Subject: Re: HASH question
Posted by Jeremy Bailin on Mon, 07 Mar 2011 19:15:56 GMT
View Forum Message <> Reply to Message

> An additional "but": If the hash keys to access the data is a vector (e.g. representing the "many of them") so would
> bethe new values (you can't assume that the new values would all be the same, right?). Assuming it was possible, do you
> think doing something like,
>
> hashtable[ids_to_change].type = array_of_new_values
>
> is o.k.? I don't (but remember, I'm a physicist who writes code so what the hell would I know? :o) That would make it
> way to easy to assign the wrong value. What if the two vectors, "ids_to_change" and "array_of_new_values", were
> different sizes? Of what if some of the keys in the "ids_to_change" vector didn't yet exist in the hash?

I do think it's okay - in fact, that's exactly what I'd expect to be able to do. And I would expect it to behave more or less in the same way as if hashtable were an array of structures which was indexed by ids_to_change: if they don't have the same number of elements, then a runtime error is thrown (a la "Array subscript for XXXX must have same size as source expression". And if there are elements of ids_to_change that are not yet in the hash, they should be created, the same as if you assigned an individual value.

Admittedly, my experience with hashes are mainly from perl, which probably colours what I think the behaviour ought to be.

-Jeremy.