A Faceted Query Engine Applied to Archaeology

julia stoyanovich, joint work with kenneth a. ross and angel janevski, columbia university

THEORETICAL BACKGROUND

Our work addresses data modeling for domains where the main complexity is concentrated in the hierarchy, with other properties of the domain, such as relationships between entities, being relatively simple.

• **Faceted Classification** treats entities or groups of entities as collections of clearly defined, mutually exclusive, and collectively exhaustive aspects, properties, or characteristics. Such aspects, properties, or characteristics are called facets[6].
• Flamenco[3] and FacetMap[2] implement search over faceted hierarchies.
• We go beyond a simple search facility, and develop a data model and a query language for the formulation of sophisticated queries over faceted hierarchies.

• An **Entity Set** is the basic conceptual structure in our data model.
• **Classes** are entity sets that have explicitly stored entities in them. Entities may belong to multiple classes. Classes have attributes and constraints associated with them. Each entity in a class must assign a value to each attribute.
• A schema defines a finite set of classes organized into a hierarchy, with a single maximal superclass, `Entity`, that adds the attribute `entityID`, a unique identifier for each entity in the system.
• **Relationships** are represented by entities. Many-to-many relatinships are represented using set-valued attributes.

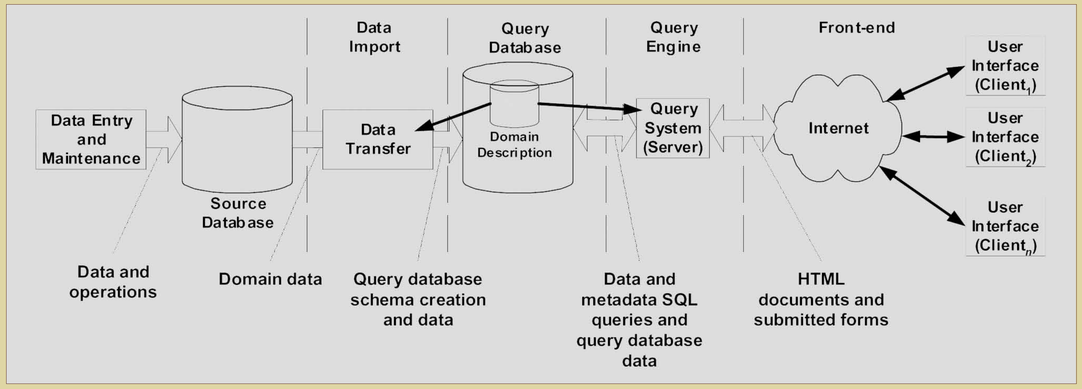
• A **Query** is formed by applying operators to entity sets to form new entity sets. During a query session, the user can refer to a previously defined entity set as a subexpression.
• We assume that a **Constraint Language** over attributes of entity sets is given, and may include equalities, inequalities, logical conjunction and disjunction. Integrity constraints may be placed on classes.
• **Entity Algebra** is a language that defines the basic set of operators. We allow the following operators on entity sets E and E', with constraints θ and θ' over attributes of E and E':
- σ_θ (E) returns all entities from E that sattisfy the condition θ
- $E \ltimes_{\theta'} E'$ returns all entities e from E for which there is some entity e' in E' such that (e, e') satisfies θ'.
- E∪E' returns all entities that are in either E or E'.
- E∩E' returns all entities that are in both E and E'.
- E – E' returns all entities that are in E but not in E'.
- Aggregation on the `entityID` attribute.

• Entity algebra has **linear** space complexity and **quadratic** time complexity.

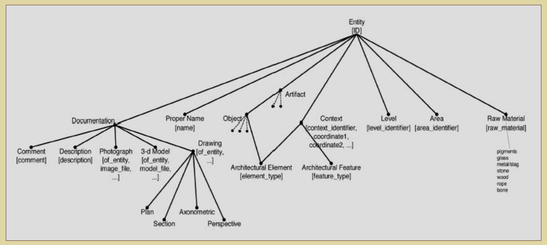
SYSTEM ARCHITECTURE

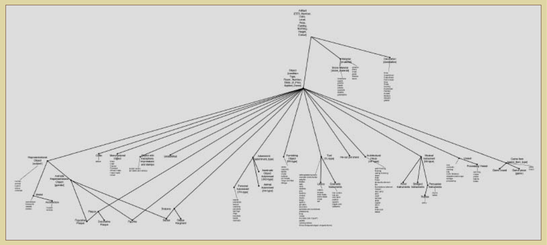
The **Faceted Query Engine** implements our faceted data model and entity algebra. The system utilize a commercial relational database to store schema meta-data and domain data in a special format.

• The Data Entry and Maintenance module allows the transfer of domain data into the Source Database.
• The Data Transfer module implements a generic procedure that translates domain-specific data into a domain-independent faceted hierarchy in accordance with the Domain Description.
• The resulting hierarchy is stored in the relational Query Database.
- `QUERY` is the central table of the schema; it stores classes, queries and operators.
- `HIERARCHY` contains `(class, parent class)` pairs for all classes in the hierarchy.
- `ATTRIBUTES` stores a list of valid attributes for each class.
• The Query Engine, implemented as a web service, interacts with the Query Hierarchy.
• The browser-based client accesses the Query Engine over the Internet to submit queries and receive query results. The user can access text, images, and multimedia information, such as QTVR.



• We collaborated with archaeologists on our team to create a faceted classification of finds from two archaeological excavations: **Memphis** in Egypt[4] and **Thulamela** in South Africa[1].
• The Thulamela dataset is a collection of iron-age finds, and includes tools, weapons, ceremonial items, personal adornments, pottery, faunal remains and metallurgical products.
• Our data collection and classification was the first efford made to systematically record and classify this data.
• The second dataset is a collection of thousands of ancient Egyptian artifacts from a six-year-long excavation in Memphis. Finds in this collection range from status objects such as scarabs and personal adornments, to musical instruments and pottery, to a variety of tools, cosmetic vessels, statues, and architectural elements.





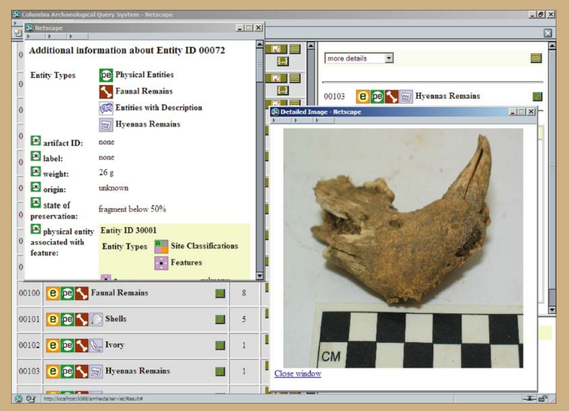
REFERENCES

1. Computational Tools for Modelling, Visualization and Analysing Historic and Archaeological Sites. http://www.learn.columbia.edu/nsf/.
2. The FacetMap Project. http://facetmap.com.
3. The Flamenco Project. http://bailando.sims.berkeley.edu/flamenco.html.
4. L. Giddy. The Survey of Memphis II. Kom Rabi'a: the New Kingdom and Post-New Kingdom Objects. The Egypt Exploration Society, London, UK, 1999.
5. K. Ross and A. Janevski. Querying faceted databases. In Proceedings of the 2004 SWDB Workshop, Toronto, Canada, August 2004.
6. B. Wynar. Introduction to Cataloging and Classification. Libraries Unlimited, Inc., 8th edition, 1992.

USER EXPERIENCE

The faceted query engine has been successful on several different levels. In terms of flexibility, we were able to transition smoothly between two distinct sites and groups of specialists. The system has also proved to be a powerful research tool. The ability to combine and save queries allows for deep investigation of objects and their contexts over time. In this way one may interrogate a group of objects in terms of their distinct properties while also charting their relationship to other entities (excavated finds, ecological remains, contexts, site features, and so on) within the system. The faceted query engine then has the potential to provide broader contextual information to what has been in the past the primarily quantitative side of archaeological research.

• Typical users of our system are domain experts, but have no knowledge of SQL or any other query language.
• The graphical user interface of the Faceted Query Engine allows the user to browse the hierarchy and access detailed informaion about individual objects.
• The user can locate entities in the dataset by any of the available facets.



• Our hierarchies of archaeological finds incorporte spatial and temporal relationships among objects, allowing the user to place objects into context and perform sophisticated data analysis.
• The user can save intermediate query results in the form of entity sets for use in future queries.
• Aggregation functionalty is available and enables the user to perform grouping operations (ex. retrieve the combined weight of pottery per architectural context).