

Utilities Reference

for release version 0.2.14

mu-sys

Name: mu-sys - *minimal exec command*
Synopsis: mu-sys [*options...*] [*file...*]

Options: 0.0.2
-l *SRCFILE* load SRCFILE in sequence
-e *SEXPR* evaluate SEXPR and print result
-q *SEXPR* evaluate SEXPR quietly
-c *JSON* environment configuration
[*file ...*] load source file(s)

mu-server

Name: mu-server - *minimal exec command*
Synopsis: mu-server [*options...*] [*file...*]

Options: 0.0.2
-v print version string and exit
-l *SRCFILE* load SRCFILE in sequence
-e *SEXPR* evaluate SEXPR and print result
-q *SEXPR* evaluate SEXPR quietly
-c *JSON* environment configuration
[*file ...*] load source file(s)

mu-exec

Name: mu-exec - *minimal exec command*
Synopsis: mu-exec [*options...*] [*file...*]

Options: 0.0.2
-v print version string and exit
-l *SRCFILE* load SRCFILE in sequence
-e *SEXPR* evaluate SEXPR and print result
-q *SEXPR* evaluate SEXPR quietly
-c *JSON* environment configuration
[*file ...*] load source file(s)

forge

Name: forge - system development tool
Synopsis: forge *command* [*options...*]

Description: 0.0.18

Usage: forge 0.0.18 *command* [*option...*]
command:
help ; this message
version ; forge version
workspace init | env ; manage workspace
build release | profile | debug ; build mu system,
; release default
bench base | current | report | clean
[--ntests=number] [--all] ; benchmark test suite
regression ; regression test suite
symbols reference | crossref | metrics
[--namespace=name] ; symbol reports,
; namespace defaults
; to mu
install ; (sudo) install mu
; system-wide
clean ; clean all artifacts
commit ; fmt and clippy,
; pre-commit checking
general options:
--verbose ; verbose operation
--recipe ; show recipe

mu-listener

Name: listener - *mu* system REPL
Synopsis: listener

Description: 0.0.2

listener is a generalized command-line REPL that can be configured to load and run in any of the *mu* namespaces.

It is intended for debugging and exploration during development.

listener has no command-line arguments, it is configured by a *JSON*-format dotfile named *.listener*. An example dotfile can be found in */opt/mu/lib/listener*. If a *listener* is not found in the current directory, the user's home directory will be searched and if found there will be used. *listener* does not require a *listener* file.

```
{
  "config": "npages: 2048",
  "namespace": "core",
  "rc": "listener.rc"
}
```

The configuration syntax can specify three options, none of which are required.

The *config* option supplies a *mu* Env configuration specification in *JSON* format, see the *mu-ref* refcard *environment* section for details. If this option is not specified, the *mu* Env will be created with default values.

The *namespace* option supplies the namespaces that should be loaded, see the *mu-ref* refcard *namespaces* section for details. If this option is not specified *mu-listener* will load only the *mu* namespace.

The *rc* option supplies the name of a source file to be loaded before the REPL runs. Additional symbols may be supplied there.