pacc
a compiler-compiler

```
Rule2   ← r:Rule3 s:Rule2 { cons(r, s) }
        / ε { 0 }

Rule3   ← And u:Rule4 { s_kid(and, u) }
        / Not u:Rule4 { s_kid(not, u) }
        / And c:Code { s_retype(guard, c) }
```

(index.html)

pacc — a compiler-compiler

Release pacc-0.3

A new stable release (article/download.html) is now available. Users of Fedora, CentOS, or RHEL can now install this release from a copr (https://copr.fedorainfracloud.org/coprs/tobygoodwin/pacc/). Of course, the development source is still available from github (https://github.com/TobyGoodwin/pacc) and gitlab (https://gitlab.com/TobyGoodwin/pacc).

pacc is a compiler-compiler, somewhat like **yacc** (or **bison**). Its input is a description of a grammar, and its output is a **C** function that recognizes strings of that grammar. The significant technical difference is this: **yacc** reads a context-free grammar (CFGs), and writes a LALR(1) parser; **pacc** reads a parsing expression grammar (PEG), and writes a packrat parser.

PEGs and packrat parsing offer several advantages over CFGs.

- There is no need for a two-level structure with a separate *lexer* (this is essentially a misfeature of CFGs - they are unable to express standard tokenization rules naturally).
- PEGs can “look ahead” in the input as far as they need to.
- Despite arbitrary look-ahead, packrat parsers are linear in time and space complexity: $O(n)$ in the size of the input (whereas LALR(1) parsers are $O(n^2)$, and fully general CFG parsing is $O(n^3)$).
- PEGs are easy to understand, and pleasant to work with.

The current stable release (article/download.html) is `pacc-0.3` (bugyō) under the GPL (http://www.gnu.org/licenses/gpl.html). This is a beta release, see the TODO list

(article/todo.html) for further information. The intention is that pacc will mature to be an industrial-strength parser-generator.

The name pacc is a recursive acronym (http://en.wikipedia.org/wiki/Recursive_acronym): **pacc: a compiler-compiler**. Needless to say, pacc's own parser (https://static.paccrat.org/pacc.txt) is written in pacc.

Last updated: 2016-08-03 21:31:34 UTC

News

Porting and packaging *2016-07-29*

One thing pacc needs is more users. And, perhaps, one way to get more users is to reduce the friction in getting started with pacc. An obvious lubricant is packaging. Read More... (entry/2016-07-29.html)

Release relief *2015-07-31*

Looking at `_pacc_coords()`, I noticed that it seemed to have the same `realloc()` bug that I'd just fixed in `_pacc_result()`. However, the "list of arrays" trick really wasn't going to work here. Read More... (entry/2015-07-31.html)

See more news articles (news.html)

 (feed)

About

Implemented in Yesod (http://yesodweb.com/)
Hosted on an IPv6 VDS (https://www.mythic-beasts.com/servers/virtual) from recommended hosting company Mythic Beasts (https://www.mythic-beasts.com/)

Email us with any problems, questions, or comments

(mailto:info@paccrat.org?Subject=Response%20to%20/)