What is the terminology for describing the various parts of the registry?

devblogs.microsoft.com/oldnewthing/20090204-00

February 4, 2009

Raymond Chen

Hives, keys, values, types, and data.

As I noted some years ago, the file that holds the registry data is called a *hive*.

A hive contains a tree of *keys*.

Keys contain a list of *values*.

Associated with each value is a *type* and *data*.

The terminology is weird and counter-intuitive thanks to the history of the registry. Back in the days before named values, you queried the data associated with (the default value of) a key by calling `RegQueryValue`, which was a rather natural name since it matches the key/value pattern. But the introduction of named *values* threw this pattern into disarray. Perhaps a better name could have been chosen for what today are known as values and data, but what's done is done and that's the name we're stuck with.

I'm sorry.

"Raymond, you idiot" section:

"Sure, Raymond, that's the historical reason why the terminology is messed up, but why hasn't anybody fixed it in the meantime?"

Well, changing terminology at this point would probably create even more confusion. For example, suppose you decide that the terminology should be changed as follows:

Old	New
key	node
subkey	subnode

value	key
type	type
data	value

I agree with you that this terminology would probably be much less confusing, but how do you get there from here? When you update all the documentation to change the terminology, how do you know that you covered everything? Do you grep for the word *key* everywhere and then decide on a case-by-case whether it should be changed to *node*? That's probably some hundreds of thousands of hits just inside the MSDN Library. (Even worse with *value*, *type*, and *data*.) And then there are all the comments in source code that are now wrong. And all the magazine articles written prior to the change are now wrong; who's going to go update them? And the existing source code needs to change `HKEY` to `HNODE` and `RegOpenKey` to `RegOpenNode`. Okay, so maybe you leave the old names around for compatibility, but now you have the problem that `RegOpenKey` returns a node, not a key, and that you pass a key name to `RegQueryValueEx`, and what the heck does `RegDeleteKey` do? Does it delete an old-key or a new-key?

Bonus chatter: There's also this thing called a *class*. I have no idea what it's for, so don't ask.

Raymond Chen

Follow