Faster Shiny apps with profiling tools

Winston Chang

RStudio

Shiny Dev Conference, January 2016

How do I make my Shiny app faster?

Why is my Shiny app slow?

Why is my R code slow?

Manual "benchmarking"

```r
library(ggplot2) # For diamonds data

plot(price ~ carat, data = diamonds)

m <- lm(price ~ carat, data = diamonds)

abline(m, col = "red")
```

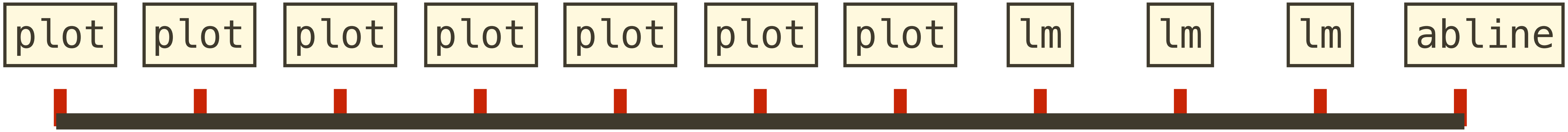
Challenges for manual benchmarking Shiny

- Difficult to run bits of code in isolation
- Giant stack traces
- Non-obvious code execution order

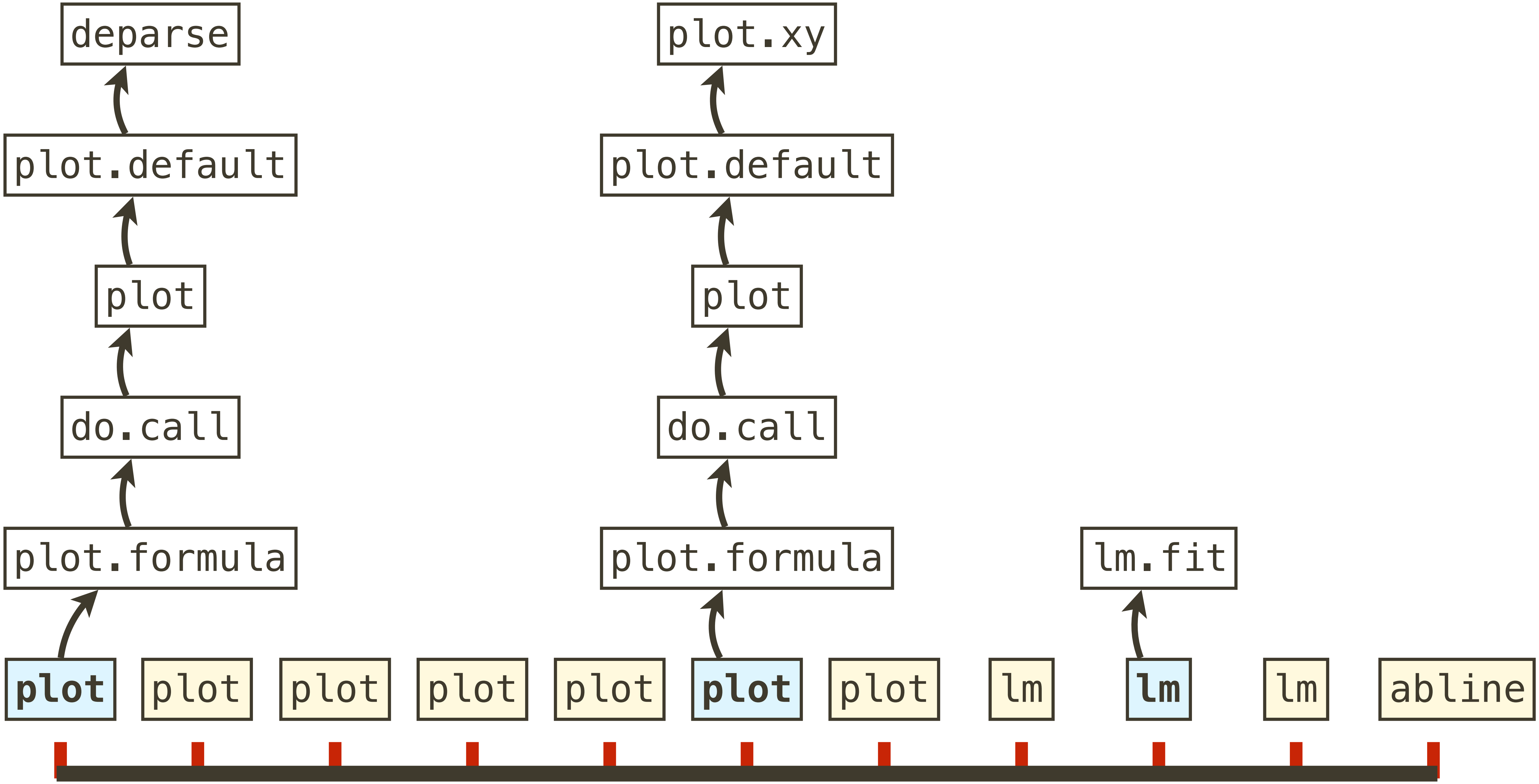
Profiling

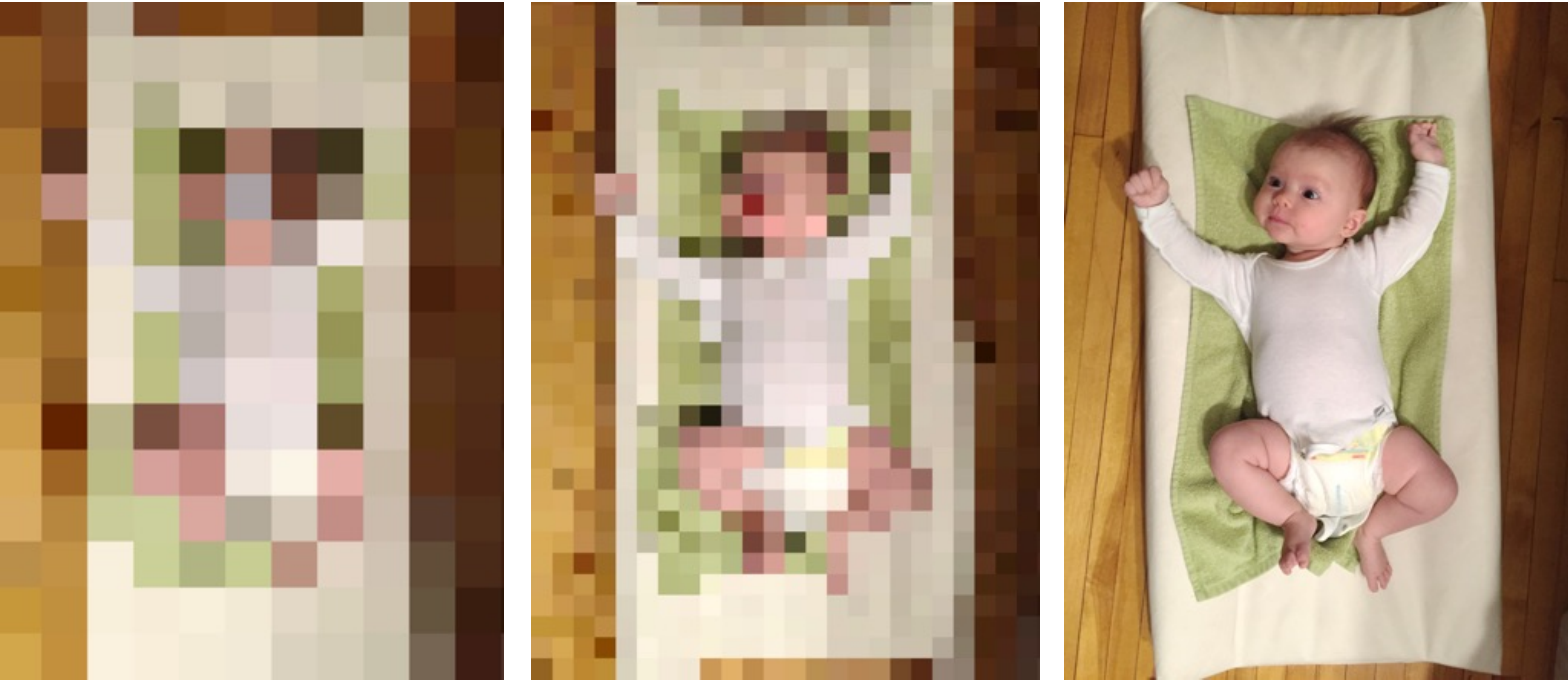
Sampling profiler

```
Rprof()       # Start profiling
plot(price ~ carat, data = diamonds)
m <- lm(price ~ carat, data = diamonds)
abline(m, col = "red")
Rprof(NULL)   # Stop profiling
```



Sampling profiler: call stack





profvis

```
devtools::install_github("rstudio/profvis")
```

```
library(profvis)

profvis({
  plot(price ~ carat, data = diamonds)
  m <- lm(price ~ carat, data = diamonds)
  abline(m, col = "red")
})
```

```r
# Generate data with 400000 rows and 150 cols
data <- as.data.frame(x = matrix(rnorm(4e5 * 150,
mean = 5), ncol = 150))

profvis({
  # Copy data
  d <- data
  # Get column means
  means <- apply(d, 2, mean)

  # Subtract mean from each column
  for (i in seq_along(means)) {
    d[, i] <- d[, i] - means[i]
  }
})
```

```r
profvis({
  d <- data
  # Four different ways of getting column means
  means <- apply(d, 2, mean)
  means <- colMeans(d)
  means <- lapply(d, mean)
  means <- vapply(d, mean, numeric(1))
})
```

```r
profvis({
  d <- data
  means <- vapply(d, mean, numeric(1))

  for (i in seq_along(means)) {
    d[, i] <- d[, i] - means[i]
  }
})
```

Profiling Shiny apps

Profiling Shiny apps: some challenges

- Difficult to run bits of code in isolation
- Giant stack traces
- Non-obvious code execution order

Limits of the R profiler

- The profiler doesn’t record time spent in system calls.
- All executed code must be within `profvis({})`, or sourced from file.

The future

- Closer integration with RStudio IDE
- More views of profiling data
- CRAN release

More information

https://github.com/rstudio/profvis

Intro: http://rpubs.com/wch/123888