THE DATA EXCHANGE FORMATS ON THERMODYNAMIC PROPERTIES OF INDIVIDUAL SUBSTANCES

Belov G.V., Iorish V.S.

IHED, IVTAN Association of RAS, Thermocentre, Izhorskaya 13/19, Moscow, Russia,
gbelov@imail.ru

The prevalence of computers, data bases and Internet technologies cause the necessity of development of the unified data exchange formats on thermodynamic properties of substances.

Here we would like to present our proposals on the subject, based on our experience of data exchange among databases on thermodynamic properties of substances.

Basic assumptions concerning users are as follows. There are two groups of physico-chemical data users:

- "Simple" user who just needs some data to solve the definite problem (engineer, researcher, student).
- "Advanced" user who develops or maintains information systems or software that needs the physico-chemical data in electronic form.

We will consider only "advanced" users.

Basic concepts are as follows.

- Data exchange formats should be determined by the data model.
- Role of language (English, Russian, XML, HTML, Basic, C++, etc.) is secondary.
- Naming conventions are important.

Data model is an integrated set of concepts to describe the data, their relations and existing constraints (for more details on data models see *Connolly T.M., and Begg C.E. Database Systems. A Practical Approach to Design, Implementation, and Management, Addison-Wesley, 1999*).

Kinds of data models

- External data model: is determined by the subject of investigation.
- Conceptual data model: reflects the logical concept of the data, independent of definite DBMS.
- Internal data model: reflects the conceptual scheme in accordance with selected DBMS.

Components of data model are

- Structure: set of rules to design the database.
- Composition: list of fields in tables of database with their types.

- Management: set of available operations with the data.
- Integrity constraints: guarantee the validity of the data.

Information on physico-chemical properties may be stored either in special data files (flat files database) or in format of one of database management systems (DBMS) format (SQL-oriented database) such as dBase, Paradox, Access, etc. SQL-oriented DBMS provide the following possibilities for data exchange:

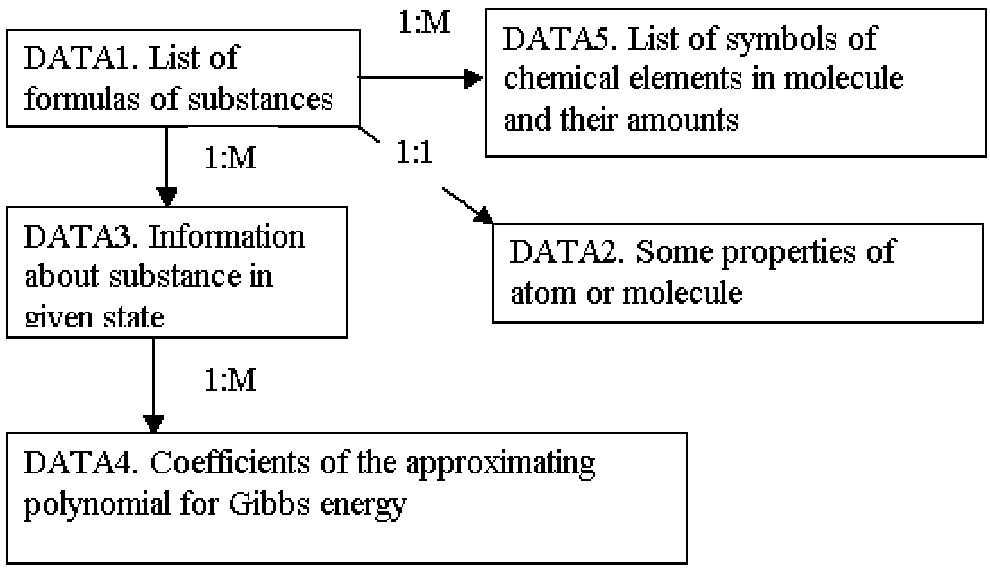
- Easy selection of information from required fields and records of the database according to given criteria.
- Easy export into CSV (text) or XML files.
- Easy import in any database.

There are many possible data export formats. CSV (or text file) is among the most simple of them. An example of such a file is presented below. Data in such format may be easily imported in any database provides the type (and meaning) of each field is known.

```
CAS Number; Date; Tfus;TMAX; dCp(298); dH(298); dS(298);
dF(298); dCp(Tfus)S; dH(Tfus)S;dS(Tfus)S
262-12-4;22.06.98;;;5;2000;3;3;8;3000;5
39227-53-7;22.06.98;;;7;2000;3;3;9;3000;5
39227-54-8;22.06.98;;;7;2000;3;3;8;3000;5
54536-18-4;22.06.98;;;7;4000;6;6;10;5000;8
50585-39-2;22.06.98;;;7;4000;6;6;10;5000;8
54536-19-5;22.06.98;;;7;4000;6;6;10;5000;8
38178-38-0;22.06.98;;;7;4000;6;6;10;5000;8
82291-26-7;22.06.98;;;7;4000;6;6;10;5000;8
```

There is no problem when all the data are stored in one table (provides the table is at least in third normal form). But when the data model is some more complex it is necessary to transfer its structure too. An example of such a structure is given below.

Thermodynamic database structure

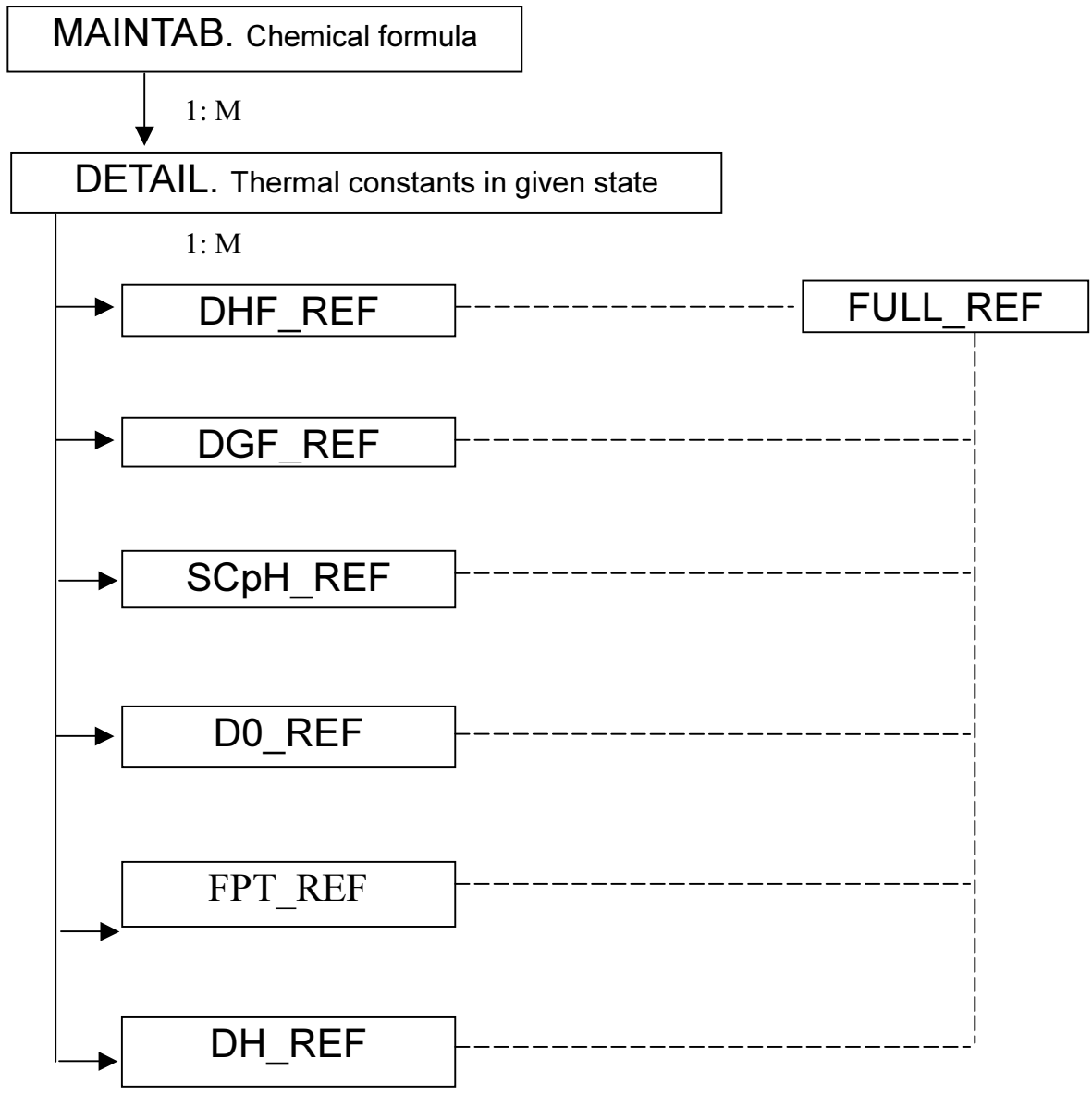


Each table (DATA1, DATA2, etc.) consists of fields that have name, type and meaning. An example of description of table composition is given below.

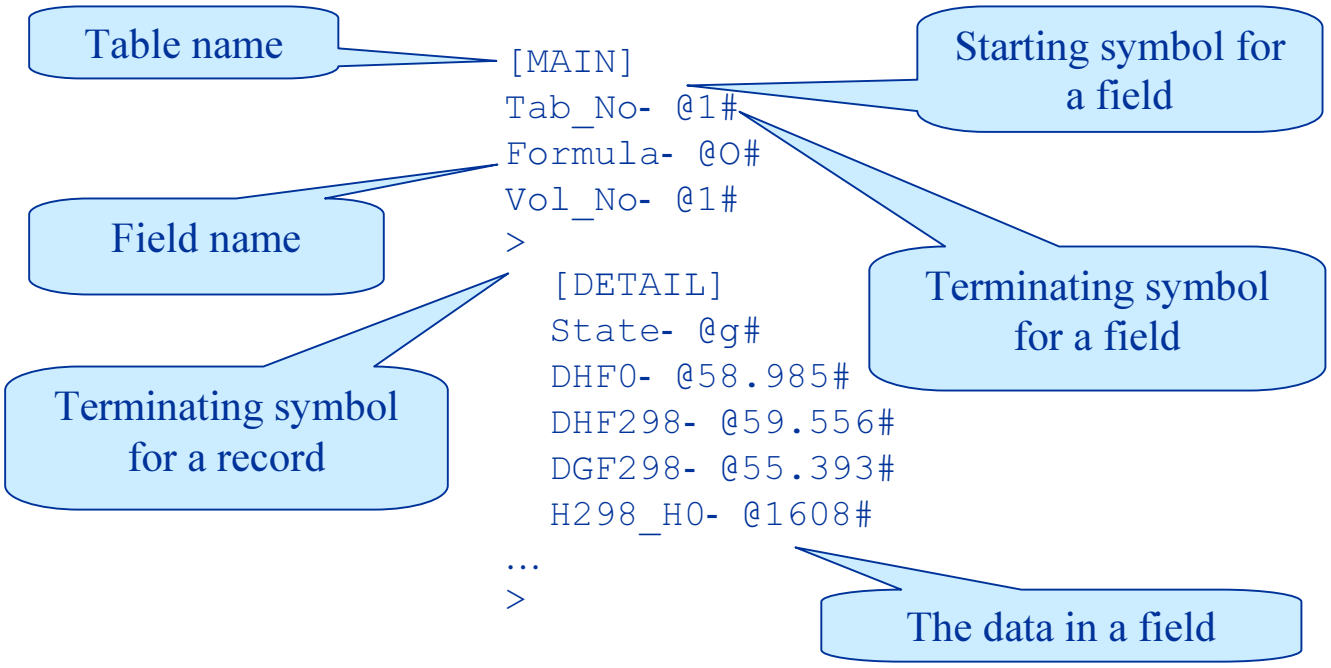
DATA1.

Field	Type	Description
D1_NO	Autoinc	Primary key
Formula	String (20)	Formula
IonCharge	Integer	Ion charge
MolMass	Float	Molecular mass
Snucl	Float	Nuclear spin component
PotIon	Float	Ionization potential
ATQTY	Integer	Amount of atoms in molecule

Below there is presented the fragment of structure of database on thermal constants of substances that contains values of the thermal constants for 26976 substances. The list of references contains more than 51500 entries. The work was supported by RFBR grant.



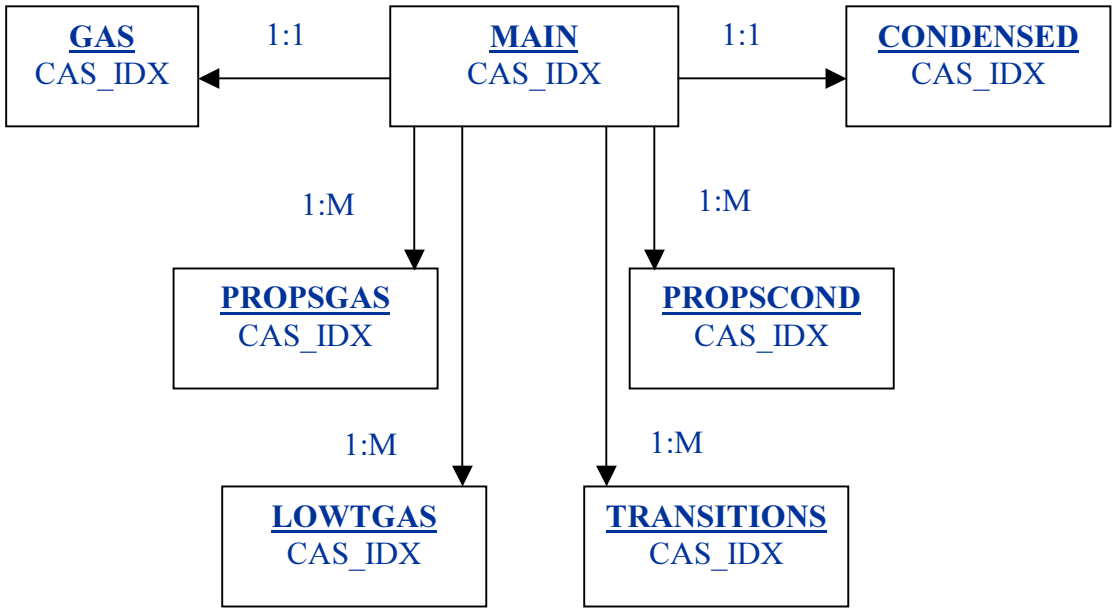
One of the problems that were met in accomplishing of the project is transfer of data from one SQL database (Paradox) into another SQL database (PosgreSQL). To solve the problem a very simple data transfer format has been developed. One can see it below.



There is nothing special about it, and this example shows only one of many possible data exchange formats.

The next pictures show the structure and example of the table composition developed in process of accomplishing of IUCODIX project.

Structure of the database for IUCODIX project



GAS (Enthalpy of formation, thermodynamic functions at room temperature and uncertainties of functions of gaseous substances)

Field	Type	Description
CAS_IDX	STRING(20)	CAS
DFH298	Float	Enthalpy of formation at 298.15K, $\Delta_f H(298)$
DFH0	Float	Enthalpy of formation at 0K, $\Delta_f H(0)$
Cp298	Float	$Cp(298)$
S298	Float	$S(298)$
H298_H0	Float	$H(298)$-$H(0)$
TMAX	Float	Upper temperature range
dCp298	Float	Uncertainty of $Cp(298)$
dH298_H0	Float	Uncertainty of $H(298)$-$H(0)$
dS298	Float	Uncertainty of $S(298)$
dF298	Float	Uncertainty of $F(298)$
dCp1000	Float	Uncertainty of $Cp(1000)$
dH1000_H0	Float	Uncertainty of $H(1000)$-$H(0)$
dS1000	Float	Uncertainty of $S(1000)$
dF1000	Float	Uncertainty of $F(1000)$
dCpTMAX	Float	Uncertainty of $Cp(T_{max})$
DHTMAX_H0	Float	Uncertainty of $H(T_{max})$-$H(0)$
dSTMAX	Float	Uncertainty of $S(T_{max})$
dFTMAX	Float	Uncertainty of $F(T_{max})$
dDHf298	Float	Uncertainty of $\Delta_f H(298)$
STATE	STRING(10)	State
DATE	STRING(8)	Date of modification

XML fields description for the table GAS

Extensible mark-up language (XML) is very popular now as a universal language of data exchange. It provides possibility to transfer data and describe their types. Now many popular database management systems have built-in functions for the data import/export in XML format. Below there is presented a fragment of the XML data description from the above table.

```
<METADATA>
      <FIELDS>
            <FIELD attrname="CAS_IDX" fieldtype="string" WIDTH="20"/>
            <FIELD attrname="DFH298" fieldtype="r8"/>
            <FIELD attrname="DFH0" fieldtype="r8"/>
            <FIELD attrname="Cp298" fieldtype="r8"/>
            <FIELD attrname="S298" fieldtype="r8"/>
            <FIELD attrname="H298" fieldtype="r8"/>
            <FIELD attrname="TMAX" fieldtype="r8"/>
            <FIELD attrname="dCp298" fieldtype="r8"/>
            <FIELD attrname="dS298" fieldtype="r8"/>
            ...
      </FIELDS>
</METADATA>
```

Our experience shows that the choice of names is very important part of the database design as inadequate name for a field may result in many errors and misunderstanding. So the field names should be self-explanatory, e.g. F, G, H, Cp, DfH298, H298_H0, etc.

One can ask: is the data model necessary. The answer is – no. See for example fragment of data in traditional IVTANTHERMO data transfer format below.

```
1*O  2                        (G)
2*1-B O2                   DIOXYGEN
3*O2=2O                                 31.99880000      493.590
4*  100.000   29.112   144.185     173.195     2.901   -253.4678
4*  200.000   29.127   164.313     193.375     5.812   -123.9807
4*  298.150   29.378   175.925     205.038     8.680    -81.2055
4*  300.000   29.387   176.105     205.220     8.735    -80.6671
4*  400.000   30.109   184.498     213.763    11.706    -58.9457
…
4*19500.000   28.208   321.559     358.981   729.718      5.8099
4*20000.000   27.837   322.504     359.690   743.728      5.8529
5*      .000             .000                .311
    21.06.90
6*  298.15 1500.00+2.49223327637E+02+2.01541175842E+01+1.03128096089E-03
  -2.26409465075E-01+1.40365325928E+02-2.95217285156E+02+3.47507507324E+02
6* 1500.00 6000.00+2.79429565430E+02+3.03885231018E+01+6.64922781289E-03
  -1.48563291878E-02+2.20714874268E+01-8.06993484497E+00+1.72535896301E+00
```

As one can see the data are “piled in a heap” (though the heap is in good order) and additional efforts are necessary to parse it. The main disadvantages of “modelless” approach, which in our opinion is not manufacturable, are

- problems with data format modification (legacy problem),
- software development is more difficult,
- higher requirements to skill of software developer.

Summing up all mentioned above, the data exchange information should contain

- data model: structure and composition of the data (including field names, types, description, units, constraints);
- data file(s) description: XML or any other language;
- data file(s) itself.

Proposals.

To simplify the data exchange it is necessary

- to develop the principles of data field names forming (naming conventions);
- to develop the list of names for all data stored in databases and make this list available via internet;
- to develop agreement on physical data units.

All information should be opened to modification, may be some technical committee should be responsible for it.