http://www.cs.man.ac.uk/~drummond/
Alternative interfaces for OWL ontologies - poster session
Alan Rector, Nick Drummond, Matthew Horridge, Hai Wang, Julian Seidenberg

OWL has a relatively small number of constructs, but they allow a huge amount of expressivity. The existing OWL plug-in maps very closely to the underlying language, and is a very powerful mechanism for editing and browsing OWL ontologies. However, learning about and developing ontologies is a daunting task - ontology tools and languages must be expressive enough to allow as much to be said as possible, while having a learning and usability curve that encourages their use.

For many users (or groups), often a particular application or style means only a subset of the full descriptive power of a language is required - especially when the aim is to extend a simpler terminology or taxonomy.
In addition, most users need support to allow speedy development and often-used patterns to be repeated quickly and without error. Even specific domains can benefit from a higher-level view of the language constructs.

Using real requirements from groups and exposure to difficulties experienced by beginners during tutorials, we are experimenting with several different interfaces which attempt some of the following:

1) Hiding the logical constructs and notation from the user
2) Selecting appropriate expressiveness for a given group
3) Abstracting away from the language by simulating some higher-level constructs

An alternative to the conditions widget is being developed, which provides an extended abstract syntax view of the conditions on a class. Although fully expressive, the decision to use abstract syntax means that descriptions and definitions can be clearer for logic-phobes. As well as the customary auto completion, white space, keywords and other text editor features, the widget provides "macros" or shortcuts for common patterns. Multiple Existential or Universal restrictions along the same property, along with closure, can be created in a very compact form. This widget hugely improves the speed at which complex or long descriptions can be made and promotes easy readability.

Work with a particular group, developing a medical terminology (read "ontology") has produced a simplified interface using several replacement widgets and a project file loader that asserts a template on an OWL file. The modified interface cuts down the allowed expressiveness, but includes mechanisms to cope with common naming issues - separating ontology resource IDs from their human readable names (and alternative synonyms/languages etc).
There is also a very restricted version of the conditions widget that only allows constructs required by the group to be used, and also attempts to hide the underlying representation as keywords instead of symbols.

We would be interested to use this opportunity to discuss some of the ways in which current pattern/macro support in the alternative syntax widget and wizards could be extended. Patterns should be perceived as "atomic units" and together can form what has become known in other systems as an "intermediate representation" - providing constructs that relate to OWL as java relates to machine language.