



T'ang Studies Society Workshop on the China Biographical Database

Harvard University
August 22-23, 2013

Sponsored by the T'ang Studies Society



Session Three:

The Design of Research Projects using Relational Database Entities and CBDB

Michael A. Fuller



In your projects, you primarily will need to think about the design of the specific “thing” you want to link to CBDB and explore.

For example, I have created a small sample database that can be used to record information on texts in a 文集.

Let's download and open it.



The workhorse (“PeopleInTexts”) table captures “People in Texts” and has the following fields:

Author	Text (255)
PersonInText	Text (255)
TextTitle	Text (255)
TextType	Text (255)
TextYear	Number (Integer)
TextSource	Text (255)
SourcePage	Text (255)
CitationNotes	Memo



And we have some initial data:

Author	PersonIn Text	Person Role	TextTitle	Text Type	Text Year	TextSource	SourcePage	CitationNotes
權德輿	劉昌	Subject	大唐四鎮北庭行軍兼涇原等州節度度支營田等使開府儀同三司檢校尚書右僕射使持節涇州諸軍事涇州刺史兼御史大夫上柱國南川郡王劉公紀功碑銘	碑銘		權德輿詩文集	1.12.189-93	
權德輿	張孝忠	Subject	唐故義武軍節度使營田易定等州觀察處置使開府儀同三司檢校司空同中書門下平章事范陽郡王贈太師貞武張公遺愛碑銘	碑銘		權德輿詩文集	1.11.183-88	
權德輿	杜亞	Recipient	與睦州杜給事書	書		權德輿詩文集	2.42.637-38	建中二年十二月九日
權德輿	杜佑	Subject	大唐銀青光祿大夫檢校司徒同中書門下平章事太清宮及度支諸道鹽鐵轉運等使崇文館大學士上柱國岐國公杜公淮南遺愛碑銘	碑銘		權德輿詩文集	1.11.178-82	
權德輿	杜佑	Subject	祭故杜岐公文	祭文		權德輿詩文集	2.49.777-78	
權德輿	馬燧	Subject	故司徒兼侍中上柱國北平郡王贈太傅馬公行狀	行狀		權德輿詩文集	1.19.296-307	



Having collected data, one needs to integrate it with CBDB data in order to avoid repeating effort.

The best way to do it, at least initially, is to copy CBDB tables into your database.

You will want:

People	ZZZ_BIOG_MAIN
Entry Data	ZZZ_ENTRY_DATA
Place Data	ZZZ_BIOG_ADDR_DATA
Kinship Data	ZZZ_KIN_BIOG_ADDR
Association Data	ZZZ_NONKIN_BIOG_ADDR
Posting Data	ZZZ_POSTED_TO_OFFICE_DATA
	ZZZ_POSTED_TO_ADDR_DATA
Text Data (?)	ZZZ_TEXT_DATA



Also, add some code tables (for convenient look-up):

Address Hierarchy	ZZZ_BELONGS_TO
Entry Codes	ENTRY_CODES
	ENTRY_TYPES
	ENTRY_CODE_TYPE_REL
Association Codes	ASSOC_CODES
	ASSOC_TYPES
	ASSOC_CODE_TYPE_REL
Place Association Codes	BIOG_ADDR_CODES
Kinship Codes	KINSHIP_CODES
Dynasties	DYNASTIES
年號	NIAN_HAO
Office Codes	OFFICE_CODES
	OFFICE_TYPE_TREE
	OFFICE_CODE_TYPE_REL



Now that you have added the CBDB tables, you'll need to link your table to them.

First, you're going to need a list of all people in your table:

Open the Query Builder

Add your table

Select the "PersonInText" field

Run the query: note that you have duplicates.

Go into SQL View

add "distinct" after "SELECT"

add "as pname" after PersonInText in the SELECT list

Rerun the query

Change the query to a "Make Table" (table = PNames1)

Close and Save the query as "My People1 Query"

Run



Open the Query Builder again

Add your table

Select the “Author” field

Go into SQL View

add “distinct” after “SELECT”

add “as pname” after Author in the SELECT list

Change the query to a “Make Table” (table = PNames2)

Close and Save the query as “My People2 Query”

Run

Copy all the records from PNames2 into PName1

Open the Query Builder again

Add PName1

Select the “pname” field

Go into SQL View

add “distinct” after “SELECT”

Close and Save the query as “My People Query”



Open the Query Builder again

Add ZZZ_BIOG_MAIN and “My People Query”

Drag the “pname” field to c_name_chn in ZZZ_BIOG_MAIN

Double-click on the link and select “All records in ‘My People Query’”

Add “PersonInText” and c_personid, c_index_year

Run the query

Change the query to a “Make Table” (table = PeopleID)

Close and save as PeopleID Query

Run the query

Open the new table PeopleID

Delete the irrelevant records

IF YOU NEED TO ADD PEOPLE TO ZZZ_BIOG_MAIN

oh well, you have to add them: give them numbers starting with 300000 and let us know, so we can add them and report their CBDB system IDs to you.



Step Two (if you wish):

Normalize the data by breaking the PeopleInTexts table into five tables:

1. PeopleID (you just created)
2. TextID
3. RoleID
4. GenreID
5. PeopleTextData (swaps most of the text for codes)



And this query:

Postings Query

The query builder shows the following table structures:

- ZZZ_POSTED_TO_ADDR_DATA**:
 - c_posting_id
 - c_personid
 - c_office_id
 - c_addr_id
 - c_addr_name
 - c_addr_chn
- PeopleID**:
 - pname
 - c_personid (Primary Key)
 - c_index_year
- ZZZ_POSTED_TO_OFFICE_DATA**:
 - c_personid
 - c_posting_id
 - c_office_id
 - c_office_pinyin
 - c_office_chn
 - c_sequence

The query grid below the tables is as follows:

Field:	pname	c_office_chn	c_addr_chn					
Table:	PeopleID	ZZZ_POSTED_TO_O	ZZZ_POSTED_TO_AI					
Sort:								
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Criteria:								
or:								



At this point, I'll get out of your way and let you start to develop a model for your project. If you have questions, I'll be circulating in the room.