Luke Palmer
2005-11-15
Linguistic Society of America FAQ.
Majid et al. **Can language restructure cognition? The case for space.**

After having a background in procedural programming languages like C++ and Perl, I remember venturing out to learn Haskell because of an open source project I wanted to work on. Haskell is a completely different language family from the other languages I knew; specifically, the functional family. But unlike Lisp and Scheme (which I had experience with), it completely forbids the programmer from falling back on procedural methods.

At this point, I could learn a new language in a day or two. Thus, I found it deeply disturbing when it took more than *two months* to learn Haskell. I spent hours and hours programming something as simple as a tic-tac-toe game. However, after a while, I started to get used to it. My speed increased, I began answering questions others asked about it, and I finally became able to work on the project.

After a while, I went back to program a few things in the languages I already knew. However, my style of programming had dramatically shifted. I was thinking about problems completely differently—the Haskell way. I abstracted things differently, factoring out algebraic structures instead (or in addition to) related data.

I have a different answer for the FAQ question: "So learning a different language won't change the way I think?" Yes, it will. In order to even be coherent in Haskell, I had to rewire my brain. Our use of language is equivalent to our ability to represent *abstractions*. If you learn a language that makes use of abstractions you are not familiar with, you will have to change the way you think to speak in that language. It is not much of a stretch for an English speaker to learn Spanish or German, because English is a descendent of both of those families. But for an English speaker to learn Arapaho is going to take some rewiring.

Frame of Reference is just one such abstraction, but it seems to be one that typically varies from family to family. In Hawaiian, an absolute language, the absolute directions are "towards the center" and "away from the center" of the island. It seems that there are no specific environmental ad-

vantages to abstracting one way or the other[1]. However, it does make sense that a language as a whole would make this distinction, as opposed to changing on an individual basis, because you wouldn't want to talk to your friend and say "it's to the north of the phone" and have your friend take a minute to calculate which way is north from the phone.

Another abstraction that was passively mentioned in the Majid et al. paper was "structure-mapping"; i.e. the method by which speakers pull out higher-order similarities in structures. This is very much akin to the change in my thought process after learning Haskell. The structures the language provides for abstraction affects the way you think about abstraction (similarly, the abstractions that are useful in your environment will make their way into the language).

---

[1]You could argue that absolute is less advantageous because it involves keeping track of your current orientation at all times; but then you could also argue that relative is less advantageous because you need to constantly calculate frames of reference. The truth is, it's only hard if that's not how you're wired to think.