The Nanosoft Package

R.W. Greenyer and P.W. Power

December 2021

1 Introduction

This package owes its beginnings, in the first instance, to Russian Physicist Dr Alexander Parkhomov, who published a paper in February 2018 entitled Multeity of Nuclides Arising in the Process of Cold Nuclear Transmutations. In this, he explained how, using a computer program for possible combinations of 280 stable nuclides, he was able to generate transmutation tables for 1389 fusion, 817 fission and 516789 transposition (2-2) reactions, all exothermic.

On the suggestion of co-author Bob Greenyer of the Martin Fleischmann Memorial Project, Dr Parkhomov generously made all these reactions publicly available in downloadable spreadsheet form. Bob then called for help to produce friendly interfaces for the data.

In response, programmer Denis LaMotte produced an easy-to-use interface for the spreadsheets while co-author Phillip Power produced an SQL based implementation of them. This latter is more complicated but it allows for more sophisticated use, especially when combined with other, publicly derivable nuclide-related tables and the use of the PHP programming language. This multi-pronged approach has evolved into the current Nanosoft package.

Dr Parkhomov extended his work to include the roles of neutrinos in these reactions and published a follow-up paper, in November 2018, entitled Multeity of Nuclides Arising in the Process of Cold Nuclear Transmutations Involving Electrons. He further expanded on his theory of low-energy neutrinos in a recent paper in Russian. Bob Greenyer offers his English translation of it here. As well, Dr Parkhomov generously released a further spreadsheet containing this new set of neutrino-including reactions and these were duly integrated with the original data and software.

Thus, the Parkhomov tables describe three types of nuclear reactions, Fusion, Fission and 2-2, the latter mostly referred to as ‘TwoToTwo’ to avoid any confusing mixtures of letters, numbers and symbols in the code. The package includes these correspondingly named programs (Fusion, Fission and TwoToTwo) which can separately query its own table. In the TwoToTwo assumption, two ‘input’ nuclides react to produce two ‘output’ nuclides. Strictly, as Dr Parkhomov points out, many of these three types are actually chains of more complicated reactions but, to avoid runaway complexity, only their stable end products and net exothermic energies are represented in the spreadsheets/tables.

To allow for separate querying of the original neutrino-less set of tables and the combined, neutrino-including set, the suffix ‘All’ was employed for the table names. While the original versions of the Fusion, Fission and TwoToTwo programs only queried their corresponding neutrino-less tables, the latest versions of these three programs will, by default, query the *.All versions of the tables (FusionAll, FissionAll and TwoToTwoAll), each of which is a super-set of the original, with the new field, “neutrino”, able to select the original table, if only set to the value ‘none’, but having two other settings – ‘left’ and ‘right’ – which can select the corresponding neutrino reactions. If the user leaves all three options ticked, the query will be on the ‘*.All’ version of the table (also dubbed the ‘super-table’).

A pair of fields, ‘nBorF’ and ‘aBorF’, each with possible values of either ‘b’ or ‘f’, was also added to each of the above Parkhomov tables. These denote whether a given nuclide is considered a Boson or a Fermion in each of its atomic and nuclear configurations. Each is set by examining the nuclide’s A and Z number. Thus, an atom’s nucleus is considered a Boson (‘b’) if it’s A number is even; if odd, a Fermion (‘f’). The atom itself is considered a Boson if its number of neutrons (A - Z) is even; if odd, a Fermion.

The nBorF and aBorF fields can be used either passively, by noting their appearances in the rows resulting from a given query and/or actively by setting them as part of the query itself. To facilitate the latter, the programs Fusion, Fission and TwoToTwo include selection boxes, whereby ‘bosons-only’, ‘fermions-only’ or ‘either’ may be chosen for each input and output nucleon to be converted to the corresponding SQL query before execution.

In addition to the exclusively Parkhomov-derived tables, the Nanosoft package includes three publicly derived tables, specifically ‘ElementPropertiesPlus’, ‘RadioNuclides’ and ‘Atomic Radii’ (whose contents were derived from the Dynamic Periodic Table, RADAR and Atomic Radii of the Nuclides sites respectively) as well as the ‘Nuclides’ table which was derived from data from Dr Parkhomov’s ‘Selected Isotopes’ table and various internet sources. Recently, an extended nuclides table, ‘NuclidesPlus’, has also been added (see below).

All these are included to allow for the many established chemical and physical properties of elements and nuclides to be incorporated into SQL queries and so refine the theory or observation that the user may be investigating. Their structures are displayed by the program Tables In Detail, while the program Show Element Data shows all data from the above tables for any element that the user may select.

Again, the universal program All Tables allows for any and all of these considerably longer and more complex table and field combinations to be explored. More new tables, or new fields in existing ones, could be added as future needs may arise.

The package makes obvious demands on the user’s own knowledge of SQL. Certainly, the longer, more complicated queries that the All Tables program is especially intended to cope with, will test this knowledge. However, there are many good books, sites and courses about SQL itself and it is not proposed to supplement them here. In fact, an intelligent user can likely acquire a good working knowledge of SQL simply by browsing the publicly available list of results from the All Results link.

In this regard, note that the MFMP is constantly monitoring the growing list of results and, in the event of a new user posting an empty result because of an SQL error, the Webmaster will typically re-run it with what he assumes to be the command the user intended. Again, as is fundamental to the MFMP, all results are open to the world.

2 The Cascades Programs

Going beyond the direct querying of a few tables, the Cascades sub-package of programs make extensive use of the PHP computer language as well as MySQL (which PHP uses to implement SQL). They seek to predict, at least qualitatively, the likely nuclides that result from ongoing (cascading) LENR reactions initiated from fusion and 2-2 reactions performed on user-designated fuel nuclides.

They evolved out of Bob Greenyer’s exploratory, step-by-step ’feeding back’ of the reaction products from these core reactions. The subsequent Cascading process was automated, subject to the setting of a number of parameters, including those necessarily aimed at simply staying within the memory and processing power of the server(!) Specifically, setting too many nuclides (over 100) to be paired off or too many loops (over 3) to explore these pairings has been known to crash the current server. All the outputs of the Cascades sub packages now include the number of seconds of time a given run took. Typical values are about 1000 seconds. This means that, while all runs of the Cascades suite carry the risk of crashing the server, it is advisable to wait and see whether it will ultimately successfully file the results, even if it may appear to have ’died’.

The user can also decide, by means of switches, whether to feed-back any given reaction product according to its bosonic or fermionic field. However, this can, in turn, be affected by whether a nuclide could form Dimers – always bosonic. Specifically, a switch in the Cascades programs may be set to reflect the fact that the elements H, D, N, O, F, Cl, Br, I, Li, Na, K, R and Cr can form Dimers and so always be fed back, if the user so chooses.

The Cascades programs also need to reflect the fact that, if the temperature is below the melting point of a given element, that element would (arguably) be unable to participate in subsequent reactions.

Accordingly, a temperature threshold box and a switch can set that criterion.

Similarly, a switch can be set as to whether or not to feed back an element that would have boiled off on or before the above temperature is reached.

Another parameters settable by the Cascades programs include minimum energy thresholds for Fusion and 2-2 reactions to be deemed significant. The program will also output the number of reactions looked for, subject to all the parameter settings, but not found.

The Cascades sub-package currently comprises the programs CascadesAll, Cascades4, Cascades5Lite and Cascades5. The four all share the same post-input software and only differ in the way the user can specify the core fuel nuclides.

Specifically, in CascadesAll (the original), the user must specify the common input SQL core of the Fusion and the TwoToTwo programs. That core command is executed for each of the two programs and all resulting output nuclides then comprise the starting list.

In Cascades4, the user must directly list the **elements** for the core fuel. The program will then find all nuclides/isotopes of each element and send these to the common post-input processing software.

In Cascades5Lite, the user must directly list all the **nuclides** for the core fuel. This more detailed list allows for particular nuclide isotopes to be excluded from those that Cascades4 will always include.

In Cascades5, the user has even greater flexibility than in Cascades5Lite. Here he/she is presented with a complete list of all entries in the chosen Nuclides tables.

The user first sets the various program-wide parameters and switches (common to all the Cascades programs) and then clicks on the ‘Refine Settings’ button to observe exactly what feedback those settings would produce for each nuclide. He/she may then further set and refine those settings as well as override their feedback effects for each by setting its Master Switch. The Master Switch is also used to specify the core fuel nuclides.

Finally, the user clicks on the ‘Find Results’ button and the post-input software common to all Cascades* programs proceeds. In this, each of the core and/or fed-back nuclides is tried in all possible pairings with itself and each of the other nuclides in both the Fusion and 2-2 modes.

This is a recursive process, in that any nuclide appearing on the RHS of a Fusion or 2-2 reaction and that also satisfies all the criteria implicit in the switch settings will be ‘fed back’ and so automatically join in the paring process. This process places limits on the server’s memory and processing power and, in practice, the user may need to experiment with the settings for maximum number of nuclides to be paired, the number of loops and the energy thresholds for inclusion in the reaction table.

3 Results

In the spirit of Open Science, all files of results are, at least initially, stored on the Nanosoft site, on the All Results page and anyone may study, copy or download them. The name assigned to each reflects, as much as possible, the SQL command used to generate it and users and visitors are free to link to any of these in their own reports and papers.

Note, however, that the site’s storage capacity is not unlimited and the Webmaster reserves the right to prune any results files at any time. That applies especially to older files and to those that have produced no rows because of errors in the SQL format. As the programs advise, users should make their own copies.

A good rule is for the user to make routine use of the ‘limit’ sub-command to obtain a preliminary idea of how a particular SQL command will perform. If the results look as expected, he/she could then re-run the command with extended limits or, equally, with extra sub commands to eliminate any irrelevant rows, thereby converging on the ideal SQL command for testing their particular idea, theory or data-set.

4 Using the Nanosoft Package

4.1 Using the ‘Big Three’ Programs

The commonest use of the package involves one of the ’Big Three’ programs: Fission, Fusion, and TwoToTwo. The user need only enter the core of the SQL command and the prefix ‘select * from ‘table”, where ‘table’ matches the particular program, will be automatically prepended. For the original Parkhomov tables, ‘table’ may variously be FissionAll, FusionAll and TwoToTwoAll. For the extended tables, it will be FissionAllNewPlus, FusionAllNewPlus and TwoToTwoAllNewPlus. These are automatically applied once the user selects which of the two reaction table sets to be used.

One user example is:

Click on the ‘Fission’ link from the main menu. Leave all default settings and enter this (minus the double quotes) into the core command box:

“E = ‘Pb’ and MeV > 10 order by MeV desc”

In this, we are querying the original (but neutrino-including) Fission table of all possible splittings of Lead, up to where the net energy released is greater than 10MeV. As is standard with ‘Big Three’ queries, the results are displayed in three tables:

1. The **Results Table** itself, which lists, in the order given (here, of MeV in descending order), all the entries that were found in the ’FissionAll’ table. In this instance, 38 rows were found.

2. All **Nuclides** Appearing in Results Table. In this instance, 39 were found, and are listed in ascending order of their Atomic Numbers (Z) and then by their Atomic Mass Numbers (A) from their entries in the ‘Nuclides’ table.

3. All **Elements** Appearing in Results Table. In this instance, 17 were found and they are listed in ascending order of their Atomic Numbers from their entries in the ‘Elements’ table.

This format of results presentation - of Rows Found, Nuclides Appearing and Elements Appearing - is common to all ”Big Three” Programs. In each run, the file containing all these, together with explanatory notes, is stored on the site and this and all previous results from all runs can be perused by anyone who clicks in the “All Results” link on the main menu.

Another example, this time querying the “FusionAll” table, can be generated by first clicking on the ‘Fission’ link on the main menu, again accepting all the default settings and entering this (minus the double quotes) into the ‘Core Query’ box:

“E1 in (‘H’, ‘D’) and E1 in (‘Ni’,‘Li’,‘Al’,‘B’,‘N’) order by MeV desc limit 100”

In this, we are querying the Fusion table to look at how either hydrogen or deuterium may react with a cocktail of Ni, Li, B and N. Note that the limit of 100 rows is not reached here.

A further example, this time querying the 2-2 table, can be generated by first clicking on the ‘TwoToTwo’ program on the main menu, again accepting all the default settings and entering this (minus the double quotes) into the ‘Core Query’ box:

“E1 in (‘H’,‘D’) and E2 in (‘Ni’,‘Li’,‘Al’,‘B’,‘N’) order by MeV desc limit 100”

In this, we are making the same query as above but on the 2-2 table

A number of points are worth noting thus far:

1. The use of ‘in’ appearing in many core SQL sub-commands. This means that the bracketed items are all logical OR’s in the sub-command. Thus, in the above entry, the sub-command “E2 in (‘Ni’,‘Li,‘Al’,‘B’,‘N’)” is a shorter form of the equivalent, but fuller “(E2 = ‘Ni’ or E2 = ‘Li’ or E2 = ‘Al’ or E2 = ‘B’ or E2 = ‘N’)” sub-command. This shorter form can also be applied to numbers. Thus, the sub-command “A1 = 1 or A1 = 2 or A1 = 4” can be replaced by the sub-command “A1 in (1,2,4)”.

2. The user should routinely use the ”limit xxx” sub-command to limit the number of rows the whole command will produce. If no use of this sub-command is found in the core command string, the program will automatically add a ”limit 1000” to that string. Any attempt to specify more than 1000 rows will be automatically overridden and ”limit 1000”, substituted.

4.2 Using the Cascades Programs

The Cascades suite is used to explore the chain of reactions – the cascade arising from an initial reaction or set of nucleons typically present in a LENR reaction chamber. Ideally, given adequate computer power, the chain reaction will continue on until no new nuclides are being added – fed back, subject to the switch settings - to the pairing list. In practice, given the server’s considerable but ultimately limited computing power, the user may need to be content with incomplete lists.

That said, the eventual pairing lists, together with the list of reactions so far found can be impressive. Two examples are:

https://nanosoft.co.nz/results/Cascades5Lite_2390101_H1Li7Al27N14Ni58Ni60Ni62B10B11_2400K.html
https://nanosoft.co.nz/results/Cascades4_2384642_HLiAlNNiB_2400K.html

In the above, we using two versions of the Cascades suite to explore essentially the same experiment. Note the inclusion of seemingly random numbers in the links. These are generated from the server time stamp and reflect the fact that all the many parameters associated with a given run of the Cascades suite are too elaborate to incorporate into the URL and it is therefore necessary to distinguish respective links to successive but ultimately distinct runs in this simple way.

4.3 Using the AllTables Program

The AllTables program can be used to explore the full range of SQL commands ranging from simple to very complicated ones.

For Example:

http://www.nanosoft.co.nz/results/select_*_from_ElementPropertiesPlusByZ.html - In this, the user is employing AllTables in its simplest mode - to explore the whole of the ElementsPropertiesPlus table.

Similarly, we could use it to explore the entire NuclidesPlus table:

http://www.nanosoft.co.nz/results/select_*_from_NuclidesPlusByZ-A.html

However, we can also make use of a nested command:

http://www.nanosoft.co.nz/results/select_*_from_FusionAllPlus_where_E1eqOandZ2inlbselect_Z_from_ElementPropertiesPlus_where_MagTypeeqFerromagneticrbByMeVDescLimit100.html

Here, we want to query the FusionAllPlus table for all instances of elemental Ferromagnetism. Note how command uses the ‘in’ sub-command.

4.4 Recent Work

All programs in the package have been reformatted so as to allow for smaller screens, such as those of a tablet or even a smartphone.

The original ‘Nuclides’ table, comprising 293 nuclides, that Dr Parkhomov based his spreadsheets on, has now been supplemented by an extended table, ‘NuclidesPlus’, comprising 324 nuclides.

NuclidesPlus is the result of allowing all nuclides with a half life of greater than about 30,000 years to be included. Specifically, the criterion was that the log_{10} of the half life in years - ‘LHL’ - should be greater than 4.5. That rule is and was not rigid, however: even in the original ’Nuclides’ table, the likes of Tritium, Co60 and Sr90, all having short half lives, but often appearing in practice, have been retained in both tables.

Based on NuclidesPlus, new extended reaction tables ‘FissionAllNewPlus’, ‘FusionAllNewPlus’ and ‘TwoToTwoAllNewPlus’ were generated. Their MeV values can be either positive (exothermic) or negative (endothermic).

To browse these (and all other relevant tables in the Nanosoft package), the AllTables program may be given a simple SQL command, such as “select * from NuclidesPlus order by Z, A”.

In parallel with the above work, the Nuclides/NuclidesPlus and Elements tables have had new fields added to them. As a way of reflecting this additional data, the new Elements table has been named ’ElementsPlus’. Also to reflect these new super-sets, the program ‘Show Element Data’ now lists entries from the NuclidesPlus and the ElementsPlus tables. In a further enhancement to all Nanosoft’s main programs - Fission, Fusion, TwoToTwo, CascadesAll, Cascades4, Cascades5Lite and Cascades5 - all now list the nuclides (from NuclidesPlus) and the elements (from ElementsPlus) appearing in their primary results tables. All these super-set tables are best compared using their Z and A values – versus their primary SQL indices (‘id’).

In future, all the above nuclide or element tables may have fields and/or rows added to them with little or no need to modify the underlying software. Thus, further fields could be added to ‘ElementsPlus’ as their relevance to LENR may be established. Similarly, further fields and nuclides could be added to NuclidesPlus although, in the case of further nuclides, it would then be necessary to extend the corresponding Reaction Tables. In this regard, a detailed explanation of how the later Reaction Tables were generated is in the Appendix.

Another recent consideration is that of reaction threshold energies. Aside from any quantum tunnelling or other catalytic assistance such as may be provided by the generation of Exotic Vacuum Objects (EVO’s), any given reaction must receive a threshold of energy in order to commence. Basic theory holds that the maximum value of this threshold is the total input binding energy, dubbed ‘BEin’, and this new field now appears in results based on the above recently extended set of tables, specifically FissionAllNewPlus, FusionAllNewPlus and TwoToTwoAllNewPlus.

In those reactions where BEin is low – typically at the lighter end of the NuclidesPlus table – both net energy-producing and useful transmutation Fusion and 2-2 reactions suggest themselves as worthy of special attention.

A related issue is that of reaction cross sections – ultimately the probabilities of the reactions occurring in the first place. However, most listed cross sections have been measured from isolated pairs of nuclides in the vacuum of a particle accelerator, rather than in the context of surrounding condensed matter. Thus, their use in these tables should, at the least, be subject to further research. In the interim, the accumulation of experimentally based quantitative measurements from nuclides in condensed matter should begin to clarify the issue.

5 Ongoing Documentation

Descriptions of various aspects of the package are available on the Notes link. By studying these notes starting with the earliest, the user can clarify any issues as well as note how the package evolved historically.

Copies of the latest version of this paper are currently located here

6 Appendix: Generation of the Reaction Tables

In the simplest reaction tables – that do not include neutrinos – two rules must be obeyed:

1. Conservation of protons.
2. Conservation of atomic numbers.

Fission Reactions: $Z \rightarrow Z1 + Z2$ and $A \rightarrow A1 + A2$

Fusion Reactions: $Z1 + Z1 \rightarrow Z$ and $A1 + A2 \rightarrow A$

And for 2-2 Reactions: $Z1 + Z2 \rightarrow Z3 + Z4$ and $A1 + A2 \rightarrow A3 + A4$

These rules obviously come with the extra constraint that the output values of Z and A must correspond to actual nuclides or - more accurately here – entries in the relevant Nuclides or NuclidesPlus table. This constraint, together with the necessity to include all possible nuclide combinations during the Reaction Table generation, especially for the 2-2 tables, was a demanding challenge for the coding – as it also undoubtedly was for Dr Parkhomov‘s original programmers.

In the case of net exothermic reactions, the total binding energy of the output nuclide(s) must exceed the total binding energy of the input nuclide(s).

Thus, for Fission Reactions: MeV = BE1 + BE2 – BE,

For Fusion Reactions: MeV = BE – (BE1 + BE2),

And for 2-2 Reactions: MeV = (BE3 + BE4) – (BE1 + BE2)

Given that these equations amount to the differences between large numbers, it is crucially important to calculate all the binding energies in the Nuclides(Plus) table with the greatest available precision.

To calculate the binding energies, the following constants, from NIST, were used:

Mn = 939.56542052, Neutron mass in MeV

Mp = 938.27208816, Proton mass in MeV

Me = 0.510998950, Electron mass in MeV

K = 931.49433355837, 1 AMU (atomic mass unit) in MeV (= 1/12 of a Carbon nucleus).

This was, in turn, derived from the assumption that the speed of light is:

c = 299,792,458 m/s

The AMU values were all taken from this table:

http://amdc.impcas.ac.cn/masstables/Ame2016/mass16.txt

The Formula used for the BE’s was

BE = Z * (Mp + Me) + (A - Z) * Mn - AMU * K

The only exception was for pure Hydrogen/Protium (Z = 1, A = 1) where the BE was set to 0.0, to reflect the fact that H^1 is not bound to any other nucleon.

To accommodate neutrinos in either the input ('left') or the output ('right'), the original 'A' constraints can be retained but, to reflect the conversion of a proton into a neutron or vice-versa, there is variously a loss of total Z on the 'left' or a gain of total Z on the 'right'. Specifically:

Fission Reactions: $Z \rightarrow Z1 + Z2 - 1$ (left) and $Z \rightarrow Z1 + Z2 + 1$ (right)

Fusion Reactions: $Z1 + Z1 \rightarrow Z - 1$ (left) and $Z1 + Z1 \rightarrow Z + 1$ (right)

And, for 2-2 Reactions: $Z1 + Z2 \rightarrow Z3 + Z4 - 1$ (left) and $Z1 + Z2 \rightarrow Z3 + Z4 + 1$ (right).

The mass-energy loss or gain implicit in the neutrino based transformation of a proton into a neutron or vice versa is Mn - (Mp + Me) and this is variously added to or subtracted from the basic binding energy difference of output and input. The kinetic energy component of the neutrinos was neglected.