic representation of places in CBDB:

Administrative Units: ADDR\_CODES

Administrative hierarchy: ADDR\_BELONGS\_DATA

## Considerations for Future Development

When the China Historical Geographic Information System project is completed, we will replace the CBDB internal geo-coding system with the CHGIS tables. **However, there is one important aspect of the CBDB approach that is not basic to CHGIS and will need to be confronted.** That is, CHGIS does not concern itself with long-time trends. It has a table that records the change from one address code to another, but this approach does not meet the needs of CBDB. Consider, for example, if the central authorities decided to combine three depopulated counties A, B, and C into one large county D and thus save administrative expenses. Suppose that thirty years later the government decided that this was a mistake and redivides D along the original boundaries into A', B', and C'. A “transitions” table would record A→D and D→A' but would have no way to reconstruct the fact that A→A'. CBDB needs to track the correspondence of administrative to physical location over time in order to provide longitudinal data. We intend to use the x-y data and the belongs-to information to define larger entities such as prefectures and circuits in terms of the set of x-y coordinates of the counties that make up the larger unit. We will use squares around those x-y coordinates to define a physical region that can be searched across time for the relevant data.

### 10. Biographical Place Information

People have many connections to place: where they were born, lived, died, and were buried, where they served in office, where they held property and ran businesses, where

they visited. Since these relations to place arise out of activities recorded in separate tables in CBDB (e.g., office holding, and possessions), the information appears in these various tables rather than in one place. The tables that record information about people and places are:

- Basic biographical information about place (BIOG\_ADDR\_DATA)
- Place of official service (POSTED\_TO\_ADDR\_DATA)
- The place where a non-kinship relation took place (ASSOC\_DATA)
- The place of an event in which people participate (EVENT\_ADDR)
- Places where people's possessions are (POSSESSION\_ADDR)
- The place where people participated in social institutions (BIOG\_INST\_DATA)

Note that at present CBDB does not preserve information about where people briefly visited, where they received their education, or where they wrote texts. CBDB does not yet provide a form to allow the user to ask questions that integrate all these sources of place information.

## 11. *Texts*

There are three major types of texts of concern to the database: inscriptional and other paleographic material, printed primary texts, and secondary scholarship (in both print and digital form). Since a work like Huang Zongxi's *Song Yuan xue'an* is both a scholarly compendium of earlier writings and a work in its own right, and since the paleographic materials also were written by authors who are of interest to the database, these distinctions for pre-modern texts of any sort are neither clear nor useful. CBDB accordingly treats all three types as **texts**. Texts have the attributes one can expect:

- title
- category of writing (inscription or manuscript/printed)
- genre (the bibliographic categories common to that period)
- current publication date
- current publisher
- current publication location

People can relate to the text in a variety of ways:

- author
- publisher
- editor
- collator
- translator
- annotator

The tables for texts are:

**Texts Codes**

Text ID  
 Text Name  
 Date of composition  
 Current status: extant or not  
 Current Publication Information (if extant)

**Text Data**

Text ID  
 Person ID (CBDB includes all attributions of a text name to a person)  
 Role ID

**Text Role Codes**

Role ID  
 Role description

**Considerations for Future Development**

Inscriptional materials have a few additional attributes recorded in separate tables that have not yet been developed because we have not yet dealt with this type of data:

alternate names  
 place where discovered  
 date of discovery  
 current location  
 source of information

These materials also can involve additional people, i.e., donors and recipient.

Since the texts can serve as sources for biographical information, CBDB records the publication information for the modern edition used, since source information for entries includes page numbers. However, CBDB does not aspire to serve as a standard reference for bibliographic information. It (at least at present and in the near future) will not list all the extant editions of texts for authors nor adjudicate which are the most reliable among those extant editions. Part of the future plans for CBDB on the web is to develop links between the database and other web resources: bibliographic sites certainly will be among such links.

**7. Events**

CBDB only recently has begun to make a concerted effort to record information about groups of people linked together by shared participation in important events. At present the list of events, which comes from the *Song shi jishi benmo* 宋史紀事本末, is heterogeneous and includes such matters as the successful crafting of peace treaties,

rebellions, the removal and restoration of empresses, and various factional disputes. Since communities were forged by common purposes discovered in these events, they are worth preserving. CBDB records events through three tables:

#### **Event-Codes**

- Event ID
- Event name
- Event date

#### **Event-Data**

- Event ID
- Event-Record ID (this is a unique number to track instances)
- Person ID
- Date (sometimes it takes a while for events to catch up with people)

#### **Event-Address**

- Event-Record ID
- Address ID (in case a person's involvement in an event is linked to more than one place.)

### **8. Possessions**

CBDB has barely begun collecting data about possessions. One aspect of the information is the type of transaction (purchase, donation, etc.) recorded in a historical text through which we know about the possession. Although CBDB tables exist, we may change their structure to accommodate the type of information we get as we collect relevant material.

### **9. Social Institutions**

People participated in the lives of their communities in many ways. A man, for example, may have served for several years as the director of an academy. That academy had students during this period: their respective roles in the academy would have served as important social links between the man and the students. The academy also had donors who contributed to its creation and upkeep and helped to define a community centered on the institution. Similar patterns appeared for Buddhist monasteries and Daoist temples.

CBDB is beginning to track this information in a way that captures the uncertainty we find in the historical sources. There are, for example, thirty-nine temples with the name Kaiyuansi 開元寺. A biographical source may tell us that Wang Anshi contributed funds to repairs at a Kaiyuansi, but we may not know (yet) which Kaiyuansi was the recipient. Other sources eventually may clarify the point, but for the moment CBDB simply records "a Kaiyuansi." There are four tables used to record this information:

**Social\_Institution\_Name\_Codes**

Institution Name ID

Institution Name

**Social\_Institution\_Codes**

Institution Name ID

Institution Code (this is a unique ID for each institution: the name may change, but the ID does not.)

Institution Type ID

Institution Dates (this includes the beginning and ending years, if known, as well as the first known and last known years)

**Social\_Institution\_Addr**

Institution Name ID

Institution Code

Address ID (this gives an approximate location by identifying an administrative unit)

XY-coordinates (this may be more precise than the coordinates associated with the Address ID. An institution may move within its locality.)

Address Type (derived from Address ID or recorded independently)

Address Dates

**Biog\_Inst\_Data**

Person ID

Institution Name ID

Institution Code (if only the name is known, CBDB assigns a 0 to this field)

Institutional Role Code

Role Dates

**SUMMARY OF TABLES IN CBDB****1. Basic Entities**

These represent the basic elements of the social world of pre-modern China. Each has a complex history and structure that are set out in additional ancillary tables. CBDB records the interaction of people with these aspects of their world in the secondary tables.

Table Name	Description
ADDR_CODES	the units in the administrative geography of China.
ADDRESSES	a convenient reference table that displays the hierarchy
ASSOC_CODES	the non-kinship social relations that connected people
BIOG_MAIN	the people of pre-modern China
ENTRY_CODES	the means by which people entered into institutions
EVENT_CODES	significant events

KINSHIP_CODES	the kinship categories of pre-modern China
OFFICE_CODES	the units of the bureaucratic organization of government
PLACE_ID	a stand-in for standard units of longitude and latitude
SOCIAL_INSTITUTION_CODES	a list of academies, monasteries, temples, etc.
STATUS_CODES	the means by which people attained social distinction
TEXT_CODES	the corpus of pre-modern writings + important secondary works

## 2. Relations between Basic Entities

Table Name	Description
ADDR_BELONGS_DATA	data for the hierarchical structure of administrative units
ALTNAMES_DATA	the many names by which people were known
ASSOC_DATA	the non-kinship relations between people
BIOG_ADDR_DATA	relations between people and administrative geography
ENTRY_DATA	the initiating relations between people and institutions
EVENTS_ADDR	relations between people, events, and places
EVENTS_DATA	relations between people and events
KIN_DATA	the kinship relations connecting people
OFFICE_TYPE_TREE	the hierarchical structure of bureaucratic organizations
POSSESSION_ADDR	the relations between people, material goods, and place
POSSESSION_DATA	“Possessions” remain at the descriptive level at present
POSTED_TO_ADDR	the relations between people, office, and place
POSTING_DATA	the container table for postings: people linked to office
POSTED_TO_OFFICE_DATA	the details of people’s connection to office
STATUS_DATA	data on a person’s place in the system of social distinctions
TEXT_DATA	the relations of people to texts
BIOG_INST_DATA	the relations of people to social institutions
BIOG_SOURCE_DATA	the list of sources used in defining the CBDB data for a person

## 3. Relationship Type Information

Table Name	Description
BIOG_ADDR_CODES	the categories of relations between people and places
ALTNAMES_CODES	the categories of names by which people were known
APPOINTMENT_TYPE_CODES	the categories of relations between people and postings: regular, acting, probationary, etc.
ASSOC_TYPES	broader categories of social relationships that organize the many non-kinship association codes into groups
ASSUME_OFFICE_CODES	indicating whether a person took up the posting
ENTRY_TYPE	broader categories of entry to organize the entry codes into groups
EXTANT_CODES	indicating degree of the source and its known existence
GENRE_CODES	the bibliographic classifications of texts
GENRE_TYPES	the broader categories of bibliographic classifications
LITERARYGENRE_CODES	the forms of literary composition

OCCASION_CODES	the events in which people participated
OFFICE_TYPES	the categories of offices
POSSESSION_ACT_CODES	the categories of relations between people and goods
SCHOLARLYTOPICS_CODES	the categories of topics of learning and scholarship
SOCIAL_INSTITUTION_TYPES	the categories of social institutions
TEXT_ROLE_CODES	the categories of relations between people and texts
YEAR_RANGE_CODES	the relative degree of exactness of a date
BIOG_INST_CODES	the roles a person plays in relation to an institution
SOCIAL_INSTITUTION_ADDR_TYPES	the type of address (actual or derived) used for an institution

#### 4. Historical Auxiliary Tables

Table Name	Description
CHORONYM_CODES	codes for the place+surname used to identify medieval clans
COUNTRY_CODES	codes for countries appearing in the data
DYNASTIES	codes for dynasties and periods
ETHNICITY_TRIBE_CODES	codes for ethnic groups appearing in the data
GANZHI_CODES	codes for the sixty two-character terms in sexagenary cycle
KIN_MOURNING	codes for all kin relations and mourning obligations in the five degrees of mourning
MEASURE_CODES	codes for quantities of goods, money, books, and space
NIAN_HAO	codes for all reign period titles
SCHOOL_CODES	codes for schools and academies (to be populated)
SOCIAL_INSTITUTION_ALTNAMES	a list of alternative names for social institutions
SOCIAL_INSTITUTION_ALTNAMES_TYPES	codes for different types of alternative names

#### 5. Analytic Auxiliary Tables

Table Name	Description
ASSOC_CODE_TYPE_REL	the relationship of specific social relations to larger categories of social relations
ENTRY_CODE_TYPE_REL	the relationship of specific modes of entry to larger categories of entry
GENRE_CODE_TYPE_REL	the relationship of specific genre codes to larger categories of genres
OFFICE_CODE_TYPE_REL	the relationship of specific offices to the office hierarchy
OFFICE_CATEGORIES	the categories of offices: rank, honorary, etc.

## Chapter 3. CBDB Tools for Analysis

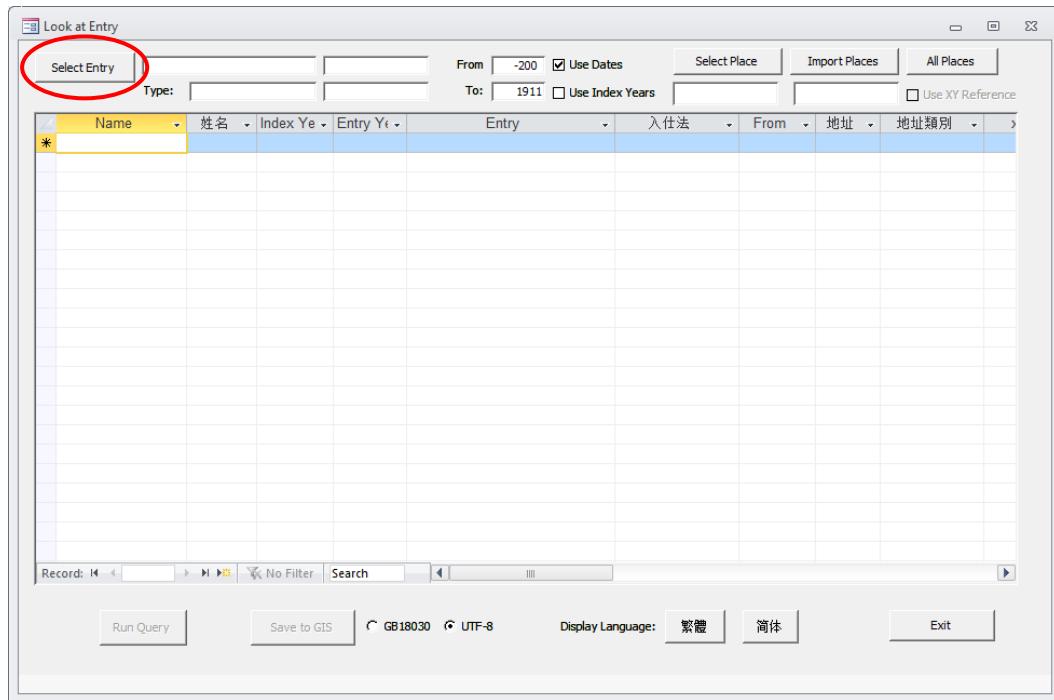
The China Biographical Database contains large amounts of information, but the information is of little value unless there are ways to analyze it. At present, the Access version of CBDB has six forms specifically designed to allow the user to query the database about important categories of information. The names of the forms describe their function.

1. **LookAtEntry** allows one to find groups of people who qualified for office through a particular route for a specified period.
2. **LookAtAssociations** allows one to find groups of people who were linked through a particular category of association
3. **LookAtOffice** allows one to look at not only the people who held particular offices but also those who held related offices subordinate to ever higher levels of bureaucratic structure.
4. **LookAtKinship** allows one to examine the kinship networks for individuals. These include both the mourning circle of the traditional Chinese kinship system and more extended sets of relations.
5. **LookAtNetworks** allows one to look at all the networks (both kinship and social relations) for an individual, a group of individuals, or a specified place.
6. **LookAtAssociationPairs** allows one to examine the intersection of ego-nets for two individuals. It locates both people connected to the two target individuals but also can identify connections at one further remove (i.e. people who had a connection with the first individual who had relations to people somehow related to the second individual).

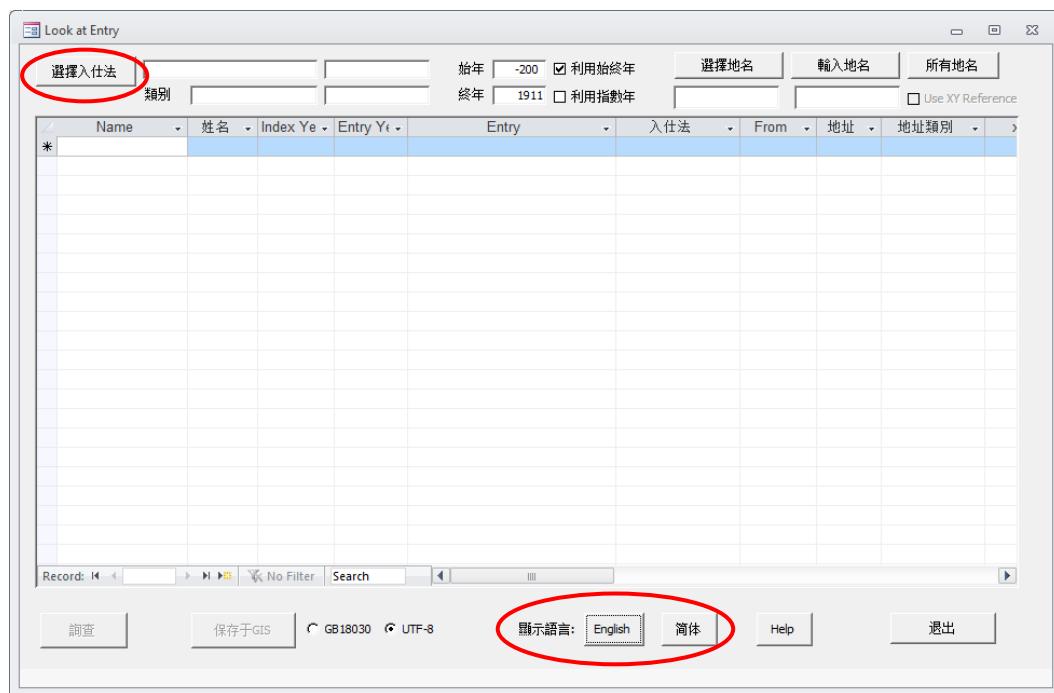
In more complicated queries, one can explore relations between groups of people by using the results of a search in one form as the input to a second form. Chapter 4, on advanced queries, considers an example of this approach. Beyond the six forms, however, Access also allows the user who is familiar with the structure of the database to make queries that can look at any and all aspects of CBDB. This process of writing SQL queries to examine the data is the second topic in Chapter 4.

## A. Using the Form LookAtEntry

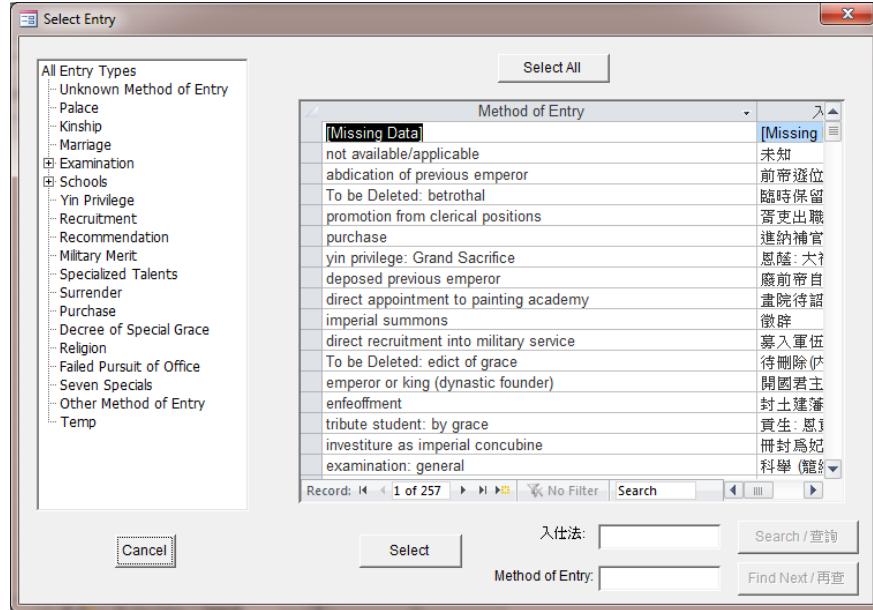
**LookAtEntry** is the simplest form. One opens it by clicking on “Query by Methods of Entry into Government” on the main page and clicks on the “Select Entry” button to choose a category:



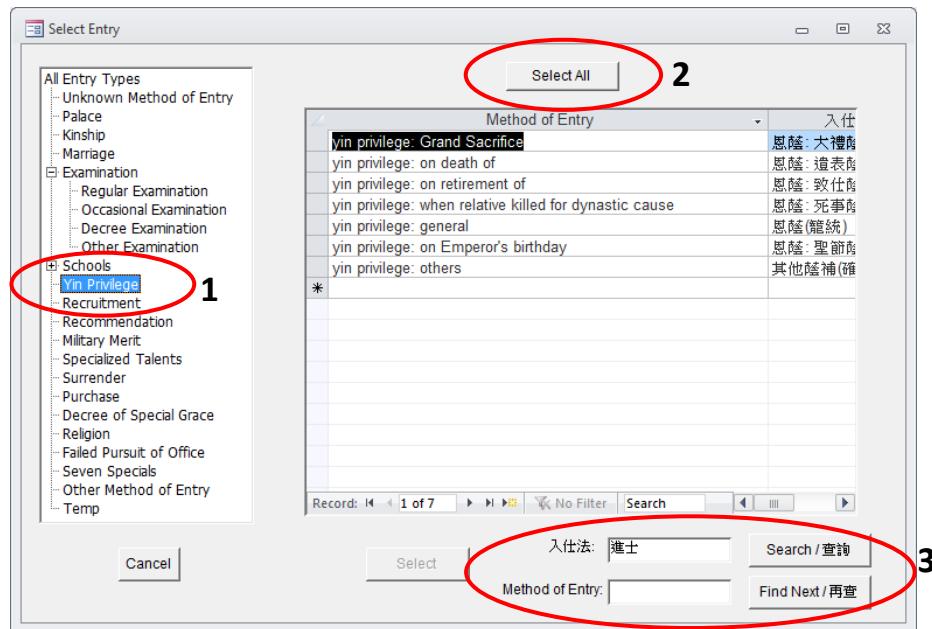
Note that all of the forms have the option to switch between English, traditional or simplified Chinese. When one click on the “繁體” label, it then gives one the option to return to English:



Clicking on the **Select Entry** button opens a form with a list of options. Since there are many different ways to attain eligibility for office. CBDB uses a collapsible tree to simplify the selection process:

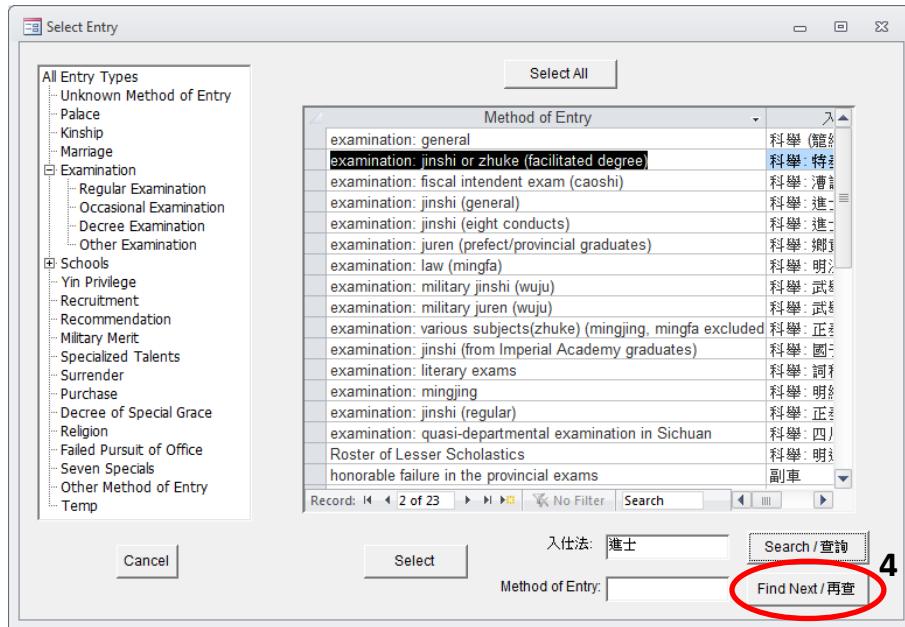


One can narrow the choices by looking at a particular general type of entry which is on the menu on the left of the window (1):

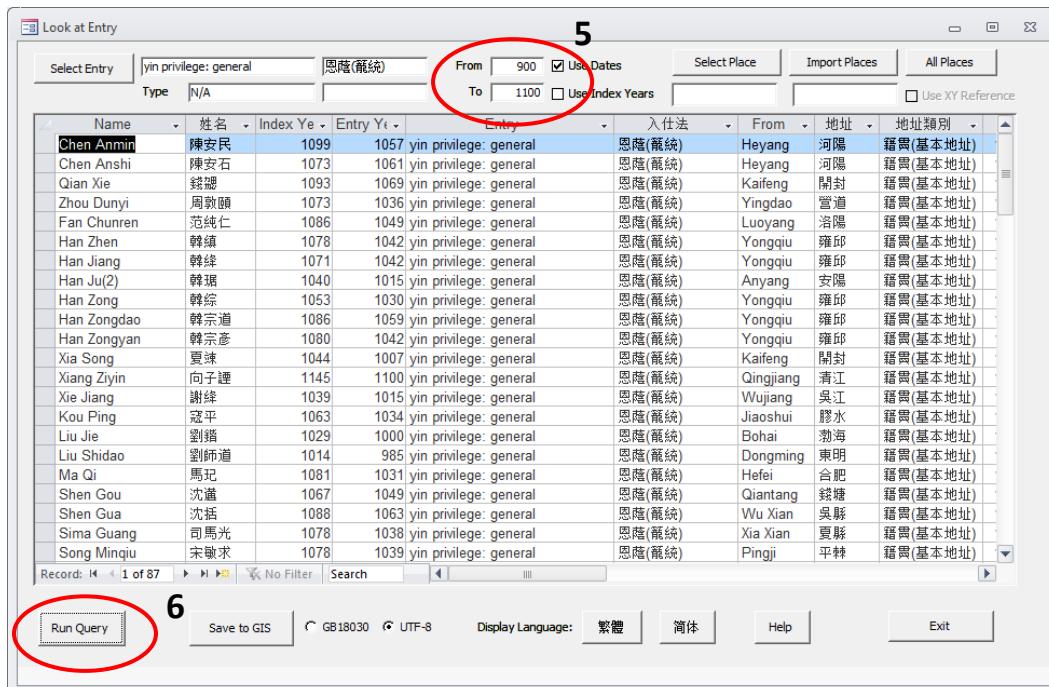


One can either select a specific method of entry from the menu on the right or select all the listed methods (2). One also can search for a specific method using the search box located on the bottom right corner (3). The searching rules for CBDB are to first look for the search

phrase at the *beginning* of the text and then look *within* the text. If CBDB finds the search, you can search for the *next* instance of the phrase (4), if the first is not what you seek by clicking on the “Find Next” button:



After one finds the method of entry and clicks **Select**, one returns to the LookAtEntry form, and can now choose the year range (5) to run the query (6):



The form allows one to choose either the entry year or the index year of the person. (The index year is included in the search if the box “Use Index Years” is selected.) Because in many

cases we do not know the entry year (given as zero in that case), it may prove useful to run the same query with the “Use Index Years” check box clicked:

Note the number of 0s in the “Entry Year” column. This approach yields 564 records, compared with just 87 when using the entry year.

The table the query produces has 15 columns:

Name (Pinyin)
Name (Chinese)
Index Year
Entry Year
Description of Entry (English)
Description of Entry (Chinese)
Place of Origin (Pinyin)

Place of Origin (Chinese)
X-coordinate of place of origin
Y-coordinate of place of origin
Count of XY coordinates

Exam Rank

Kinship Relation

Kin Name (Pinyin)
Kin Name (Chinese)

CBDB looks for the person's index-place, but if is not recorded, it takes the first place listed in the BIOG\_ADDR\_DATA table

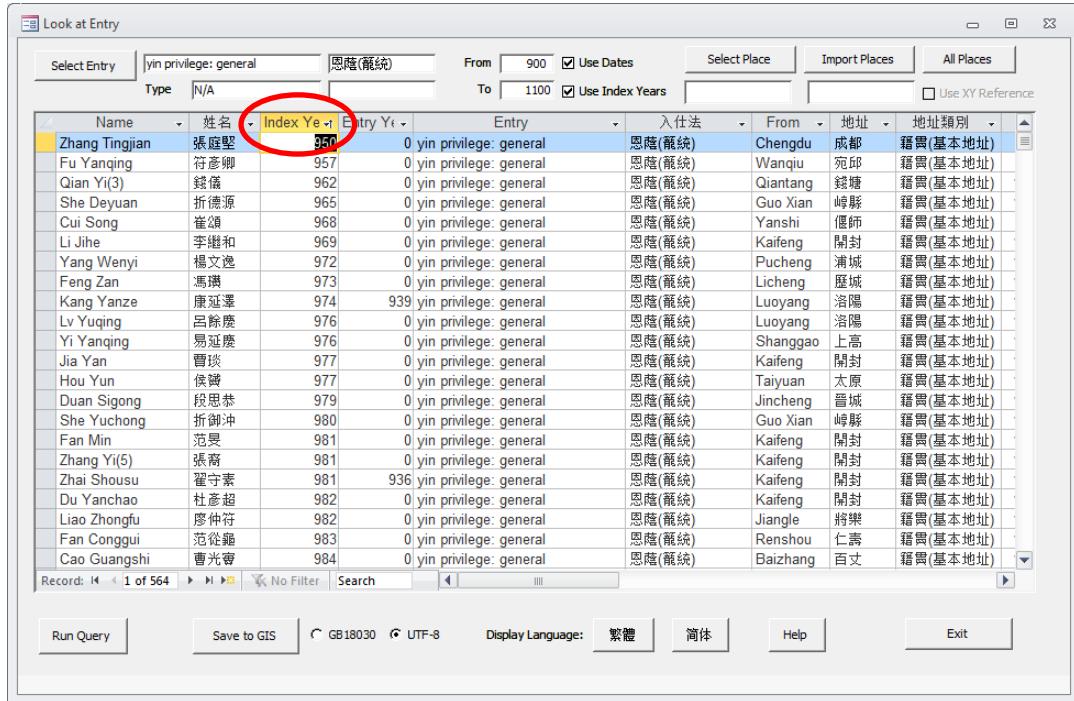
At present CBDB uses centroids to define locations but will use administrative seats

CBDB calculates how many people in the table share the same place of origin

Given, if known. It obviously is irrelevant for people who enter through *yin* privilege.

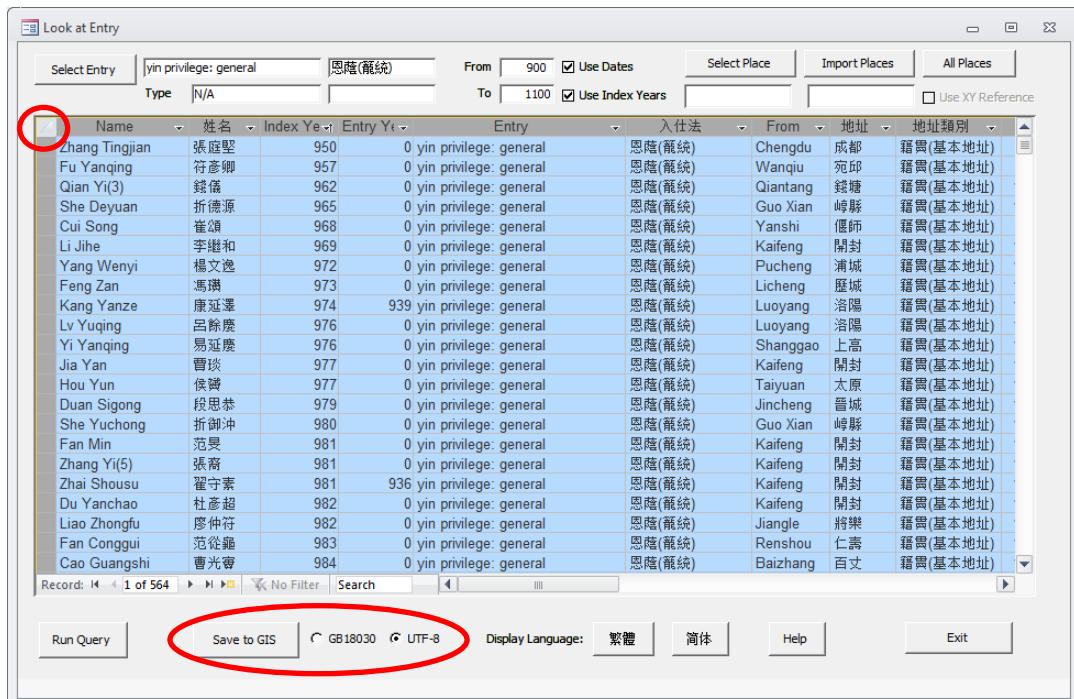
*Yin* privilege allows a person to become eligible for service based on the merit of a relative. If CBDB knows who that relative was and what the relationship was, the table gives this information

One can **sort** the table using any of these columns. For example, “Index Year” may be useful. *Left-Click* on the column name “Index Year” to select the column and then *Right-Click* to choose from the sorting options:



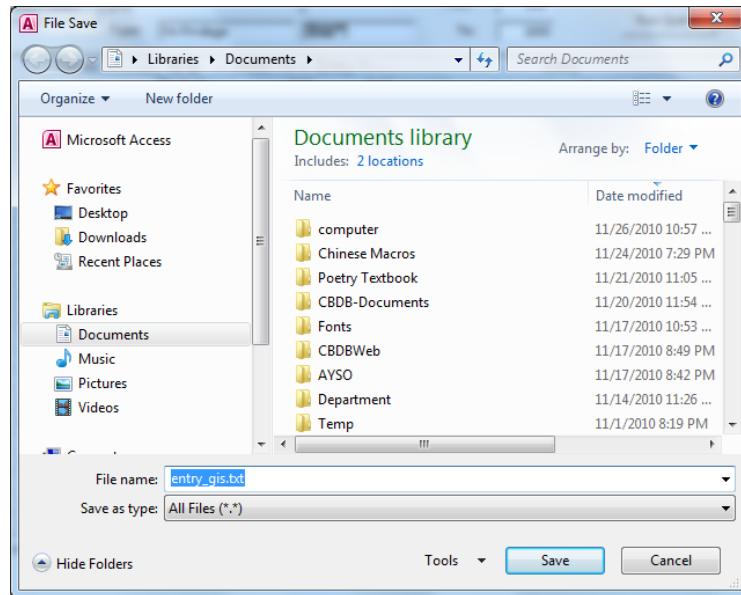
Name	姓名	Index Year	Entry Year	入仕法	From	地址	地址类别
Zhang Tingjian	張庭堅	950	0 yin privilege: general	恩蔭(龍統)	Chengdu	成都	籍貫(基本地址)
Fu Yanqing	符彥卿	957	0 yin privilege: general	恩蔭(龍統)	Wanqui	宛邱	籍貫(基本地址)
Qian Yi(3)	錢儀	962	0 yin privilege: general	恩蔭(龍統)	Qiantang	錢塘	籍貫(基本地址)
She Deyuan	折德源	965	0 yin privilege: general	恩蔭(龍統)	Guo Xian	崞縣	籍貫(基本地址)
Cui Song	崔頌	968	0 yin privilege: general	恩蔭(龍統)	Yanshi	偃師	籍貫(基本地址)
Li Jihe	李繼和	969	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Yang Wenyi	楊文逸	972	0 yin privilege: general	恩蔭(龍統)	Pucheng	浦城	籍貫(基本地址)
Feng Zan	馮瓛	973	0 yin privilege: general	恩蔭(龍統)	Licheng	庭城	籍貫(基本地址)
Kang Yanze	康延澤	974	939 yin privilege: general	恩蔭(龍統)	Luoyang	洛陽	籍貫(基本地址)
Lv Yuqing	呂餘慶	976	0 yin privilege: general	恩蔭(龍統)	Luoyang	洛陽	籍貫(基本地址)
Yi Yanting	易延慶	976	0 yin privilege: general	恩蔭(龍統)	Shanggao	上高	籍貫(基本地址)
Jia Yan	賈琰	977	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Hou Yun	侯蕡	977	0 yin privilege: general	恩蔭(龍統)	Taiyuan	太原	籍貫(基本地址)
Duan Sigong	段思恭	979	0 yin privilege: general	恩蔭(龍統)	Jincheng	晉城	籍貫(基本地址)
She Yuchong	折御沖	980	0 yin privilege: general	恩蔭(龍統)	Guo Xian	崞縣	籍貫(基本地址)
Fan Min	范曼	981	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Zhang Yi(5)	張裔	981	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Zhai Shousu	翟守素	981	936 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Du Yanchao	杜彥超	982	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Liao Zhongfu	廖仲符	982	0 yin privilege: general	恩蔭(龍統)	Jiangle	將樂	籍貫(基本地址)
Fan Conggui	范從龜	983	0 yin privilege: general	恩蔭(龍統)	Renshou	仁壽	籍貫(基本地址)
Cao Guangshi	曹光審	984	0 yin privilege: general	恩蔭(龍統)	Baizhang	百丈	籍貫(基本地址)

If one wishes to save the table, the simplest method is to **select** the entire table by clicking on the small box in the upper left-hand corner. Then save to the clipboard with **Ctrl-C**.

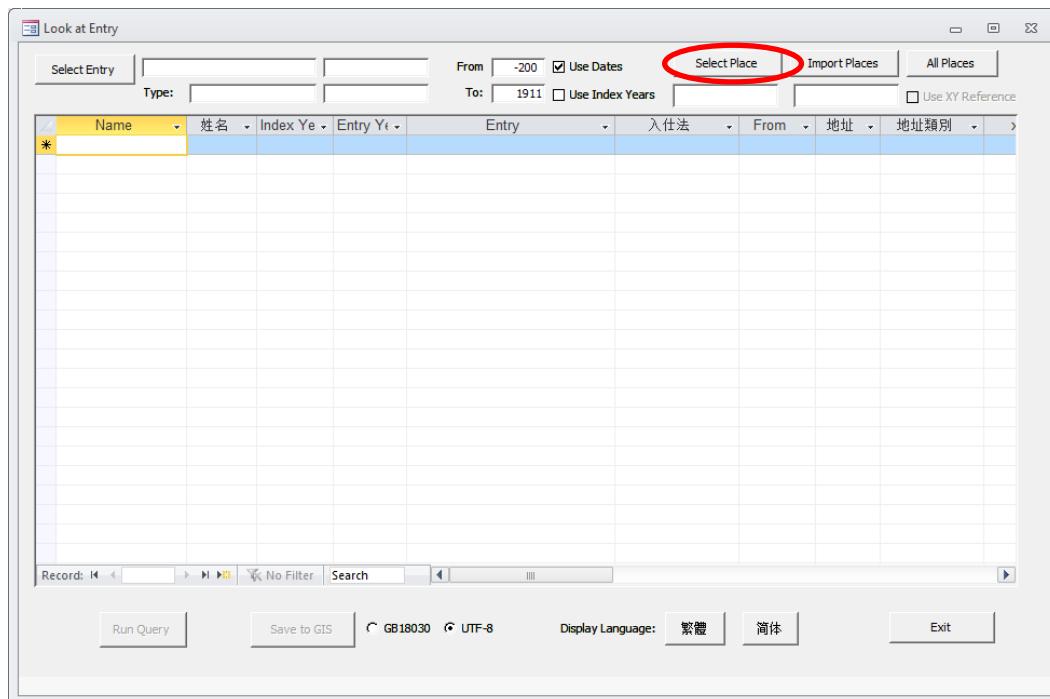


Name	姓名	Index Year	Entry Year	入仕法	From	地址	地址类别
Zhang Tingjian	張庭堅	950	0 yin privilege: general	恩蔭(龍統)	Chengdu	成都	籍貫(基本地址)
Fu Yanqing	符彥卿	957	0 yin privilege: general	恩蔭(龍統)	Wanqui	宛邱	籍貫(基本地址)
Qian Yi(3)	錢儀	962	0 yin privilege: general	恩蔭(龍統)	Qiantang	錢塘	籍貫(基本地址)
She Deyuan	折德源	965	0 yin privilege: general	恩蔭(龍統)	Guo Xian	崞縣	籍貫(基本地址)
Cui Song	崔頌	968	0 yin privilege: general	恩蔭(龍統)	Yanshi	偃師	籍貫(基本地址)
Li Jihe	李繼和	969	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Yang Wenyi	楊文逸	972	0 yin privilege: general	恩蔭(龍統)	Pucheng	浦城	籍貫(基本地址)
Feng Zan	馮瓛	973	0 yin privilege: general	恩蔭(龍統)	Licheng	庭城	籍貫(基本地址)
Kang Yanze	康延澤	974	939 yin privilege: general	恩蔭(龍統)	Luoyang	洛陽	籍貫(基本地址)
Lv Yuqing	呂餘慶	976	0 yin privilege: general	恩蔭(龍統)	Luoyang	洛陽	籍貫(基本地址)
Yi Yanting	易延慶	976	0 yin privilege: general	恩蔭(龍統)	Shanggao	上高	籍貫(基本地址)
Jia Yan	賈琰	977	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Hou Yun	侯蕡	977	0 yin privilege: general	恩蔭(龍統)	Taiyuan	太原	籍貫(基本地址)
Duan Sigong	段思恭	979	0 yin privilege: general	恩蔭(龍統)	Jincheng	晉城	籍貫(基本地址)
She Yuchong	折御沖	980	0 yin privilege: general	恩蔭(龍統)	Guo Xian	崞縣	籍貫(基本地址)
Fan Min	范曼	981	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Zhang Yi(5)	張裔	981	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Zhai Shousu	翟守素	981	936 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Du Yanchao	杜彥超	982	0 yin privilege: general	恩蔭(龍統)	Kaifeng	開封	籍貫(基本地址)
Liao Zhongfu	廖仲符	982	0 yin privilege: general	恩蔭(龍統)	Jiangle	將樂	籍貫(基本地址)
Fan Conggui	范從龜	983	0 yin privilege: general	恩蔭(龍統)	Renshou	仁壽	籍貫(基本地址)
Cao Guangshi	曹光審	984	0 yin privilege: general	恩蔭(龍統)	Baizhang	百丈	籍貫(基本地址)

One then can paste the table into any program that accepts the format. If one wishes to save the results to a file readable by ArcGIS, one selects the coding for the file and clicks on the **Save to GIS** button at the bottom of the form. The table is saved to a text file, by default, "entry\_gis.txt."



If one wishes to explore the mode of entry for people from a particular region, one uses the **Select Place** button in the upper right part of the form:



This opens the **Select Address** form. One can search for a place name using the search box:

Name	地名	Fir	Last Year	Belongs to	Belongs to	Belongs to	Belongs to	Address
Kaide Fu	開德府	1106	1109	京畿路	宋朝			11124
Kaide Fu	開德府	1110	1113	河北東路	宋朝			11171
Kaide Fu	開德府	1114	1119	京畿路	宋朝			11125
Kaide Fu	開德府	1120	1128	河北東路	宋朝			11572
<b>Kaifeng</b>	<b>開封</b>	<b>960</b>	<b>1052</b>	<b>開封府</b>	<b>宋朝</b>			<b>100658</b>
Kaifeng	開封	1949	2005	開封市	河南省	中華人民共和國		749
Kaifeng	開封	1127	1234	開封府	南京路	金朝		3079
Kaifeng	開封	960	1052	開封府	宋朝			10991
Kaifeng	開封	1053	1119	開封府	京畿路	宋朝		11028
Kaifeng	開封	1235	1387	汴梁路	河南行中書省	元朝		17447
Kaifeng	開封	712	907	汴州	河南道	唐朝		14748
Kaifeng	開封	1912	1949	河南省	中華民國			10036
Kaifeng	開封	1120	1126					10992
Kaifeng Bingbeidao	開封兵備道	1368	1643	河南諸道	明朝			303067
Kaifeng Fu	開封府	1127	1234	南京路	金朝			3078
Kaifeng Fu	開封府	1368	1643	河南布政司	明朝			5028
Kaifeng Fu	開封府	1430	1643	河南布政司	河南巡撫	明朝		5028
Kaifeng Fu	開封府	1053	1119	京畿路	宋朝			11027
Kaifeng Fu	開封府	1644	1911	河南省	清朝			8074
Kaifeng Fu	開封府	960	1052	宋朝				10990
Kaifeng Shi	開封市	1949	2005	河南省	中華人民共和國			169
Kaifeng Shixiaqu	開封市轄區	1949	2005	開封市	河南省	中華人民共和國		748
Kaiguang	開光	725	907	銀州	關內道	唐朝		14660

For example, if we search for the address “Kaifeng”, note that there are many addresses for Kaifeng. We will select the one active from 960 to 1052 by clicking on the corresponding row and clicking **Select** will return back to the main window, where we can run another query.

Name	姓名	Index Ye	Entry Ye	Entry	入仕法	From	地址	地址類別
Chen Zhizhong	陳執中	1049	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Qian Xie	錢勰	1093	1069	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Zhang Chengyi	張誠一	1085	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Jia Yan	賈琰	977	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Fan Min	范曼	981	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Xia Anqi	夏安期	1058	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Xia Song	夏竦	1044	1007	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Gao Zunhui	高遵惠	1097	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Li Zhaozhou	李昭肅	1059	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Li Zhaochu	李昭述	1050	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Lu Shimin	陸師閔	1099	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Lv Gongru	呂公孺	1090	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Lv Gongbi	呂公弼	1066	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Su Qi	蘇耆	1035	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Su Xie	蘇澥	1093	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Du Qi	杜杞	1050	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Wang Jizhong	王繼忠	1003	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Wang Jing	王漘	1081	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Wang Tian	王田	1050	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Wang Hao	王鴻	1039	1009	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Yang Chongxun	楊崇勳	1035	986	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)
Chen Zhigu	陳執古	1051	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貫(基本地址)

Using the single address for Kaifeng from 960 to 1052 produces 104 people. If one wishes to look at Kaifeng more broadly, return to the **Select Address** form and enter “Kaifeng” into the Filter text box and then click the **Filter** command button (on the right of the search box).

This gives a list of all places that begin with the word “Kaifeng.” (Using the Chinese, here 開封, is better to avoid the possibility of homonyms.) Then click the **Select ALL Filtered** command button:

Name	地名	Fir	Last Year	Belongs to	Belongs to 1	Belongs to 2	Belongs to 3	Belongs to 4	Address l
Kaifeng Shi	開封市	1949	2005	河南省	中華人民共和國				169
Kaifeng Shixiaqu	開封市轄區	1949	2005	開封市	河南省	中華人民共和國			748
Kaifeng	開封	1949	2005	開封市	河南省	中華人民共和國			749
Kaifeng Fu	開封府	1127	1234	南京路	金朝				3078
Kaifeng Fu	開封府	1127	1234	開封府	南京路	金朝			3079
Kaifeng Fu	開封府	1368	1643	河南布政司	明朝				5028
Kaifeng Fu	開封府	1430	1643	河南布政司	河南巡撫	明朝			5028
Kaifeng Fu	開封府	1644	1911	河南省	清朝				8074
Kaifeng	開封	1912	1949	河南省	中華民國				10036
Kaifeng Fu	開封府	960	1052	宋朝					10990
Kaifeng	開封	960	1052	開封府	宋朝				10991
Kaifeng	開封	1120	1126						10992
Kaifeng Fu	開封府	1053	1119	京畿路	宋朝				11027
Kaifeng	開封	1053	1119	開封府	京畿路	宋朝			11028
Kaifeng	開封	712	907	汴州	河南道	唐朝			14748
Kaifeng	開封	1235	1367	汴梁路	河南行中書省	元朝			17447
Kaifeng	開封	960	1052	開封府	宋朝				100658
Kaifeng Bingbeidao	開封兵備道	1368	1643	河南諸道	明朝				303067

This will return you to the main LookAtEntry form, with all the Kaifeng codes selected:

Select Entry		yin privilege: general	From	900	<input checked="" type="checkbox"/> Use Dates	Select Place	Import Places	All Places
Type:	N/A	To:	1100	<input checked="" type="checkbox"/> Use Index Year	<input type="button" value="[[Kaifeng]]"/>	<input type="button" value="[[Filter]]"/>	<input type="checkbox"/> Use XY Reference	
Name	姓名	Index Ye	Entry Ye	Entry	入仕法	From	地址	地址類別
Chen Zhijian	陳知儉	1080	0	yin privilege: general	恩蔭(籤統)	Guancheng	管城	籍貴(基本地址)
Chen Zhizhong	陳執中	1049	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Qian Xie	錢勰	1093	1069	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Zhang Chengyi	張誠一	1085	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Jia Yan	賈琰	977	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Fan Min	范曼	981	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Han Zhen	韓縝	1078	1042	yin privilege: general	恩蔭(籤統)	Yongqiu	雍邱	籍貴(基本地址)
Han Jiang	韓絳	1071	1042	yin privilege: general	恩蔭(籤統)	Yongqiu	雍邱	籍貴(基本地址)
Han Zong	韓宗	1053	1030	yin privilege: general	恩蔭(籤統)	Yongqiu	雍邱	籍貴(基本地址)
Han Zongshi	韓宗師	1098	0	yin privilege: general	恩蔭(籤統)	Yongqiu	雍邱	籍貴(基本地址)
Han Zongdao	韓宗道	1086	0	yin privilege: general	恩蔭(籤統)	Yongqiu	雍邱	籍貴(基本地址)
Han Zongda	韓宗達	1086	1059	yin privilege: general	恩蔭(籤統)	Yongqiu	雍邱	籍貴(基本地址)
Han Zongyan	韓宗彥	1080	0	yin privilege: general	恩蔭(籤統)	Yongqiu	雍邱	籍貴(基本地址)
Xia Anqi	夏安期	1080	1042	yin privilege: general	恩蔭(籤統)	Yongqiu	雍邱	籍貴(基本地址)
Xia Song	夏竦	1044	1007	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Gao Zunhui	高遵惠	1097	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Li Zhaoou	李昭肅	1059	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Li Zhaochu	李昭述	1050	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Liu Shidao	劉師道	1014	985	yin privilege: general	恩蔭(籤統)	Dongming	東明	籍貴(基本地址)
Lu Shimin	陸師閔	1099	0	yin privilege: general	恩蔭(籤統)	Kaifeng	開封	籍貴(基本地址)
Lv Jingchu	呂景初	1061	0	yin privilege: general	恩蔭(籤統)	Suanzao	酸棗	籍貴(基本地址)

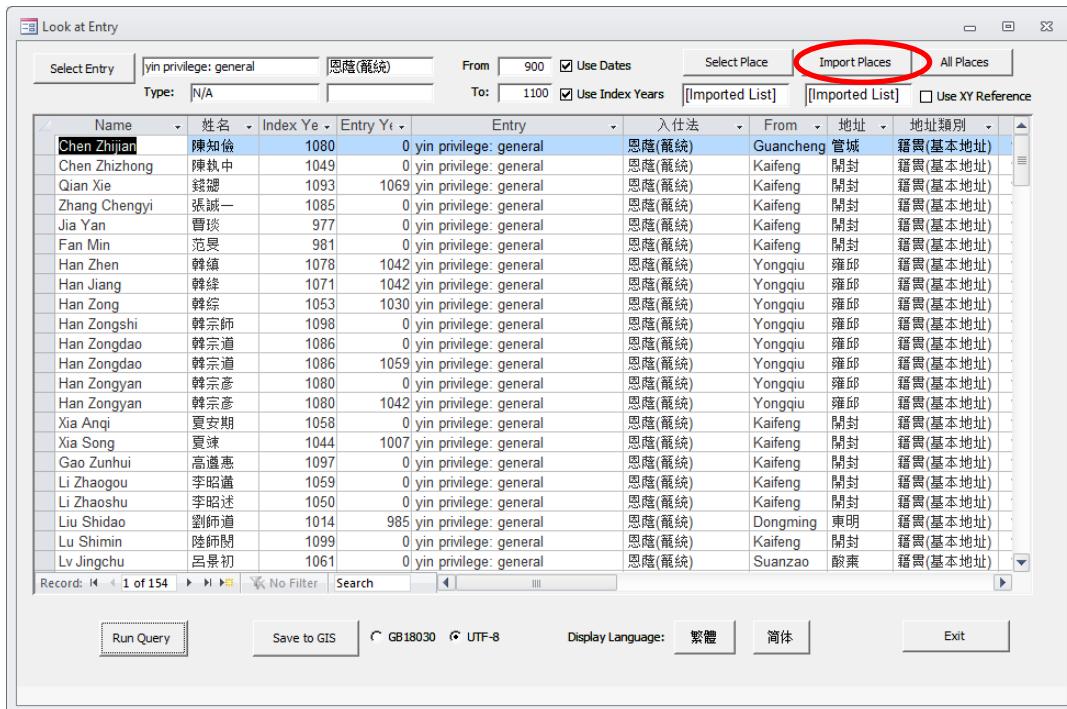
The **Place** text fields will show the filter term in a pair of square brackets, here “[[Kaifeng]].”

Running the query now produces 155 records for people from Kaifeng whose index years were between 900 and 1100 and who entered government service through the *yin* privilege.

If you wish to limit the number of codes for Kaifeng even further, you can select all the filtered records in the **Select Address** form, paste the records into a new text file, delete the records you do not want, and move the Address IDs to the first item on each remaining line. When CBDB reads the list, it looks for just those initial numbers and ignores everything else:



Now click on the Import Places button in the LookAtEntry form, select the file, and run the query. (CBDB gives a warning when it reads the list of IDs and finds an invalid ID.)



This approach produces the result one expected. Note that the table includes people from Yongqiu 雍邱 and Dongming 東明: these are administrative units subordinate to Kaifeng and thus automatically included in the search. (One can save the text files of address code to reuse them in other searches.)

There is one additional approach to searching by name that handles the problem of when a place name might change and thus be excluded from a list using names. During the Later Jin (936-947), for example, Kaifeng was called by its old name, Bianzhou. CBDB allows

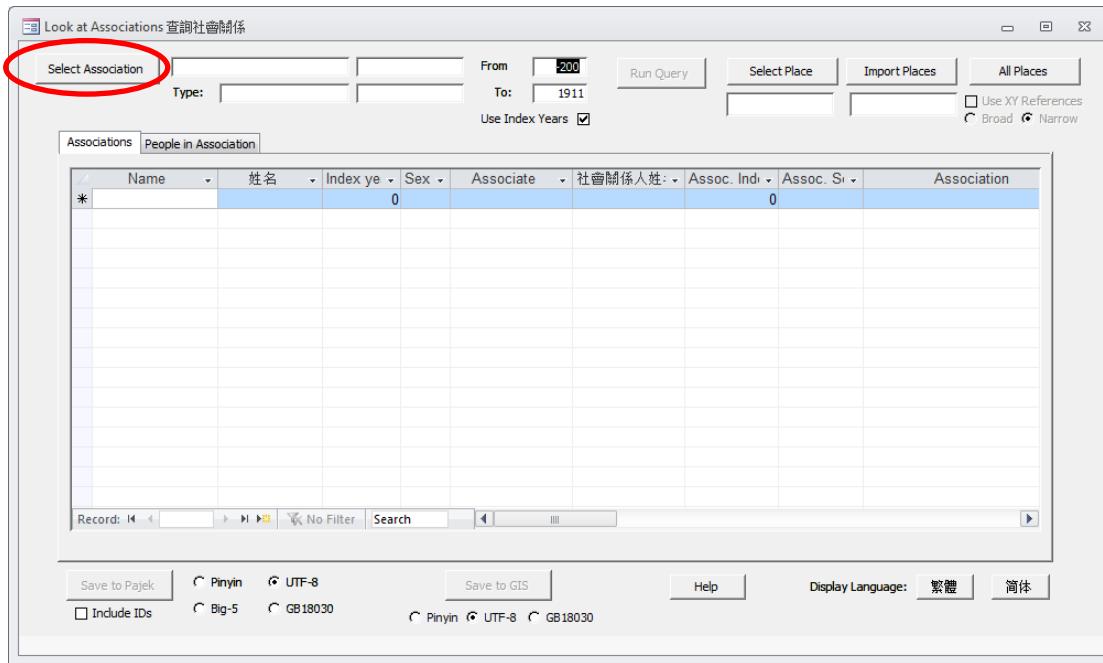
one to pick one address (or a filtered or imported list of addresses) and, based on its longitude and latitude, to find all the administrative units throughout the specified time period that were close to that unit. If one uses the set of addresses selected by the “Filter” feature and checks the “Use the XY Reference” check-box, one finds 171 people.

The screenshot shows the 'Look at Entry' window with the following details:

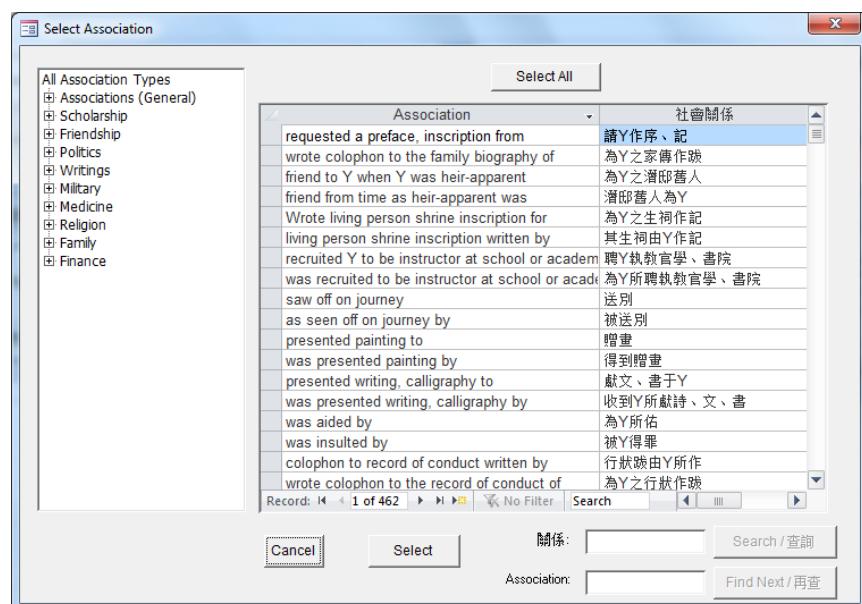
- Search Criteria:**
  - From: 900, To: 1100
  - Use Dates
  - Use Index Years
  - Use XY Reference (highlighted with a red circle)
  - [[KaiFeng]] (selected filter)
  - [[[Filter]]]
- Table Headers:** Name, 被姓, Index Ye, Entry Ye, Entry, 入仕法, From, 地址, 地址類別
- Data Rows:** A list of 171 entries, each containing a name, family name, index year, entry year, entry type, method of entry, reign period, city, and address category. For example, Chen Zhijian (陳知儉) has an index year of 1080, entry year of 0, entry type of 'yin privilege: general', and was admitted via '恩賜(籲統)' during the 'Guancheng' reign period in Kaifeng.
- Bottom Buttons:** Run Query, Save to GIS, Display Language (Traditional Chinese / Simplified Chinese), and Exit.

## B. Using the Form LookAtAssociations

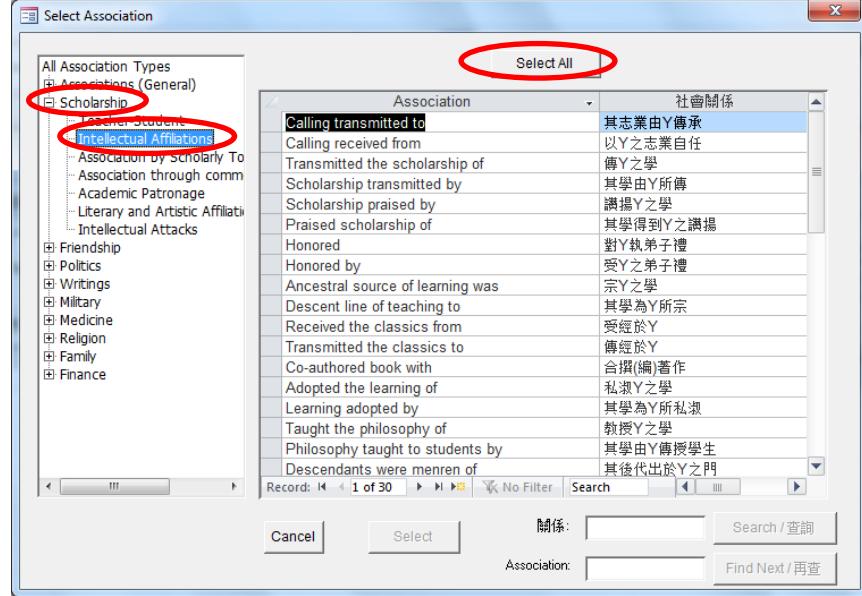
**LookAtAssociations** allows one to look at the people who have participated in particular associations or categories of associations. After opening the form, one clicks on “Select Association” to choose the type of association one wants to investigate.



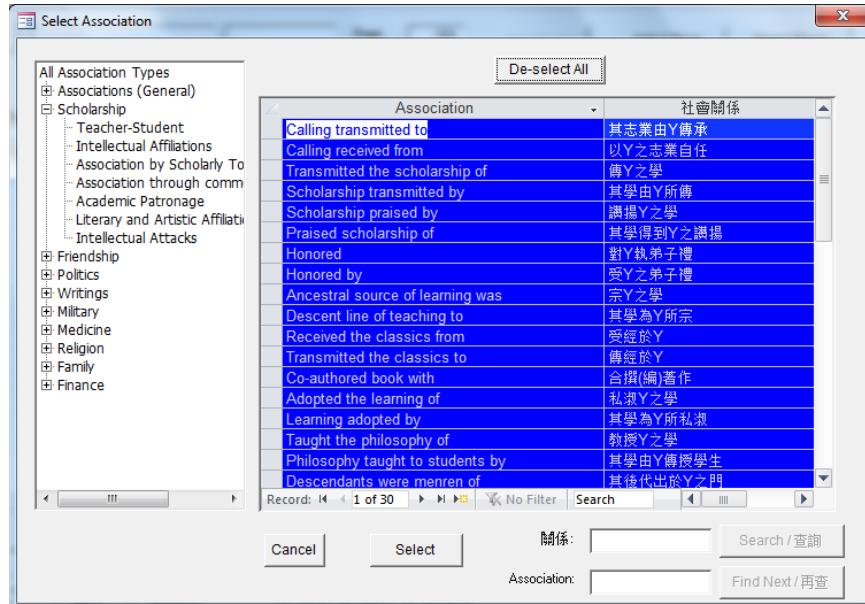
There are over four hundred categories of associations, so CBDB allows one to pick by type and subtype.



Consider the “Scholarship” associations from the list on the left. Under “Scholarship” there are seven subtypes. The subtype “intellectual affiliations” in turn has eight categories of associations:



One can select all of the subtypes simply by clicking on the “Select All” command button at the top and then clicking on the “Select” button at the bottom center:



Note that, as with entry, one can search for terms in the table of associations in both English and Chinese (using the search box at the bottom right corner) and search again if the first item found is not what you are looking for. In the screenshot below, I have selected *all intellectual affiliation associations* (1) as the subtype of association. I then chose the year between 800 and 1400 (2), and ran the query (3):

The screenshot shows the 'Look at Associations' window. At the top, there are input fields for 'Select Association' (set to '[All]'), 'Type' (set to 'Intellectual Affiliations / 學術交往'), 'From' (set to '800'), 'To' (set to '1400'), and a 'Run Query' button. A red circle highlights the 'From' and 'To' input boxes. Another red circle highlights the 'Run Query' button. Below the search bar is a table titled 'Associations' with columns: Name, 姓名, Index ye, Sex, Associate, 社會關係人姓, Assoc. Indi, Assoc. Si, and Association. The table lists numerous entries, such as Zha Yue associated with Feng Fang, with index years ranging from 1181 to 1059. The bottom of the window includes options for saving to Pajek, GIS, or CSV, and language settings for Pinyin, UTF-8, Big-5, and GB18030.

Although dates are a part of the ASSOC\_DATA table, we do not have date information for most associations, and **LookAtAssociations** uses the *index year* of the individuals to see whether they fall within the specified beginning and end dates.

Using the index year of people, however, introduces a significant limitation at the same time that it allows one to focus on specific timeframes: people for whom CBDB does not have an index year simply disappear from the results. This CBDB allows one to search without using the index years by unchecking the **Use Index Years** box directly below the input boxes for years:

This screenshot shows the same 'Look at Associations' window, but with the 'Use Index Years' checkbox (highlighted by a red circle) unchecked. As a result, the results table contains fewer entries, specifically 488 records instead of 330. The table structure is identical to the previous screenshot, listing associations between people like Li Zhong and Huang Tingjian, with their index years and descriptions.

Note that the results rise from 330 to 488. One can sort on index years after doing the search to look for the relevant associations.

The **Associations** table in **LookAtAssociations** has 19 columns to display the types of information recorded in the **ASSOC\_DATA** table:

Name (Pinyin)																			
Name (Chinese)																			
Index Year																			
Sex																			
Associated Person's Name (Pinyin)																			
Associated Person's Name (Chinese)																			
Associated Person's Index Year																			
Associated Person's Sex																			
Association Category (English)																			
Association Category (Chinese)																			
Association Count																			This gives the number of objects or events that established the association
Address (English)																			This is the index address, if known. Otherwise, CBDB selects the first record of address information on the person.
Address (Chinese)																			
X-coordinate																			These are the coordinates for the centroid for the address above.
Y-coordinate																			
Associate's Address (English)																			
Associate's Address (Chinese)																			
Associate's X-coordinate																			
Associate's Y-coordinate																			
Kinship Relation (English)																			The next four columns are for associations created through actions for the sake of a kin
Kinship Relation (Chinese)																			
Kin Name (pinyin)																			
Kin Name (Chinese)																			
Associate's Kinship Relation (English)																			The next four columns are for associations created through actions for the sake of the associate's kin
Associate's Kinship Relation (Chinese)																			
Associate's Kin Name (pinyin)																			
Associate's Kin Name (Chinese)																			
Distance																			If CBDB has the coordinates for the place identification for both people, it calculates the great-circle arc distance between them (in kilometers).

In addition to the table of associations, **LookAtAssociations** also provides a table listing all the people involved in the association one is investigating. One views this table simply by clicking on the **People in Association tab**. This table provides information about association with place.

The screenshot shows a software window titled "Look at Associations 檢查社會關係". The interface includes a toolbar with buttons for "Select Association", "Run Query", "Import Places", and "All Places". There are also fields for "From" (900), "To" (1200), and "Type" (Oppositional military as: 軍事對抗). A checkbox "Use Index Years" is checked. Below the toolbar is a tab bar with "Associations" (highlighted with a red circle) and "People in Association". The main area is a grid table with the following columns: Name, 姓名, Index Year, From, 翡貫, X, Y, Sex, and XY count. The data grid contains 24 rows of information. At the bottom of the grid are buttons for "Record", "Save to Pajek", "Save to GIS", and encoding options (Pinyin, UTF-8, Big-5, GB18030). There are also buttons for "繁體" and "簡體".

Name	姓名	Index Year	From	籍貫	X	Y	Sex	XY count
Chen Shiqing	陳世卿	1012	Sha Xian	沙縣	117.781792	26.39761	M	
Zhang Jun	張浚	1156	Hengyang	衡陽	112.597038	26.901617	M	
Fan Xiang	范祥	1060	Sanshui	三水	108.32904	35.190163	M	
Han Qi	韓琦	1067	Anyang	安陽	114.345497	36.098343	M	
Xue Liangpeng	薛良朋	1175	Rui'an	瑞安	120.630691	27.790512	M	
Hu Zonghui	胡宗回	1108	Wujin	武進	119.952301	31.782784	M	
Lei Youzhong	雷有終	1005	Heyang	郃陽	110.151016	35.242584	M	
Liu Ping	劉平	1023	Kaifeng	開封	114.34333	34.785477	M	
Lv Yihao	呂頤浩	1130	Linhai	臨海	121.120598	28.843132	M	
Mei Zhihi	梅軾禮	1127	Pujiang	浦江	119.883675	29.458708	M	
Ming Hao	明鎬	1048	Anqiu	安邱	119.1974	36.42905	M	
Sang Jingxun	桑景詢	1128	Gaoyou	高郵	119.437332	32.79071	M	
Tang Dongye	湯東野	1135	Danyang	丹陽	119.569855	31.995802	M	
Zeng Xiaoyun	曾孝蘊	1109	Jinjiang	晉江	118.589905	24.909641	M	
Zeng Wei	曾謂	1130	Jinjiang	晉江	118.589905	24.909641	M	
Wang Shiyong	王時雍	1130	Renshou	仁壽	104.143768	30.004446	M	
An Fu	安扶	1126	Kaifeng	開封	114.34333	34.785477	M	

This table has nine columns:

- Name (pinyin)
- Name (Chinese)
- Index Year
- Sex
- Place Association (pinyin)
- Place Association (Chinese)
- X-coordinate
- Y-coordinate
- XY-count

One can save the address information for display through a GIS program by clicking on the **Save to GIS**. Since association data provides an implicit social network linking the groups of people connected by the category of association being examined, one can save the network for analysis in the Pajek format by clicking on the **Save to Pajek** button. The Pajek format is one standard format for visualization in social network analysis (SNA), and many programs can read it and convert it to other formats. CBDB allows files for both GIS programs and for Pajek to be saved in different text encodings to enable the use of Chinese characters. Note that there is an option to include the Person ID with the node information in the Pajek files.

The default display for both nodes and edges uses color-coding to indicate degree of distance from the target person or people:

- White = the target nodes;
- Blue = nodes directly connected to them
- Green = node distance of 2
- Orange = node distance of 3
- Yellow = node distance of 4
- Red = node distance of 5 or more

Like the **LookAtEntry** form, **LookAtAssociations** allows one to look at associations for people from a particular place or from a particular list of places. However, as an experiment, **LookAtAssociations** has an additional option if one chooses to use the XY Reference feature. One choose either a narrow bounding box to define administrative units close to the units one has chosen, or one can choose a slightly larger box that may include additional units by clicking on the radio button labeled “Broad” under the “Use XY References” check box. In this case, CBDB uses the Kaifeng administrative unit in the Northern Song, and it turns out that the results are the same if one chooses either “Narrow” or “Broad.”

The screenshot shows the 'Look at Associations' query interface. At the top, there are several input fields: 'Select Association' (dropdown), 'Type' (dropdown set to 'Intellectual Affiliations / 學術交往'), 'From' (text box '800'), 'Run Query' (button), 'Select Place' (button), 'Import Places' (button), 'All Places' (button), 'To' (text box '1400'), 'Use Index Years' (checkbox), 'Kaifeng' (text box), '開封' (text box), and 'Use XY References' (checkbox checked). Below these are two tabs: 'Associations' (selected) and 'People in Association'. The main area is a grid table with columns: Name, 姓名, Index ye, Sex, Associate, 社會關係人姓, Assoc. Indi, Assoc. Si, and Association. The table lists various historical figures and their associations. At the bottom, there are buttons for 'Save to Pajek', 'Pinyin' (radio button), 'UTF-8' (radio button selected), 'Save to GIS', 'Help', 'Display Language' (dropdown set to '繁體'), '简体', 'Include IDs' (checkbox), 'Big-5' (radio button), 'GB18030' (radio button), and 'Pinyin' (radio button), 'UTF-8' (radio button selected), 'GB18030' (radio button).

Name	姓名	Index ye	Sex	Associate	社會關係人姓	Assoc. Indi	Assoc. Si	Association
Chen Zhizhong	陳執中	1049	M	Wang Zhi	王致	0	M	memorialized the writings of
Chen Zhizhong	陳執中	1049	M	Wang Zhi	王致	0	M	memorialized the writings of
Liu Ban	劉攽	1083	M	Sima Guang	司馬光	1078	M	Co-authored book with
Chao Yongzhi	晁諭之	1124	M	Shao Pu	邵溥	1148	M	Honored by
Du Hao	杜鷟	997	M	Xing Bing	邢昺	991	M	Co-authored book with
Yan Wenyi	閻文應	1006	M	Lv Yijian	呂夷簡	1038	M	Shared "same way" with
Chang Anmin	常安民	1102	M	Cai Shu	蔡叔	1145	M	Honored by
Zhang Guan	張觀	1044	M	Guo Zhen	郭稹		M	Co-authored book with
Li Fang	李昉	984	M	Song Bai	宋白	995	M	Co-authored book with
Li Fang	李昉	984	M	Hu Meng	扈蒙	974	M	Co-authored book with
Qian Jingchen	錢景谌	1080	M	Wang Anshi	王安石	1080	M	Honored
Gou Zhongzheng	句中正	988	M	Wu Xuan	吳鉉	989	M	Co-authored book with
Hu Meng	扈蒙	974	M	Li Mu	李穆	984	M	Co-authored book with
Guo Zhen	郭稹		M	Li Ruogu	李若谷	1029	M	Co-authored book with
Guo Zhen	郭稹		M	Song Xiang	宋庠	1056	M	Co-authored book with
*		0				0		

Record: 14 1 of 15 < > No Filter Search |

Save to Pajek  Pinyin  UTF-8 Save to GIS Help Display Language: 繁體 简体  
 Include IDs  Big-5  GB18030  
 Pinyin  UTF-8  GB18030

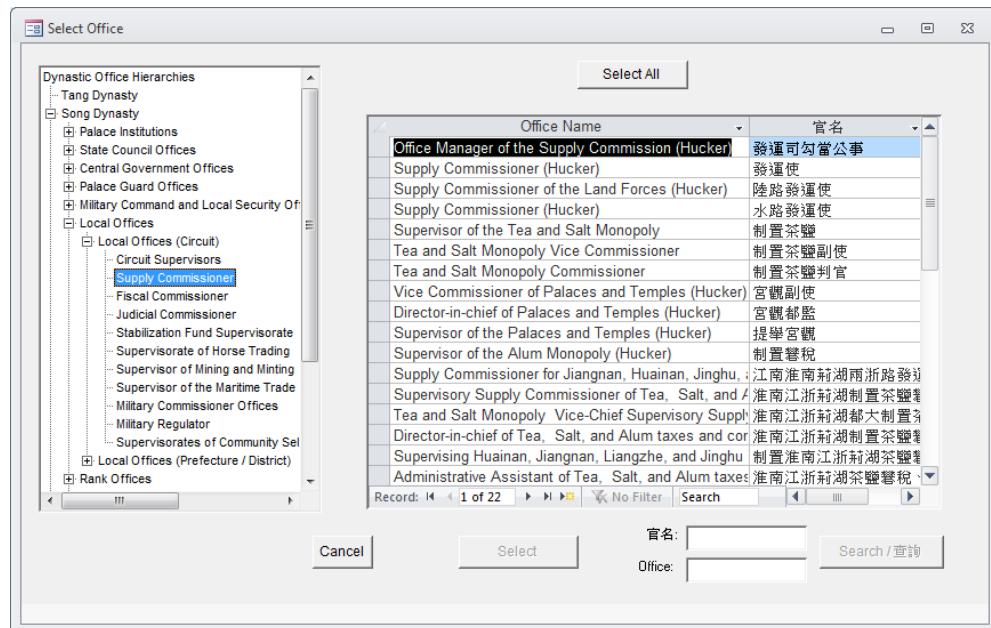
### C. Using the Form **LookAtOffices**

The bureaucratic system of imperial China was complex, and it evolved over time. As a result, CBDB at present has over six thousand office codes and will certainly have many more as the database extends its coverage to all of pre-modern China. Thus a central challenge in offering a useful approach to the examination of people's roles in office is how to aggregate the plethora of offices into larger units for analysis. **LookAtOffice** provides both hierarchical and functional groupings. When one opens **LookAtOffice**, it looks much like the other simple analytic forms. One clicks on the **Select Office** button on the top left to begin.

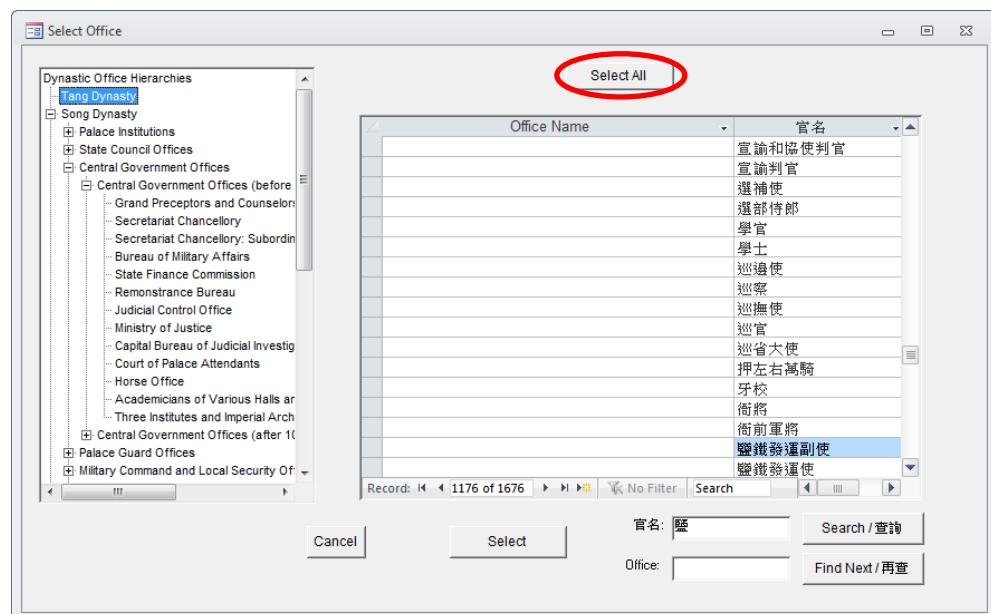
The screenshot shows the 'Look at Office Holding' application window. At the top, there is a search bar with fields for 'Select Office' (with a red circle around it), 'Type', and date ranges ('From' set to 200, 'To' set to 1911). Below the search bar are two tabs: 'Office Postings' and 'People in Office'. The main area is a grid table with columns labeled 'Name', '姓名', 'Index Ye', 'Sex', 'Addr Type', '地名類', 'Place (Person)', and '地名(人)'. The first row of the grid has an asterisk (\*) in the 'Name' column. At the bottom of the grid, there are navigation buttons for 'Record' (with page numbers 14, <, >, >>), a 'No Filter' button, and a 'Search' button. Below the grid, there are two sets of buttons: 'Save Offices to GIS' and 'Save People to GIS', each with options for 'GB18030' and 'UTF-8'. On the right side of the bottom bar, there are language selection buttons for '繁體' (Traditional Chinese) and '简体' (Simplified Chinese).

The “Select Office” form displays a tree of office categories. The first node, “Bureaucratic Structure” shows the organizational hierarchy organized by dynasty. One can view—and select—the offices at any level of structure. The first form shows all the offices associated with the Supply Commissioner at the circuit level.

One also can search for particular terms. Because there are not yet tree structures to explore bureaucratic organization of dynasties other than the Song, searching by the Chinese term (most records do not yet have English equivalents) is the best way to find particular offices in other dynasties. Searching for “salt” 鹽 (see the second form below) eventually leads one to Tang dynasty offices. (Because the Song offices were added first, they are the first offices found in the search routine.) One can determine which node on the tree is current by jiggling one of the scroll bars (not a great design, but it works):



Offices in the Supply Commissioner's Office in the Song Dynasty



A Tang office containing the character 鹽 ("salt")

If one looks for the word “Chancellery,” the *second* record brings one to the Vice Grand Councilor’s Office in the Song dynasty. One then can select all of the records for offices in the Vice Grand Councilor’s Office by clicking on “Select All:”

The form below shows a query selecting all postings to offices associated with the Vice Grand Councilor in CBDB, beginning in 200 B.C. and ending in 1911 and does not use index years. Because at present office codes are tied to dynasty, in fact all the results are from the Song:

The query generates two tabbed pages of results. The first, **Office Postings**, displays information about all the postings to the offices being examined. The second, **People in Office**, lists the people who were appointed to the offices. This list of people is particularly useful if one wishes to then import it into the **LookAtNetworks** form to explore the social

networks connecting the people who held a particular office. (One clicks on the square in the upper left corner to select all the records, copies them (Ctrl-c), and pastes them to a text file.

The screenshot shows a software window titled "Look at Office Holding". At the top, there are search fields for "Select Office" (set to "[All]"), "Type" (set to "Vice Grand Councilor Of 執政門"), and date filters "From -200" and "To 1911". Below these are two tabs: "Office Postings" (which is selected and highlighted with a red circle) and "People in Office". The main area is a grid table with 28 columns. The columns are: Name, 姓名, Index Year, Address, 地名, X, Y, Female, and Addr Type. The data grid contains approximately 361 rows of historical records. At the bottom of the window, there are buttons for "Save Offices to GIS", "Save People to GIS", and language options "繁體" and "简体". Encoding options "GB18030" and "UTF-8" are also present.

The table “Office Postings” has 28 fields:

- Person ID
- Person Name (pinyin)
- Person Name (Chinese)
- Index Year
- Sex
- Address ID
- Address (pinyin)
- Address (Chinese)
- X coordinate of Address
- Y coordinate of Address
- Posting ID
- Office ID
- Office (translation)
- Office (Chinese)
- First year of appointment
- Last year of appointment
- Dynasty (useful in cases where the years are very uncertain)
- Appointment type (regular, provisional, etc.)
- Appointment type (Chinese)
- Information on assumption of office (accepted, declined, etc.)
- Information on assumption of office (Chinese)
- Office Address ID
- Office Address (pinyin)
- Office Address (Chinese)

X coordinate of Office Address

Y coordinate of Office Address

XY count (number of postings) for the Office Address

Notes

The table “People in Office” contains the usual information about people:

Person ID

Person Name (pinyin)

Person Name (Chinese)

Index Year

Sex

Address ID

Address (pinyin)

Address (Chinese)

X coordinate of Address

Y coordinate of Address

XY count (number of people) for the Address

Because one might want to look at the spatial distribution of either the postings or the people who held the posts, the **LookAtOffice** form provides ways to save both to files that can be read by GIS software. One can specify either UTF-8 or GB18030 encoding at the bottom left of the form:

**Look at Office Holding**

Name	姓名	Index Ye	Sex	Addr Type	地名類	Place (Person)	地名(人)
An Tao	安肅	1102	False	Basic Affiliation	籍貫(基本地址)	Kaifeng	開封
Chen Zhizhong	陳執中	1049	False	Basic Affiliation	籍貫(基本地址)	Kaifeng	開封
Chen Kangbo	陳康伯	1156	False	Basic Affiliation	籍貫(基本地址)	Yiyang	弋陽
Chen Shengzhi	陳升之	1070	False	Basic Affiliation	籍貫(基本地址)	Dantu	丹徒
Chen Shu(2)	陳恕	1004	False	Basic Affiliation	籍貫(基本地址)	Kaifeng	開封
Chen Shu(2)	陳恕	1004	False	Basic Affiliation	籍貫(基本地址)	Kaifeng	開封
Chen Yaozuo	陳堯佐	1022	False	Basic Affiliation	籍貫(基本地址)	Guancheng	管城
Cheng Kan	程戡	1056	False	Basic Affiliation	籍貫(基本地址)	Yangdi	陽翟
Cheng Kejun	程克俊	1148	False	Basic Affiliation	籍貫(基本地址)	Fuliang	浮梁
Cheng Kejun	程克俊	1148	False	Basic Affiliation	籍貫(基本地址)	Fuliang	浮梁
Cheng Lin	程琳	1047	False	Basic Affiliation	籍貫(基本地址)	Luoyang	洛陽
Cheng Lin	程琳	1047	False	Basic Affiliation	籍貫(基本地址)	Luoyang	洛陽
Qian Duanli	錢端禮	1168	False	Actual Residence	落籍(實際居住)	Linhai	臨海
Quan Bangyan	權邦彥	1133	False	Basic Affiliation	籍貫(基本地址)	Wuyuan	婺源
Zhan Dafang	詹大方	1148	False	Basic Affiliation	籍貫(基本地址)	Sui'an	遂安
Zhang Que	張憲	1128	False	Basic Affiliation	籍貫(基本地址)	Leshou	樂壽
Zhang Que	張憲	1128	False	Basic Affiliation	籍貫(基本地址)	Leshou	樂壽
Zhang Que	張憲	1128	False	Basic Affiliation	籍貫(基本地址)	Leshou	樂壽
Zhang Zhibo	張知白	1028	False	Basic Affiliation	籍貫(基本地址)	Qinonchi	清池

Record: 1 of 463 | No Filter | Search | 繁體 | 簡體

**Save Offices to GIS**      **Save People to GIS**

GB18030     UTF-8

GB18030     UTF-8

## D. Using the Form LookAtKinship

Queries involving kinship are more complex than queries examining categories of association or modes of attaining eligibility for office. Since the information on kinship for an individual usually contains just a few records, CBDB begins with those records and then looks at the kinship information available for all the kin listed for the initial person. CBDB repeats this search for the kin, the kin of the kin, the kin of the kin of the kin, and so on, until specified criteria are met. First is simply a limit to the number of search iterations to allow. Usually 5000 loops are adequate. Second are limits on the distance of the kinship relations being examined:

**Max. Ancestor Gen.** specifies how many generations of ancestors to include. One's father's generation is 1; the grandfather is 2, great-grandfather 3, and so on.

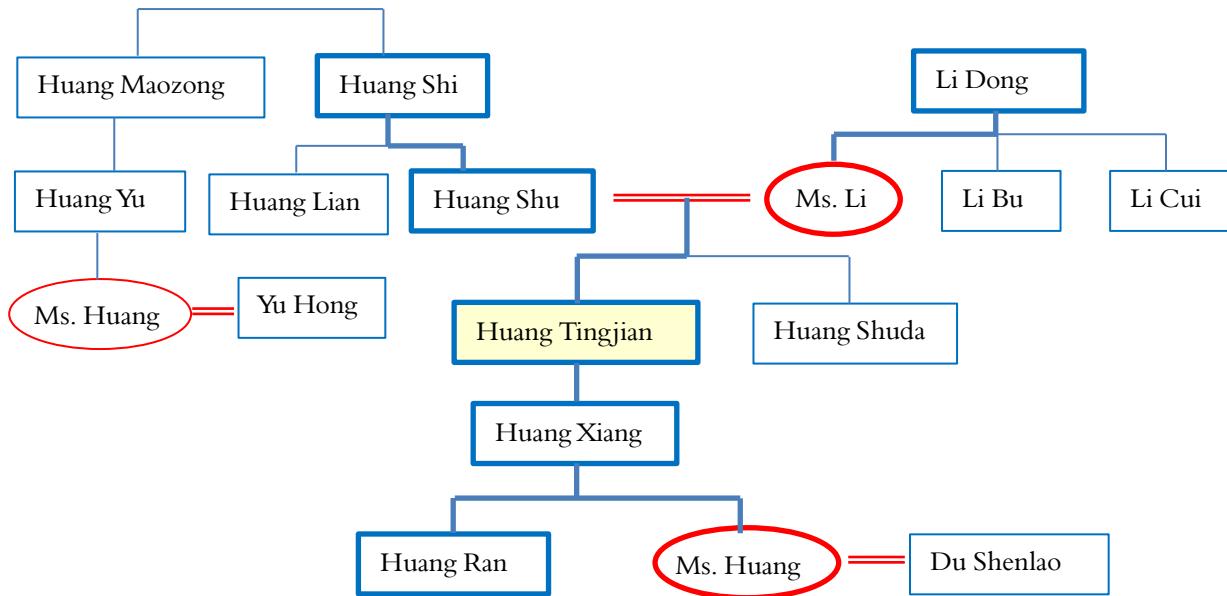
**Max. Descend. Gen.** specifies how many generations of descendants to include.

One's children's generation is 1, grandchildren 2, great-grandchildren 3, and so on.

**Max. Collateral Kin** limits how many horizontal moves are allowed. For example, one's wife's sister has one unit of "marriage" distance and one unit of "collateral" distance. One's wife's sister's husband's brother has two units of "marriage" distance and two units of "collateral" distance.

**Max. Marriage Dist.** limits how many links defined by marriage are allowed in the search. One's wife's sister's husband has two units of "marriage" distance.

To visualize these distinctions, consider the partial kinship network:



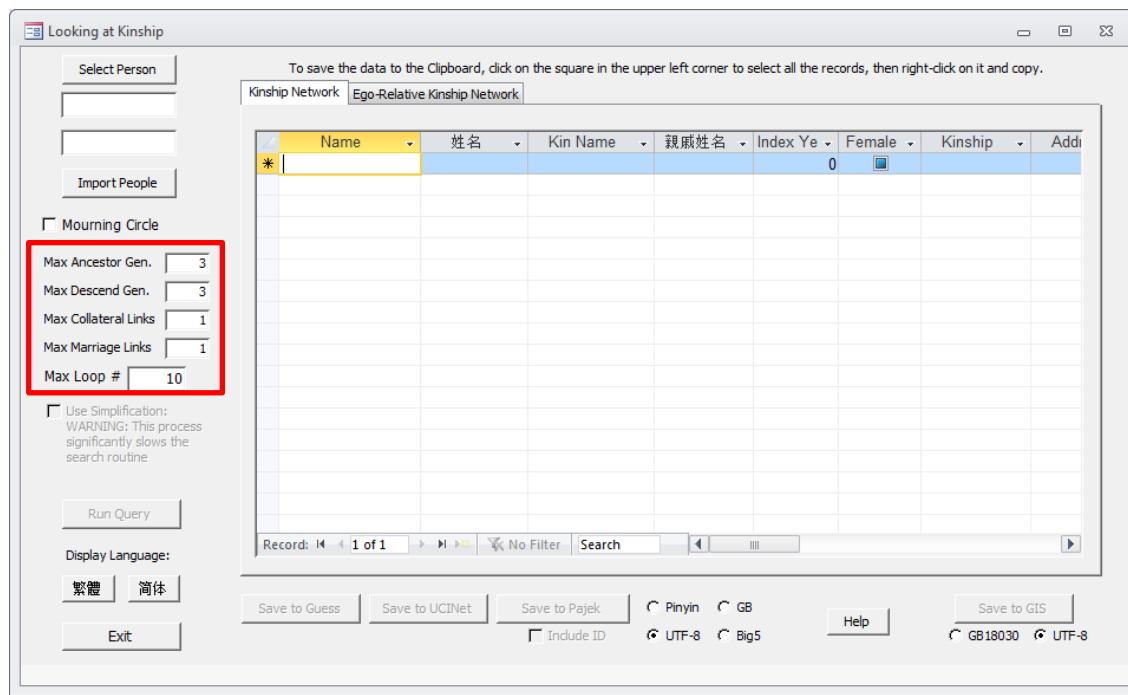
For Huang Tingjian, the squares and ovals with thick lines show direct lineal descent (fathers and mothers, sons and daughters). The double lines are marriage links. All other single lines

mark collateral relations. In the measurement system used in LookAtKinship:

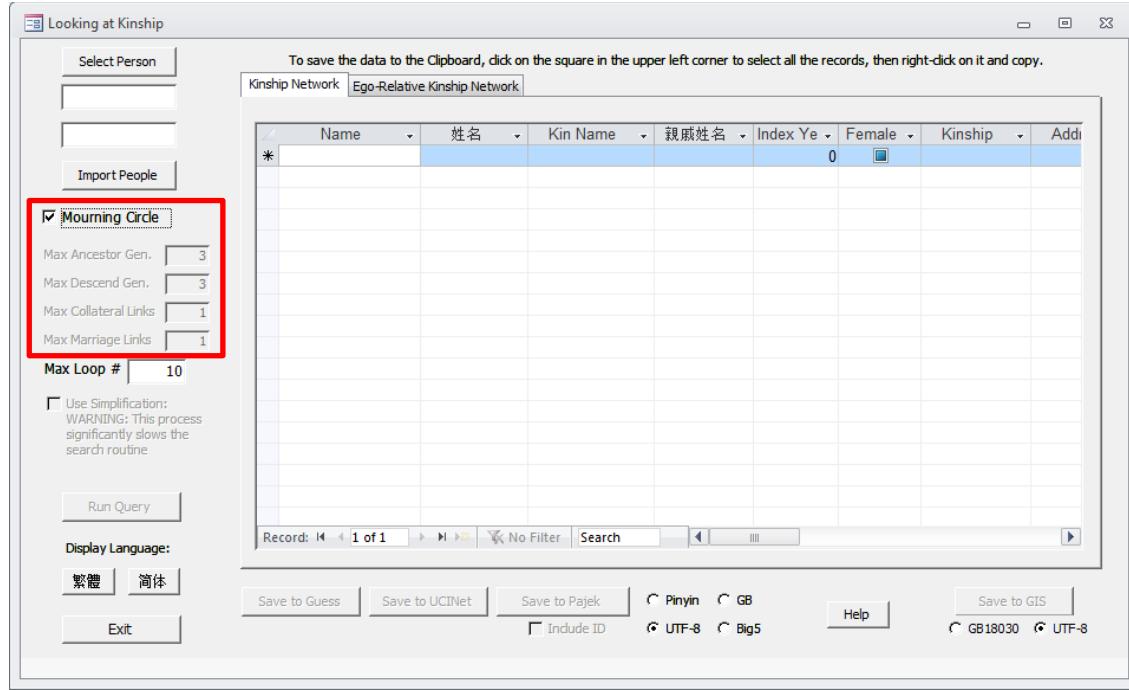
Huang Yu 黃育	is FFBS	(Up = 2, Down = 1, Collateral = 1)
Yu Hong 余宏	is FFBSDH	(Up = 2, Down = 2, Collateral = 1, Marr. = 1)
Li Cui 李萃	is MB	(Up = 1, Collateral = 1, Marr. = 1)
Du Shenlao 杜莘老	is SDH	(Down = 2, Marr. = 1)

Because LookAtNetwork keeps looking through a very large table of kinship relations until the distance limits are reached, the kinship table produced by the search can grow very large. Therefore please note:

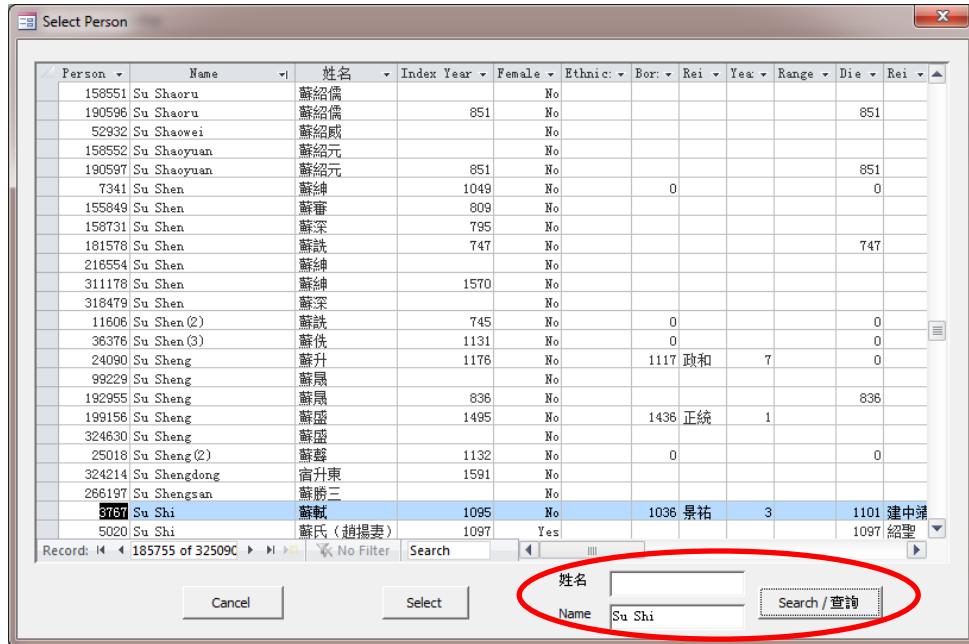
**WARNING: searching for extended degrees of collateral and marriage distance may result in a very large dataset**



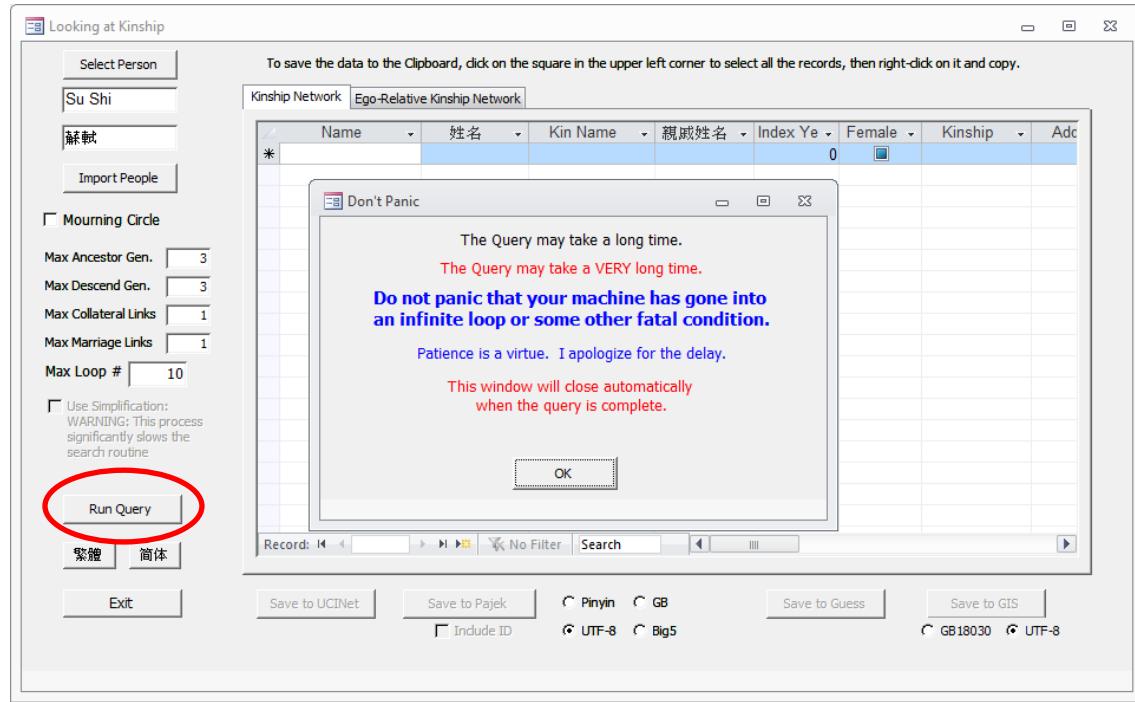
Another standard concern in Chinese kinship studies is to examine the so-called “mourning circle” defined by five degrees of kinship relation. LookAtKinship allows one to simply click on the “Mourning Circle” check-box to reconstruct what is known in the database about kin who are part of an individual’s mourning circle. When one selects “Mourning circle,” however, the four limit parameters are preset and therefore deactivated.



To examine kinship relations, one first clicks on the “Select Person” command button on the top left corner, which will open a form with a list of all the people in the database. One can search for a person using either Chinese characters or *pinyin*. (Sometimes, for a second search, the form finds an unrelated person: just click **Search** again to reset the results.)



Once one has selected the person, one sets the search limits (or chooses the Mourning Circle) and clicks the **Run Query** command button to start the search. Because the search can take a long time, CBDB opens a window to encourage patience.



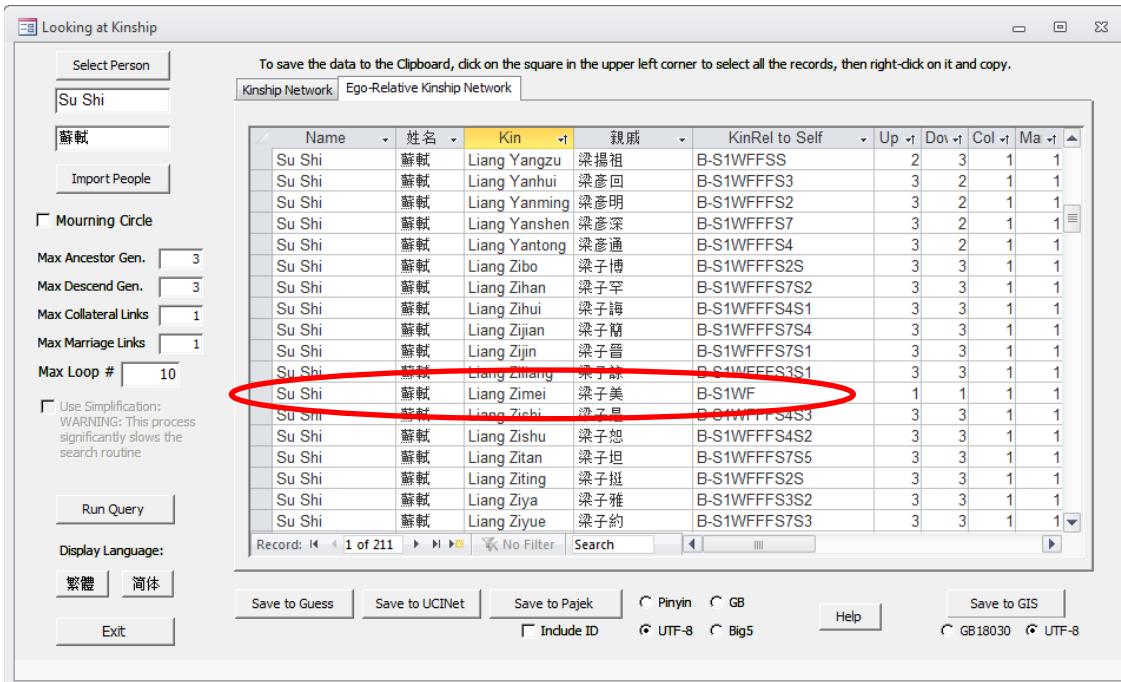
When the search finishes, there are two tables one can examine. The first, **Kinship Network**, lists all the kinship relations discovered through the search:

Name	姓名	Kin Name	親戚姓名	Index Ye	Female	Kinship	A
Li Xiangyu	李相如	Chao Buzhi	晁浦之	1159	<input type="checkbox"/>	MB	Juyi
Zhang Mian	張沔	Zhang Shen	張誠	1042	<input type="checkbox"/>	S2	Wu
Zhang Yuanbi	張元弼	Zhang Mian	張沔	1106	<input type="checkbox"/>	FF	Wu
Huang Shi	黃璽	Zhang Dun	章惇	1099	<input type="checkbox"/>	MB	Che
Zhang Dun	章惇	Zhang Jie	章傑	1094	<input type="checkbox"/>	SS	Puc
Zhang Dun	章惇	Zhang Jin	章僅	1094	<input type="checkbox"/>	SS	Puc
Han Zongshu	韓宗恕	Han Zhen	韓鎮	1098	<input type="checkbox"/>	F	Yon
Huang Shi	黃璽	Huang Haoqian	黃好謙	1099	<input type="checkbox"/>	F	Che
Huang Zai	黃宰	Huang Ziyou	黃子游	1117	<input type="checkbox"/>	S	Wai
Liang Yanchang	梁彥昌	Liang Shi	梁適	1068	<input type="checkbox"/>	F	Xuc
Liang Yanhui	梁彥回	Liang Zilang	梁子諒	1066	<input type="checkbox"/>	S1	Xuc
Su Chi	蘇遲	Liang Zimei	梁子美	1135	<input type="checkbox"/>	WF	Jinh
Liang Yanchang	梁彥昌	Liang Yangzu	梁揚祖	1068	<input type="checkbox"/>	SS	Xuc
Ouyang Xiu	歐陽修	Ouyang Xingshi	歐陽興世	1068	<input type="checkbox"/>	SS	Xinz
Su Dai	蘇迨	Ouyang Xiu	歐陽修	1110	<input type="checkbox"/>	WFF	Cha
Pu Shidao	蒲師道	Pu Zongmeng	蒲宗孟	1053	<input type="checkbox"/>	S	Nan
Su Zhe	蘇軾	Su Chi	蘇子瞻	1098	<input type="checkbox"/>	S1	Mei
Su Shi	蘇軾	Su Zhe	蘇軾	1095	<input type="checkbox"/>	B-	Mei

This table has nineteen columns:

Name (pinyin)  
 Name (Chinese)  
 Kin Name (pinyin)  
 Kin Name (Chinese)  
 Index Year of Kin  
 Sex of Kin  
 Kinship Relation  
 Address of Person (pinyin)  
 Address of Person (Chinese)  
 Address Type  
 X-Coordinate  
 Y-Coordinate  
 Address of Kin (pinyin)  
 Address of Kin (Chinese)  
 X-Coordinate of Kin Address  
 Y-Coordinate of Kin Address  
 Address Type of Kin Address  
 Notes  
 Distance (great-circle arc distance in kilometers between the addresses)

The second table, **Ego-Relative Kinship**, describes the kinship relation between each person in the first table and the person selected at the very beginning:



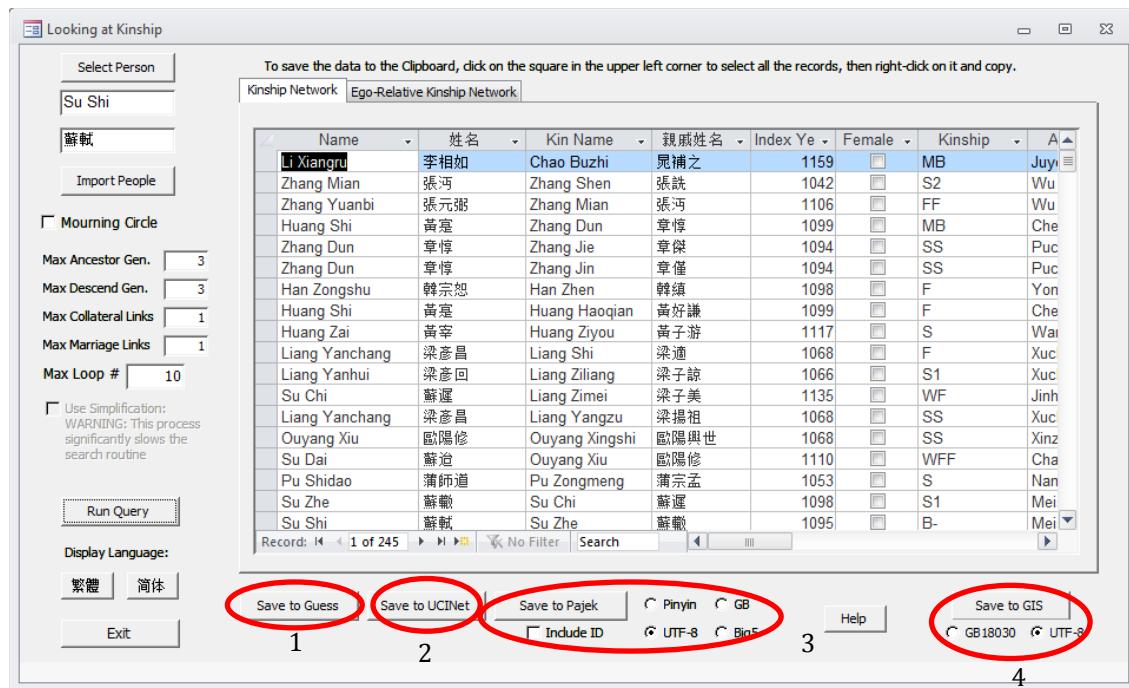
The screenshot shows a software window titled "Looking at Kinship". On the left, there is a sidebar with options like "Select Person" (set to "Su Shi"), "Import People", and various filtering and search parameters. The main area displays a table titled "Kinship Network" with the following columns: Name, 姓名 (Name), Kin, 親戚 (Relationship), KinRel to Self, Up, Dow, Col, and Ma. The table lists numerous entries for Su Shi and other family members, such as Liang Yangzu, Liang Yanhui, Liang Yanming, Liang Yanshen, Liang Yantong, Liang Zibo, Liang Zihan, Liang Zihui, Liang Zijian, Liang Zijin, Liang Zilang, Liang Zimei, Liang Zhibi, Liang Zishu, Liang Zitan, Liang Ziting, Liang Ziya, and Liang Ziyue. A red oval highlights the entry for Liang Zimei (梁子美) in the "Kin" column. The bottom of the window includes buttons for "Save to Guess", "Save to UCI Net", "Save to Pajek", "Pinyin", "GB", "Include ID", "UTF-8", "Big5", "Help", "Save to GIS", "GB18030", "UTF-8", and "Exit".

For example, Liang Zimei 梁子美 is Su Shi's younger brother's son's wife's father. The path one traverses to reach Liang Zimei requires one to go over one collateral link (B-), down one generational link (S1), over one marriage link (W), and finally up one generation (F). Thus the kinship distance metrics are {1,1,1,1}. (The “Ego-Relative Kinship” table has an additional

column that gives a raw path that shows how CBDB simplified the kinship relations, but since this task proves to beyond any simple algorithm, CBDB has disabled the function.)

As is true for all the other tables in all the other forms, if one clicks on the upper left corner of either table in this form, one can select all the records in the table, which then can be cut and pasted into other programs. Also, right-clicking on any of the column headings allows one to sort on that column.

Finally, one can export the kinship data to four different types of files. The first three are different formats of Social Network Analysis (SNA) files: **GUESS** (1), **UCINet** (2), and **Pajek** (3) with various character code options and the option to include ID in the labels. The fourth is for GIS visualization: the program can save the file for **ArcGIS** (4) with two different code options.



The default display for both nodes and edges in the SNA output files uses color-coding to indicate degree of distance from the target person:

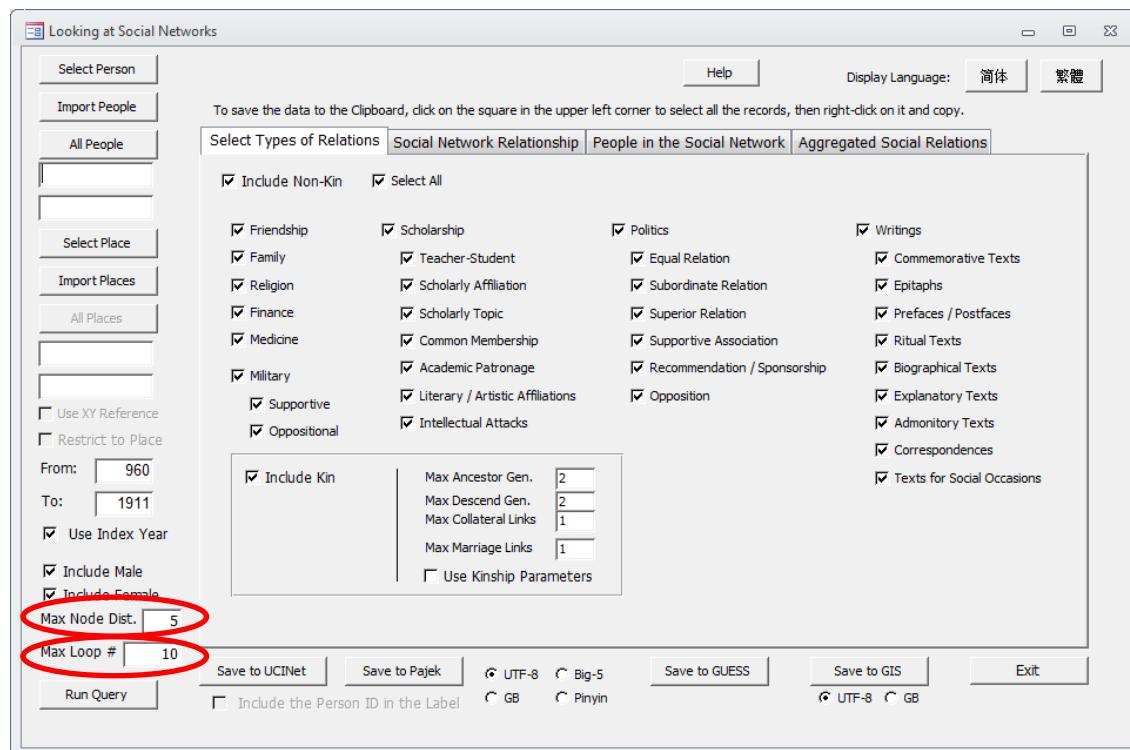
- White = the target node;
- Blue = nodes at a summed kinship distance of 1
- Green = nodes at a summed kinship distance of 2
- Orange = nodes at a summed kinship distance of 3
- Yellow = nodes at a summed kinship distance of 4
- Red = nodes at a summed kinship distance of 5 or more

## E. Using the Form LookAtNetworks

**LookAtNetworks** is the most powerful querying interface developed for the Access version of CBDB. It allows the user to explore social networks defined both by kinship ties and by other forms of social relations. It allows the user to select a person or to import a list of people produced by other queries. It similarly allows the user to start with a place or to import a list of places. **LookAtNetworks** allows the user to select the particular forms of social association to investigate, and it allows one to set the range of years to consider. As in **LookAtKinship**, the queries in **LookAtNetwork** are iterative: the query produces an initial group of people and then looks at the relevant connections between these people and others in the database. Each cycle adds more people, whose associations then produce yet more people.

**LookAtKinship** has five metrics to limit the search, but **LookAtNetworks** has just two: a *maximum loop count* (how many times the query iterates through the list of people), and a *maximum node distance*. This distance is the number of links between a person in the network and members of the group of people identified by the first step in the search process. If the user selects a particular person, then all distances are measured from that person. If one starts with a list of people, then all the people on that list serve as starting points. If one starts with a place or list of places, then the people initially identified as associated with that place or those places serve as the starting points.

**WARNING: Higher node distances take time and may result in a very large dataset**



NOTE: A query set at a node distance of 1 will result in a) all the people associated with the selected person(s) and b) all the associations between the people in the network. This particular network is called an ego network. One can sort these relationships in the query

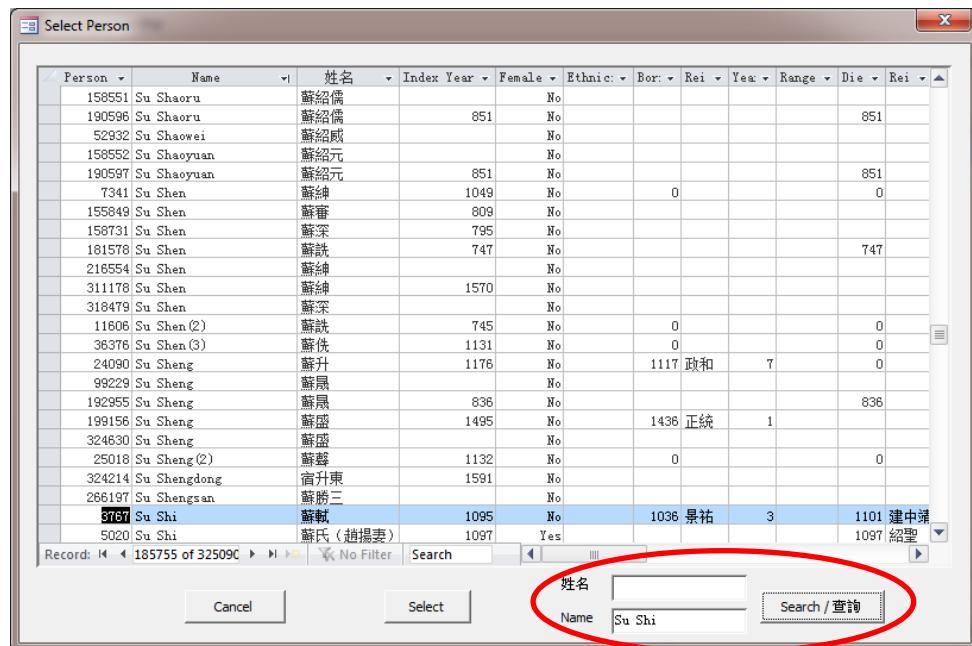
results table, and one can delete any records one does not wish to export for further analysis.

Running a query begins with selecting the elements to investigate.

## 1. Begin with People

### A. Select a Person

If one wants to look at the social networks which link a particular person to others, one can click on the **Select Person** command button to open a list of people:



The screenshot shows a Windows-style dialog box titled "Select Person". It contains a large grid table with columns for Person ID, Name, 姓名 (Name), Index Year, Female, Ethnic, Bor., Rei., Year, Range, Die., and Rei. The grid lists numerous entries for Su Shi, with the entry "5767 Su Shi 蘇軾" highlighted. At the bottom of the grid, there is a search bar with a "姓名" label, a text input field containing "Su Shi", and a "Search / 檢索" button. The entire search input area is circled in red.

Since the list contains all the people in the database, one can search for a particular person using the search box in the lower right corner.

### B. Import a List of People

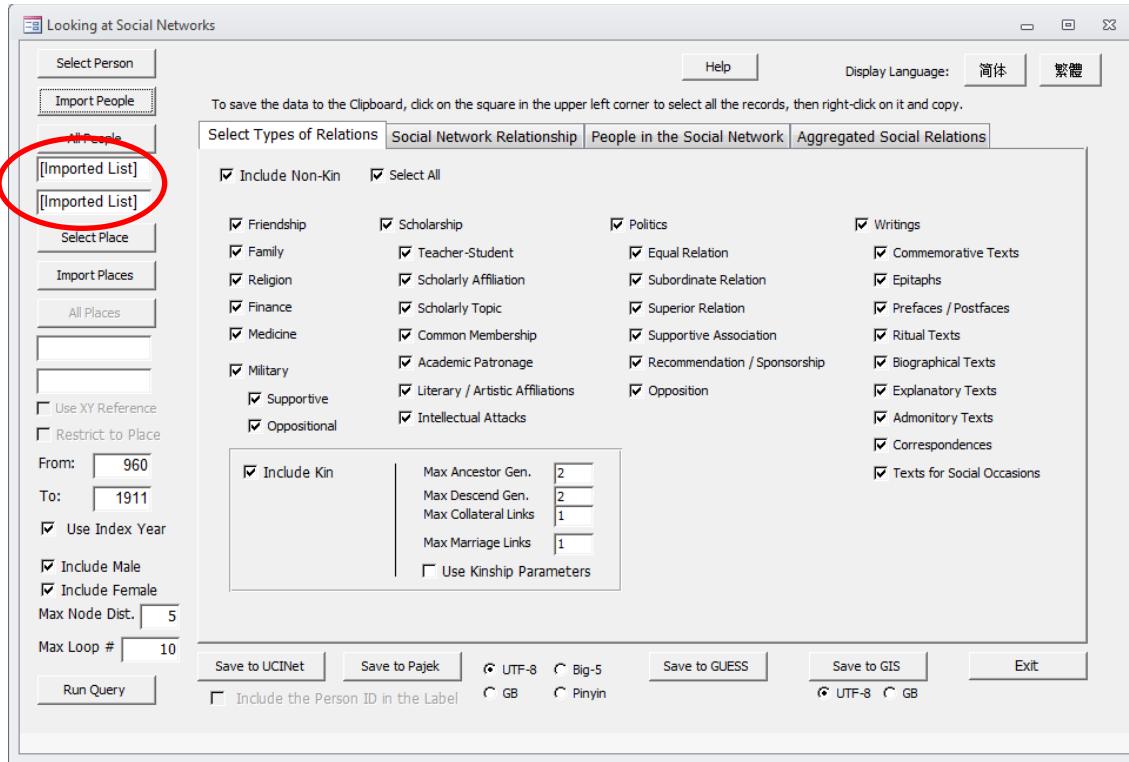
A second, very useful way to consider social networks is to import a group of people sharing common characteristics identified by other queries. For example, one could start with people in the Song dynasty who became eligible for office through a legal examination. One copies the results of the **LookAtEntry** query to a **text** file (a .txt file can be created in Word or Excel) and edits the results:



The screenshot shows a Windows Notepad window titled "LawExam.txt - Notepad". The content is a tab-separated text file with columns for Person ID, Name, and various numerical values. The row with Person ID 1684 is circled in red. The data includes names like Li Chaoyin, Yang Zhongchen, Zhao Fan, Cui Taifu, Wang Guo, etc., along with their corresponding IDs and values.

Person ID	Name	姓	名	年	性別	年	年	年	年	年	年
31463	Li chaoyin	李	朝	730							
22514	Yang Zhongchen	楊	仲	1090							
25877	Zhao Fan	趙	范	1194							
1684	Cui Taifu	崔	台	1086							
35453	Wang Guo	王	果								
44296	Zhu weiyue	祝	維								
39531	Ximen Chengyun	西	門	成	允	1019					
45356	Xu Zun	許	遵			1068					
11580	Chen Gui	陳	規			1131					
20149	Lin Yan	林	炎			1253					
13312	Zha Tao	查	陶			996					
3959	Wang Yi	王	衣			1133					
773	xu Ziyan	徐	子			1189					

Note that one can include a line of labels as well as other textual information: the program will simply ignore this line. One can use tabs or commas to delimit fields. The **one crucial datum** that CBDB extracts from each line is the Person ID which *must be the very first item of the line*. After one clicks on the **Import People** command button, selects the file, and **LookAtNetworks** successfully reads the file, the form will look like:



The two boxes that give the person's name in pinyin and Chinese will state “[Imported List].”

## 2. Begin with Place

### A. Select a Place

When one clicks on the **Select Place** command button, one opens a form to allow one to select a particular place. Since there are at present over 21,000 place names, the form provides a search utility to find a name either in Chinese or *pinyin*. Since the same administrative unit has separate records as it changes its boundaries over time, the table of place names provides additional information to help the user identify which ID to use in the search. If one selects an administrative unit larger than a county, CBDB automatically identifies the lower-level units (and their IDs) associated with the selected place. If one later specifies a time period for the search that includes years before or after the years during which the selected place existed, CBDB identifies the earlier or later units that shared the same PLACE regions (identified through the ADDR\_PLACE table).

Name	地名	Fir	Last Year	Belongs to	Belongs to I	Belongs to	Belongs to	Belongs to	Address J
Kaibian	開邊	960	966	戎州					13528
Kai cheng	開城	1368	1643	平涼府	陝西布政司	明朝			5254
Kai cheng	開城	1427	1643	平涼府	陝西布政司	陝西巡撫	明朝		5254
Kai cheng	開成	1280	1367	開成州	陝西行中書省	元朝			17688
Kai cheng Zhou	開成州	1280	1367	陝西行中書省	元朝				17687
Kai de Fu	開德府	1110	1119	京畿路	宋朝				101003
Kai de Fu	開德府	1110	1126	河北東路	宋朝				101003
Kaifeng	開封	1053	1119	開封府	京畿路	宋朝			100658
Kaifeng	開封	1053	1119	開封府	宋朝				100658
Kaifeng	開封	1194	2005	開封市	河南省	中華人民共和國			749
Kaifeng	開封	1127	1234	開封府	南京路	金朝			3079
Kaifeng	開封	1912	1949	河南省	中華民國				10036
Kaifeng	開封	1235	1367	汴梁路	河南行中書省	元朝			17447
Kaifeng	開封	712	907	汴州	河南道	唐朝			14748
Kaifeng Bingbeidao	開封兵備道	1368	1643	河南諸道	明朝				303067
Kaifeng Fu	開封府	1430	1643	河南布政司	河南巡撫	明朝			5028
Kaifeng Fu	開封府	1368	1643	河南布政司	明朝				5028
Kaifeng Fu	開封府	1127	1234	南京路	金朝				3078
Kaifeng Fu	開封府	1053	1119	宋朝					11027
Kaifeng Fu	開封府	1053	1119	京畿路	宋朝				11027
Kaifeng Fu	開封府	1844	1911	河南省	清朝				8074
Kaifeng Shi	開封市	1949	2005	河南省	中華人民共和國				169
Kaifeng Shixiaqu	開封市轄區	1949	2005	開封市	河南省	中華人民共和國			748

Record: 14 < 9597 of 25469 > No Filter Search

Cancel Select Place Name: Kaifeng Search / 搜尋

地名: [ ] 地名: [ ] Filter

Place Name: Kaifeng Place Name: [ ] Clear Filter

As described in the section on **LookAtEntry**, one can also use the **Filter** feature to select a group of address all beginning with a specified word or phrase.

### B. Import a List of Places

Sometimes it is more useful to work with a set of Address IDs to precisely define the area for which one wants to study the social networks. Importing a list of Address ID works the same way as importing Person IDs. CBDB disregards lines that begin with text as well as all text the follows after the Address ID: it simply requires that the ID be the first item on the line.

Unless one clicks the **Restrict to Place** check box, the selection of a place or list of places *only influences the first step* of locating an initial group of people around whom to build a social network. After the first round of locating people with a connection to the specified place(s), CBDB searches for the sorts of associations selected as the next part of the query process.

If one uses both people and place as the starting point for a query, CBDB looks for people from the specified place who had connections to the selected person or group of people that matched the categories of association specified for the search.

### C. Use XY Reference

As in the other forms, CBDB allows the one to use the longitude and latitude of the place(s) one has selected to identify other relevant administrative units for the specified time period. One clicks on the **Use XY Reference** check box to activate this feature.

## 3. Determine the Time Period

This is straight-forward: simply fill in the beginning and ending years for the *index years* of people to be considered for the search.

#### 4. Select the Node Distance

One needs to be careful: the number of people found by the search procedure can grow exponentially with the increase in node distance. It is a good practice to start conservatively with a small node distance. In the example search discussed below, using the nine people who became eligible for service through law examinations, a node distance of three produces over 5000 relationships.

#### 5. Set the Maximum Number of Iterations

The search procedure is slow with CBDB's large dataset, and one might want to experiment with a relatively small "Max Loop #."

#### 6. Select Kin, Non-Kin, Male and Female

The LookAtKinship form does not allow one to look at kinship relations for a group of people imported through a list, so LookAtNetworks provides an alternative approach to examining kinship. One selects "Kin" and de-selects "Non-Kin." There also may be time when one wants to eliminate associations (kinship or social) based on females, or one may want to examine networks strictly among women. LookAtNetworks allows the user to select these options.

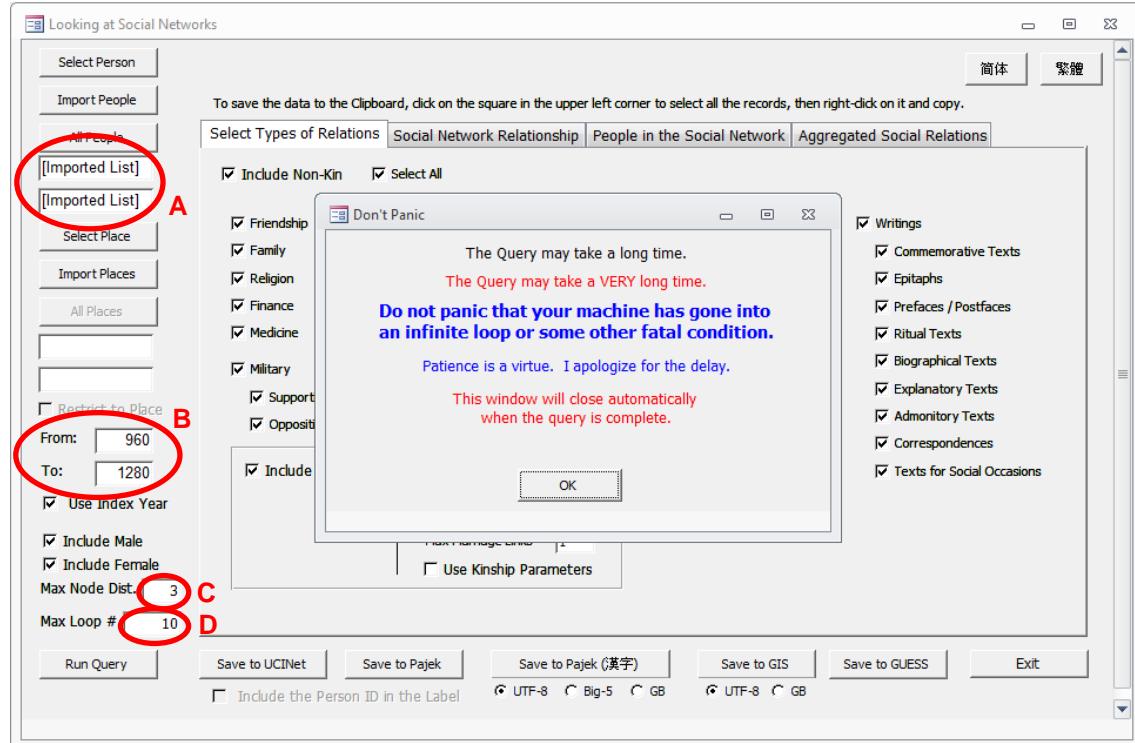
#### 7. Select Types of Non-Kinship Relations

Because there are many, many categories of non-kinship relationship, most of which are of little importance in a particular query, one can limit the search to selected large groups of associations. These are:

- Friendship
- Family
- Religion
- Finance
- Medicine
- Military
- Scholarship
- Politics
- Writing

The last four types of non-kin associations have further selectable subdivisions. "Military" has two, "Scholarship" seven, "Politics" six, and "Writings" nine. One can mix the types of associations as one wishes. Once selected, these limits to the range of associations remain active through the entire search process.

Once all of these decisions have been made, one runs the query. It may take a long time. The example examined below uses the list of people (A) who entered service through the law examination. The first version selects the years 960 through 1280 (B) with a maximum node distance of 3 (C) and a maximum loop count of 10 (D) but does not constrain either the kinship or the non-kinship associations and allows all possible types of association.



The result is a network with 10,985 people participating in 38,737 relations.

Name	姓名	Linked to	社會關係人姓	Kin/N	Link
An Dun	安惇	An Fang	安邦	K	S
An Dun	安惇	An Tao	安袁	N	Coalition member of
An Dun	安惇	Zhang Dun	章惇	N	Coalition associate of
An Dun	安惇	Zhang Dun	章惇	N	Supported
An Dun	安惇	Sima Guang	司馬光	N	Disagreed with views of
An Dun	安惇	Sima Guang	司馬光	N	did harm to
An Dun	安惇	Cai Jing	蔡京	N	Coalition associate of
An Dun	安惇	Cai Jing	蔡京	N	Served in the same bureau
An Dun	安惇	Zou Hao	鄒浩	N	Opposed or attacked
An Dun	安惇	Cai Bian	蔡卞	N	Supported
An Dun	安惇	An Yaochen	安堯臣	K	K+1(male)
An Dun	安惇	An Bin	安邠	K	S
An Dun	安惇	Li Jie	李駢	N	Opposed by or attacked by
An Dun	安惇	Zhang Shiliang	張士良	N	Opposed or attacked
An Dun	安惇	Zhang Shiliang	張士良	N	did harm to
An Dun	安惇	Shi Yu	石豫	N	Recommended
An Dun	安惇	Xiahou Mao	夏侯茂	N	Purged
An Dun	安惇	An Jiao	安郊	K	S1
An Dun	安惇	An Bang	安邦	K	S2

**Looking at Social Networks**

To save the data to the Clipboard, click on the square in the upper left corner to select all the records, then right-click on it and copy.

Select Types of Relations										Social Network Relationship		People in the Social Network		Aggregated Social Relations	
Name	姓名	Female	Indi	AddrID	From	籍貫	X	Y	Pers						
An Dun	安惇	<input type="checkbox"/>	1101	13551	Qujiang	渠江	106.631752	30.474428							
An Fang	安邡	<input type="checkbox"/>	1136	13551	Qujiang	渠江	106.631752	30.474428							
An Tao	安燾	<input type="checkbox"/>	1102	11028	Kaifeng	開封	114.34333	34.785477							
Zha Dao	查道	<input type="checkbox"/>	1014	12853	Xiuning	休寧	118.175774	29.7811909							
Zha Yue	查籥	<input type="checkbox"/>	1181	13100	Jiangling	江陵	112.190773	30.350445							
Chai Chengwu	柴成務	<input type="checkbox"/>	993	11251	Jiyin	濟陰	115.282862	34.990951							
Chao Qianzhi	晁謙之	<input type="checkbox"/>	1154	12889	Qianshan	鉛山	117.776588	28.208446							
Chao Gongwu	晁公武	<input type="checkbox"/>	1162	12889	Qianshan	鉛山	117.776588	28.208446							
Chao Gongmai	晁公邁	<input type="checkbox"/>	1140	12967	Linchuan	臨川	116.351341	27.984781							
Chao Buzhi	晁補之	<input type="checkbox"/>	1110	11263	Juye	鉅野	116.088707	35.3962097							
Chao Duanyan	晁端彥	<input type="checkbox"/>	1092	11028	Kaifeng	開封	114.34333	34.785477							
Chen Anmin	陳安民	<input type="checkbox"/>	1099	11478	Heyang	河陽	112.785721	34.840869							
Chen Anshi	陳安石	<input type="checkbox"/>	1073	11478	Heyang	河陽	112.785721	34.840869							
Chen Andao	陳安道	<input type="checkbox"/>	1072	11478	Heyang	河陽	112.785721	34.840869							
Chen Chong	陳充	<input type="checkbox"/>	1073	0	[Unknown]	[未詳]									
Chen Zhengtong	陳正同	<input type="checkbox"/>	1137	13921	Sha Xian	沙縣	117.781792	26.39761							
Chen Ji(2)	陳機	<input type="checkbox"/>	1158	12701	Dantu	丹徒	119.44429	32.20649							
Chen Jue	陳桷	<input type="checkbox"/>	1150	12781	Pingyang	平陽	120.556442	27.675718							
Chen Zhiyan	陳志研	<input type="checkbox"/>	1000	11140	Qingshang	青唐	119.644240	34.76076							

Record: 14 < 1 of 10985 | No Filter | Search | Run Query | Save to UCINet | Save to Pajek | Save to Pajek (漢字) | Save to GIS | Save to GUESS | Exit |  Include the Person ID in the Label |  UTF-8  Big-5  GB |  UTF-8  Big-5  GB

Many of the pairs of people in this list have more than one relationship between them, so CBDB also produces a table in the **Aggregated Social Relations** tab with just one record for each pair of people that gives the number of relations between them:

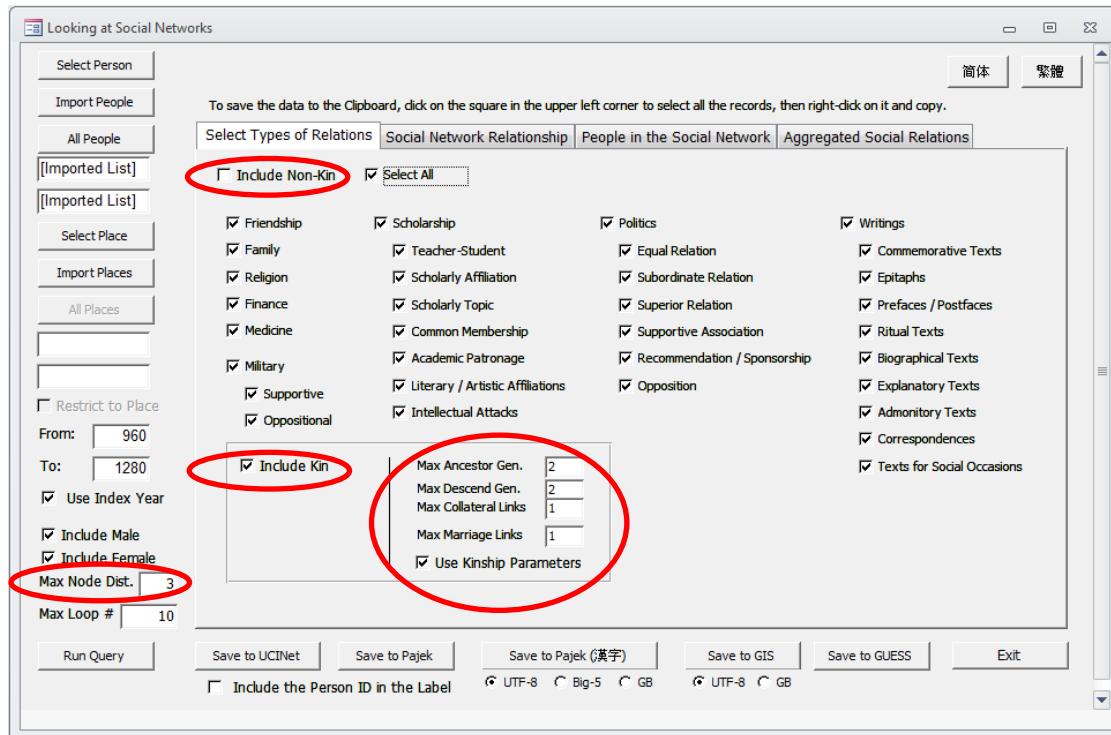
**Looking at Social Networks**

To save the data to the Clipboard, click on the square in the upper left corner to select all the records, then right-click on it and copy.

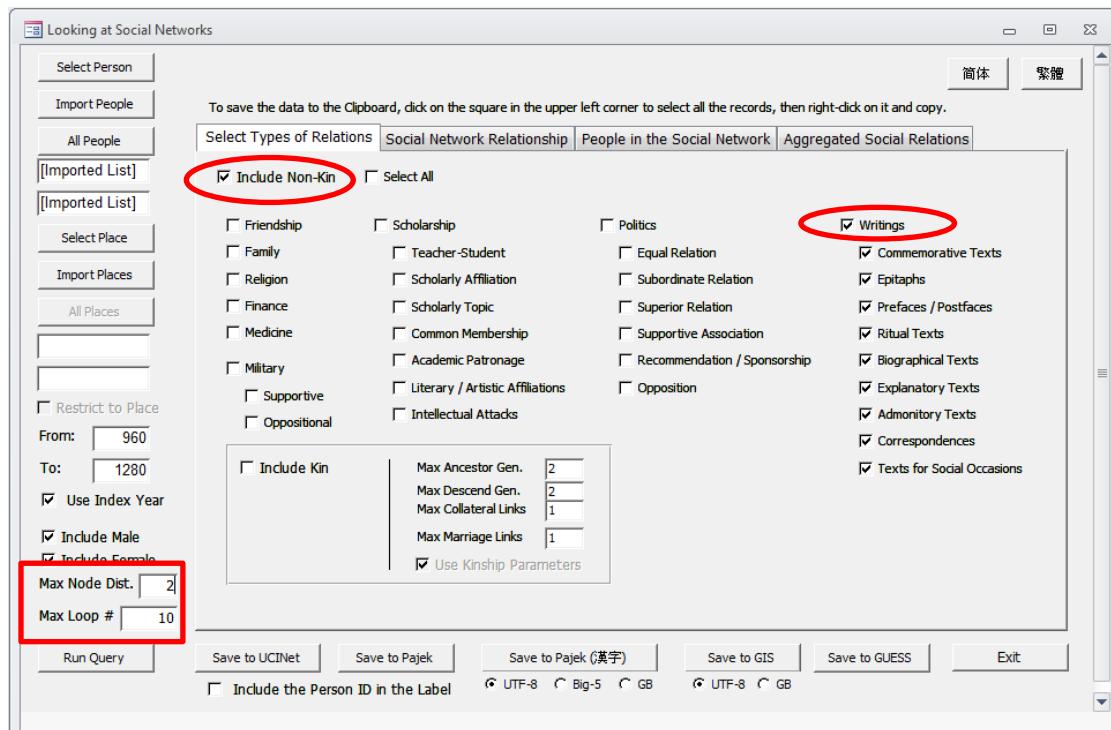
Select Types of Relations					Social Network Relationship		People in the Social Network		Aggregated Social Relations	
Name	姓名	Linked to	社會關係人姓	Link						
An Dun	安惇	An Fang	安邡	K:S						
An Dun	安惇	An Tao	安燾	N:Coalition member of						
An Dun	安惇	Zhang Dun	章惇	Multiple associations merged						
An Dun	安惇	Sima Guang	司馬光	Multiple associations merged						
An Dun	安惇	Cai Jing	蔡京	Multiple associations merged						
An Dun	安惇	Zou Hao	鄒浩	N:Opposed or attacked						
An Dun	安惇	Cai Bian	蔡卞	N:Supported						
An Dun	安惇	An Yaochen	安龜臣	K:K+1(male)						
An Dun	安惇	An Bin	安邠	K:S						
An Dun	安惇	Li Jie	李階	N:Opposed by or attacked by						
An Dun	安惇	Zhang Shiliang	張士良	Multiple associations merged						
An Dun	安惇	Shi Yu	石豫	N:Recommended						
An Dun	安惇	Xiahou Mao	夏侯茂	N:Purged						
An Dun	安惇	An Jiao	安郊	K:S1						
An Dun	安惇	An Bang	安邦	K:S2						
An Tao	安燾	Chang Anmin	常安民	N:Supported						
Zha Dao	查道	Fan Zhigu	樊知古	N:Recommended by						
Zha Dao	查道	Cha Yuanfang	查元方	K:F						
Zha Dao	查道	Li Xuji	李虛己	N:Served in the same bureau with						

Record: 14 < 1 of 27427 | No Filter | Search | Run Query | Save to UCINet | Save to Pajek | Save to Pajek (漢字) | Save to GIS | Save to GUESS | Exit |  Include the Person ID in the Label |  UTF-8  Big-5  GB |  UTF-8  Big-5  GB

If the network is too large, one can examine more narrowly defined networks. If one looks just at kinship relations for the group, CBDB finds 144 people linked through 165 relations (with total node distance of 3 and constraints on the kinship distance for the relationships).



If one looks just at associations formed through writing with a maximum node distance of 2 and excludes kinship, CBDB discovers 3443 aggregated relations among 1191 people:



The screenshot shows the 'Looking at Social Networks' window. On the left, there are several buttons and input fields for filtering: 'Select Person', 'Import People', 'All People', '[Imported List]', '[Imported List]', 'Select Place', 'Import Places', 'All Places', and 'Restrict to Place'. Below these are dropdowns for 'From' (960) and 'To' (1280), and checkboxes for 'Use Index Year', 'Include Male', 'Include Female', 'Max Node Dist.' (set to 2), and 'Max Loop #' (set to 10). At the bottom left of the main area, there is a status bar with the text 'Records: 14 of 7064'.

Name	姓名	Linked to	社會關係人姓	Kin/N	Link
Xu Zun	許遵	Wang Anshi	王安石	N	Was sent a reply by
Wang Yi	王衣	Qi Chongli	綦崇禮	N	Epitaph written by
Ximen Chengyun	西門成允	Liu Zhi	劉摯	N	Epitaph written by
Ximen Chengyun	西門成允	Liu Zhi	劉摯	N	Epitaph written by
Wang Yi	王衣	Qi Chongli	綦崇禮	N	Epitaph written by
Chen Gui	陳規	Zhang Shi	張栻	N	Postface of book written by
Xu Ziyan	徐子寅	Lou Yue	樓鑑	N	Biographical sketch (xingzh)
Xu Ziyan	徐子寅	Lou Yue	樓鑑	N	Biographical sketch (xingzh)
Xu Ziyan	徐子寅	Lou Yue	樓鑑	N	Sacrificial prayer written by
Xu Ziyan	徐子寅	Lou Yue	樓鑑	N	Sacrificial prayer written by
Xu Ziyan	徐子寅	Lou Yue	樓鑑	N	Epitaph written by
Chen Gui	陳規	Yang Jian	楊簡	N	Preface of book by
Chen Gui	陳規	Zhu Xi	朱熹	N	Postface of book written by
Zhu Xi	朱熹	Chen Mi	陳宓	N	Invocation prayer written by
Zhu Xi	朱熹	Chen Shou	陳守	N	Sent letter to
Zhu Xi	朱熹	Chen Ding	陳定	N	Epitaph written for
Zhu Xi	朱熹	Chen Ding	陳定	N	Epitaph written for
Zhu Xi	朱熹	Chen Mi	陳宓	N	Sacrificial prayer written by

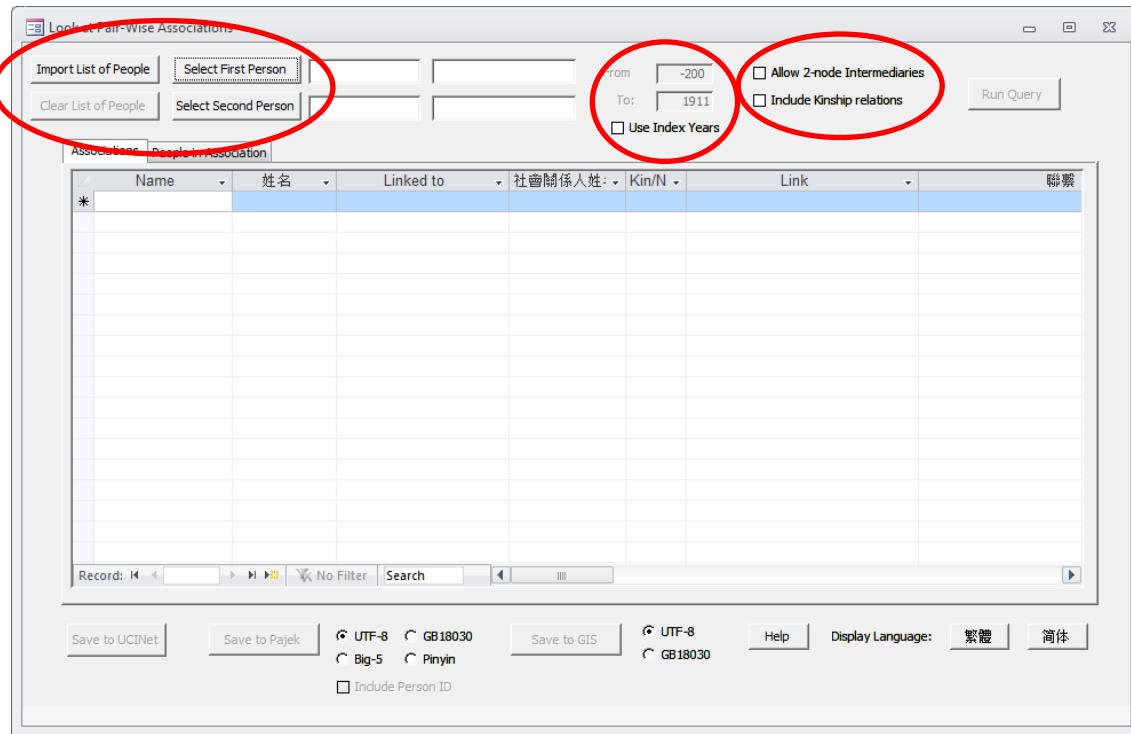
The results seem promising: not too many links, and not too few. However, if one scrolls to the right in the table of results and right-clicks on the header of the field called “Edge Distance” to sort the records, one will discover that only the first nine records connect the initial group of people who became eligible for service through legal examinations with other individuals. (These are relations with an “edge distance” of 0, i.e., directly linked to the original list.) Only five of the initial thirteen people have any associations defined by writings, and these links are to only seven people. Of those seven associations, five are to people (Liu Zhi, Yang Jian, Zhu Xi, Zhang Shi, and Lou Yue) who have vast social networks who contribute most of the relations in the social network. Thus it perhaps is better to return to the larger set of unrestricted relations among 3964 people and use the tools of social network analysis to sort through the data.

**LookAtNetworks** provides ways to output the results of a query to three different SNA programs: **UCINet**, **Pajek**, and **GUESS**. Because Pajek has begun to support Chinese characters, CBDB allows the output to Pajek to be in either of three coding systems: UTF-8, Big-5, and GB. GIS software also supports Chinese characters, but how they are handled differs depending on the regional settings of one’s computer. The default display for both nodes and edges in the SNA output files uses color-coding to indicate degree of distance from the target person:

- White = the target node (the UCINet files set these to Black)
- Blue = nodes at a summed distance of 1
- Green = nodes at a summed distance of 2
- Orange = nodes at a summed distance of 3
- Yellow = nodes at a summed distance of 4
- Red = nodes at a summed distance of 5 or more

## F. Using the Form LookAtAssociationPairs

At times one wants to consider whether there were any social links between two individuals or among members of a group of people identified through criteria other than those of kinship or social network. One could use **LookAtNetworks** to generate the social network of one person and see at what point the other person or people appear as part of the network. However, the Access version of CBDB provides a tool to directly examine if there were any connections without going through the general network search.



The form is simple. First one (1) either chooses two individuals or imports a list of people using the procedure describe for other forms above, then (2) the range of index years for the people in the relations, if desired, and finally, (3) the type of permissible relations. The options for relationships are:

1. **Allow 1-node Intermediaries:** That is, people who are directly linked to both (or, for imported lists, two) of the selected people: [Person A] — Node1 — [Person B]. In this case one leaves the check box for two-node intermediaries unchecked.
2. **Allow 2-node Intermediaries:** Here one allows people linked to one person who in turn have links to people linked to the second person (or to another person on the imported list): [Person A] — Node1 — Node2 — [Person B]. In this case one clicks on the check box for two-node intermediaries to select the option.
3. **Include Kinship relations:** The default is simply to look at social (non-kinship) relations connecting people, but kinship also can be important, and the form allows on to examine the role of kinship relations in the social network.

## ONE NODE INTERMEDIARY SEARCHES

For example, if one explores the links between Su Shi 蘇軾 and Cheng Yi 程頤, allowing only people directly linked to both of them finds 211 associations among twenty people.

The screenshot shows the "LookAtPairWiseAssociations" application window. At the top, there are two search fields: "Select First Person" with "蘇軾" (Su Shi) and "Select Second Person" with "程頤" (Cheng Yi). Below these are several filter options: "From" set to "-200", "To" set to "1911", and checkboxes for "Allow 2-node Intermediaries" (unchecked), "Include Kinship relations" (unchecked), and "Use Index Years" (unchecked). The main area displays a table titled "Associations" with columns: Name, 姓名, Linked to, 社會關係人姓, Kin/N, Link, and a Link column. The table lists 211 records. A red circle highlights the "Associations" tab and the first few rows of the table. Another red circle highlights the "Record: 14 < 1 of 211" status bar at the bottom. The bottom of the window contains various export options like "Save to UCINet", "Save to Pajek", and "Save to GIS", along with encoding settings (UTF-8, GB18030, Big-5, Pinyin) and language selection buttons for Chinese characters.

As in **LookAtNetworks**, the form provides two output tables: “Associations” for the relationships, and another, “People in Associations,” for the people in the relations

This screenshot shows the same application window, but the "People in Association" tab is selected. The table has columns: Name, 姓名, Female, Indi, AddrID, From, 籍貫, X, Y, Person, Node Dis, and XY cc. It lists 20 records. A red circle highlights the "People in Association" tab and the first few rows of the table. Another red circle highlights the "Record: 14 < 1 of 20" status bar at the bottom. The bottom of the window contains the same export and language options as the previous screenshot.

As with the other forms, one can save the results of a search by clicking on the grey square in the upper left hand corner of the table to select all the records and then using Ctrl-C:

Name	姓名	Linked to	社會關係人姓	Kin/N	Link	
Lv Gongzhu	呂公著	Sima Guang	司馬光	N	Friend of	友
Su Shi	蘇軾	Xie Jingwen	謝景溫	N	Impeached by	被Y彈劾
Su Shi	蘇軾	Xie Jingwen	謝景溫	N	Impeached by	被Y彈劾
Su Shi	蘇軾	Chao Yuezhi	晁說之	N	Recommended	推薦
Sima Guang	司馬光	Fan Zuyu	范祖禹	N	Recommended	推薦
Sima Guang	司馬光	Fan Zuyu	范祖禹	N	Recommended	推薦
Fan Zuyu	范祖禹	Yang Guobao	楊國寶	N	Recommended	推薦
Su Shi	蘇軾	Fan Zuyu	范祖禹	N	Recommended by	被Y推薦
Cheng Yi	程頤	Sima Guang	司馬光	N	Recommended by	被Y推薦
Cheng Yi	程頤	Sima Guang	司馬光	N	Recommended by	被Y推薦
Li Zhichun	李之純	Wen Yanbo	文彥博	N	Recommended by	被Y推薦
Chao Yuezhi	晁說之	Fan Zuyu	范祖禹	N	Recommended by	被Y推薦
Sima Guang	司馬光	Chao Yuezhi	晁說之	N	Praised or admired by	被Y欣賞/器重
Su Shi	蘇軾	Liu Tingshi	劉庭式	N	Praised or admired	欣賞/器重
Cheng Yi	程頤	Yang Guobao	楊國寶	N	Praised or admired	欣賞/器重
Cheng Yi	程頤	Liu Tingshi	劉庭式	N	Praised or admired	欣賞/器重
Su Shi	蘇軾	Li Zhichun	李之純	N	Supported by	得到Y的支持
Sima Guang	司馬光	Wen Yanbo	文彥博	N	Prefaced book by	為Y所著書作序

One also can sort on a column of the table by clicking on the column (in this case, “Name”) to select it, then right-clicking to choose the type of sort:

Name	Subform	id to	社會關係人姓	Kin/N	Link	
Lv Gongzhu			司馬光	N	Friend of	友
Su Shi			謝景溫	N	Impeached by	被Y彈劾
Su Shi			謝景溫	N	Impeached by	被Y彈劾
Su Shi			晁說之	N	Recommended	推薦
Sima Guang			范祖禹	N	Recommended	推薦
Sima Guang			范祖禹	N	Recommended	推薦
Fan Zuyu			楊國寶	N	Recommended	推薦
Su Shi			范祖禹	N	Recommended by	被Y推薦
Cheng Yi			司馬光	N	Recommended by	被Y推薦
Cheng Yi			司馬光	N	Recommended by	被Y推薦
Li Zhichun			文彥博	N	Recommended by	被Y推薦
Chao Yuezhi			范祖禹	N	Recommended by	被Y推薦
Sima Guang			晁說之	N	Praised or admired by	被Y欣賞/器重
Su Shi			劉庭式	N	Praised or admired	欣賞/器重
Cheng Yi			楊國寶	N	Praised or admired	欣賞/器重
Cheng Yi			劉庭式	N	Praised or admired	欣賞/器重
Su Shi			李之純	N	Supported by	得到Y的支持
Sima Guang			文彥博	N	Prefaced book by	為Y所著書作序

One also can select a block of records to save by clicking the mouse on the left-hand grey column of the first record in the block and then, with the left-click button still held down, dragging the mouse down the grey column to the last record in the desired group:

The screenshot shows the 'Look at Pair-Wise Associations' window. In the 'Associations' tab, a block of records is selected, indicated by a red rectangle around the first few rows. The selected records include entries for Cheng Yi, Fu Bi, Fan Zuyu, and Han Jiang, among others. The 'Link' column contains various relational terms like 'Sacrificial prayer written for', 'Sent letter to', 'Recommended', etc. At the bottom of the table, it says 'Record: 14 1 46 of 211'. Below the table, there are buttons for saving to UCI Net, Pajek, or GIS, and language options for English (UTF-8, GB18030, Big-5, Pinyin) and Chinese (繁體, 簡體). A checkbox for 'Include Person ID' is also present.

However, note that the entry directly below the selected block includes Dai Biaoyuan 戴表元 (1244-1310), a late Southern Song figure. If one wishes to narrow the search to intermediate nodes who are roughly contemporaneous with the target people, one can use **index years** to limit the search. If one limits the index years to a range between 1050 and 1120, one finds fourteen people with 108 relations connecting them:

The screenshot shows the same 'Look at Pair-Wise Associations' window, but with a different set of parameters. The 'From' field is set to 1050 and the 'To' field is set to 1120. The 'Use Index Years' checkbox is checked. The result is a much smaller set of records, with a red circle around the bottom of the table indicating the reduced number of records (1 of 108). The rest of the interface is identical to the previous screenshot, including the 'Associations' tab, the list of people, and the bottom buttons for saving and displaying language.

If one then includes kin of either Su Shi or Cheng Yi who have a social connection to the other, then one discovers one additional connection but, in this case, no additional people:

The screenshot shows the 'Look at Pair-Wise Associations' window. The 'Select First Person' field contains '蘇軾' (Su Shi) and the 'From' field is set to '1050'. The 'Select Second Person' field contains '程頤' (Cheng Yi) and the 'To' field is set to '1120'. The 'Include Kinship relations' checkbox is checked. The results table lists various associations, including several entries for 'Lv Gongzhu' (呂公著) and 'Wen Yanbo' (文彥博), which are highlighted with red boxes. The status bar at the bottom indicates 'Record: 14 1 of 109'.

## TWO NODE INTERMEDIARY SEARCHES

If one broadens the search to allow two intermediary links to connect the target people, the network becomes more complicated: The program reveals 1268 relations among 115 people with index years between 1050 and 1120:

The screenshot shows the 'Look at Pair-Wise Associations' window with the 'Allow 2-node Intermediaries' checkbox checked. The 'Select First Person' field contains '蘇軾' (Su Shi) and the 'From' field is set to '1050'. The 'Select Second Person' field contains '程頤' (Cheng Yi) and the 'To' field is set to '1120'. The results table is much larger, indicating 1268 records. The status bar at the bottom indicates 'Record: 14 1 of 1268'.

## SEARCHES USING LISTS

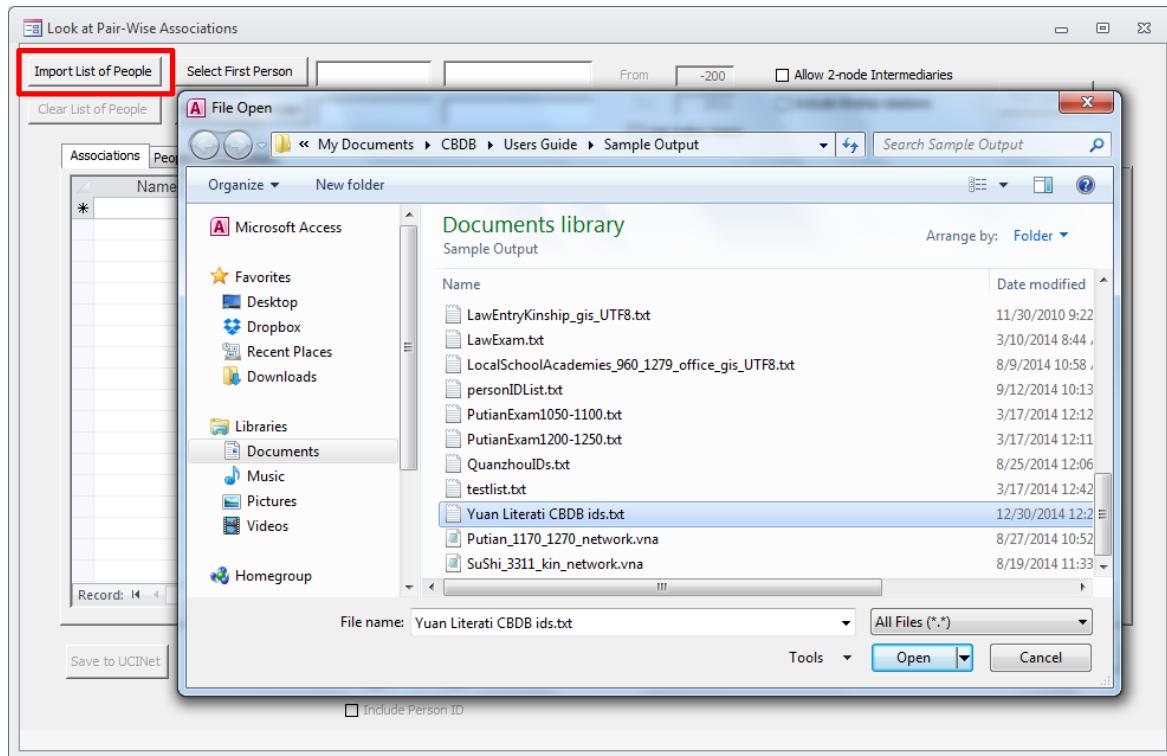
If one wants to look for connections within a larger group of people chosen by other criteria, the form allows one to import a list of person IDs. Here one looks at Jinhua men who from the Yuan dynasty who have extant collections. As in all lists for importing people, CBDB ignores lines that do not start with numbers and simply extracts the initial numbers and ignores any subsequent text on a line:

```

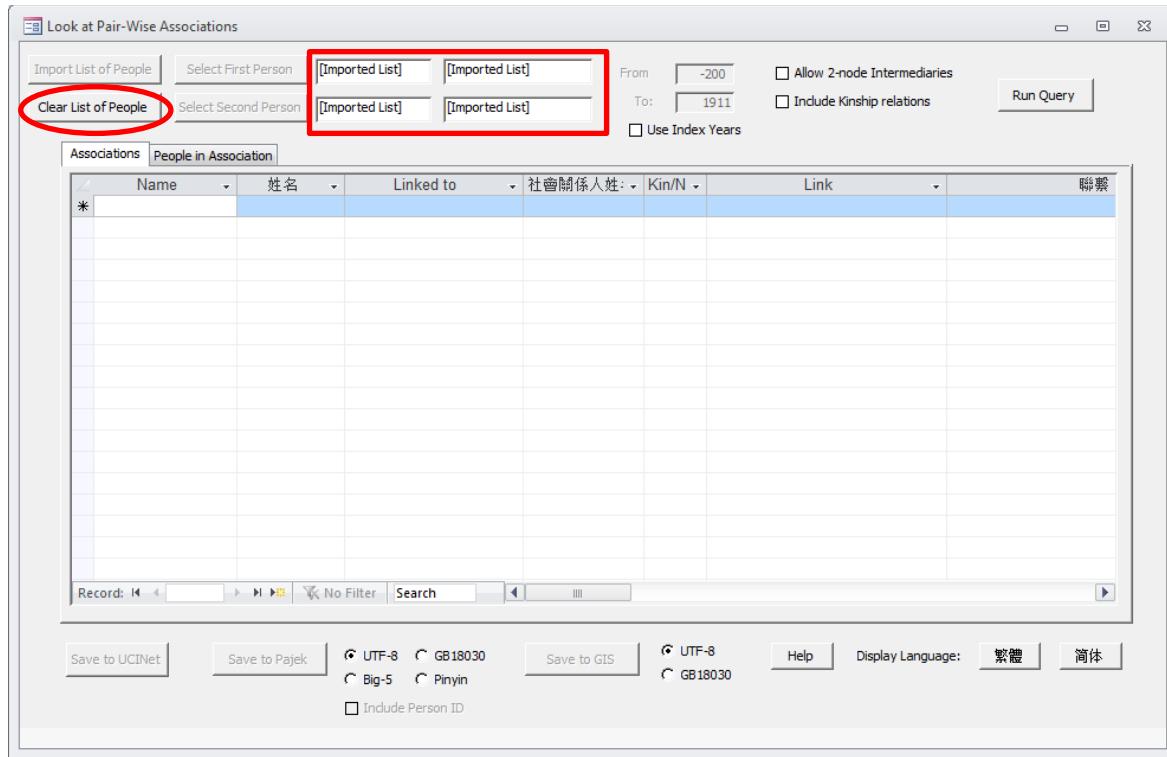
Yuan Literati CBDB ids.txt - Notepad
File Edit Format View Help
Yuan Jinhua literati with extant collections|
0010727
0010706
0010731
0010733
0028195
0027957
0010740
0028426
0028142
0028143
0027852
0010097
0010726
0028517
0028688
0041460
0028498
0027951
0035430
0028673
0035614
0107337
0034380

```

One clicks on the **Import List of People** command button and locates the file:



If the file is successfully read, the form indicates that the names are from an imported list:



To clear the list and return to selecting people through the two **Select** command buttons, simply click on the **Clear List of people** command button.

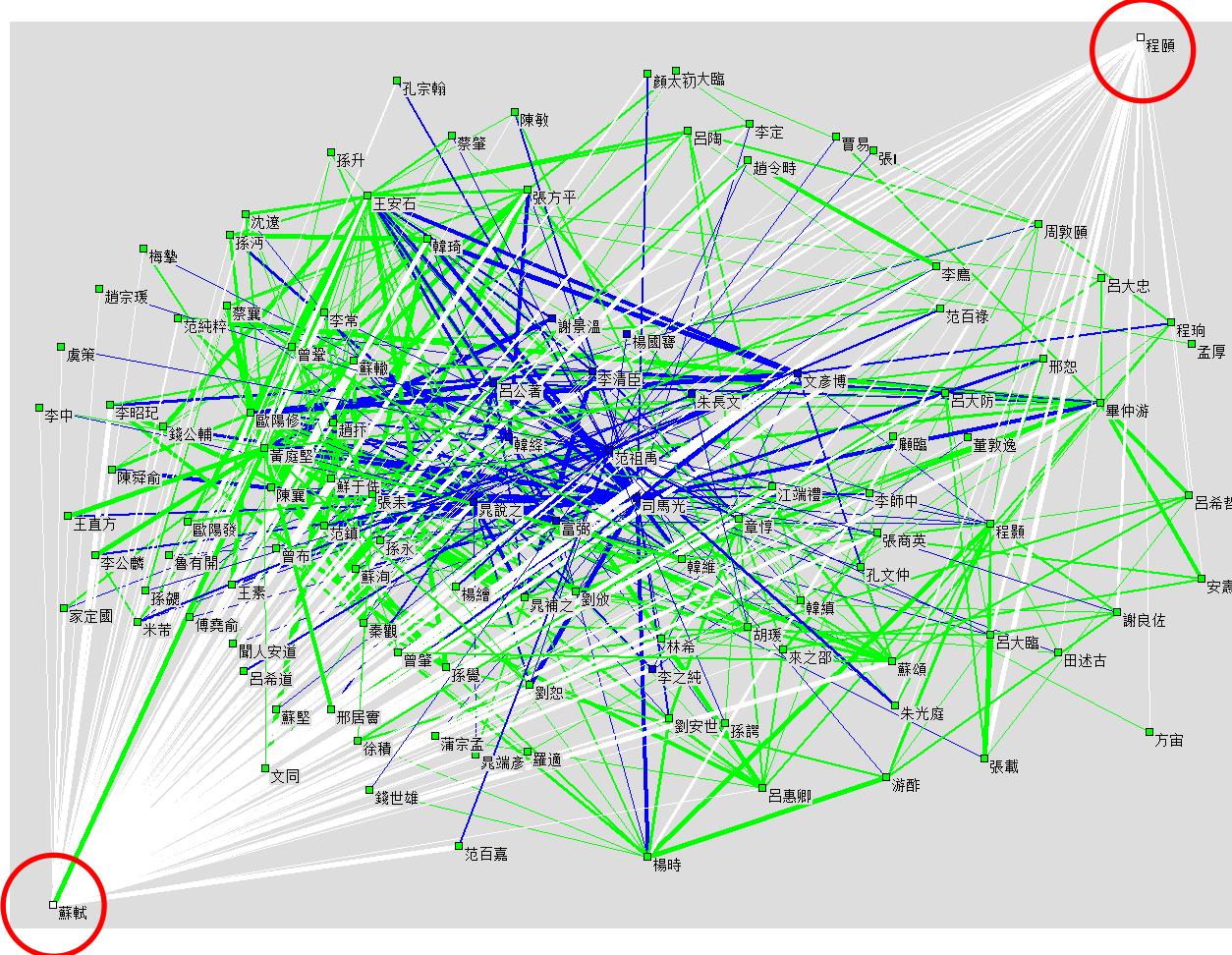
Once one has imported the list, the search procedures are the same. In this case, the query is set to look for one-node intermediaries with index years between 1250 and 1400 and produces 1532 associations among 180 people:

Name	姓名	Linked to	社會關係人姓	Kin/N	Link	聯繫
Xu Heng	許衡	Xu Heng	許衡	N	Ancestral stele or records written by	為Y作世系碑記
Xu Heng	許衡	Xu Heng	許衡	N	Ancestral stele or records written by	世系碑記由Y所作
Xu Heng	許衡	Xu Heng	許衡	N	Presented literary composition as	贈詩、文
Xu Heng	許衡	Xu Heng	許衡	N	Received literary composition as	收到Y的贈詩、文
Xu Heng	許衡	Su Tianjue	蘇天爵	N	Preface book by	為Y所著書作序
Xu Heng	許衡	Su Tianjue	蘇天爵	N	Calligraphy or painting postscripter	書、畫由Y作跋
Xu Heng	許衡	Su Tianjue	蘇天爵	N	colophon to writings was composed	詩文跋由Y所作
Xu Heng	許衡	Ouyang Xuan	歐陽玄	N	Funerary stele written by	神道碑由Y所作
Xu Heng	許衡	Cheng Jufu	程鉅夫	N	School stele written by	學記〈書院記〉由Y所
Xu Heng	許衡	Zhang Yanghao	張義浩	N	School stele written by	學記〈書院記〉由Y所
Xu Heng	許衡	Zhu Derun	朱德潤	N	School stele written by	學記〈書院記〉由Y所
Wu Cheng	吳澄	Wu Cheng	吳澄	N	Epitaph written by	墓誌銘由Y所作
Wu Cheng	吳澄	Wu Cheng	吳澄	N	Epitaph written for	為Y作墓誌銘
Wu Cheng	吳澄	Wu Cheng	吳澄	N	Congratulatory note sent to	向Y致賀
Wu Cheng	吳澄	Wu Cheng	吳澄	N	Congratulatory note received from	從Y處收到賀詞 (occasional)
Wu Cheng	吳澄	Wu Cheng	吳澄	N	Kuangji (Grave Inscription) written	墓誌由Y所作
Wu Cheng	吳澄	Wu Cheng	吳澄	N	Wrote kuangji (Grave Inscription) for	為Y作墓誌
Wu Cheng	吳澄	Yu Ji(2)	虞集	N	Preface of book by	書序由Y所作

## OUTPUT TO SNA AND GIS PROGRAMS

Like the other forms, LookAtAssociationPairs can generate files for use with Pajek and with GIS visualization programs. The output tables for Associations and People are the same as those in LookAtAssociations. Please consult the information in that section of the User's Guide.

Allowing the form to list all the relations between the 1-node and 2-node intermediaries between Su Shi and Cheng Yi who have index years between 1050 and 1120 intermediaries produces a network that can be imported into Pajek.



The default display for both nodes and edges in the SNA output files uses color-coding to indicate degree of distance from the target person and the type of connections:

### Nodes

White = the target nodes;

Blue = nodes that serve as 1-node intermediaries

Green = nodes that serve as 2-node intermediaries

### Edges

from target nodes

from 1<sup>st</sup> order to 2<sup>nd</sup> order nodes

between 2<sup>nd</sup> order nodes (except for one mysterious line to Su Shi)

The output files aggregate the associations between people, and the width of the lines reflects the number of associations between nodes.

# Chapter 4: Advanced Query Techniques

The Access version of CBDB permits a variety of increasingly complex and powerful approaches to analyzing the data. The first level of advanced query simply is to use the output from one form as the input for a second search. The next step, taken when one has become relatively familiar with the data structures in CBDB, is to use the Access built-in Query Design form to create free-form queries. As one's command of the concepts of SQL (Structured Query Language) deepens, one can create ever more sophisticated queries. This chapter considers one example of using the output from CBDB forms as input for other queries and then introduces the basic ideas of SQL and illustrates them through an example that requires two steps in query design.

## A. Kinship Networks for Examination Graduates in Putian, Fujian during the Song

One question in the study of social history during the Song dynasty is whether local elites remained stable and controlled access to the cultural resources needed to gain entrance to official status or whether there was in fact social mobility where marginal families managed to join the elite stratum through the educational success of their sons. To explore this question, one can look at the kinship structures for those who entered government service through examinations in localities at different times during the Song and see if there is any change in organization. In our example, we consider Putian in Fujian during two periods: 1050-1100 and 1200-1250. We first use the **LookAtEntry** form:

1  
2  
3  
4  
5

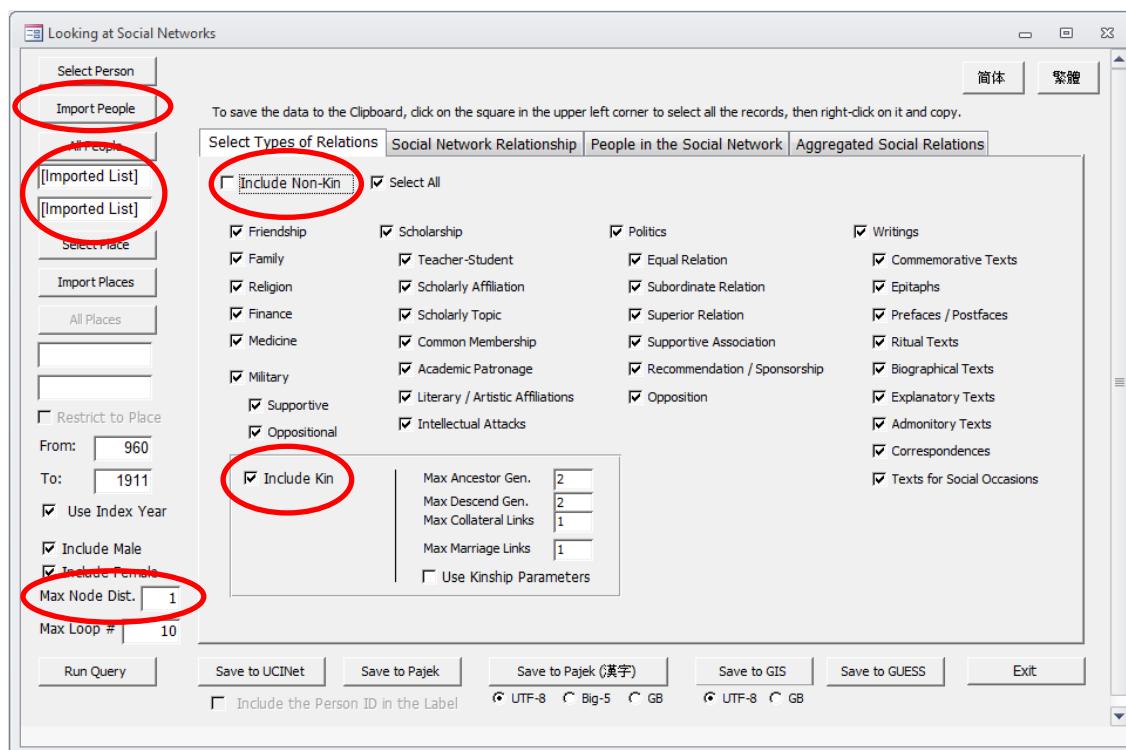
From	籍貫	X	Y	XY Count	Beam Rank	Relationship	Kin Name	親戚姓名	Person ID
Putian	莆田	119.01110	25.433954	113					25902
Putian	莆田	119.01110	25.433954	113	0				25905
Putian	莆田	119.01110	25.433954	113	0				25910
Putian	莆田	119.01110	25.433954	113	0				25936
Putian	莆田	119.01110	25.433954	113					26033
Putian	莆田	119.01110	25.433954	113					26038
Putian	莆田	119.01110	25.433954	113	0				26047
Putian	莆田	119.01110	25.433954	113	0				26048
Putian	莆田	119.01110	25.433954	113	0				26049
Putian	莆田	119.01110	25.433954	113	0				26050
Putian	莆田	119.01110	25.433954	113	0				26054
Putian	莆田	119.01110	25.433954	113	0				26065
Putian	莆田	119.01110	25.433954	113	0				26066
Putian	莆田	119.01110	25.433954	113	0				26067
Putian	莆田	119.01110	25.433954	113	0				26068
Putian	莆田	119.01110	25.433954	113	0				26069
Putian	莆田	119.01110	25.433954	113	0				26070
Putian	莆田	119.01110	25.433954	113					38983
Putian	莆田	119.01110	25.433954	113					42755
Putian	莆田	119.01110	25.433954	113					25902
Putian	莆田	119.01110	25.433954	113	1				25818
Putian	莆田	119.01110	25.433954	113					43485

Record: 1 of 113 | No Filter | Search | Run Query | Save to GIS | GB18030 | UTF-8 | 繁體 | 简体 | Exit |

The procedure is:

- (1) Use **Select Entry** to choose all types in the category of “Examination” 科舉門.
- (2) Set the range of examinations first to 1050-1100. (Here I show 1200-1250.)
- (3) Use **Select Place** to choose Putian 莆田 during the Song Dynasty.
- (4) Run the Query
- (5) Move the horizontal scroll bar to the right until you see “Person ID,” then click on the label “Person ID” to select all the entries in the column. Use **Ctrl-c** to copy the column to memory.
- (6) Create a text file with an appropriate name in an appropriate folder and paste the IDs into the file.
- (7) Repeat this process but change the years to 1200-1250 and create a second text file.

Once you have the two text files of the IDs of people from Putian who entered government through examination for the two specified periods, open the form **LookAtNetworks** and have the form read the file for 1050-1100:

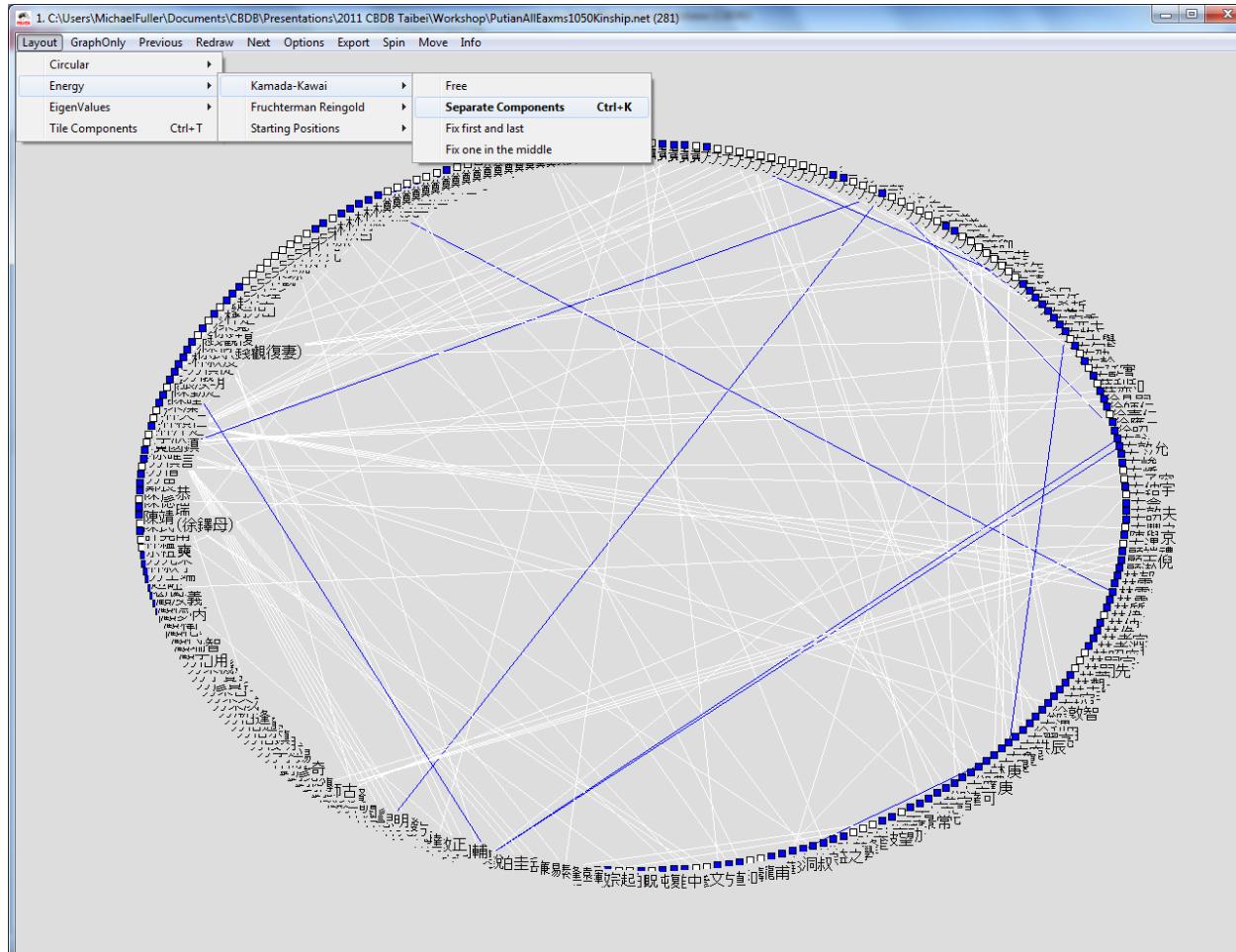


Here, the procedure is:

- (1) Import your text file with the **Import People** command button in the top left corner.
- (2) You will get confirmation that the list was correctly imported when you see “[Imported List”]
- (3) Set the node distance to 1: we want to look only at directly connected people.
- (4) In this case, we want to look at just kinship networks, so unclick “Non-Kin.”

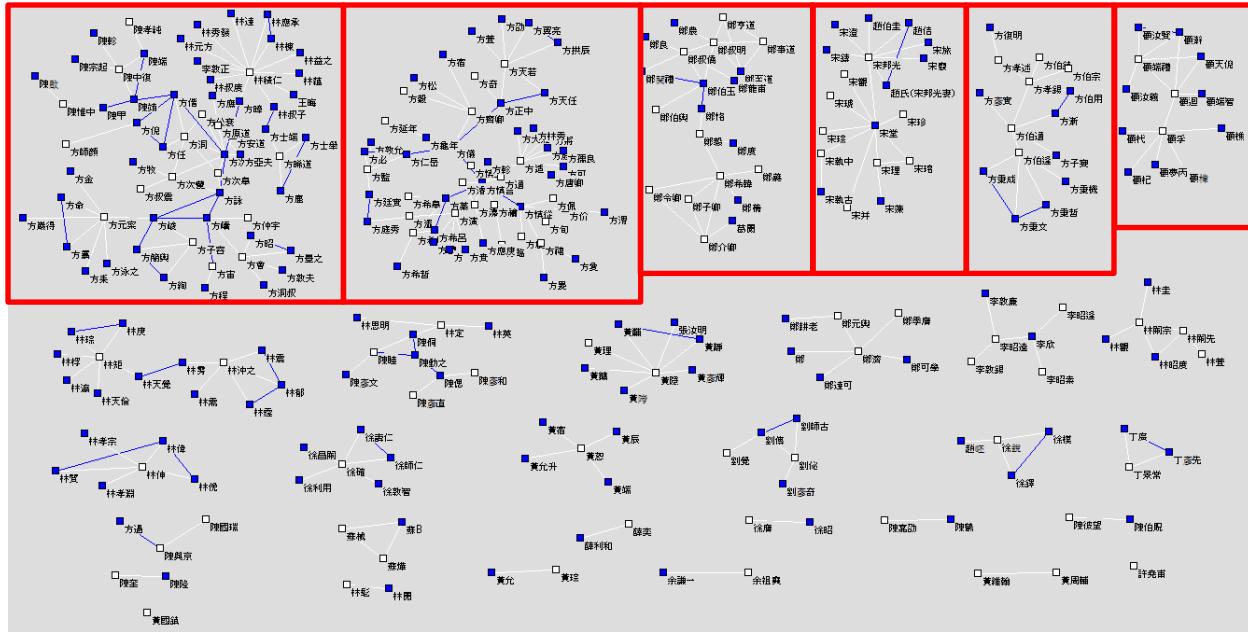
- (5) After you run the query, save the results into a **Pajek** file that uses UTF-8 encoding.  
 (6) Repeat the process for the people from 120-1250 and create a second Pajek file.

Open **Pajek** and “Draw” the results:

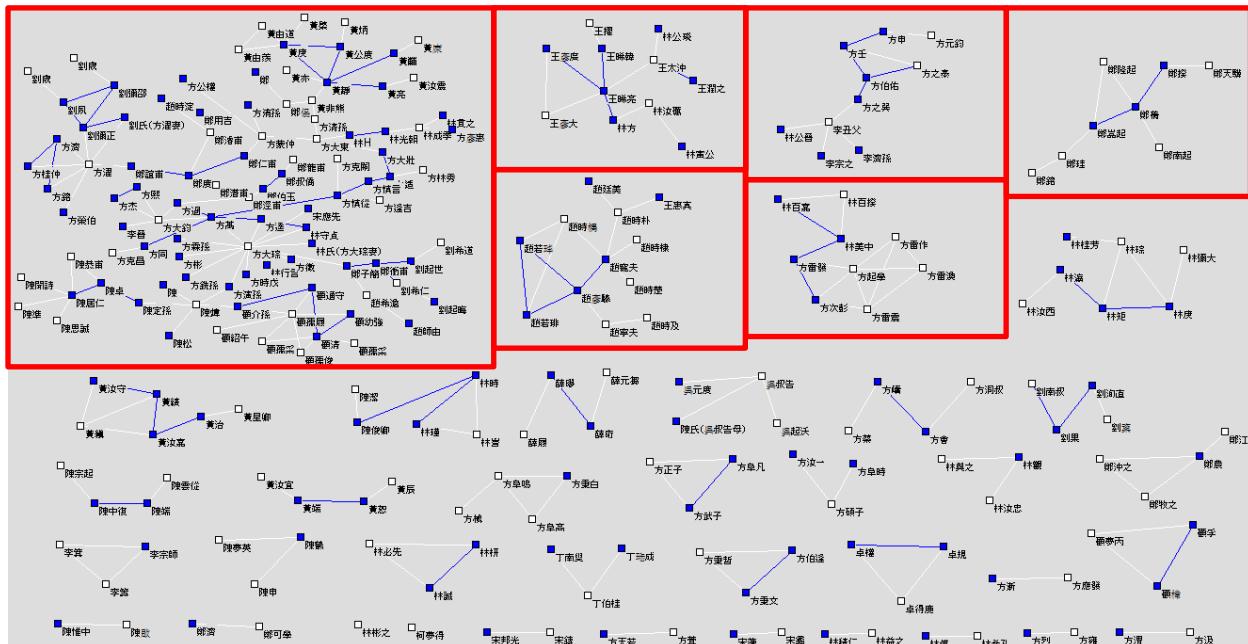


The initial layout for visualizing networks in Pajek is “Circular.” A more useful layout for looking at groups of kinship networks is to select “Separate Components” in the “Kamada-Kawai” layout listings.

When one select and closely looks at the components of the kinship networks for men from Putian who passed an examination for the years 1050-1100 and 1200-1250, one gets:



Putian Examination Kinship Networks, 1050-1100



Putian Examination Kinship Networks, 1200-1250

Note that by the later period, the “principal component” (the largest component in the network) has grown to include not only a Fang 方, Chen 陳, and Lin 林 clan, but also members of Zheng 鄭 and Gu 顧 clans. The Song 宋 surname largely has disappeared. In the diagrams, the white nodes are the men who passed the examinations, and the blue squares are their kin. The white lines connect the examinees to their kin and to each other, while the blue lines connect kin to one another.

## B. Using the Access Query Designer

Another extremely powerful capacity built into Access is the ability to design SQL queries to look at the CBDB data from whatever angle you wish. There are a few concepts to master, but the **Query Designer** in Access allows end-users to begin to explore the data without any knowledge of **SQL (Structured Query Language)**. As you become more familiar with queries, you can learn more about the formalisms to help you work with the data better.

In order to use the **Query Designer**, you will need some knowledge of the tables in CBDB and their relations to one another. We have simplified some of the tasks by creating a set of tables that are “denormalized,” that is, where we have added descriptive fields to explain the codes in the fields that rely on IDs. For example, the table BIOG\_ADDR\_DATA records lists of places associated with individuals: where they were born, where their “basic affiliation” was, where they moved, where they were buried, etc. The key information for each record, however, is a set of three codes: a person ID, an address ID, and an address type ID. We have created a second table, **ZZZ\_BIOG\_ADDR\_DATA**, that takes information from other tables (BIOG\_MAIN, ADDR\_CODES, BIOG\_ADDR\_CODES) to give the name of the person, the name of the place, and the description of the type of address, along with other useful data. Using these tables with descriptions and codes simplifies the task of building a useful query. The tables are:

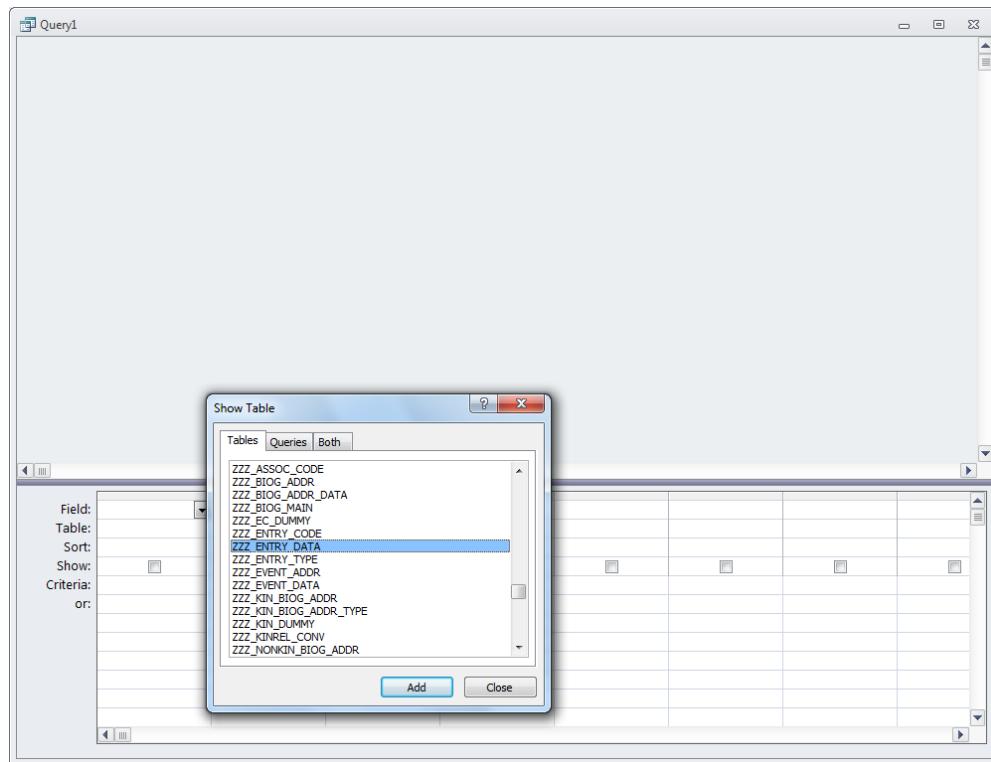
1. ZZZ\_ALT\_NAME\_DATA  
(fills in alternate name type)
2. ZZZ\_BIOG\_ADDR\_DATA  
(fills in address and address type)
3. ZZZ\_BIOG\_MAIN  
(fills in nianhao, ethnicity)
4. ZZZ\_ENTRY\_DATA  
(fills in entry type)
5. ZZZ\_KIN\_BIOG\_ADDR  
(this is the table for kinship, but it also provides the main entry for biographical address)
6. ZZZ\_NONKIN\_BIOG\_ADDR  
(this is the table for associations, but it also provides the main entry for biographical address)
7. ZZZ\_POSTED\_TO\_ADDR\_DATA  
(fill in address information)
8. ZZZ\_POSTED\_TO\_OFFICE\_DATA  
(fills in office information)
9. ZZZ\_TEXT\_DATA  
(fills in text data)

## I. An Example:

### The mode of entry into government of near kin of the successful *jinshi* degree candidates of the 1148 examination

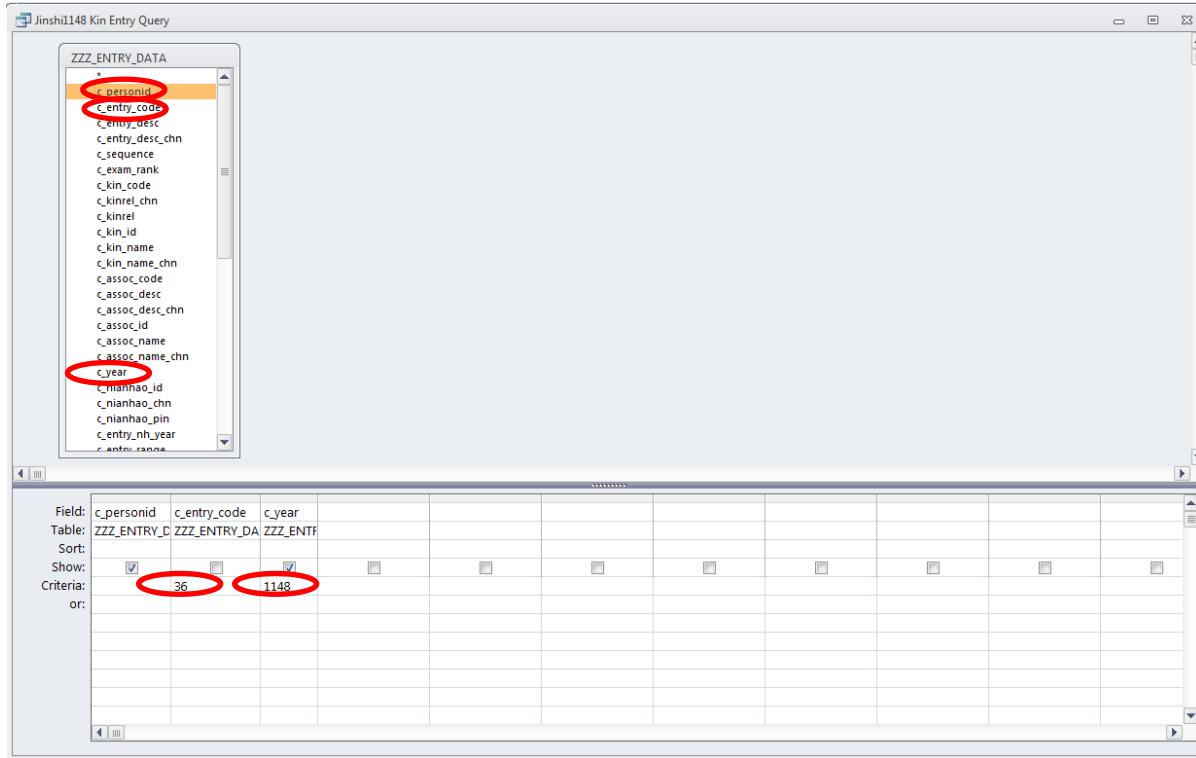
How might one use an SQL query to determine how many of the people who passed the *jinshi* examination in 1148 (for which we have a complete list) had close relatives who had entered government service?

1. In the **Create** menu (next to the **Home** tab at the top of the main screen) , Select **Query Design**:

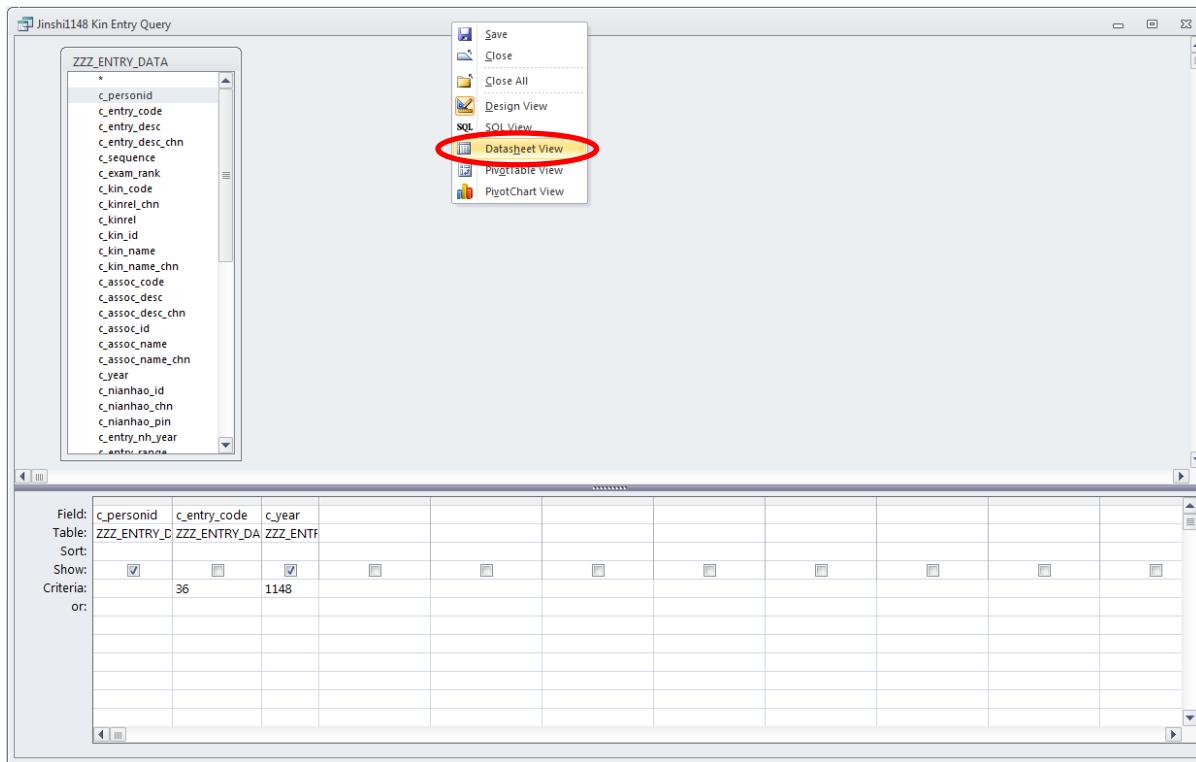


In the “Show Table” window, select **ZZZ\_ENTRY\_DATA** and click **Add**

2. Double-click on **c\_personid**, **c\_entry\_code**, and **c\_year** to specify the code 36 (*jinshi*), and the year 1148.



3. Check the results by right-clicking on the top border of the Query form and switching to **Datasheet View**:



4. There are 273 records. (Please note that as CBDB adds data, these results will change.)

c_personid	c_year
466	1148
70	1148
601	1148
3990	1148
7201	1148
667	1148
1714	1148
1286	1148
3166	1148
3317	1148
8139	1148
8159	1148
10131	1148
10572	1148
10702	1148
10938	1148
11187	1148
11280	1148
11341	1148
11358	1148
11416	1148
11572	1148
11873	1148
12302	1148
13230	1148
13280	1148
13286	1148
13438	1148
13464	1148
13477	1148
13650	1148
13951	1148
13994	1148
14094	1148
14399	1148

5. Now add the kinship table ZZZ\_KIN\_BIOG\_ADDR by clicking on **Show Table** along the Query Tools menu at the top of the screen and select ZZZ\_KIN\_BIOG\_ADDR from the "Show Table" window:

- Create a **link** between the two tables by clicking on c\_personid in ZZZ\_ENTRY\_DATA and dragging it to the c\_personid in ZZZ\_KIN\_BIOG\_ADDR. The query builder may ask you to confirm that you want to select only those pairs of records from the two tables which share the same person IDs.
- From the kinship table, add the following fields:  
 c\_person\_name\_chn (the name of the person identified by c\_personid)  
 c\_node\_id (the ID of the relative)  
 c\_node\_chn (the name of the relative)  
 c\_upstep (the number of generations up in the kinship relation)  
 c\_dwnstep (the number of generations down in the kinship relation)  
 c\_marstep (the number of marriage relations involved in the kinship relation)  
 c\_colstep (the number of brother/sister relations involved in the kinship relation)  
 c\_link\_desc (the English description of the kinship relation)  
 c\_link\_chn (the Chinese description of the kinship relation)
- Set the limit for generations up (c\_upstep) to 2 (i.e., FF, FFB, etc.)  
 Set the limit for generations down (c\_dwnstep) to 0 (i.e, we want to lookonly at ancestors)  
 Set the limit for affines (c\_marstep) to 0

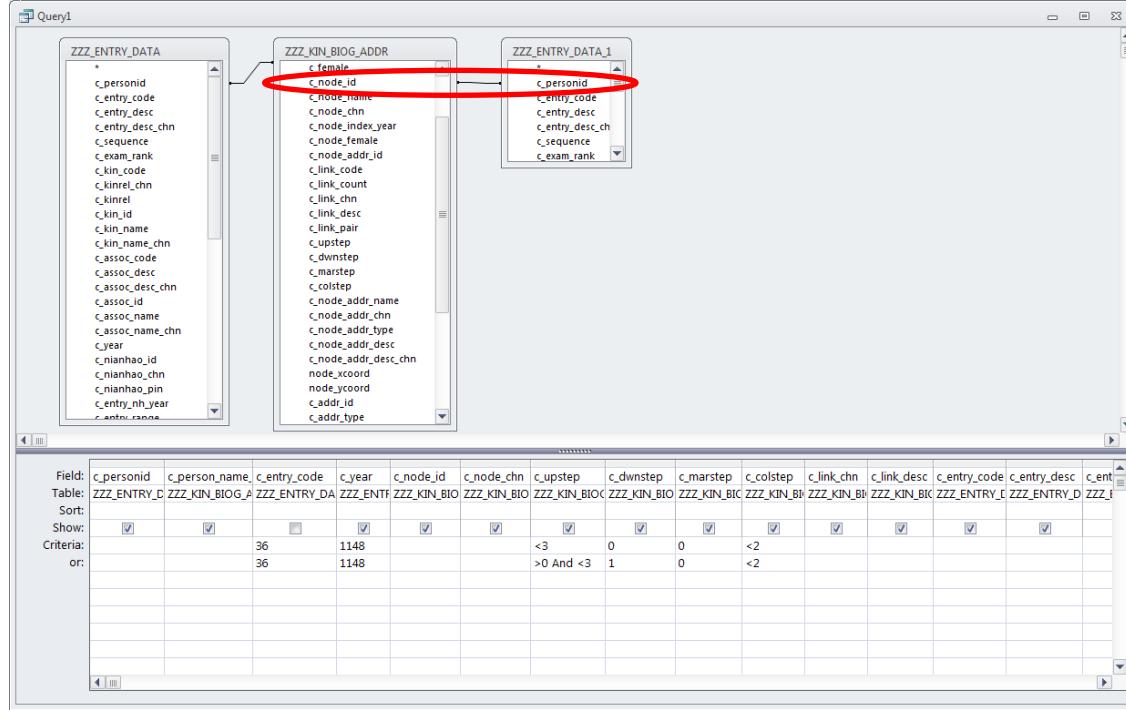
Set the limit for brother/sister (`c_colstep`) to at most 1

- d. Repeat this process, but allow cousins (i.e. FBS or FFBS: 1 down step, at least 1 up)

6. Check the results: There are 319 relatives that meet the criteria

c_personid	c_person_na	c_year	c_node_id	c_node_t	c_upstep	c_dwnstep	c_marstep	c_colstep	c_link_chn	c_link_des
70	陳良弼	1148	23954	陳儀	1	0	0	0	父	F
466	蔣櫟	1148	3233	蔣之美	1	0	0	0	父	F
466	蔣櫟	1148	461	蔣之奇	1	0	0	1	從父;伯叔父	FB
601	方師尹	1148	23968	方勗	1	0	0	0	父	F
667	韓彥直	1148	8050	韓世忠	1	0	0	0	父	F
667	韓彥直	1148	3331	韓彥古	0	0	0	1	弟	B-
1286	陸升之	1148	3630	陸靜之	0	0	0	1	兄	B+
1286	陸升之	1148	13462	陸長民	1	0	0	0	父	F
1286	陸升之	1148	7051	陸幼	2	0	0	0	祖父	FF
1714	鄒櫓	1148	3888	鄒浩	1	0	0	1	從父;伯叔父	FB
1714	鄒櫓	1148	12591	鄒洞	1	0	0	0	父	F
3166	張宗元	1148	10223	張子厚	1	0	0	0	父	F
3317	馮用休	1148	23981	馮伉	1	0	0	0	父	F
3990	王萬修	1148	20097	王萬全	1	1	0	1	從兄弟;堂兄弟	FBS
3990	王萬修	1148	20098	王萬章	1	0	0	0	族兄弟	K(male)
3990	王萬修	1148	23951	王棟	1	0	0	0	父	F
7201	朱江	1148	16698	朱潤	1	1	0	1	從弟	FBS-
7201	朱江	1148	23659	朱信	1	0	0	0	父	F
8139	董德元	1148	22870	董獎	1	0	0	0	父	F
8159	魏師遜	1148	23133	魏師心	0	0	0	1	兄	B+
8159	魏師遜	1148	23132	魏	1	0	0	0	父	F
10131	吳彥夔	1148	23440	吳城	1	0	0	0	父	F
10572	吳邵年	1148	10571	吳表臣	1	0	0	0	父	F
10702	王師愈	1148	10686	王澄	1	0	0	0	父	F
10938	梁南一	1148	10900	梁克家	2	0	0	0	族子	K+1(male)
10938	梁南一	1148	23570	梁杰	1	0	0	0	父	F
11187	傅知新	1148	11181	傅希龍	1	0	0	1	從父;伯叔父	FB
11280	方縉	1148	11273	方深道	1	0	0	0	父	F
11341	薛燦	1148	23531	薛師彌	1	0	0	0	父	F
11358	龔夢良	1148	12537	龔元	0	0	0	1	弟	B-
11358	龔夢良	1148	23685	龔齊晉	1	0	0	0	父	F
11358	龔夢良	1148	11357	龔茂良	0	0	0	1	兄	B+
11416	方簡肅	1148	20287	方綯	1	0	0	0	父	F
11572	張溥	1148	23501	張降	1	0	0	0	父	F
11873	胡師徐	1148	23965	胡世徵	1	0	0	0	父	F

7. Now add a ***second*** version of the ZZZ\_ENTRY\_DATA table and link that table to the ZZZ\_KIN\_BIOG\_ADDR table by making c\_node\_id = c\_personid:

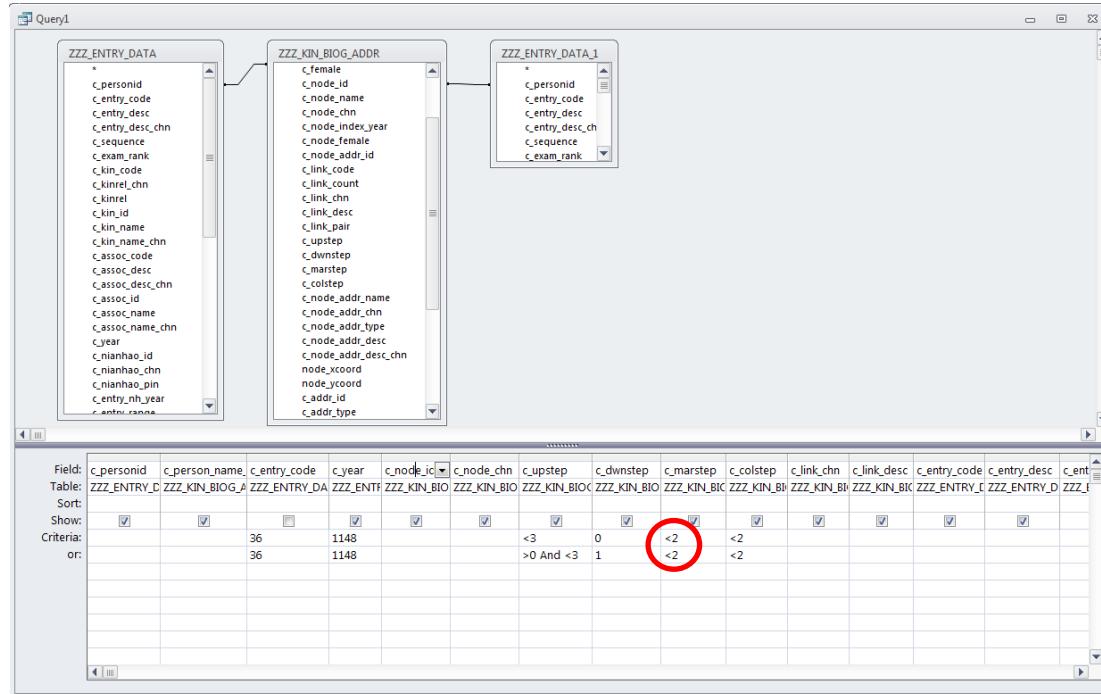


8. Add the two fields c\_entry\_desc and c\_entry\_desc\_chn from ZZZ\_ENTRY\_DATA\_1 (to get the mode of entry of the kin) and check the results:

c_personid	c_person_name	c_year	c_node_id	c_node_chn	c_upstep	c_dwnstep	c_marstep	c_colstep	c_link_chn	c_link_desc	c_entry_desc	c_entry_desc_chn	c_entry_desc	
466	蔣瓈	1148	461	蔣之奇	1	0	0	1	從父;伯叔父	FB	36 examination: ji	科舉:進士(清)		
466	蔣瓈	1148	3233	蔣之美	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
466	蔣瓈	1148	461	蔣之奇	1	0	0	1	從父;伯叔父	FB	248 Decree examin	科舉制舉:督		
3990	王萬修	1148	20097	王萬全	1	1	0	0	1	從兄弟;堂兄弟	FBS	36 examination: ji	科舉:進士(清)	
3990	王萬修	1148	20098	王萬章	1	0	0	0	0	K(male)	36 examination: ji	科舉:進士(清)		
7201	朱江	1148	16698	朱浦	1	1	0	1	從弟	FBS-	36 examination: ji	科舉:進士(清)		
667	韓彥直	1148	8050	韓世忠	1	0	0	0	父	F	77 military merit	軍昌轉補		
1714	鄒樗	1148	3888	鄒浩	1	0	0	1	從父;伯叔父	FB	36 examination: ji	科舉:進士(清)		
1286	陸升之	1148	3630	陸靜之	0	0	0	0	兄	B+	118 yin privilege: g	恩蔭:龍統		
1286	陸升之	1148	13462	陸長民	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
10572	吳邵年	1148	10571	吳表臣	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
10702	王師愈	1148	10686	王登	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
10938	梁南一	1148	10900	梁克家	2	0	0	0	族子	K+1(male)	36 examination: ji	科舉:進士(清)		
11187	傅知新	1148	11181	傅希龍	1	0	0	1	從父;伯叔父	FB	36 examination: ji	科舉:進士(清)		
11280	方鉅	1148	11273	方深道	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
11358	龔夢良	1148	11357	龔茂良	0	0	0	1	兄	B+	36 examination: ji	科舉:進士(清)		
11358	龔夢良	1148	12537	龔元	0	0	0	1	弟	B-	29 examination: ji	科舉:特奏名		
11416	方罷	1148	20287	方綱	1	0	0	0	父	F	37 examination: ji	科舉:進士(/)		
13286	劉安世	1148	13285	劉思	1	0	0	0	父	F	94 honorific title	封贈		
13286	劉安世	1148	13287	劉之淵	1	0	0	0	族兄弟	K(male)	39 examination: ji	科舉:鄉貢舉		
13438	詹允宗	1148	13437	詹林宗	0	0	0	1	弟	B-	36 examination: ji	科舉:進士(清)		
13464	陸光之	1148	13462	陸長民	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
13477	王佐	1148	22224	王公袁	0	0	0	1	弟	B-	36 examination: ji	科舉:進士(清)		
13477	王佐	1148	13476	王彥彥	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
13650	陳大方	1148	13648	陳正肅	1	0	0	0	父	F	14 imperial sum	徵辟		
13951	洪邦直	1148	10156	洪皓	2	0	0	1	從祖;伯叔祖	FFB	36 examination: ji	科舉:進士(清)		
14094	葛部	1148	14093	葛立中	1	0	0	0	父	F	33 examination: ji	科舉:舉進士		
14407	陳舉善	1148	15236	陳嘉善	0	0	0	1	弟	B-	36 examination: ji	科舉:進士(清)		
14420	陳豐	1148	14419	陳贊	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
14819	趙彥恂	1148	364	趙公稱	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
14913	趙像之	1148	14912	趙叔翹	1	0	0	0	父	F	74 grace given to	宗子該恩		
14914	趙闡之	1148	14912	趙叔翹	1	0	0	0	父	F	74 grace given to	宗子該恩		
17050	舒黼	1148	17049	舒下	1	0	0	0	父	F	36 examination: ji	科舉:進士(清)		
17485	鮑慎履	1148	23368	鮑潤卿	0	0	0	1	兄	B+	36 examination: ji	科舉:進士(清)		
19595	李彥穎	1148	22972	李衡衛	0	0	0	1	兄	B+	36 examination: ji	科舉:進士(清)		

85 kin from the 273 initial degree recipients have data on how they entered officialdom

9. Simply adding a 1 to the c\_marstep will allow one to look at affinal relations as well. Using the criterion “<2” means that a c\_marstep of either 0 or 1 in the record will be acceptable:



This produces 6 additional records for a total of 91.

Record: 1 of 91

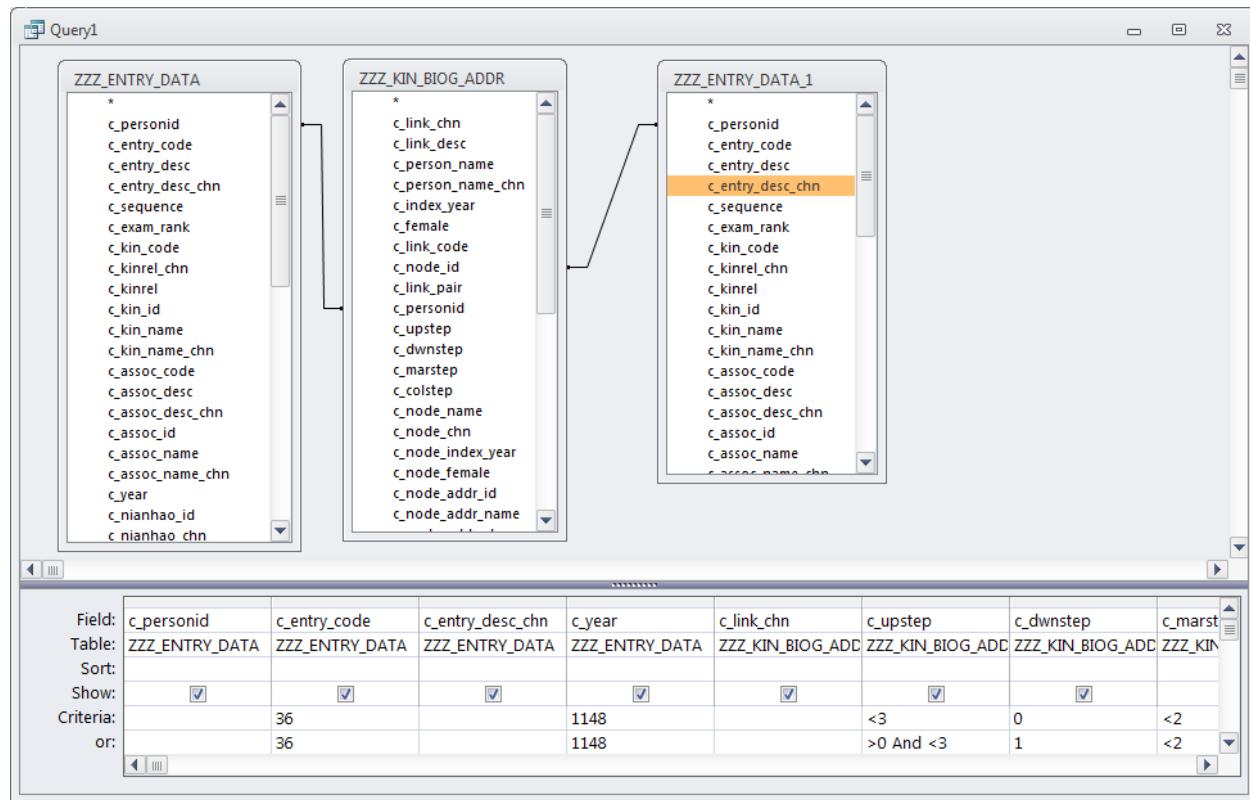
c_personid	c_person_name	c_year	c_node_id	c_node_chn	c_upstep	c_dwnstep	c_marstep	c_colstep	c_link_chn	c_link_desc	c_entry_code	c_entry_desc
466	蔣璫	1148	461	蔣之奇	1	0	0	1	從父;伯叔父	FB	36	examination: ji 科舉:進士(書)
466	蔣璫	1148	3233	蔣之美	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
466	蔣璫	1148	461	蔣之奇	1	0	0	1	從父;伯叔父	FB	248	Decree examin 科舉制舉:督
3990	王萬修	1148	20097	王萬全	1	1	0	1	從兄弟;堂兄弟	FBS	36	examination: ji 科舉:進士(書)
3990	王萬修	1148	20098	王萬章	1	0	0	0	族兄(男)		36	examination: ji 科舉:進士(書)
3990	王萬修	1148	8050	韓世忠	1	0	1	0	岳父	WF	77	military merit 軍昌轉補
7201	朱江	1148	16698	朱淵	1	1	0	1	從弟	FBS-	36	examination: ji 科舉:進士(書)
667	韓彥直	1148	8050	韓世忠	1	0	0	0	父	F	77	military merit 軍昌轉補
1714	鄒櫟	1148	3888	鄒浩	1	0	0	1	從父;伯叔父	FB	36	examination: ji 科舉:進士(書)
1286	陸升之	1148	3630	陸靜之	0	0	0	1	兄	B+	118	yin privilege: g 恩蔭(寵統)
1286	陸升之	1148	13462	陸長民	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
3166	張宗元	1148	7046	劉光世	1	0	1	0	岳父	WF	77	military merit 軍昌轉補
3317	馮用休	1148	8050	韓世忠	1	0	1	0	岳父	WF	77	military merit 軍昌轉補
10572	吳邵年	1148	10571	吳表臣	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
10702	王師愈	1148	10686	王登	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
10938	梁南一	1148	10900	梁克家	2	0	0	0	族子	K+1(male)	36	examination: ji 科舉:進士(書)
11187	傅知新	1148	11181	傅希龍	1	0	0	1	從父;伯叔父	FB	36	examination: ji 科舉:進士(書)
11280	方彊	1148	11273	方深道	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
11358	龔夢良	1148	11357	龔茂良	0	0	0	1	兄	B+	36	examination: ji 科舉:進士(書)
11358	龔夢良	1148	12537	龔元	0	0	0	1	弟	B-	29	examination: ji 科舉:特奏名
11416	方簡興	1148	20287	方綱	1	0	0	0	父	F	37	examination: ji 科舉:進士(書)
13230	毛惠直	1148	15903	羅綏	1	0	1	0	岳父	WF	39	examination: ji 科舉:鄉貢舉
13286	劉安世	1148	13285	劉思	1	0	0	0	父	F	94	honorable title 封贈
13286	劉安世	1148	13287	劉之湧	1	0	0	0	族兄弟	K(male)	39	examination: ji 科舉:鄉貢舉
13438	詹允宗	1148	13437	詹林宗	0	0	0	1	弟	B-	36	examination: ji 科舉:進士(書)
13464	陸光之	1148	13462	陸長民	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
13477	王佐	1148	22224	王公袞	0	0	0	1	弟	B-	36	examination: ji 科舉:進士(書)
13477	王佐	1148	13476	王俊彥	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
13650	陳大方	1148	13648	陳正彙	1	0	0	0	父	F	14	Imperial sum 雜辟
13951	洪邦直	1148	10156	洪皓	2	0	0	1	從祖;伯叔祖	FFB	36	examination: ji 科舉:進士(書)
14094	葛邵	1148	14093	葛立中	1	0	0	0	父	F	33	examination: f 科舉:舉進士
14407	陳舉善	1148	15236	陳嘉善	0	0	0	1	弟	B-	36	examination: ji 科舉:進士(書)
14420	陳豐	1148	14419	陳贊	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
14819	趙彥恂	1148	364	趙公稱	1	0	0	0	父	F	36	examination: ji 科舉:進士(書)
14819	趙彥恂	1148	14912	趙叔饗	1	0	0	0	父	F	74	grace given to 宗子該恩

## II. Some Useful Additional Procedures for Queries

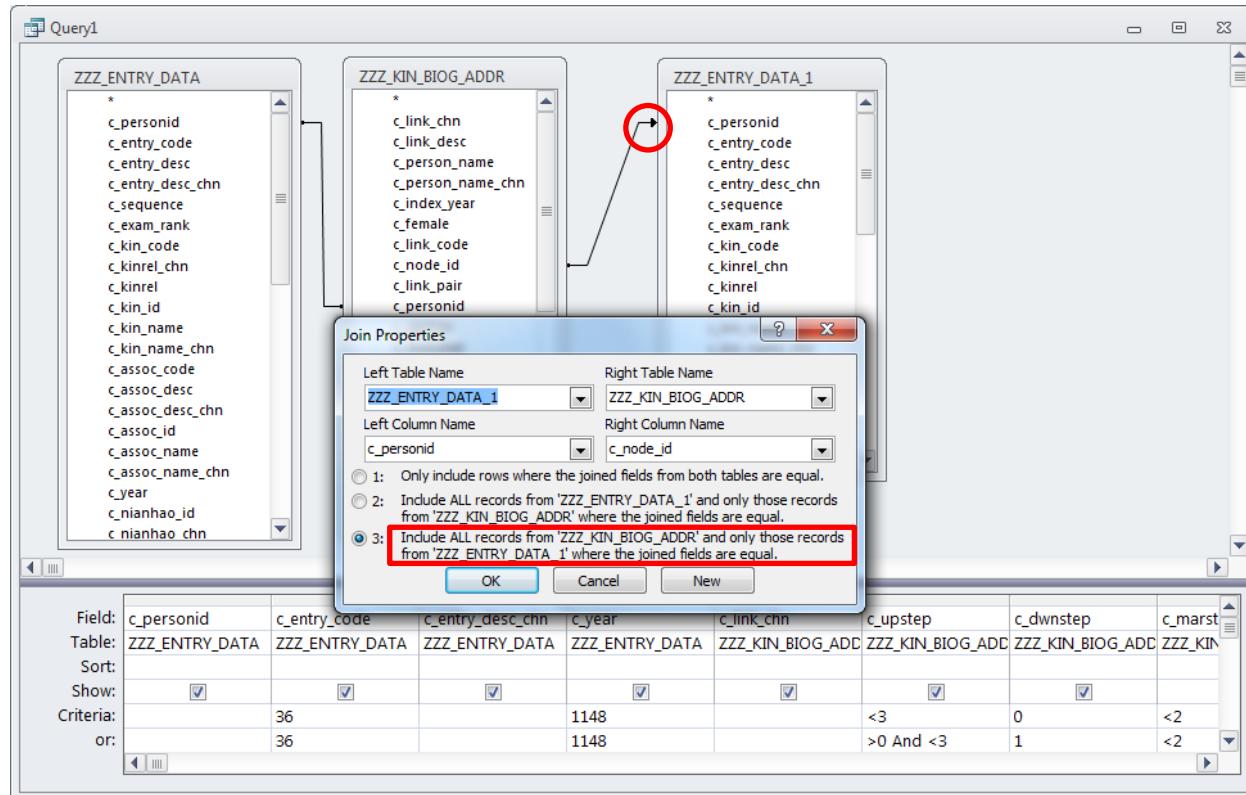
### A. Null Information can be Useful

In the above query, we dealt only with those relatives for whom information about their mode of entering government service was known. Suppose, however, that we wanted a list of all the relatives as well as the available information about their mode of entering service. Such a list helps clarify the percentage for whom we have data.

Our initial design looked like:



We need to change the way Access selects its records. To do this we need to modify the link between the **entry data** for the kin and the **kin** themselves, which we created by equating **c\_node\_id** (i.e., the ID for the kin) in **ZZZ\_KIN\_BIOG\_ADDR** with **c\_personid** in **ZZZ\_ENTRY\_DATA\_1**, the second copy of **ZZZ\_ENTRY\_DATA** you added to the query. To modify that link, double-click on the line connecting **c\_node\_id** and **c\_personid**. This will open a dialog box:



Select option 3 and click OK. Note the arrow pointing to c\_personid. This arrow indicates a “right join” in the language of SQL.

When we execute the query, we get records for all the initial 319 kin.

As you become familiar with the Query Builder, you should begin to look at the SQL code to see how Access structures the queries because sometimes it is easier to directly modify the SQL code, as when one adds “distinct” to the intial select statement. The SQL code for a connection **where there are equal values in both linked fields** of the tables uses an **“inner join.”**

```

1. FROM ZZZ_ENTRY_DATA AS ZZZ_ENTRY_DATA_1 INNER JOIN
2.      ZZZ_KIN_BIOG_ADDR INNER JOIN ZZZ_ENTRY_DATA
3.          ON ZZZ_KIN_BIOG_ADDR.c_personid = ZZZ_ENTRY_DATA.c_personid
4.      ) ON ZZZ_ENTRY_DATA_1.c_personid = ZZZ_KIN_BIOG_ADDR.c_node_id;

```

That is, the inner block (lines 2 and 3) one picks all the records in ZZZ\_KIN\_BIOG\_ADDR and ZZZ\_ENTRY\_DATA where the IDs match, and the outer block (line 1 and 4) matches the result of the inner block with records from ZZZ\_ENTRY\_DATA\_1 where the c\_node\_id of ZZZ\_KIN\_BIOG\_ADDR matches the person ID in ZZZ\_ENTRY\_DATA\_1.

In contrast, in order to allow us to see **ALL the records for kin in ZZZ\_KIN\_BIOG\_ADDR and only those records in ZZZ\_ENTRY\_DATA\_1 where the c\_node\_id matches the c\_personid**, we use a “right join” in the outer block. The SQL code for this linkage is:

```

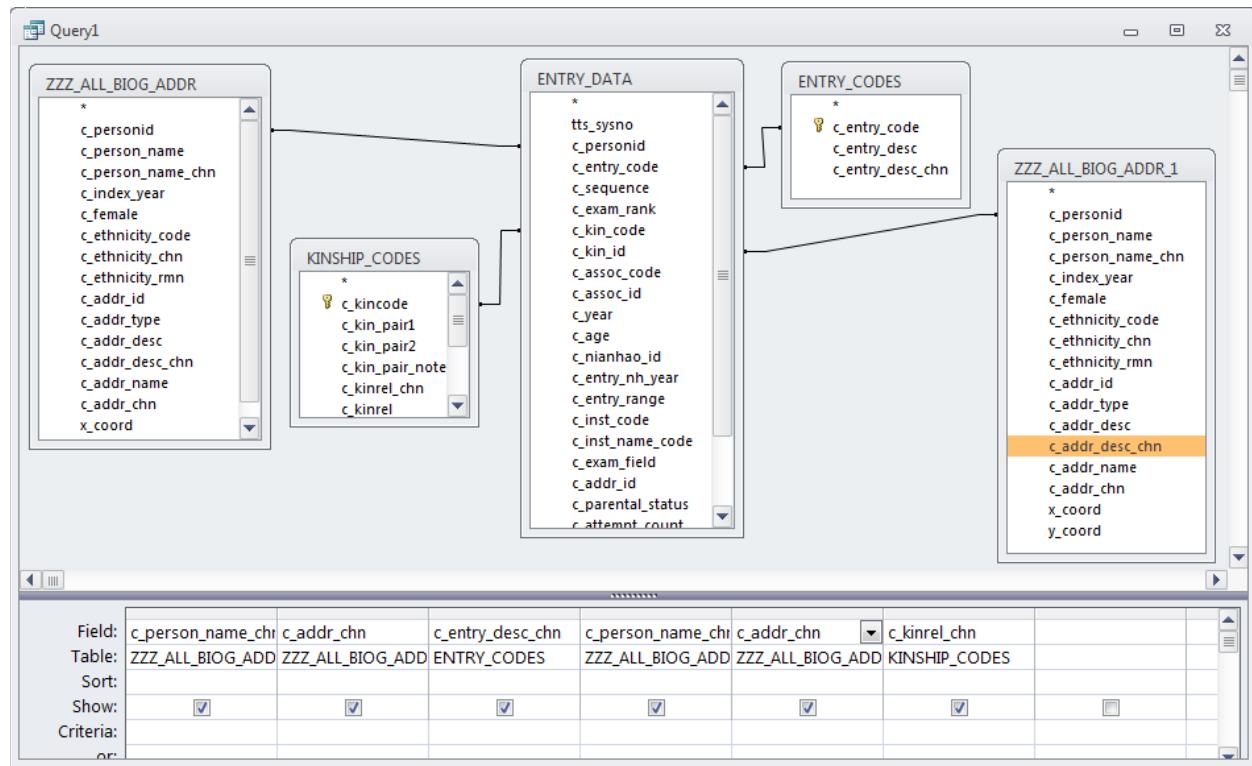
1. FROM ZZZ_ENTRY_DATA AS ZZZ_ENTRY_DATA_1 RIGHT JOIN
2.      ZZZ_KIN_BIOG_ADDR INNER JOIN ZZZ_ENTRY_DATA
3.          ON ZZZ_KIN_BIOG_ADDR.c_personid = ZZZ_ENTRY_DATA.c_personid
4.      ON ZZZ_ENTRY_DATA_1.c_personid = ZZZ_KIN_BIOG_ADDR.c_node_id;

```

## B. How to Reference Information about Many People who are Parts of a Single Table

Although the “denormalized” tables save you work in finding information about people who are parts of those tables, they do not include all possible information. More generally, you will be able to create more flexible queries if you are not restricted to the set of denormalized tables that CBDB provides.

Let's consider a simple problem: suppose you wanted place associations for the kin who played a role in entry into government service. The table that has our BEST association of people with place (as opposed to BIOG\_ADDR\_DATA, which records all associations) is ZZZ\_ALL\_BIOG\_ADDR. Let us begin with ENTRY\_DATA rather than ZZZ\_ENTRY\_DATA:

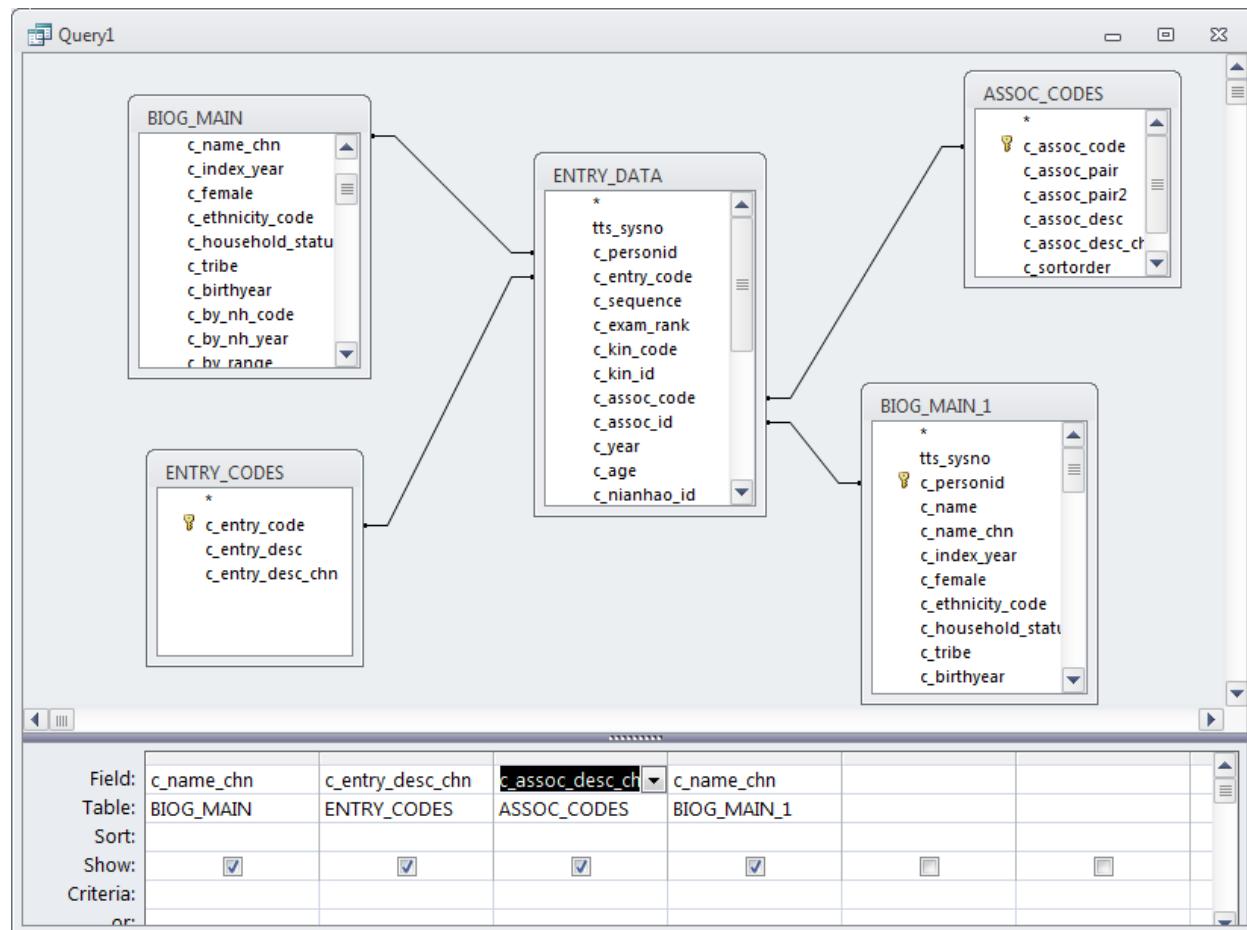


Note that since we are using ENTRY\_DATA, we need to add KINSHIP\_CODES to get the kinship description and ENTRY\_CODES to get the mode of entry. As described earlier, we use the second reference to ZZZ\_ALL\_BIOG\_ADDR to get the name of the kin as well as the name of the place.

In one final example of using two references to a table, we want to take a first step in looking at the people who are NOT kin who assisted in someone's entry into government service. Here we use entirely basic tables.

Note that we use two references to BIOG\_MAIN and need to add both ENTRY\_CODES and

ASSOC\_CODES to understand how the person entered service and what the relation between him and the associate was:



As it turns out, CBDB does not yet have much data on this mode of entering service:

The screenshot shows the Microsoft Access Query1 window displaying the results of the query. The results are as follows:

BIOG_MAIN	c_entry_desc_chn	c_assoc_des	BIOG_MAIN
吳師仁	薦舉(保任)	推薦	陳襄
胡瑗	薦舉(保任)	被薦	范仲淹
扈彥珂	恩蔭: 遺表蔭補	恩主是Y	王建立
杜復	薦舉(保任)	未詳	史浩
湯楷	薦舉(保任)	被Y推薦	石抹宜孫
何瑛	科舉: 制舉(寵統)	未詳	何瑛

### C. The TableFields Table

For getting information on additional people involved in various types of social interactions, you need to know which fields in a table refer to IDs for people. When in doubt, you can open the TableFields table from the list of tables on the left of the main Access interface and look for the fields in the table you want to explore. Those that have “BIOG\_MAIN” in the “foreign key” column and “c\_personid” in the ForeignKeyBase column a refer to people.<sup>1</sup> For example, in ASSOC\_DATA, we have:

AccessTblNm	AccessFldNm	IndexOnField	DataFormat	NULL_allowed	ForeignKey	ForeignKeyBase
assoc_data	c_addr_id		Long	<input checked="" type="checkbox"/>	ADDR_CODES	c_addr_id
assoc_data	c_assoc_claimer_id		Long	<input checked="" type="checkbox"/>	BIOG_MAIN	c_personid
assoc_data	c_assoc_code	Primary	Long	<input type="checkbox"/>	ASSOC_CODES	c_assoc_code
assoc_data	c_assoc_count		Integer	<input checked="" type="checkbox"/>		
assoc_data	c_assoc_day		Integer	<input checked="" type="checkbox"/>		
assoc_data	c_assoc_day_gz		Integer	<input checked="" type="checkbox"/>	GANZHI_CODES	c_ganzhi_code
assoc_data	c_assoc_id	Primary	Long	<input type="checkbox"/>	BIOG_MAIN	c_personid
assoc_data	c_assoc_intercalary		Binary	<input checked="" type="checkbox"/>		
assoc_data	c_assoc_kin_code	Primary	Long	<input type="checkbox"/>		
assoc_data	c_assoc_kin_id	Primary	Long	<input type="checkbox"/>	BIOG_MAIN	c_personid
assoc_data	c_assoc_month		Integer	<input checked="" type="checkbox"/>		
assoc_data	c_assoc_nh_code		Integer	<input checked="" type="checkbox"/>	nian_hao	c_nianhao_id
assoc_data	c_assoc_nh_year		Integer	<input checked="" type="checkbox"/>		
assoc_data	c_assoc_range		Integer	<input checked="" type="checkbox"/>	year_range_codes	c_range_code
assoc_data	c_assoc_year		Integer	<input checked="" type="checkbox"/>		
assoc_data	c_inst_code		Integer	<input type="checkbox"/>	SOCIAL_INSTITUTION_CODES	c_inst_code
assoc_data	c_inst_name_code		Integer	<input type="checkbox"/>	SOCIAL_INSTITUTION_NAME_CODE	c_inst_name_code
assoc_data	c_kin_code	Primary	Long	<input type="checkbox"/>	KINSHIP_CODES	c_kincode
assoc_data	c_kin_id	Primary	Long	<input type="checkbox"/>	BIOG_MAIN	c_personid
assoc_data	c_litgenre_code		Integer	<input checked="" type="checkbox"/>	literarygenre_codes	c_lit_genre_code
assoc_data	c_notes		Memo	<input checked="" type="checkbox"/>		
assoc_data	c_occasion_code		Integer	<input checked="" type="checkbox"/>	OCCASION_CODES	c_occasion_code
assoc_data	c_pages		Text	<input checked="" type="checkbox"/>		
assoc_data	c_personid	Primary	Long	<input type="checkbox"/>	BIOG_MAIN	c_personid
assoc_data	c_sequence		Integer	<input checked="" type="checkbox"/>		
assoc_data	c_source		Long	<input type="checkbox"/>	TEXT_CODES	c_textid
assoc_data	c_text_title		Text	<input type="checkbox"/>		
assoc_data	c_topic_code		Integer	<input checked="" type="checkbox"/>	SCHOLARLYTOPIC_CODES	c_topic_code

Among all these, the following are IDs of people:

- c\_assoc\_claimer\_id (the ID of the person claiming the existence of the association)
- c\_assoc\_id (the ID of the associate)
- c\_assoc\_kin\_id (the ID of the kin of the associate through who the association exists, if any)
- c\_kin\_id (the ID of the kin of the main person in the record through who the association exists, if any)
- c\_personid (the person whom the record is about)

<sup>1</sup> In a normalized database, “foreign key” simply refers to those fields that use the IDs defined (as primary keys) in other tables.

# Appendix: Installing CBDB Files

## A. Installing the MS Access Files

In order to keep the database files within the two gigabyte limit for Microsoft Access files, CBDB is divided into two files: a “Base” file with the most basic tables, and a “User” file with the user interface and the “denormalized” files used for searches. The User file draws on the tables in the Base file as “linked tables.” When you install the CBDB files, you will need to reestablish the link between the User and Base files.

1. Create a folder into which to extract the two files that you have downloaded from the CBDB website. Extract the files.
2. Double-click on the User file to open it in Microsoft Access. You will see:

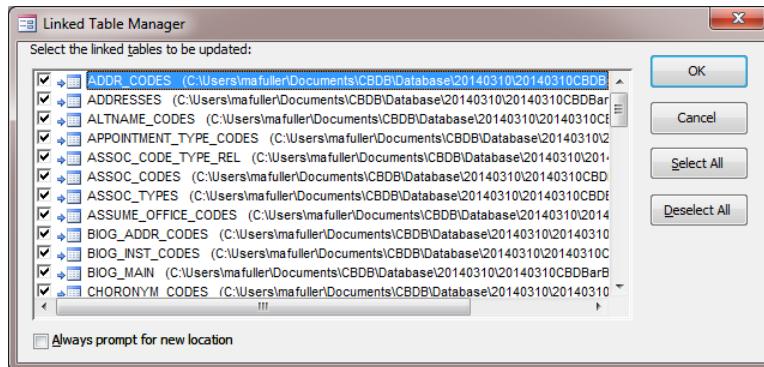


Note the arrows next to most of the tables in the list on the left side of the screen. The arrow indicates that the table is a *linked* table from the Base file.

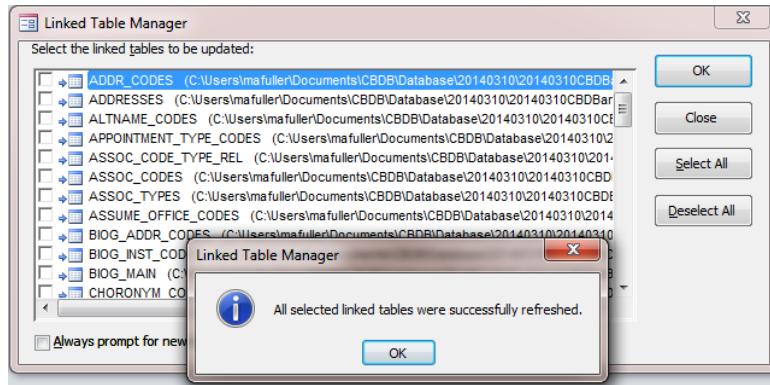
3. Right-click on any linked table, and you will see a list of optional actions:



- Click on “Linked Table Manager.” You will see a list of all the linked tables. Access gives the source file for each table in parentheses next to it.



- Click on “Select All” and then on “OK.” This will open a standard “Open File” window. Go to the folder in which you installed the CBDB files and click on the Base file.
- Click on “Open File.” Access will open a message window to tell you that the links between the two tables have been refreshed:



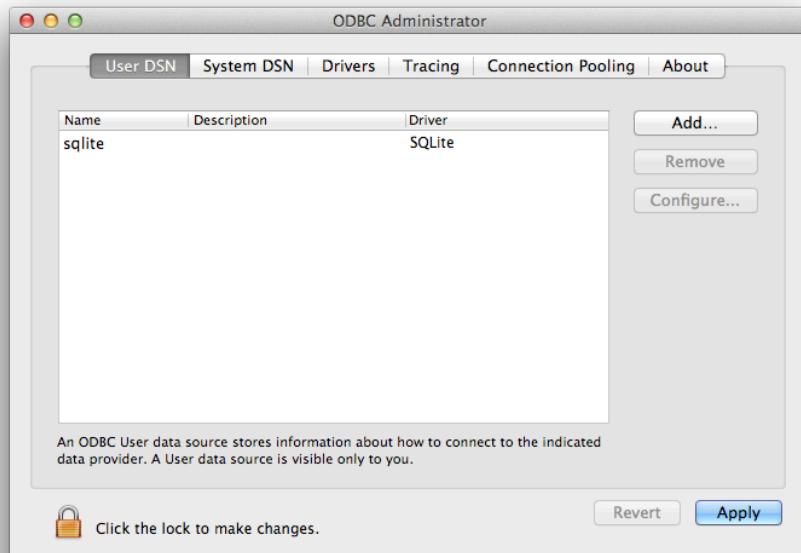
7. The User file is now ready to use.

## B. Installing the SQLite CBCB database on a Macintosh

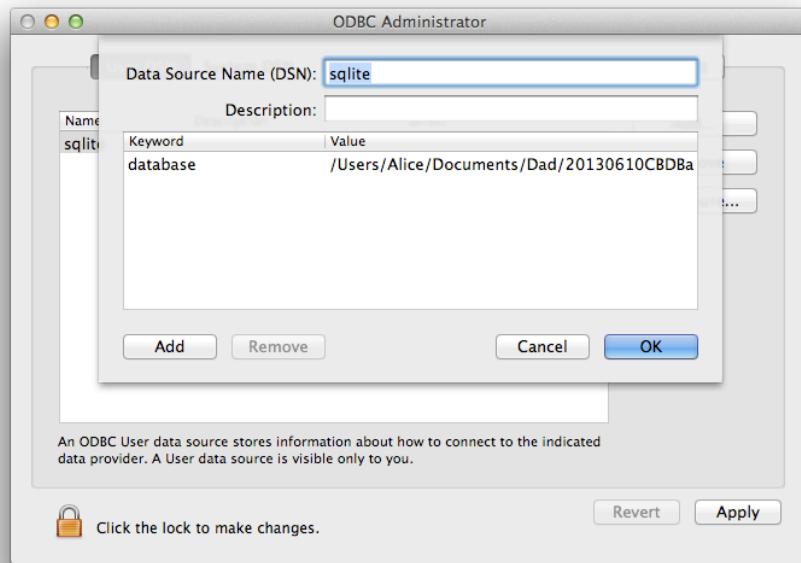
For Apple users (or Linux users, who probably do not need these instructions), there is a stand-alone version of the CBDB database using the SQLite format.

For any database file to be used in a Macintosh system, the operating system needs a connector between the file and the standard ODBC (Open Database Connectivity) interface. In order to make this connection, you will need the Mac ODBC Administrator and the ODBC driver for SQLite. (You may need to download these from the web, or you may decide to leave these steps to your information technology specialist, if you have access to one. The ODBC driver for SQLite can be downloaded from <http://www.ch-werner.de/sqliteodbc/> ).

1. Install the Macintosh ODBC Administrator and the driver for SQLite.
2. In “Finder,” go to Utilities and open the ODBC Administrator.
3. Go to User DSN and add “CBCBFull” as an SQLite database:



4. Click on “Configure” to set up the connector:



5. Add the keyword “database” and use the full path for the database file as the “value.”
6. Click on OK. The window will close. Then click on Apply.
7. The SQLite version of CBDB should be ready to use with OpenOffice or whatever software interface you prefer.