Programming Concepts as Heuristics for Finding Biological Function: a Case Study with Metaprogramming and V(D)J Recombination

Premise:

Computer programming is currently the most common way in which human designers create coded, functional systems. It is reasonable to assume that, since humans are created in the image of God, and God has created the world in order that we may understand it, that we can apply our limitted knowledge of how coded systems work to understanding God's own systems.

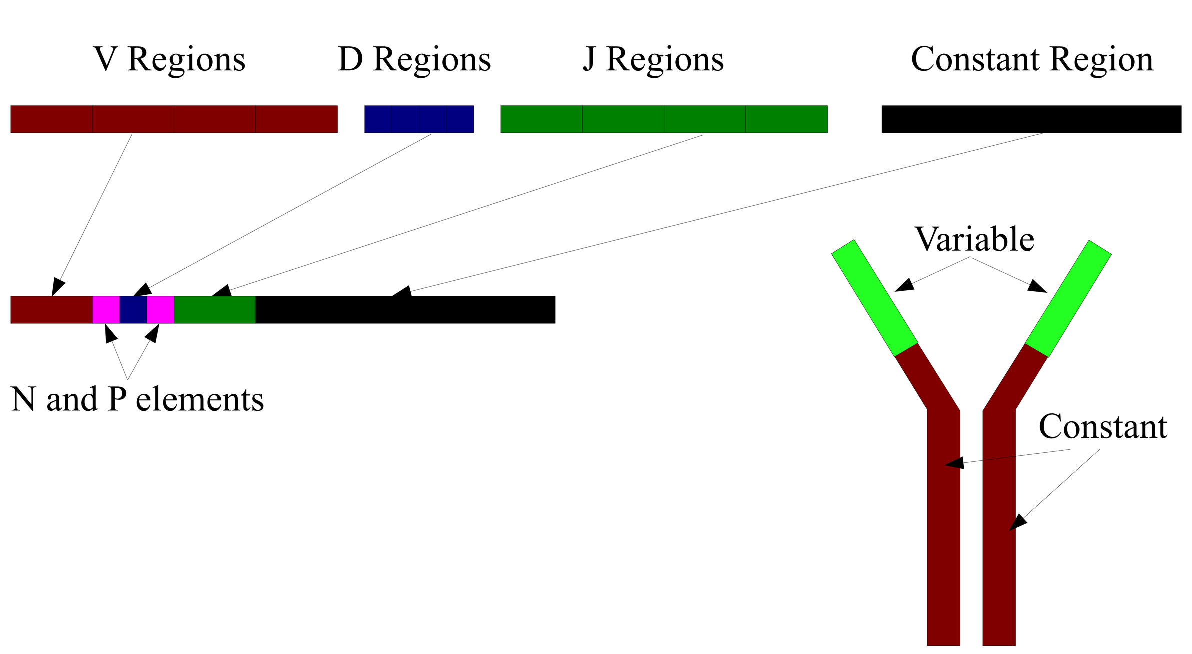
Metaprogramming:

Metaprogramming is a technique used by programmers to enable easier coding for specialized problems. Metaprogramming systems allow a programmer to use a domain-specific language, which is then rearranged and recombined to translate the domain-specific language into a standard language.

An non-computer example of this would be architectural drawings. An architect would sit down with their customer and decide, for example, where in the building to place electrical sockets. A computer program could then take that drawing, and create a new specification for the electrician that includes all of the details of where to run the wires based on pre-coded rules. The input specification from the architect is the metaprogram, and the computer program is the metaprogramming system, and the electrician's specification is the target language.

V(D)J Recombination:

V(D)J recombination takes many different gene segments from a host and recombines them into functional antibodies. There are four different types of genes – V (variable) genes, D (diversity) genes, J (joining) genes, and C (constant) genes. To make a heavy chain antibody the cell takes one each of a V, D, J, and C gene and combine them together to generate the antibody gene. Using this mechanism, just a few genes can generate millions of antibody genes.



Predicting the Function of N and P elements in V(D)J Recombination

One of the more mysterious aspects of V(D)J recombination is the origin of the nucleotides in the joints between each segment. These elements are termed N (non-templated) and P (palindromic) elements. These nucleotides are not part of the genes coding for V, D, or J. However, their current function is largely unknown. They are considered "random" with the function of generating additional antibody diversity.

However, these nucleotides are less mysterious if they are viewed in the context of a metaprogramming system. Metaprogramming systems rearrange a program in a domain-specific language into another general-purpose language. That appears to be exactly what is happening here. The domain-specific language is the V, D, and J elements with their recombination signal sequences. They are rearranged into an antibody gene which is then transcribed using the standard cellular transcription mechanisms.

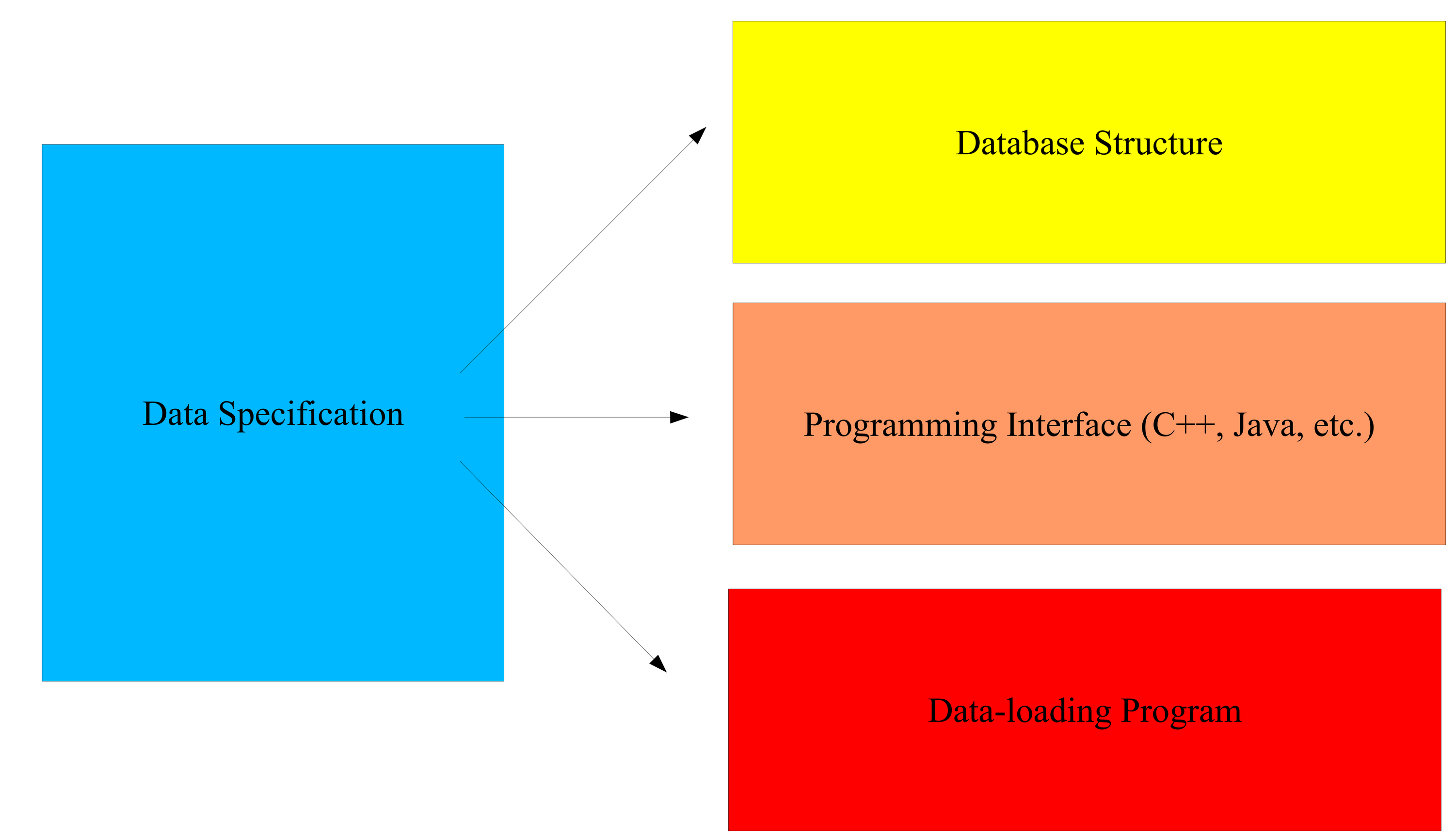
Metaprogramming systems often do more than just rearrange existing parts. They often add bits of code to make all of the rearranged parts work together correctly. Looking at V(D)J recombination from a metaprogramming perspective, it is reasonable to postulate that *the function of N and P elements is to help the pieces of the antibody gene to work together structurally and functionally*.

Evidences For:
- Certain mouse antibodies generate an essential arginine triplet during V(D)J recombination not provided by the original template segments

Evidences Against:
- Some antibodies recombine in multiple ways
- Some recombinations are unproductive

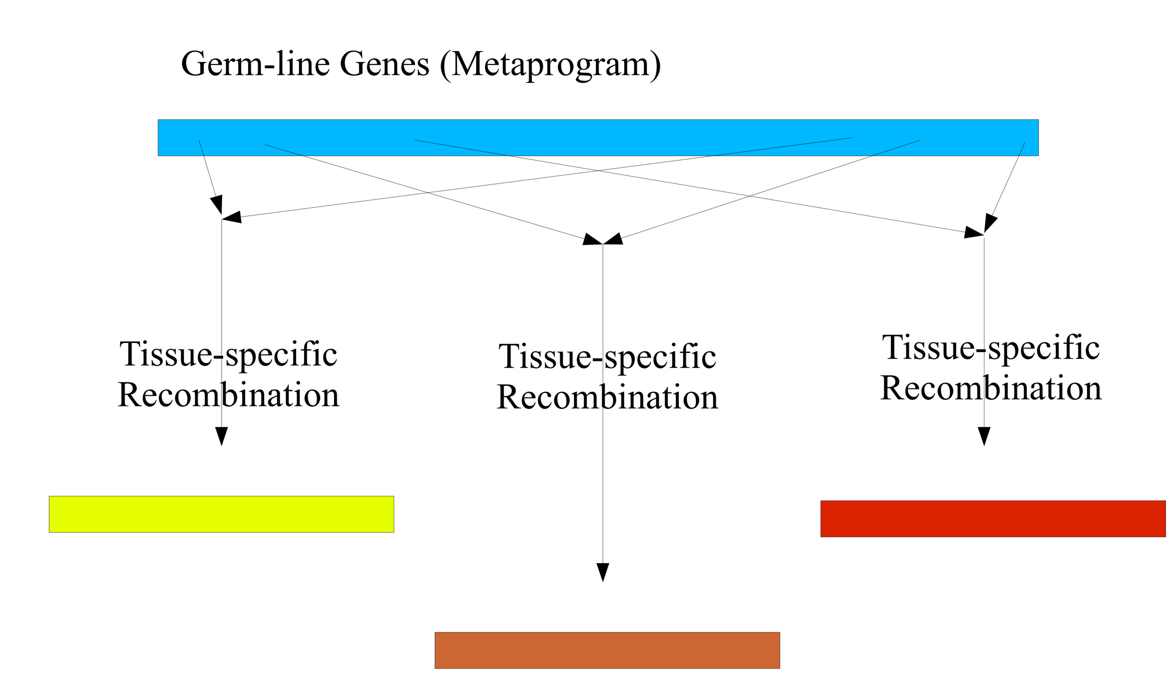
Enterprise Metaprogramming and Recombinational Systems

One recent advancement in computer science is the advent of enterprise metaprograms. An enterprise metaprogram is a metaprogram which feeds as an input into multiple different programs. An example is a data specification that could be transformed into a database table structure, a set of programming interfaces, and a data-loading program.



If human design concepts are applicable to the genome, we might expect that such a design pattern might occur within some genome architecture. This would be realized by having a single set of genes which served as a recombination template for multiple, interacting subsystems. A single change in the template gene would affect all three systems simultaneously, such that they would continue to interoperate despite their differences. This would allow cell-directed mutations to use a single change to have large-scale system changes in the genome.

What Enterprise Metaprogramming Would Look Like in a Genome



Recombined genes in each tissue are able to work in concert with one another, because they are from a common source. Directed or undirected changes in the source would affect all the genes so they could continue to interact correctly.