Daylight

Chemical
Information
Systems, Inc.

Enterprise-level cheminformatics.

Thor, Merlin, tdt’s, Thorfilters

Summer School 2004

Documentation Reference

http://www.daylight.com/dayhtml/doc/admin/

Chemical nomenclature

…Even before the science of chemistry as we know it evolved, it was important to report discoveries and present theories either orally or through printed publications. To do so, it was recognized very early that a special, controlled language would be necessary. This language is called chemical nomenclature and today it is expected not only to reveal the atoms present, but also how these atoms are arranged in the molecule and the chemical relationships with other chemical substances…

Nomenclature of Organic Compounds Principles and Practice (2001)
Second Edition Edited by Robert B. Fox and Warren H. Powell page 1

“Name it to tame it”

- Unique names for substances have long been a goal for chemists.
- However the two major players, Chemical Abstracts Service (CAS) and the International Union of Pure and Applied Chemistry (IUPAC) cannot even agree on the spelling of the names of the most fundamental substances, the elements.
  - CAS aluminum, cesium
  - IUPAC aluminium, caesium
  - Neither spell sulphur correctly!!

SMILES, a unique name?

- SMILES was developed to address the problem of uniquely naming the vast majority of substances.
  - SMILES uses *only* the internationally agreed element symbols to name the elements.
  - SMILES encodes the relationship between atoms and the configuration of chiral centres.
  - Canonical SMILES act as unique names (identifiers) for all *substances* which can be accurately represented by a single valence bond structure.
  - Canonical SMILES act as unique names for virtually all valence bond *structures.*

Identifiers and data

- The SMILES contains information about the structure only. The language continues to be enriched to enable the SMILES to accurately represent what a compound *is*. I.e. it is an ***identifier.***
- Other information exists which is *about the identifier.* This we refer to as ***data***. This should not under any circumstances be embedded in the SMILES. E.g.
  - Molecular formula
  - Molecular weight
  - Depiction i.e. 2D coordinates
- Data is associated with the appropriate identifier using thor data trees or tdt's.

Thor data trees

- A thor data tree has the following format

  $tag<*identifier*>
  tag1<*data*1>
  tag2<*data*2>
  tag3<*data*3>
  …
  tagn<*data*n>
  |

- As the tags are arbitrary the data tree is not a universal representation of the information.

Daylight

Chemical
Information
Systems, Inc.

Enterprise-level cheminformatics.

Thor data tree example

```
$SMI<OC(CCN1CCCCC1)(C2CCCCC2)c3ccccc3>
FP<U..428+..UFLVQE+.3I6E.E.80V8...EVGcFEEE.....V22kM.2W.++.21EO.0
U0.+oE..EO01.+6660E7.+0.UU.1..42.04..E.2U1.0U.2.2.c0.V+0E.AA..E7Y+..6.
..+...F.UE6.....E06.EVW...26c30..6E0+..2;1024;155;1024;155;1;>
PAMW<301.47>
PMF<C20H31NO>
2D<4.62,-4.08,4.62,-3.08,3.00,-3.51,1.76,-3.08,0.85,-3.08,0.35,-3.94,-0.65,-
3.94,-1.15,-3.08,-0.65,-2.22,0.35,-2.22,5.62,-3.08,6.17,-2.20,7.16,-2.20,7.62,-
3.08,7.09,-3.97,6.08,-3.97,4.62,-2.08,5.49,-1.58,5.49,-0.57,4.62,0.08,3.75,-
0.57,3.75,-1.58;>
|
```

- Thor data trees currently use only the ASCII character set.
- Internally they are regular objects

Yet more identifiers

- There are lots of identifiers associated with a particular substance.
  - CAS Number
  - NCI Number
  - IUPAC Name
  - Common name
- These may or may not have data associated with them.
- However it is imperative to ensure that data is associated with the appropriate identifier. E.g.
  - Biological test data is associated with the sample identifier, not with the structure. In this case the structure is a datum about the sample id.

A non-SMILES rooted tdt

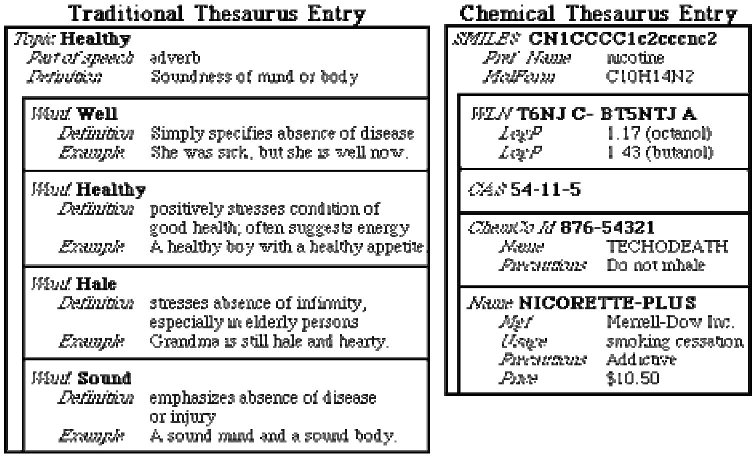
```
$VCS_ID<T125>
VCS_SUPP<LOPAC>
VISM<Cl.OC(CCN1CCCCC1)(C2CCCCC2)c3ccccc3>
2D<7.47,-0.29,4.62,-4.08,4.62,-3.08,3.00,-3.51,1.76,-3.08,0.85,-3.08,0.35,-3.94,-0.65,-
3.94,1.15,3.08,0.65,2.22,0.35,2.22,5.62,3.08,6.17,2.20,7.16,2.20,7.62,3.08,7.09,3.97,6.
08,3.97,4.62,2.08,5.49,1.58,5.49,0.57,4.62,0.08,3.75,-0.57,3.75,-1.58;>
ID<237>
CATNUM<T-125>
/NAME<Trihexyphenidyl hydrochloride>
/POSITION<RK003-C11>
ACTION<Antagonist >
/CLASS<CHOLINERGIC>
SELECTIVITY<Muscarinic>
DESCRIPTION<"Muscarinic receptor antagonist; centrally acting anticholinergic">
AMW<337.93>
MF<C20H32ClNO>
/PISM<OC(CCN1CCCCC1)(C2CCCCC2)c3ccccc3>
|
```

Creating data models

- Now we have a clear concept of identifiers and data we can build appropriate data models.
- These can be built into any framework, from an Excel spread sheet upwards.
- Of particular interest to us are Thor and Oracle®.
- Thor is designed to be a chemistry database system so the structure is of supreme importance.
- In an Oracle® database the chemical structure has no particular significance, and most of the time is treated just like any other piece of text. It only becomes chemically relevant data when queried using the functions in DayCart™.

The thesaurus model

- Thor is an acronym for **TH**esaurus **O**riented **R**etrieval
- The unique SMILES acts as a unifying concept under which the identifiers can be grouped.
- Note, as with a traditional thesaurus, there is no need for the other identifiers to be unique.
- This model is built into the thor server. It can be implemented in Oracle®.



Traditional Thesaurus Entry

Topic: **Healthy**

Part of speech	adverb
Definition	Soundness of mind or body

Word **Well**

Definition	Simply specifies absence of disease
Example	She was sick, but she is well now.

Word **Healthy**

Definition	positively stresses condition of good health; often suggests energy
Example	A healthy boy with a healthy appetite.

Word **Hale**

Definition	stresses absence of infirmity, especially in elderly persons
Example	Grandma is still hale and hearty.

Word **Sound**

Definition	emphasizes absence of disease or injury
Example	A sound mind and a sound body.

Chemical Thesaurus Entry

SMILES **CN1CCCC1c2cccnc2**

Pref Name	nicotine
MolForm	C10H14N2

WLN **T6NJ C- BT5NTJ A**

LogP	1.17 (octanol)
LogP	1 43 (butanol)

CAS **54-11-5**

ChemCo Id **876-54321**

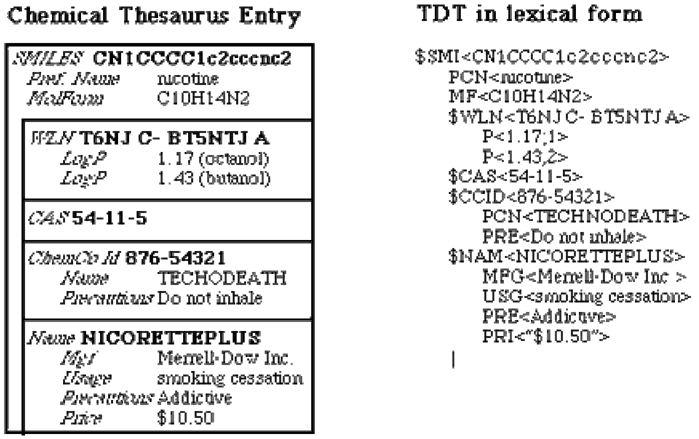
Name	TECHODEATH
Precautions	Do not inhale

Name **NICORETTE-PLUS**

Mfr	Merrell-Dow Inc.
Usage	smoking cessation
Precautions	Addictive
Price	$10.50

Building a tree

- The identifiers along with their associated data, become subtrees in the main SMILES rooted trees.
- As implemented, only the SMILES is allowed to have subtrees. The root of a tree is not required to be a SMILES



Chemical Thesaurus Entry

```
SMILES CN1CCCC1c2cccnc2
  Pref. Name    nicotine
  MolForm       C10H14N2
  WLN T6NJ C- BT5NTJ A
      LogP          1.17 (octanol)
      LogP          1.43 (butanol)
  CAS 54-11-5
  ChemCo Id 876-54321
      Name          TECHODEATH
      Precautions Do not inhale
  Name NICORETTEPLUS
      Mfg           Merrell-Dow Inc.
      Usage         smoking cessation
      Precautions Addictive
      Price         $10.50
```

TDT in lexical form

```
$SMI<CN1CCCC1c2cccnc2>
   PCN<nicotine>
   MF<C10H14N2>
   $WLN<T6NJ C- BT5NTJ A>
      P<1.17;1>
      P<1.43;2>
   $CAS<54-11-5>
   $CCID<876-54321>
      PCN<TECHNODEATH>
      PRE<Do not inhale>
   $NAM<NICORETTEPLUS>
      MFG<Merrell-Dow Inc >
      USG<smoking cessation>
      PRE<Addictive>
      PRI<"$10.50">
   |
```

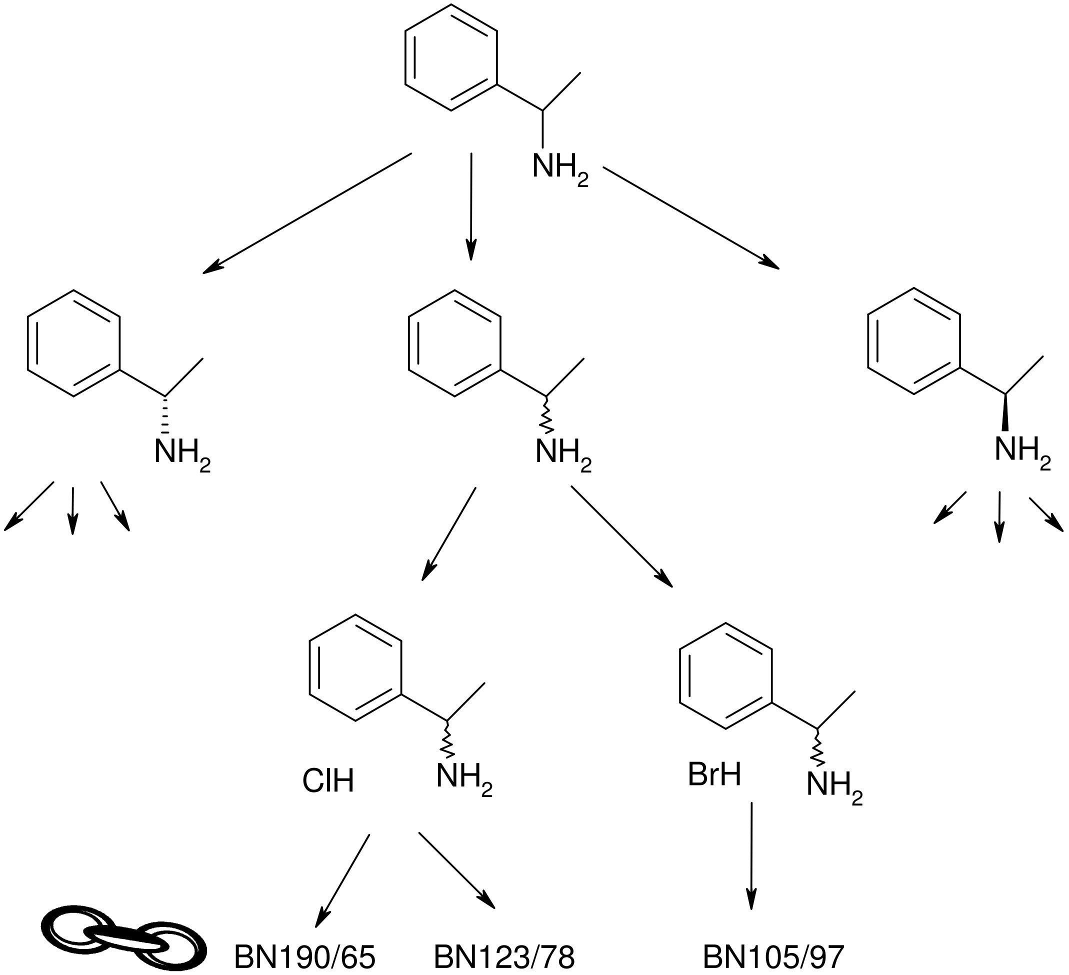
Daylight
Chemical
Information
Systems, Inc.

Enterprise-level cheminformatics.

Why a thesaurus?

- If the only reason to have a thesaurus were to keep track of names and identifiers, then there would be no need for a chemical thesaurus. A SMILES may be regarded as a word and it would simply be added.
- What is important is that a natural hierarchy exists in structures.
  - C[C@@H](N)c1ccccc1 and C[C@H](N)c1ccccc1both have the same unique SMILES CC(N)c1ccccc1
  - Moreover CC(N)C(O)c1ccccc1, C[C@H](N)C(O)c1ccccc1, C[C@@H](N)C(O)c1ccccc1, CC(N)[C@H](O)c1ccccc1, CC(N)[C@@H](O)c1ccccc1, C[C@H](N)[C@H](O)c1ccccc1, C[C@H](N)[C@@H](O)c1ccccc1, C[C@@H](N)[C@H](O)c1ccccc1, C[C@@H](N)[C@@H](O)c1ccccc1 have the same unique SMILES.
  - They are automatically merged along with their data into the appropriate tree

Structure Hierarchy



Storage and retrieval

- Thor is a *storage and retrieval* system. It is important to distinguish this from a *searching* systems: Thor *does not search* for chemical information (this is the job of the Merlin system.)
- By *storage and retrieval*, we mean that Thor has the ability to *store* a tdt using the identifiers it contains, and later *retrieve* that same tdt using any identifier within it. Although this sounds simple, Thor's *thesaurus orientation* makes this a powerful method which is extremely fast at storing and retrieving chemical information. The time it takes to access your information is the same no matter how large your database grows. Storage and retrieval takes place in milliseconds, even in databases containing millions of entries.

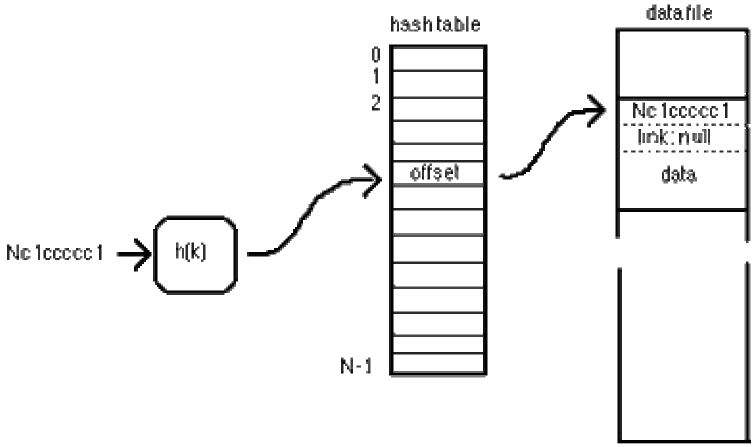
Hash tables

- By using the unique SMILES of a molecule as the molecule's primary identifier (the tdt's "main topic"), Thor is able to eliminate all searching during data retrieval. All data are looked up directly in a *hash table*.
- Hashing begins with a *hash function*, h(K,N), which takes a string of characters, K, and converts it (via a pseudo-randomizing algorithm) into a number between zero and N-1.
- Using a hash function h(K,N), data records on the computer's disk can be accessed directly: The hash value is used to access a *hash table*, which contains the desired record's location in the *data file*. Except in the case of hash collisions , only two disk accesses are required (one if the hash table is *cached*).

Daylight Chemical Information Systems, Inc.

Enterprise-level cheminformatics.

Hash-based retrieval



Clashes and cross-reference

- It is possible for two different SMILES to produce the same hash number. This is termed a collision. The thor server uses standard computer science procedures to resolve these.
- If a non-SMILES identifier is used to look up, the “cross-reference” yields the SMILES which is/are at the root of the page(s) with that identifier.
- In both the above cases a small number of disc accesses will be needed to retrieve the data.

Ambiguous words

- In a traditional thesaurus the word “well” appears on several pages
  - Greatly
  - Receptacle
  - Lowness
  - Depth
  - Excavation
  - Water
  - Flow
  - Well
  - Store
  - Aright
  - Healthy
  - Skilfully
- The context allows us to ascertain which meaning is appropriate.

Ambiguous identifiers

- In a chemical thesaurus, the “Available Chemicals Directory” there are 10 compounds with the name “007”
  - Fc1ccc(cc1)C(=O)c2ccc(F)cc2 007
  - OC(=O)c1c(cccc1C(F)(F)F)C(F)(F)F 007
  - Oc1ccccc1O 007
  - [O-][N+](=O)c1cccc(C=O)c1 007
  - Nc1ccc(C(=O)O)c(Cl)c1 007
  - FF 007
  - COC(=O)C1=C(C)NC(=C(C1c2ccccc2[N+](=O)[O-])C(=O)OC)C 007
  - CCOC(=O)C1=C(C)NC(=C(C1c2cccc(c2)[N+](=O)[O-])C(=O)OC)C 007
  - O[Mg]O.O[Al](O)O 007
  - CC1(C)C2CC1C(C)(O)C(=O)C2 007
- The context of the query allows us to ascertain which is appropriate.

Exploring data

- Whilst for many purposes, e.g. compound registration, inventory etc, the Thor functions of *store and retrieve* are sufficient, it is sometimes necessary to ask other *chemically-significant* questions like
  - What *superstructures* of my target do I have?
  - What *substructures* of my target do I have?
  - What *tautomers* of my target do I have?
  - What compounds, *similar* to my target, do I have?
- In addition there are related non-chemical questions like
  - How many active compounds have come from Asinex
  - Which commercial drugs have MW < 200 or > 350

The Merlin system

- This requirement for *data exploration* by a series of *ad hoc* queries led to the development of the Merlin system.
- Like the Thor system, Merlin provides access to a Thor database. However, the "view" Merlin gives of the database is quite different than Thor's view: Thor is a "microscope" for the database that provides a detailed view of individual datatrees, whereas Merlin might be thought of as a "macroscope" that performs operations on the database as a whole.

In-memory searching

- The basic idea behind Merlin is that data in a computer's main memory can be manipulated roughly five orders of magnitude faster than data on its disk.
- But this simple comparison, while impressive, does not capture the real differences between disk- and memory-based searches:
  - With disk-based systems, you formulate a search carefully, because it can take minutes to days to get your answer back. With Merlin it is usually much faster to get the answer than it is to think up the question. This has a profound effect on user's attitudes towards the EDA system.
  - In disk-based systems, you typically approach with a specific question, often a question of enough significance that you are willing to invest significant effort to find the answer. With Merlin, it is possible to "explore" the database in "real-time" - to poke around and see what is there. Searches are so fast that users adopt a whole new approach to exploratory data analysis.

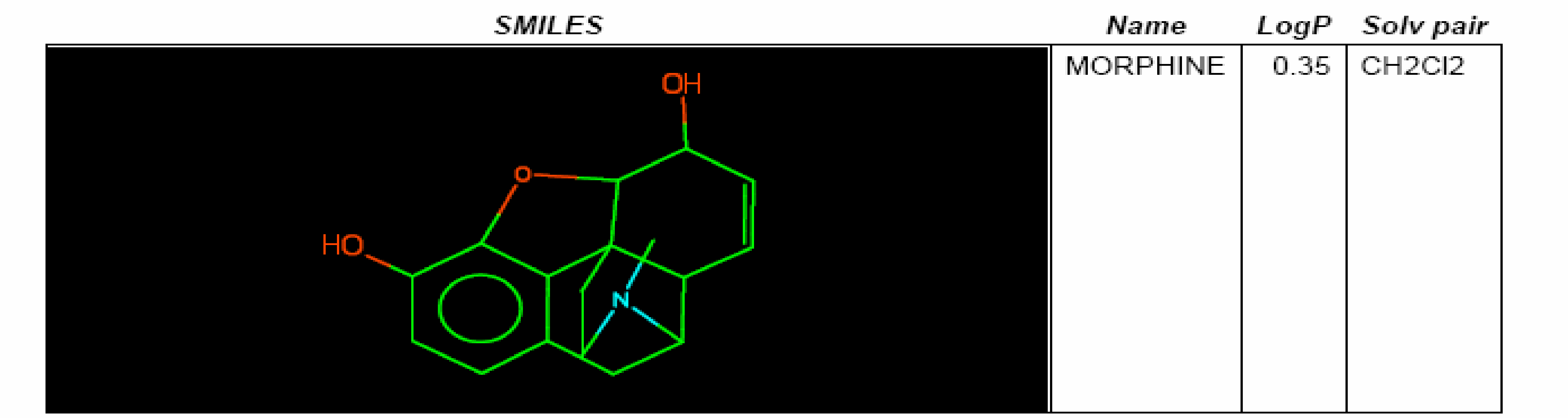
Database operations

- There are three basic database operations in Merlin:
  - **Searching**: Merlin provides a number of search services. You can search for particular text or "ranges" of text (e.g. names, properties, activity, etc.), numeric ranges, similar molecular structures, substructures, and superstructures.
  - **Sorting**: Merlin can sort information using a variety of "comparison functions," including numeric, alphabetic, molecular formula, CAS number, etc.
  - **Selecting**: You can use several techniques to select items of interest "by hand."

Chemical spreadsheets

- Merlin presents the database as a "chemical spreadsheet" called a *pool*.. The data are seen in *rows* and *columns*. The results of searches are stored in *hitlists*.
  - A **pool** is a THOR database that is loaded into the computer's memory.
  - A **row** is Merlin's representation of a THOR datatree. That is, all data in one row are from a single TDT in one database.
  - A **hitlist** is an ordered subset of the rows in the pool. Sorts and searches modify hitlists: A search adds or deletes rows from a hitlist, and a sort changes the order of the rows in a hitlist.
  - A **hit** is a row that is currently in a hitlist.
  - A **column** is a "vertical slice" through the pools, and contains data of one particular datatype. For example, a column might be for the datatype "Name," the column would contain a name from each row of the pool.

Caveat



SMILES	Name	LogP	Solv pair
	MORPHINE	0.35	CH2Cl2

- The intersection of a row and column is called a **cell**.
- The content of the cell may be dependent on the content of adjacent cells.
- Whilst the integrity of this relationship is maintained in thor which retrieves data as it was stored, some merlin *clients* such as **xvmerlin()** do not maintain this relationship.
  - In the example above the logP value is that for octanol.

Servers and clients

- It is important to note that both THOR and Merlin are client/server systems.
- The server's role is relatively straightforward:
  - It has exclusive access to all databases that it opens
  - It performs all database transactions.
- Clients never access a database directly, but rather access all data via a server.
- The server can serve many clients at once, so normally only one server runs at a time on each computer, no matter how many clients are running.
- Users can write their own client programs using the toolkits

Thor anatomy

- A Thor chemical database, usually thought of as a single entity, is made up of as many as four databases, to store datatypes, indirect references, monomer definitions and chemical information.
  - **datatypes**
    - The *datatypes-definitions database* contains the definitions of all datatypes. Datatypes are frequently common to all databases at a particular site (i.e. all regular databases refer to the same datatypes database); this makes the data from all databases intercompatible.
  - **indirect data**
    - The optional *indirect-data database* contains the expansions for indirect data. Like datatypes databases, indirect-data database are often shared by several related regular databases.
  - **monomers**
    - The optional *monomer-definition database* contains monomer definitions for combinatorial-library (mixture) databases that use the CHUCKLES and CHORTLES languages.
  - **chemical data**
    - The *regular*, or *chemical-information database*, contains data about chemicals.

Thor database files

- **description**
  - The *description* file (suffix **.THOR**, also called the *header* file) describes the database. It contains the names of the other files, timestamps for each of the constituent files, the database's encrypted passwords, and the names of the databases where datatypes and indirect data can be found.
- **lockfile**
  - A *lockfile* (suffix **.LCK**) is present whenever a Thor server has the database open. It contains the process ID of the Thor server, and prevents two Thor servers from opening the same database.
- **primary data**
  - The *primary data file* (suffix **.DP**) contains all of the data in the database; this is where Thor datatrees are stored. A database can be completely rebuilt from the contents of this file.
- **primary hash**
  - The *primary hash file* (suffix **.HP**) contains the hash table that allows "order one" (constant time) access of the primary data via SMILES.
- **cross ref.**
  - The *cross-reference file* (suffix **.DX**) contains a cross-reference for each non-SMILES identifier; each non-SMILES identifier is listed with the SMILES of each TDT in which the non-SMILES identifier appears.
- **cross-ref. hash**
  - The *cross-reference hash file* (suffix .HX) contains the hash table that allows "order one" access of non-SMILES identifiers.

Normalisations

- For the hash look-up system to work identifiers must be normalised.
- A list of normalisations is available here
- Note that some of these are data normalisations. For instance it is imperative that the 2D coordinates of a structure remain ordered correctly when a SMILES is normalised.

Daylight
Chemical
Information
Systems, Inc.

Enterprise-level cheminformatics.

Thor datatypes

- Datatype definitions are expressed as TDTs. This requires that there be at least a minimal set of predefined (truly universal) datatypes (also known as "hard-coded" or "bootstrap" datatypes).
- A full list is available here
- This set is used to define the meanings of the user-defined tags.
- Example files are in $DY_ROOT/data/datatypes/
- The tdt file is loaded into the _datatypes database
- This database can be accessed by thor clients like a regular database.

Thor indirect data

- *Indirect data* are data that are stored separately from the regular data in a database.
- Indirect datafields are defined via the "normalization" specification by adding the INDIRECT specification to the normalizations.
- A field thus marked will contain an *indirect reference* rather than the actual data of interest.
- When the TDT is retrieved from the database, an i*ndirect-reference expansion* takes place:
  - The indirect reference is looked up in an auxillary database, and the expansion data replaces the original indirect reference.

Thor administration clients

- sthorman
- thorchange
- thorcrunch
- thordbinfo
- thordbping
- thordelete
- thordestroy
- thordiff
- thordump
- thorlist
- thorload
- thorlookup
- thorls
- thormake
- thorping
- thorwho

Daylight
Chemical
Information
Systems, Inc.

Enterprise-level cheminformatics.

Merlin administration clients

- merlindbping
- merlinload
- merlinls
- merlinping
- merlinsmartstalk
- merlinwho

End-user query tools

- mcl

- xvmerlin
- xvpcmodels
- xvthor

Daylight
Chemical
Information
Systems, Inc.

Enterprise-level cheminformatics.

Database management

- Virtually all database management is carried out by client programs which pass instructions to the appropriate server. However
  - *In extremis* **thordump()** will dump the contents of the *.DP *file* as tdt's.
  - As the tdt's are ASCII characters the database content can be manipulated externally *via* editors and text utilities such as **awk()** and **grep(),** and the database rebuilt. For example, thor_find_and_replace() replaces the entire content of a subtree.
  - By tying identifiers to data, as in the thor model, the integrity of the data is not compromised by a updates and changes.

Practical exercises

- Build a database using the $DY_ROOT/data/example* files
- Load into merlin
- Set fingerprint size to be 2048 fixed width
- Reload into merlin
- Count number of SMILES in file
- Count number of WLN’s in file
- Retrieve records with more than one WLN
- Make merlin produce a new row for each WLN
- Add $DY_ROOT/data/logpstar00 compounds to database, along with their clogP and CMR values