Software Architecture

cURL

Daniel Stenberg

by Vanesa Alonso, Patricia García and Silvia Suárez

LibcURL

Release cycles

Contribute

History

Benefits

Security issues and layers

Further information

HISTORY

BEGINNINGS
Updated currency rates with HTTP GET

mid 90s

1998

URL GET -> cURL
Gopher and FTP support

LIBCURL AND PHP
More users, more bugs. TLS support

2000

2001-now

PROTOCOLS AND APPLE
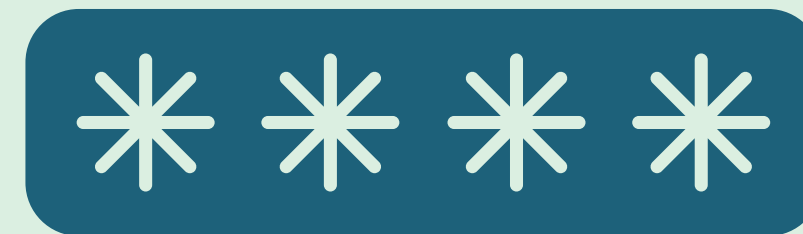More protocols and Apple adoption

LIBCURL

CLIENT-SIDE URL

Free, portable, thread-safe

SUPPORT FOR PROTOCOLS

FTP, HTTP, HTTPS, IMAP, SMPTP

SUPPORT FOR OTHER FEATURES

HTTP form-based upload, user-plus password authentication, etc.

HOW TO CONTRIBUTE?

GOT AN IDEA?

Use the mailing lists and share it!

ALREADY GOT SOME CODE?

Submit a pull request and discuss with the development team!

RELEASE CYCLE

8 week cycles
Releases on Wednesdays
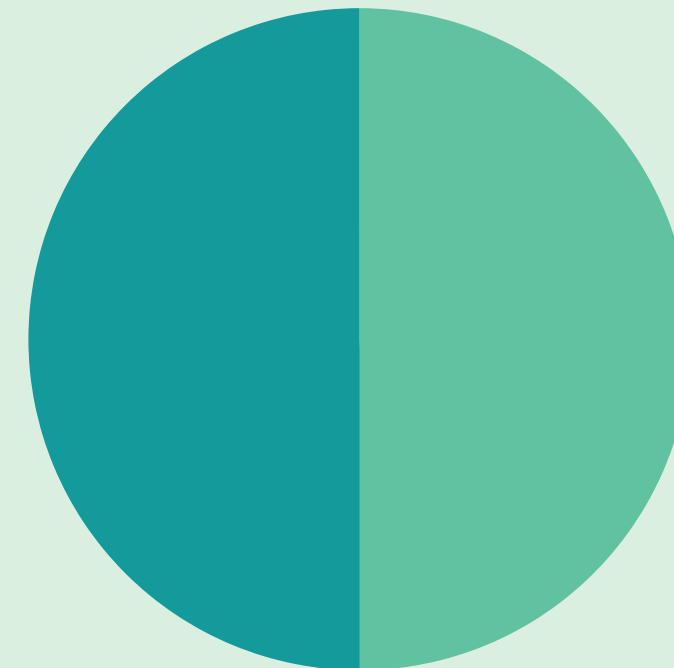**Open window on Mondays**

1ST HALF

- **Open features.**
- **Changing entities**

2ND HALF

- **Bug fixing**

BENEFITS OF USING CURL

- Highly versatile
- Beneficial for evaluating endpoint functionality
- Provides detailed output for the data sent and received
- Strong error logging capabilities

SECURITY ISSUES AND LAYERS

KEEP TRACK OF SECURITY PROBLEMS

When it was inserted, fixed and how

C-RELATED PROBLEMS

HTTP functionality also implemented in Rust.

TOO MANY LAYERS, APPS AND LANGUAGES

A lot of time spent searching for errors.

FURTHER INFORMATION (I)

REASONS ON WHY CURL IS DEVELOPED IN C

- In the nineties there was no other choice to make a portable library or tool than C
- C has made it possible to really make cURL and Libcurl deportable available everywhere

FURTHER INFORMATION (II)

HTTP has been the protocol which has taken them the most time tO implement

Apple included it in mac OS in 2001 becoming one of the first non-Linux OS that actually adopted cURL as a standard

It was confirmed to be used in the Mars helicopter mission in 2021