Short Essay

I've selected the Red Hook Alumni Association (RHAA) database for my discussion. The RHAA database has been a work in progress for nearly ten years. The project brings together "data" from many different sources ranging from class lists, graduation brochures, yearbooks, reunion mailings, and early RHAA spreadsheet records. The raw data was extracted, transformed to standardized formats, and loaded into the emergent database. Through queries and reports, the data may now be presented as "information."

The database currently has records for 7,905 individuals. This total varied widely during the early stages of transforming and normalizing the data: literally thousands of redundant were identified in the first passes; several hundred missing records were identified from analog sources and added to the database; and hundreds of records continue to be added each year.

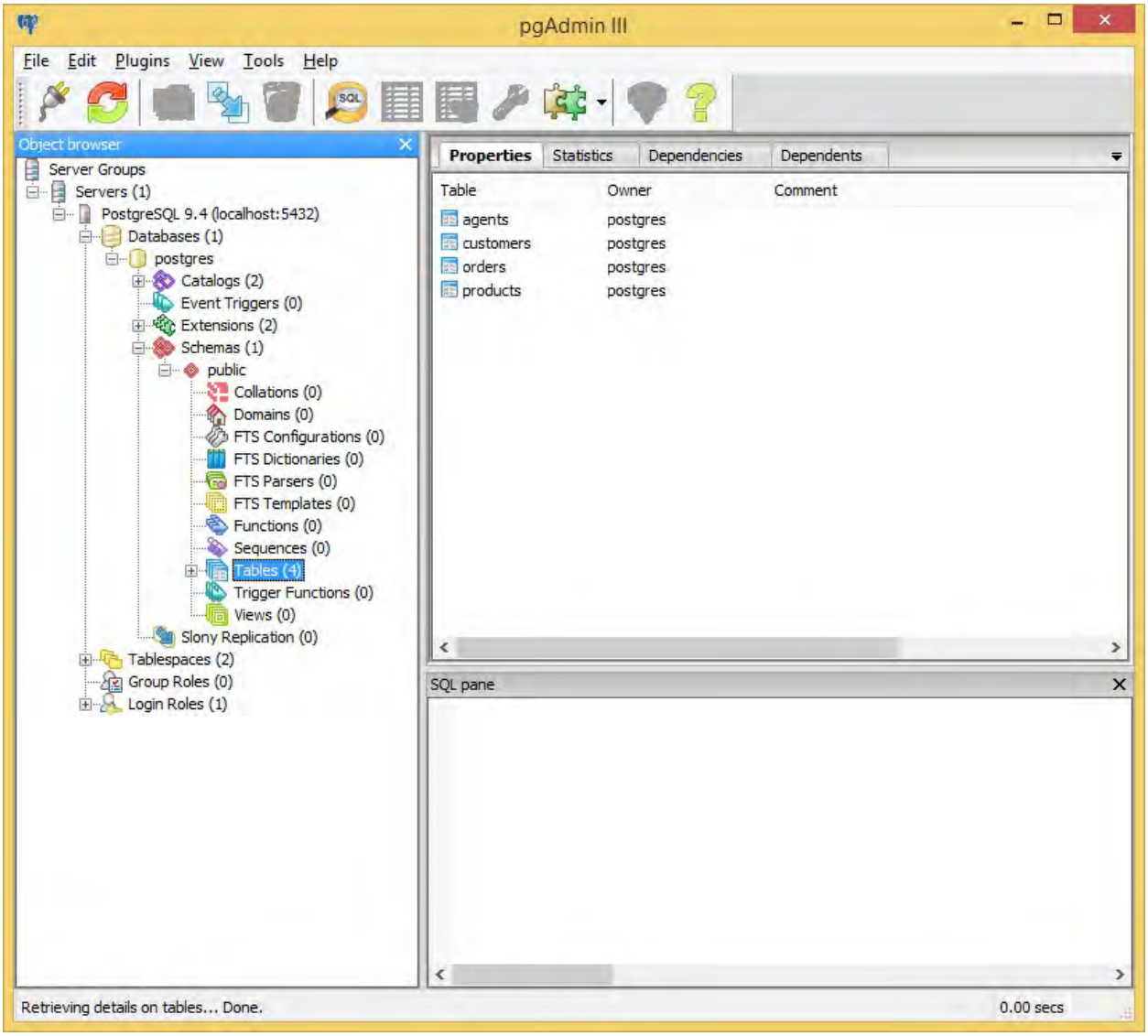
The information now available allows the Red Hook Alumni Association to generate specialized mailing lists, track membership and dues payments, and most importantly, has facilitated an increase in the number of scholarships and grants offered. The online database format allows the various committee members to add and update records through a standardized interface.

Short Essay

The hierarchical data model is considered to be the first data model, utilized in IBM's 1960s Information Management System (IMS). The hierarchical model organizes data into a tree structure. Individual records have (single) parent and (multiple) child records. Our text explains that the hierarchical model operated at the physical interface model, preventing access through higher levels interfaces of coding. The network data model allows individual records to have (multiple) parent and (multiple) child records, naturally providing for many-to-many relationships. This is commonly explained as a graph view of the data sets.

Relational database systems introduce the concept of the data viewed in the context of, well, duh… relationships. The relational model clearly and consistently defines the structure of the data, and limits operations on the data to a set of established and common instructions. This allows programmers to work at the higher interface levels, without concern for the underlying physical structure or platform. The constraints of relational, structured query language (SQL) databases are in fact primary components facilitating simple coding and high performance.

The "semistructured" data model of XML is schemaless, or at least, lacks a formal, universal schema; the data is said to be self-describing. The XML model is very flexible and fluid, allowing attributes to be included at will. The semistructured model makes structuring queries more difficult. I expect it also introduces performance issues. What do I think of XML as a model for data storage? Frankly, it offends my sensibilities, sir - particularly upon noting that our text cites Rotus Notes as an exemplar of its implementation.


pgAdmin III
File Edit Plugins View Tools Help
Object browser
Server Groups
Servers (1)
PostgreSQL 9.4 (localhost:5432)
Databases (1)
postgres
Catalogs (2)
Event Triggers (0)
Extensions (2)
Schemas (1)
public
Collations (0)
Domains (0)
FTS Configurations (0)
FTS Dictionaries (0)
FTS Parsers (0)
FTS Templates (0)
Functions (0)
Sequences (0)
Tables (4)
Trigger Functions (0)
Views (0)
Slony Replication (0)
Tablespaces (2)
Group Roles (0)
Login Roles (1)
Properties
Statistics
Dependencies
Dependents
Table
Owner
Comment
agents
postgres
customers
postgres
orders
postgres
products
postgres
SQL pane
Retrieving details on tables... Done.
0.00 secs