Mathematical synthesis

Peter White*
Motorola Corp., USA

At the QED '95 conference in Warsaw, I presented the work we are doing here at Motorola to synthesize software and hardware from mathematical specifications. We are in the process of evaluating the Specware toolset, which is currently under development at the Kestrel Institute in Palo Alto, California. The Specware system has the following components:

1) An algebraic specification language, strongly typed, including subsorts and quotient sorts.

2) Specification morphisms, to describe the relationship between specifications.

3) Category theoretic operations such as colimit to combine specifications, providing for features such as parameters to specifications.

4) Sheaf / category theoretics refinement operations of specifications, allowing successive introduction of details. For example, a queue specification could be refined to a partial map, which could be refined to some list implementation for LISP code generation.

5) A theorem prover to discharge the proof obligations generated during the specification and refinement phases.

6) A graphical user interface, so that specifications can be manipulated using pretty pictures.

We are having success with describing systems using Specware, but the code generations proves to be arduous.

I thought I was able to contribute the perspective of *a-user-of* the theorem provers to the conference. From the perspective of a user, I think that

*e-mail: `peter@opus.geg.mot.com`

features that make life easier (such as types and heavy duty set theory) are necessary, mathematicians will not use a system that is hard to use. Eventually, the system will require an interface that allows the notation to be as free wheeling as that found in textbooks. Perhaps an acceptance test would be to see if someone doing tensor analysis, complete with all of the tricks they play on superscripts and subscripts, would like to check his work with the system. Tensor analysis might also be a good domain since it is so easy to make an error with all of those tiny super/sub scripts.

I think another theme of the conference is to try and find a way to unify the different logics and theorem provers, in such a way that everyone is working to the same end, the creation of a large database of mathematical knowledge that is useful to industry and to mathematicians.