

Yacc, CodeWarrior, Plugin

Tristero Computer Systems P/L
(ACN 067 057 809)
P.O. Box 120,
Northcote,
Victoria 3070,
Australia

Yacc 1.9

Yacc 1.9 for CodeWarrior is based largely on Berkeley Yacc 1.9 and was converted to the Metrowerks CodeWarrior IDE by Jeff Laing. Code generation was modified specifically for CodeWarrior and a number of the more obscure “syntax error” type errors that Berkeley Yacc produces have been enhanced to provide more information.

Yacc 1.9 is a fat plugin which should be compatible with all versions of CodeWarrior (that support plugins) up to CW8 (which is the latest at this point in time). Balloon Help and AppleScript support are included for all options.

Installation:

If you are reading this document, you must have already determined that Yacc is released as DiskDoubler self-expanding archive. To install Yacc, simply copy the contents of the “-> Compilers” folder into the Compilers folder in your CodeWarrior Plugins folder and the contents of the “-> Preference Panels” into the Preference Panels folder. Each “->” folder should only contain one file.

(hint: The CodeWarrior Plugins folder looks like this  on a standard release)

Documentation:

Yacc provides a standard CodeWarrior preferences panel which is selectable by selecting “Yacc” under the “Code Generation” item (≥ 1.5) or by hitting on the  icon (in IDE < 1.5).

The following table describes each of the options that is selectable from the preferences panel.

The column “Un*x switch” refers to the command line flag which is traditionally associated with this particular field.

Field	Un*x switch	Meaning
Generate Header	-d	If selected, Yacc will produce a header file which contains the contents of the %union as a struct which can be included into lexical analysers such as those produced by [f]lex. It will also include #define statements for every token which is defined in the grammar and their corresponding numeric values.
Generate Description	-v	If selected, Yacc will produce a report which describes the productions of the source grammar.
Include #lines	-l	If selected, Yacc will include # lines in the output C source code; these can be used by some development systems to help the compiler keep track of the actual source of errors.
Include debug code	-t	If selected, Yacc will include code which can be used to debug the execution of the generated parser.
Separate Code & Tables	-r	If selected, Yacc will generate two output files, one for code and one for data; the default behaviour is to include both in the same output file.
File Prefix	-b	All files generated by Yacc will include this value as a prefix. Files are: ?.tab.c -- combined code + data ?.code.c -- parser code ?.data.c -- parser tables ?.output -- informative report where ? is replaced by the specified prefix.
Symbol Prefix	-p	All globally symbols generated by Yacc will be prefixed by this value, thus ensuring that no namespace collisions occur with other application code.

For more information on Yacc, I recommend the book “lex & yacc” by John R. Levine, Tony Mason & Doug Brown, published by O’Reilly & Associates (ISBN 1-56592-000-7). [End of plug] For a port of Lex, see your info-mac archive (in **dev/flex-251-cw.hqx**) or the Apprentice 4 CD-ROM from Celestin (in **Apprentice:Source Code:C:Applications:Flex-CW 2.5.1**)

Shareware:

Whilst my previous release of standalone Yacc for the Macintosh (still available on info-mac (in **dev/byacc-19-cw.hqx**) and the Apprentice 4 CD-ROM (in **Apprentice: Source Code:C:Applications:BYacc-CW 1.9**)) was basically a gift to the net, this plugin is explicitly not. This time, it comes with the hefty shareware fee of 15 of your major currency (thats dollars, pounds, etc not pennies or yen). Feel free to send cash or cheques payable to "Tristero Computer Systems" at the address shown above.

If you don't think its worth the money, please don't continue to use it. Feel free to use the standalone Yacc instead which is free.

Permission is explicitly given for this software to be included on compilation CD-ROMs and other collections provided that Tristero Computer Systems receive a fully registered copy of the disk it appears on. Metrowerks should give us a call. ;-)

Revision History:

This is the first public release of CW Yacc.

Bugs:

Yacc is basically still a port of Berkeley Yacc 1.9 which is freely available on the net. I have very little understanding of its internals other than as a satisfied user, so if you have problems with its grammar analysis capabilities or with the quality of the parsers it generates, sorry, but there is very little I can do for you although problems of the form "size_t is wrong" or "it uses int when it should use short int" are probably resolvable.

If you find bugs in the IDE port, feel free to report them to <jeffl@suburbia.net.au> and I will see what I can do.

To do: (but don't expect to see them on the net if I get no feedback)

- Implement %expect ala Bison
- Allow multiple parsers per project, probably by adding additional % directives that allow the overriding of the filename and symbol prefixes.
- Lift parser skeleton out of code and make it easier to customise (probably as TEXT resources inside the plugin)
- Improve the error reporting
- Build a support library for using Yacc parsers as IDE plugins.
- Port Flex
- Retrofit improvements back onto the standalone Yacc (yeah, right, like thats worth doing :-
()

Thanks

Behind every piece of hobbyist software must be a number of support people, all of whom

deserve thanks. My wife Pauline and daughter Catriona for putting up with me disappearing down to the back room when I should be keeping them company, Julie Coburn (wherever you are) for convincing the boss that I needed this PowerMac at home and Ron Liechty (MWRon) for handling my CW queries in the most friendly yet efficient manner.