# Implementation of a Classification Hierarchy for the GeneTests/GeneClinics Genetic Testing Databases

J. Edwards, PhD, PE[1], P. Tarczy-Hornoch, MD[1,2]
[1]Pediatrics, [2]Biomedical & Health Informatics
University of Washington, Seattle, WA

*The combination of a) our changing understanding of genotypic and phenotypic classification of diseases and b) the rapid growth and expansion of the number of entries in two databases targeted toward clinicians resulted in the need to develop a flexible dynamic hierarchical classification system for genetic disorders. The two databases making use of this classification schemas are the GeneClinics (GC) database – www.geneclinics.org and the GeneTests (GT) database – www.genetests.org The GC and GT databases serve respsectively as the users manual and yellow pages of genetic testing. The GeneTests/GeneClinics (GT/GC) classification hierarchy is maintained as a simple set of parent/child relationships in a relational database. The hierarchy is generated in real time in response to a user request. It is not maintained as a set of members with relationships defined by characters that are parsed to determine the structure of the tree. The GT/GC classification hierarchy entries are handled as objects by the data maintenance and search tools and may have a number of attributes and associations that create a rich tool for defining and examining genetic disorders.*

## INTRODUCTION

The enormous growth in raw sequence data from the Human Genome project has resulted in a smaller but parallel growth in the availability of genetic testing. This growth exceeds the ability of the word of mouth system of finding out for which genetic diseases testing is available. Furthermore the rapid growth of the genetic testing industry makes it hard for non-expert clinicians to keep up to date with the appropriate application of genetic testing. This is becoming even more of an issue as the genetic basis for common genetic disorders is becoming clear. No longer is genetic testing the domain of geneticists – it is becoming an issue that will soon impact most health care providers[1]. The GT and GC databases were designed to address these needs.

*GeneClinics and GeneTests clinical genetic databases:* The GT database[2] was established in 1993 for clinicians searching for information about the availability of genetic testing. GT currently (March, 2002) contains 525 laboratories offering clinical tests for 540 disorders and research tests for 750 disorders. A clinical test is one offered by a lab that is CLIA certified and is intended for clinical use. A research test is one not intended for patient care – instead the primary intent is to permit the laboratory offering the test to investigate the genotypic basis of a given phenotype or set of related phenotypes. The GC database[3] was established in 1997 for clinicians searching for up to date information on the application of genetic testing for specific clinical conditions. It consists of entries that are expert authored, peer reviewed and regularly updated. The entries focus on the use of testing for diagnosis, management, and genetic counseling of specific inherited diseases including links to patient resources, policy statements, PubMed, and Genomic Databases. As of March 2002 the GC database contained 139 reviews (which cover a much larger number of genetic disorders).

*Need for a flexible classification hierarchy*: Our understanding of the causal relationship between genotype and phenotype is a dynamic process thus any classification system needs to be flexible to support this. The initial solution was a simple tree construct which permitted subdividing broader clinical phenotypes (e.g., Charcot-Marie-Tooth Hereditary Neuropathies) into phenotypically distinct subtypes (e.g., CTM 1, CMT 2, CMT 4, CMT X). In turn these phenotypic subtypes were divided into distinct genotypic subtypes (e.g., CMT 1A [PMP-22 mutations], 1B [P0 mutations], etc.) which were linked via a causality relationship to the underlying genotypic information. Over time, the GT/GC experience with complex phenotype and genotype relationships (such as spinal muscular atrophy and the FGFR-related craniosynostoses) revealed that the rules of nature and medicine were not so simple and that the model needed to be more flexible by not assuming a simple tree structure and by not assuming that only leaves (terminal nodes) could be genotypic categories linked by causality relationships to genotype. Our approach to this problem addresses most of the considerations pointed out by Robin and Biesecker in their wish list for a multiaxis nomenclature system for genetics[4].

## BACKGROUND

The GT/GC classification hierarchy was implemented to build and display relationships between the core GT/GC entries (tests and reviews respectively) and a

core classification hierarchy for genetic disorders. The solution described here is a directed graph which a) permits nodes ("diagnostic categories") to be either genotypic or phenotypic, b) allows the ability to split a given diagnostic category into children unified either by phenotype (e.g., CMT 1) or by genotype (e.g., PMP-22 mutations leading to both CMT 1A and HNPP); and c) permits genotypic categories (e.g., spinal muscular atrophy caused by SMN mutations) to have phenotypic children (i.e., SMA 1, SMA 2, SMA 3, SMA 4 defined by age of onset and rate of progression).

Since molecular testing is directed directly at genotype (and only indirectly at phenotype), a constraint in this new model is that a specific test can only be linked to a diagnostic category defined by genotype. For clinical tests ([T]), names (diagnostic categories, synonyms and their children) are "genotypic" categories whenever possible; for research testing ([R]), entries may be "genotypic" if the causative gene(s) is (are) known, or "phenotypic" if unknown. A "genotypic" category is defined as a category unified by common genotype thus for a single gene single disease entry the "genotypic" and "phenotypic" category is the same.

The purpose of the hierarchy is to display the relationship between disorders sharing clinical features and/or genetic mechanisms. The current hierarchy is simple, but robust enough to 1) facilitate differential diagnosis; 2) support the understanding of disease lumping and splitting by phenotype or genotype that accompanies gene discovery; 3) reflect the concept that clinical testing is by genotype only and research subjects are identified by either genotype and/or phenotype; 4) display in a single view search results indicating clinical test ([T]) availability, research ([R]) studies and use of clinical testing in patient care. For example, Alport Syndrome (Figure 1) is a kidney disorder with hearing loss that can be inherited in an autosomal dominant, autosomal recessive or X-linked recessive manner; these three forms are caused by alterations in three different genes (COL4A3, COL4A4 and COL4A5). The hierarchy is an efficient way to alert the user that a review ([REV]) for all forms of Alport Sydrome is available, but testing is available for the X-linked form only.

The hierarchy exists in the database as a table of parent/child relationships. A given entry may have one or more parents and one or more children. A child of an entry can be a parent of another and vice verse. This allows a theoretically unlimited number of generations of relationship members. The tree that is displayed as a result of a search is constructed on the fly when the database is queried. Structuring the hierarchy as sets of parent/child relationships facilitates management of the data by allowing the data curators to concentrate on direct relationships between hierarchy entities. The algorithm that builds the tree manages the interrelationships between the data elements.

Hereditary Hearing Loss and Deafness [REV]
- Nonsyndromic Hearing Loss and Deafness [R]
  - Nonsyndromic Hearing Loss+Deafness (Mitochondrial)[T,R]
    - Aminoglycoside-Induced Deafness [T,R]
  - Nonsyndromic Hearing Loss+Deafness, Autosomal Dom [R]
    - DFNA 3 (Connexin 26) [T,R,REV]
  - Nonsyndromic Hearing Loss+Deafness, Autosomal Rec[T,R]
    - DFNB 1 (Connexin 26) [T,R,REV]
  - Nonsyndromic Hearing Loss and Deafness, X-Linked
- Syndromic Hearing Loss and Deafness
  - Mitochondrial Disorders (Assc. w/ Hearing Loss+Deafness)
    - Diabetes and Hearing Loss [T,R]
    - Kearns-Sayre Syndrome [T]
    - MELAS [T,R,REV]
    - MERRF [T,R]
    - NARP [T]
  - Syndromic Hearing Loss+Deafness, Autosomal Dominant
    - **Alport Syndrome, Autosomal Dominant**
    - Branchiootorenal Syndrome [T,R,REV]
    - Neurofibromatosis 2 [T,R,REV]
    - Stickler Syndrome [R,REV]
      - Stickler Syndrome Type I [T]
      - Stickler Syndrome Type II [T]
      - Stickler Syndrome Type III
    - Waardenburg Syndrome
      - Waardenburg Syndrome Type I [T,R]
      - Waardenburg Syndrome Type II [R]
      - Waardenburg Syndrome Type III
      - Waardenburg Syndrome Type IV
  - Syndromic Hearing Loss+Deafness, Autosomal Recessive
    - **Alport Syndrome, Autosomal Recessive**
    - Jervell and Lange-Nielsen Syndrome [R]
      - LQT 1 [T,R]
      - LQT 5 [T,R]
    - Pendred Syndrome [T,R,REV]
    - Refsum Disease
      - Refsum Disease, Adult [T,R]
      - Refsum Disease, Infantile [T,R]
    - Usher Syndrome
      - Usher Syndrome Type 1 [R,REV]
      - Usher Syndrome Type 2 [R,REV]
      - Usher Syndrome Type 3 [R]
  - Syndromic Hearing Loss and Deafness, X-Linked
    - **Alport Syndrome, X-Linked [T,R]**
    - DFN 1
    - Norrie Disease [T,R]
      - X-Linked Familial Exudative Vitreoretinopathy

**Figure 1: Hearing Loss and Deafness Hierarchy (T=clinical testing, R=research testing, REV=Review).**

## DESIGN OBJECTIVES

The current GT database, administration tools, and search engine have been in production since December 1997. The GC database has been in production since 1998. The two databases, maintained by separate content teams, were combined in 2001 into a merged

Web site. The GT/GC hierarchy had to accommodate the existing architecture of the combined databases with minimal database schema and program changes.

The GT/GC hierarchy needed graphical maintenance tools and a revised Web-based search interface. The maintenance tools are used by both the GT and GC teams; however, each team needs a different view of the database to maintain its specific data. The GT team uses tools to add, edit or delete a laboratory, test package or clinic. The GC team has tools to add, edit or delete a review. To help maintain the database, reports are needed to facilitate viewing data. The public view of the hierarchy must be logical and self-explanatory. Users should be able to use it with a minimum amount of explanation.

## SYSTEM DESCRIPTION

*The hierarchy table:* The hierarchy exists is stored as in a table with self referential pointers in a relational database - *entry_entities*. The *entry_entities* table has four columns: *entry_entity_id*, *entry_id*, *entities_id*, *entity_id*. The *entities_id* is a child of the *entry_id* and the *entity_id* calls out the type of the entry. The *entry_entity_id* field contains a number that is unique for each row - it is a primary key. The *entry_id* field is a foreign key containing the entities_id of the GT/GC entity - it is a foreign key. The *entities_id* field contains the id of the child GT/GC entity. The *entity_id* field contains a number designating the type of entity, for example, 2 refers to a disease, 13 to a review.

*The name table:* A GT/GC entry exists as a field in the name table. *The table has eight columns: name_id, entities_id, name, full, type, metaphone, entity_id* and *edit_date*. The *entities_id* is the entities_id the unique id of a particular entry. It is the same for all rows associated with a given entry. The *name* column contains the text of the entry's name. The *full* column contains the same text that is in the *name* field in upper case and is used for case-insensitive searches. The *type* field contains a character that defines the type of name the row holds. There are five types of names: primary name (N), synonym(Y), overview(O), shortname(S), hidden or virtual synonym(V). The name is the name of a disorder or a review. The overview category is the primary name of an overview. The shortname is the name of the directory in which the review or overview XML and HTML files are stored. The hidden or virtual synonym is used to enable the entry of an alternative spelling of an entry to facilitate searching. For example, 'DFNA1' may be entered as a hidden synonym of 'DFNA 1' permitting a user to find the entry using either term. The hidden synonym is shown only if it is the term that matched a user's query. The *metaphone* term is the result of passing the primary name through an algorithm that strips all non-alphanumeric characters converting the remaining letters into a phonetic pattern[5]. When a user does a `sounds like' search the search term is passed through the metaphone algorithm before it is sent to search the database using the *metaphone* field. The *entity_id* field contains a number that defines the entity type. There are eight *entity_ids* that are stored in table name: disease, locus, gene, product, package, variant, service, and review.

*Creating a classification entry:* Parent child relationships are created using the `add/edit GT/GC entry' tool. The tool is a java applet and is run as a client in a java enabled browser, such as Netscape or Internet Explorer, on a database maintainer's PCs. The `add/edit GT/GC entry' panel is used to enter the name, synonyms, and external references of the entry. The entry can also be associated with children using the `children' subpanel. Adding child relationships to the entry defines the entries' hierarchical relationships. An entry can be associated with entries that also have children and be a grandparent of the children of its children. If the entry is flagged as a feature, its children are defined as the set of entries associated with that feature. The entry's name is added to the name table and its child associations are added to the *entry_entities* table. Existing parent and child relationships are displayed in an entry tool panel for ready reference.

*Associating GT and GC entities with GT/GC entries:* GC reviews and GT entries are associated with GT/GC entries using the add/edit review and add/edit package panels (tools) respectively. A list of GT/GC entries is provided based on a query by GT/GC entry name or id. When a list item is selected it is added to the GT/GC entries that are associated with the given GC review or GT entry. When the GC and GT entities are associated with a GT/GC entry they inherit all attributes of that entry, including: GT/GC parent/child relationships, external reference links (OMIM, PubMed), gene symbols, feature categorization, phenotype/genotype classification, primary hierarchy status, and names (primary, synonyms and hidden synonyms). Primary hierarchies are identified by the GT/GC data curators as disorders that may be specific interest to users. These attributes may be used as search criteria, influence the search display, and may be displayed in search results.

*GT and GC entities viewing rules:* The fact that not all nodes in the hierarchy are visible to all users at all times adds an additional challenge to the display algorithm. GT and GC entities have attributes regarding status of a particular entry. Users have affiliation and role attributes that are used by the search algorithm to decide if a given user can view a given GT or GC entry given the entry's status attributes. The initial search results are checked to ensure that the displayed links

lead to viewable data. In this way dead end links are avoided. For example, GC reviews can be active, draft or inactive. An active review is publicly viewable; any user can access it. A draft review is viewable by authors, reviewers and GC staff. A link to an inactive review will not show up in any search results. GC review status is maintained in the add/edit review administration panel. GT entries consist of a laboratory linked to a test. Laboratories may also request that clinically available tests be viewed only by health care professionals. A laboratory may be flagged as inactive when its information is found to be out of date. Links to GT entries that have inactive labs are not shown in the search results.

*Searching and displaying the hierarchy entries:* In this section we describe the algorithm used to search and display the hierarchy. The complexity of the hierarchy and the viewing rules required the adoption of a recursive approach. The fact that a given node may have zero or more children and zero or more parents adds another layer of complexity. The search begins when the user submits a query string using a Web browser. The query form parameters are passed to a jakarta-tomcat servlet engine (http://jakarta.apache.org) by an apache web server (http://httpd.apache.org/). The access servlet class puts the parameters into a hashtable, obtains a database connection from the connection pool and calls the search class. The search class prepares a SQL query based on the search parameters. The query is submitted to the Oracle 8.1 database (http://www.oracle.com/) using JDBC (http://java.sun.com/). Match objects are created from the resultant data set and added to a matches vector. If a entry is matched with the primary name and if its synonyms also match the query string, synonym matches are ignored and the primary name match object is used. If the match is a synonym or a hidden synonym, the primary name is appended to the synonym name to help users to understand that the match was on a synonym. If the matches vector size equals zero, a failed search message is returned.

After the set of matches is obtained, parent and child relationships are added to each match. Hashtables of parent and child ids enable the relationships to be built without accessing the database for each match thereby speeding up the process. The database is accessed to retrieve detailed information about a match only in the case that one of the hashtables indicates that the action is necessary. If the match has viewable references, the reference URLs are included as a attribute of the match. If a match has no viewable references and no parents or children that have viewable references, that match is flagged as a 'no show'.

Now the set of all matches that are related to the search request contain references to their parents and children. The next step is construction of the hierarchy tree(s) from the matches. The children of each match are recursively tested against each match. If a match has a child that is a match, the child match is merged into the match. A match may be a child of one or more matches. In some cases a match's parent is not already a match. In these cases a new parent match is created and the matches of that common parent are merged as children matches of that common parent. The search results are now prepared for display. If there are no matches with URL links, a failed search message is returned. Primary hierarchies are identified so they be returned at the top of the results. The match objects are sorted by name, a result table is generated, and returned to the user.

## CURRENT STATUS

The GT/GC hierarchy was put into production July 2001. The GT/GC database contains some 40,000 users, 139 reviews, 525 laboratories, 915 disorders, 4,400 laboratory/test entries, and 1,050 clinics. There are over 900 distinct Laboratory Directory searches per day and about 2,000 reviews per day are viewed. Feedback regarding the hierarchy from end users (both genetics professionals and other healthcare providers) has been very favorable. The curators of the GT/GC databases (the content teams) have been very satisfied and thus far have not identified genetic disorders than can not be represented using this hierarchical model.

## DISCUSSION & CONCLUSION

The GT/GC classification hierarchy attributes include:

- the GT/GC classification hierarchy is stored as parent/child relationships
- the hierarchy tree is built in real time
- the entry attributes may be changed easily as knowledge of a disorder evolves
- the database schema is designed to facilitate the addition of attributes and associations
- the object oriented design of the administration applets and search algorithm provide flexibility to respond to evolving data and search requirements

The GT/GC classification hierarchy is built in real time when a user queries the GT/GC database. The GT/GC hierarchy relationships exist in the database as a table of parent/child relationships. The hierarchy is constructed in response to a query. This approach differs from the most widely used medical nomenclature hierarchies, such as MeSH (Medical Subject Headings)[6]. In both cases someone must maintain the entry relationships. The difference is in the way the relationship data is stored and user requested views are constructed. The primary hierarchy of medical nomenclature currently in use is MeSH

(Medical Subject Headings), the controlled vocabulary of biomedical terms for the National Library of Medicine. Subject specialists index 4,300 biomedical journals with appropriate Medical Subject Headings. The Subject Headings are pre-arranged hierarchically, with narrower terms arranged under broader terms. The MeSH terms are maintained as a set of entries that are identified and associated using "Tree Numbers". A given entry must explicitly have a tree number for each of its positions in the MeSH tree. This hierarchy exists independently of any specific search for information by users.

One advantage to this external hierarchy is its utility as a guide in the construction of an information search. However, the static nature of the system has several disadvantages. In the biomedical field, where new developments and terminology are constantly being introduced, it is very difficult to keep an externally maintained system up to date. Because MeSH is focused on a system rather than on individual entries, interelationships between hierarchies may not have been established or be apparent. Searches may result in missed terms because the data is must be linked to a predefined hierarchy rather than one created dynamically in response to a specific search request as the GT/GC hierarchies are.

Using the GT/GC technique, the parent/child relationships are defined and used to dynamically construct the hierarchy. MeSH uses a predefined hierarchy. If MeSH terms were placed in hierarchical format based on their parent/child relationships, a potentially powerful dynamic system could result. The hierarchy would become more complex and separate trees would be interrelated based on the term hierarchical relationships. Broader terms would automatically end nearer the top of the tree and narrower terms at the bottom. Some terms would automatically be part of multiple trees but this would be easy to see and maintain. Data would be still be indexed with MeSH by being associated to the appropriate terms. But there would be no need for a predefined tree. A hierarchy of terms would be generated and returned to the user in response to a query. The result set could be limited by hierarchy nodes selected by the user.

GT/GC entries can have attributes assigned, such as genotype/phenotype classification, related genes, feature categorization, external references - such as OMIM and PubMed links, citations, and primary hierarchy status. These attributes add richness to the entries and can provide additional information that can help a user obtain a desired search result. MeSH terms could also have object oriented properties if a similar approach was used to maintain and build a MeSH term hierarchy.

The GT/GC approach enables those who maintain the GT/GC hierarchy to focus on the direct relationships between GT/GC entries. As entries are added and associated with each other an implicit tree structure is constructed. If an entry is associated with several parents or children it will appear in the branch that contains those relatives when the hierarchy tree is constructed in response to a user's query. By design the hierarchy can contain an indefinite number of generations. Also reviews and tests may have many attributes, such as phenotype and genotype, and are linked with other entities. This results in a complex set of associations that are used to enable a variety of query strategies and user views. The database schema, administrative applet and search classes are designed to facilitate the addition of attributes and associations.

## ACKNOWLEDGEMENTS

Thanks to Chris Beahler, Health Sciences Libraries, University of Washington for providing information on commonly used medical information hierarchies. The GeneClinics and GeneTests databases are supported by a number of agencies and institutes including NLM, NHGRI, NCI, MCH, and the DOE.

## REFERENCES

1. Emery, J, Hayflick, S. "The challenge of integrating genetic medicine into Primary Care". BMJ 322:1027-30, 2001
2. Tarczy-Hornoch, P, Covington, ML, Edwards, J, Shannon, P, Fuller, S, Pagon, RA. "Creation and Maintenance of Helix, a Web Based Database of Medical Genetics Laboratories, to Serve the Needs of the Genetics Community". Jour Amer Med Inform Assoc, Fall Symposium Suppl, 341-5, 1998
3. Tarczy-Hornoch, P, Shannon, P, Baskin, P, Espeseth, M, Pagon, RA. "GeneClinics: a hybrid text/data electronic publishing model using XML applied to clinical genetic testing". J Am Med Inform Assoc. 7:267-76, 2000
4. Robin, N, Biesecker L. "Considerations for a multiaxis nomenclature system for medical genetics". Genet. Med. 290-3, 2001
5. Lawrence, P, "Lawrence Philips' Metaphone Algorithm", http://aspell.sourceforge.net/metaphone/ 2000
6. Humphreys BL, Lindberg DA, Schoolman HM, Barnett GO. "The Unified Medical Language System: an informatics research collaboration". J Am Med Inform Assoc. 1-11, 1998