

Project Quincy, Annotated Diagram individuals http://projectquincy.rubyforge.org Lead Developer: Jean Bauer Records basic information about a biographical specific person, including a postings standardized name for authority work footnotes and the state they are most organizational occupations correspondence associated with. relationships Records what roles a person held places Joins two individuals in a specific id primary key over his or her lifetime, including occupation titles relationship type for a given period string name military ranks, professions, political Lists the roles (or occupations) a person of time. appointments, marital status, etc. sex string can hold over his or her lifetime. birth date date id primary key death date date individual id 1 foreign key id primary key id primary key state id foreign key individual id 2 foreign key individual id foreign key name string notes text relationship type id foreign key occupation title id foreign key foreign key occupation_type_id created at datetime start year integer start year integer notes text datetime updated at end year integer end year integer datetime created at sort name string notes text notes text updated at datetime boolean american datetime created at datetime created at boolean birth day known updated at datetime updated at datetime birth month known boolean birth year known boolean occupation types death_day_known boolean Groups occupation titles for relationship types death month known boolean easier searching. One-way designation from death year known boolean individual 1 to individual 2. id primary key For example, John Adams to John Quincy Adams is a name string residences parent relationship type, notes text assignments whereas John Quincy Adams datetime created at (postings to John Adams is a child updated at datetime module) relationship. residence types **letters** id primary key (correspondence name string module) organizations locations text notes validations (organizational (places created at datetime (footnotes module) module) updated at datetime module)

Project Quincy, Annotated Diagram http://projectquincy.rubyforge.org **Lead Developer: Jean Bauer** biographical postings footnotes organizational correspondence places empires in_empires continents states in_states **letters** (correspondence module) organizations (organizational module) assignments (postings module) **locations** in_regions regions

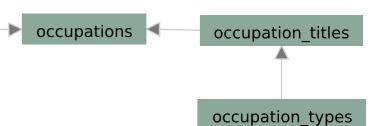
individuals

Records basic information about a specific person, including a standardized name for authority work and the state they are most associated with.

id	primary key
name	string
sex	string
birth_date	date
death_date	date
state_id	foreign key
notes	text
created_at	datetime
updated_at	datetime
sort_name	string
american	boolean
birth_day_known	boolean
birth_month_known	boolean
birth_year_known	boolean
death_day_known	boolean
death_month_known	boolean
death_year_known	boolean

validations (footnotes module)





residences

Where people lived over the course of their lives, including where they were born and where they died.

id	primary key
individual_id	foreign key
location_id	foreign key
residence_type_id	foreign key
start_year	integer
end_year	integer
birth_place	boolean
death_place	boolean
notes	text
created_at	datetime
updated_at	datetime

residence_types

Increased specificity (renting, owning, if the person lived there year round or only part time, etc.)

id	primary key
name	string
temporary	boolean
notes	text
created_at	datetime
updated_at	datetime

Project Quincy, Annotated Diagram http://projectquincy.rubyforge.org Lead Developer: Jean Bauer biographical

postings footnotes

organizational correspondence

places

validations (footnotes module)

continents

A list of the five inhabitable continents: North America, South America, Europe, Asia, and Africa.

idprimary keynamestringcreated_atdatetimeupdated_atdatetime

empires

Groups of states. For example: England, Scotland, Ireland, and Canada are all in the British Empire or the United States of America is a collection of states (Massachusetts, Pennsylvania, etc.).

idprimary keynamestringnotestextcreated_atdatetimeupdated_atdatetime

individuals (biographical module)

states

The largest continuous unit of local administration. This could be a country, province, territory, or a state in the American Union. States can be grouped into Empires.

idprimary keynamestringcontinent_idforeign keynotestextcreated_atdatetimeupdated_atdatetime

in empires

Allows states to be included in multiple empires over time.

primary key id foreign key state id empire id foreign key start year integer end year integer notes text created at datetime updated at datetime

in_states

Allows locations to be included in multiple states over time.

id primary key location id foreign key state id foreign key start year integer end year integer notes text created at datetime updated at datetime assignments (postings module)

letters

module)

(correspondence)

locations

organizations

(organizational

module)

A single place, whether it is a town, city, or address. It has a latitude and longitude value for plotting on a map.

idprimary keynamestringlatfloatlongfloatnotestextcreated_atdatetimeupdated_atdatetime

regions

Any grouping of locations that does not fall within standard political boundaries (i.e. states). Examples include: the Old Southwest, the Mediterranean Basin, the Ohio River Valley, etc.

idprimary keynamestringnotestextcreated_atdatetimeupdated_atdatetime

in_regions

Allows locations to be included in multiple regions at once.

id primary key
location_id foreign key
region_id foreign key
notes text
created_at datetime
updated_at datetime

Project Quincy, Annotated Diagram http://projectquincy.rubyforge.org Lead Developer: Jean Bauer biographical postings

footnotes

organizational correspondence

places

assignment_titles

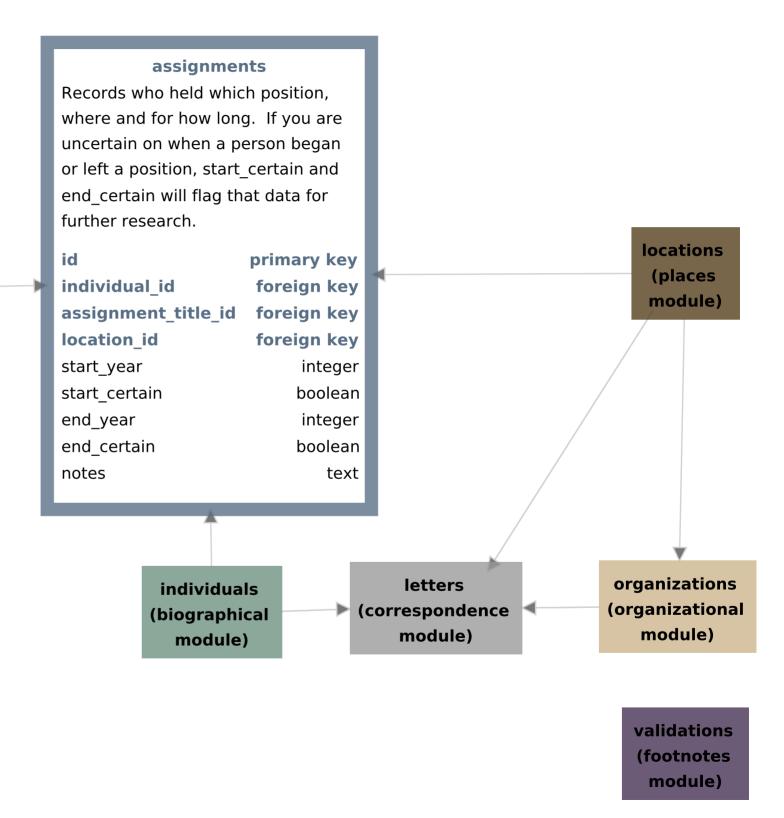
A list of all the different assignments held by individuals.

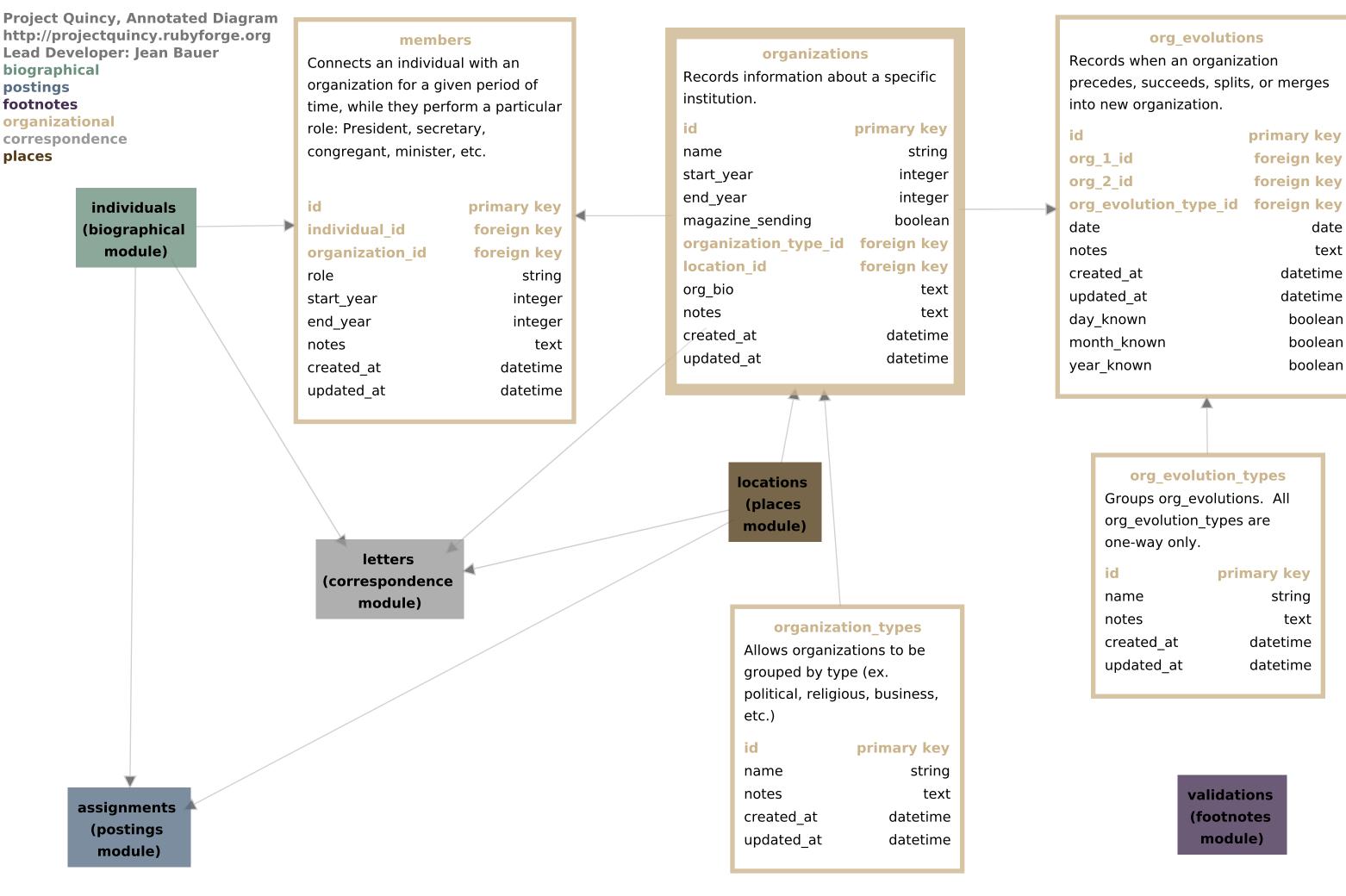
idprimary keynamestringassignment_type_idforeign keytemporarybooleancommissionedbooleannotestextcreated_atdatetimeupdated atdatetime

assignment_types

Groups the possible assignments for ease of searching.

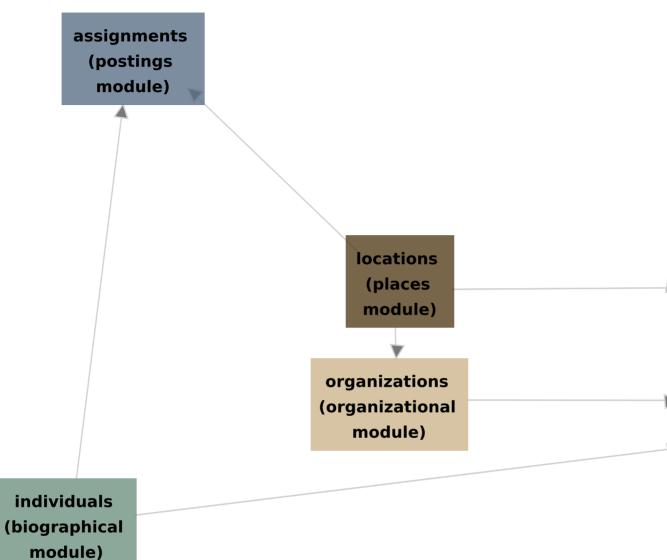
idprimary keynamestringnotestextcreated_atdatetimeupdated_atdatetime





Project Quincy, Annotated Diagram http://projectquincy.rubyforge.org Lead Developer: Jean Bauer biographical postings footnotes organizational correspondence places

validations (footnotes module)



letters

Metadata about a letter: sender and recipient (individual or group), location to and from, dates (sent, received, archived). The database does not store text or images of letters.

id	primary key
from_individual_id	foreign key
from_organization_id	foreign key
from_location_id	foreign key
to_individual_id	foreign key
to_organization_id	foreign key
to_location_id	foreign key
circular	boolean
date_sent	date
date_received	date
date_docketed	date
notes	text
created_at	datetime
updated_at	datetime
title	string
sent_day_known	boolean
sent_month_known	boolean
sent_year_known	boolean
received_day_known	boolean
received_month_known	boolean
received_year_known	boolean
docketed_day_known	boolean
docketed_month_known	boolean
docketed_year_known	boolean

Project Quincy, Annotated Diagram http://projectquincy.rubyforge.org Lead Developer: Jean Bauer biographical postings footnotes organizational correspondence

places

bibliographies

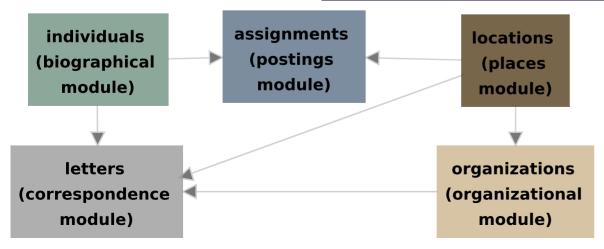
A list of works, including a bibliographic citation allowing the user to double check all information in the database.

id primary key
entry text
notes text
created_at datetime
updated_at datetime

citations

Gives the location of the data within a given source.

primary key id string title bibliography id foreign key string pages canonic url string notes text created at datetime updated at datetime



validations

Validations is a table unlike any other in the database; it does not directly link to any of the records in the database. Instead, it records the type of validatable entity (model name) and that entity's unique identifier. This way, the validations table can "link" to as many (or as few) tables in the database as you require. The validations also table records the user who entered the data, and the source from which the data is validated. The "supports" field will record whether a given citation supports on contradicts the data in auestion.

id primary key validatable type string validatable id integer supports boolean user id foreign key citation id foreign key text notes created at datetime datetime updated at

users

All the registered users in the database. This table can be expanded to support many authentication strategies.

idprimary keynamestringnotestextcreated_atdatetimeupdated_atdatetime