

Guile

Is

OK

hello

andy wingo

<http://wingolog.org/>

guile co-maintainer

(along with ludovic courtès)

proper macros

before: defmacro and s-exps

now: syntax-case and syntax

- with source locs and names

- modules & hygiene

- opt, kw, rest args, case-lambda

a personal revelation

guile compiles now

custom vm

features we want/need

- partial continuations
- multiple values
- appropriate fast-paths

compiler in scheme of course

native compilation planned

eval in scheme

uniform semantics

shares stack w/ compiled code

- multiple return values
- tail calls
- stack traces

bootstrap from c

r6rs

gradual integration
- modules, hygiene, library

minor things mostly ok
- letrec*, bytevectors

modules share namespace

some irritations too

gnu & scheme

unix or the lisp machine?

activate the sleeper cell

guile & emacs

implement elisp for guile runtime

allow scheme (& js, lua) in emacs

no rewrites

status

in summary

guile ($\geq 1.9.11$) is ok!

questions?