|  |
| --- |
| University of Advancing Technology |
| Taxonomy for Programming Languages |
| Emerging Programming Languages |
|  |
| **Caleb J Thompson** |
| **10/11/2010** |

|  |
| --- |
|  |

Table of Contents

[Nominal Language Taxonomy 3](#_Toc274568153)

[Acronyms. 3](#_Toc274568154)

[Backronyms. 3](#_Toc274568155)

[Related to Predecessor’s Name. 3](#_Toc274568156)

[Other. 3](#_Toc274568157)

[Use of NLT. 3](#_Toc274568158)

[Example Classifications 3](#_Toc274568159)

[Essential Elements. 3](#_Toc274568160)

[Differences from other paradigms. 4](#_Toc274568161)

[Representative languages. 4](#_Toc274568162)

[Resources. 4](#_Toc274568163)

## Nominal Language Taxonomy

The Nominal Language Taxonomy (NLT) is a system that categorizes languages based upon their names. A name says a lot about a programming language. Its history, inspirations, and the personalities of its designers are just a few. NLT divides languages into those whose names are acronyms, backronyms, related to their predecessors’ names, and those that are none of the above.

### Acronyms.

Acronyms in a programming language’s name are often succinct and descriptive summaries of the language’s design philosophy or primary focus. Take for example ALGOL, which is an acronym for ALGOrithmic Language. The name tells exactly what the language was designed to be used for.

### Backronyms.

The community using the language often creates Backronyms. Because of this, they can be even more useful as descriptors than Acronyms. A Backronym can often be much more critical of the language than would be an Acronym created by a language’s designer, who is biased toward it. For example, Lisp has a backronym “Lots of Infuriating Silly Parentheses”. Backronyms can also be relatively devoid of meaning, such as Ada’s “Another Damn Acronym”. Languages can be classified as both Acronym and Backronym, such as Perl: “Practical Extraction and Report Language” is the designer’s Acronym, but the community has created several Backronyms including “Pathetic Excuse for a Real Language” and “Pathologically Eclectic Rubbish Lister” (which appears on the man page for Perl).

### Related to Predecessor’s Name.

Languages whose names relate to their predecessor’s name provide some background into the language’s roots, and often include information concerning the differences between the languages. Examples include C and C++, which derive from a language named B and C, respectively.

### Other.

Languages in this category simply don’t fall into the other categories. Languages which end up here will assist in refactoring the taxonomy to be more inclusive, possibly by adding additional classifications.

### Use of NLT.

Classification of a language using NLT requires researching its name and its predecessor. Once relevant information on a language has been discovered, classification can begin. Languages can be in up to two categories. A language is classified as combination of Acronym, Backronym, or Related to Predecessor’s Name. The Other classification is exclusive as it is only used when a language does not fit into at least one of the other categories.

## Example Classifications

