

What I did on my summer holiday

Daniel Silverstone
dsilvers@digital-scurf.org
IRC: Freenode//Kinnison

What I did on my summer holiday

Daniel Silverstone
dsilvers@digital-scurf.org
IRC: Freenode//Kinnison

...also...

Luiz Henrique de Figueiredo
lhf@tecgraf.puc-rio.br
IRC: Freenode//lhf

Lexing and parsing and how to perform evil.

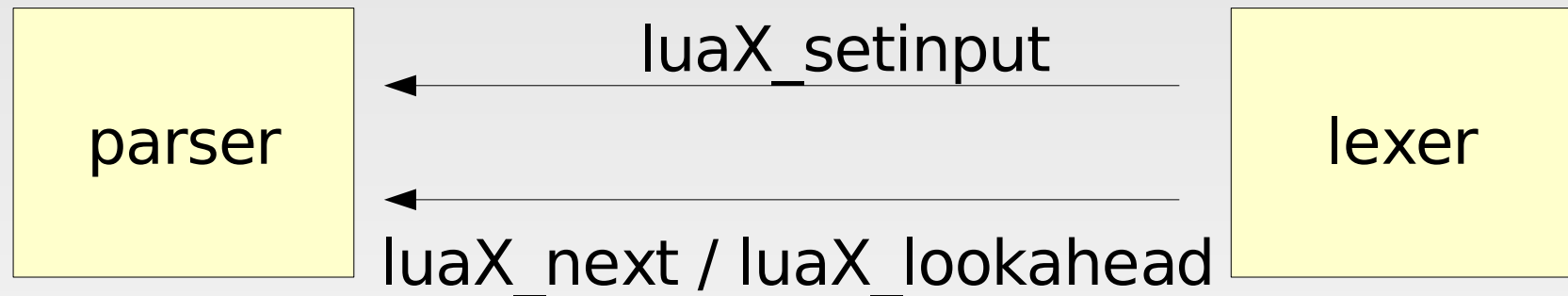
- What does a lexer do?
- What does a parser do?
- How can we subvert this with the minimum of C code?

The token pump

- Very small change to C
- Simple way to activate the pump and process tokens.
- Producer/Consumer model lends itself to non-trivial token stream manipulation very simply

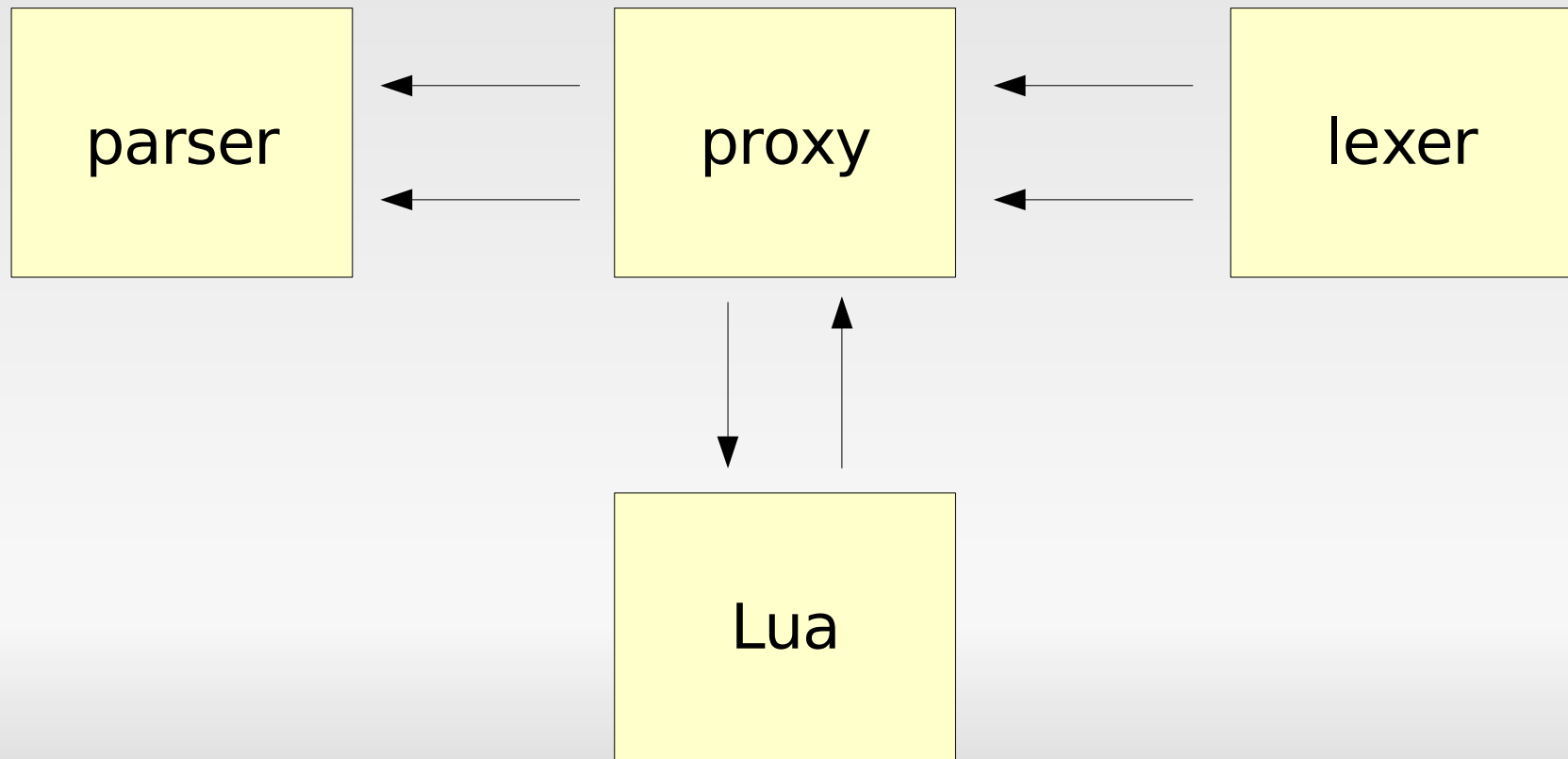
How the pump attaches

- Does not change Lua core
- Needs to access core objects and functions



How the pump attaches

- Does not change Lua core
- Needs to access core objects and functions



Okay, but what about more complex evil?

- We have lua, how much more complex do you want?
- Try to avoid reimplementing Lua's parser
- What about coroutines?

So, how about an example

Simple pump example ->

Pah, too easy, something
more clever please?

<- Macro example

Prove it's useful

Mutation “operators” are somewhat easier

John Belmonte's exception idea

Oh dear, you're going to hell

- Most probably
- But imagine how powerful an idea token filtering is.
- Also, Ihf is coming with me if I do, it's his fault I did this.

What comes next?

- There's no include directive
- There's no token pasting
- I did no overloaded macros, they should be trivial though
- There're plenty of more evil token filtering possibilities