

Automating our work away

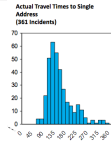
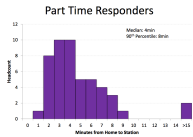
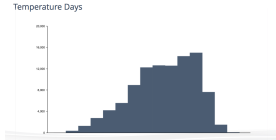
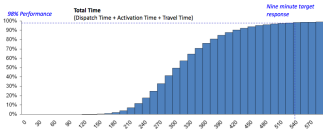
One consulting firm's experience with RMarkdown

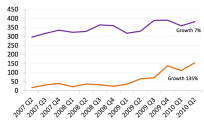
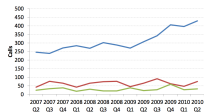
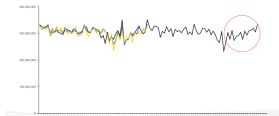
Finbarr Timbers

1. About Darkhorse Analytics
2. Problems we faced
3. How we solved those problems
4. How you can improve R(Markdown) adoption



DARKHORSE
ANALYTICS





Problems

1. It takes a lot of time to replicate, which is a problem when rework happens (>50 slides!). We've had 3 month projects with 1 month consumed by rework.

Problems

1. It takes a lot of time to replicate, which is a problem when rework happens (>50 slides!). We've had 3 month projects with 1 month consumed by rework.
2. All styled differently. We need to edit every plot individually.

Problems

1. It takes a lot of time to replicate, which is a problem when rework happens (>50 slides!). We've had 3 month projects with 1 month consumed by rework.
2. All styled differently. We need to edit every plot individually.
3. No documentation on how to replicate.

Instead? R!

What we did:

What we did:

1. Use RMarkdown to write template (“best practice”) slides.

What we did:

1. Use RMarkdown to write template (“best practice”) slides.
2. Made the slides accessible to the company (we used Git + Bitbucket).

What we did:

1. Use RMarkdown to write template (“best practice”) slides.
2. Made the slides accessible to the company (we used Git + Bitbucket).
3. Made it easy to run in other environments (we used installation scripts; moving to Docker)

What we did:

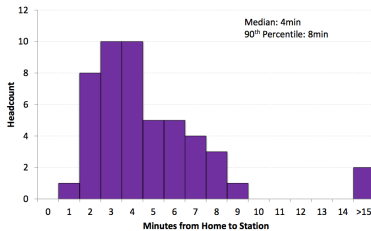
1. Use RMarkdown to write template (“best practice”) slides.
2. Made the slides accessible to the company (we used Git + Bitbucket).
3. Made it easy to run in other environments (we used installation scripts; moving to Docker)
4. Iterate (we used Git).

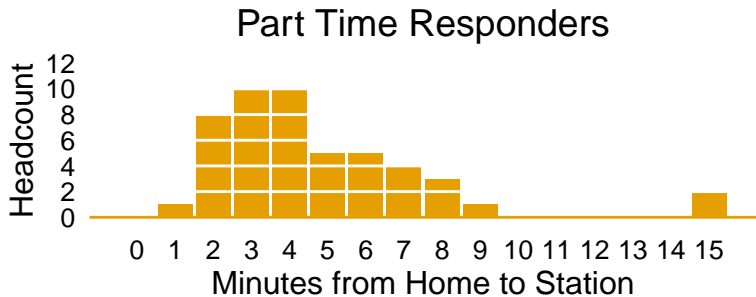
What we did:

1. Use RMarkdown to write template (“best practice”) slides.
2. Made the slides accessible to the company (we used Git + Bitbucket)
3. Made it easy to run in other environments (we used installation scripts; moving to Docker)
4. Iterate (we used Git).
5. Publish as a package (yet to happen).

```
darkhorse_histogram = function(df, x_var, x_breaks,
  x_label, y_var, y_breaks, y_label, title,
  col="#E69F00") {
  ggplot(df, aes_string(x = x_var, y = y_var)) +
    ggtitle(title) +
    geom_bar(fill=col, stat = "identity") +
    geom_hline(yintercept= 0, colour=col) +
    darkhorse_theme() +
    ylab(y_label) + xlab(x_label) + scale_y_continuous
    geom_hline(
      yintercept=y_breaks[2:length(y_breaks)],
      color="white") +
    scale_x_continuous(breaks=x_breaks)
}
```

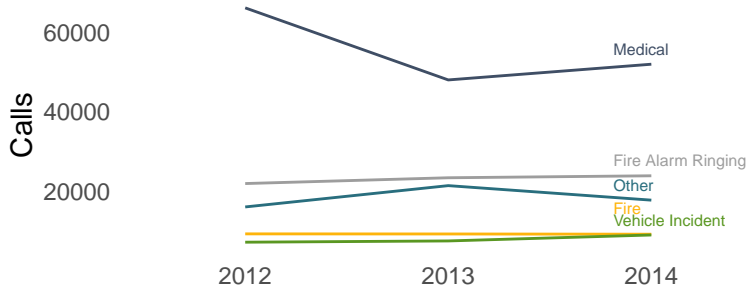
Part Time Responders





Also: SQL

```
SELECT
    *
FROM (SELECT
    CASE
        WHEN Call_Type NOT IN ('Medical', 'Fire', 'Fire Alarm P
        ELSE Call_Type
    END AS CallType,
    CallID,
    Year
FROM ERA_TFS.dbo.Calls_Template
WHERE Qualified_Event = 1) t
PIVOT (COUNT(CallID) FOR Year IN ([2012], [2013], [2014]))
```



1. **Make it great.**

1. **Make it great.**
2. **Make it easy.**

1. **Make it great.**
2. **Make it easy.**
3. **Make it profitable.**