

---

Subject: Re: NTL and C4251

Posted by [jmansion](#) on Tue, 25 Apr 2006 12:33:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

>BTW, when developing Esc interpreter, I have never had any >formal grammar for the language  
- descent parser C++ code is >as good as the formal grammar description.

Ugh! Do your spurs jingle!

I think that's a mistake.

AST -> Abstract Syntax Tree

CoCo/R -> a recursive descent parser generator. Pat Terry had a book on the C/C++ version which is now out of print so the book is online. The Java and C# versions make life a doddle, and Pat has a new (and very good IMO) book on them.

Info is here: <http://www.ssw.uni-linz.ac.at/Research/Projects/Coco/>

I utterly dispute any suggestion that its not a timesaver. Its just so easy to iterate the language design itself, and it makes writing 'little languages' a pleasure. The output is quite readable, and will probably not be too different to what you write by hand.

I still favour a hand built scanner and LEMON for my current project, because I think I can go fastest this way. Tho I harbour a concern that actually my carefully written scanner will be only slightly faster than I'd get from re2c or ragel.

We'll have to agree to differ.

---