Best of Both Worlds
Relational Databases and Statistics
Collect data

Load data

Filter, transform & aggregate data

Analyze & Plot

Publish paper
Collect data

Load data

Filter, transform & aggregate data

Analyze & Plot

Publish paper
data <- dbGetQuery(conn,"
  SELECT t1,COUNT(t1) AS ct FROM (  
    SELECT CAST(flux as integer) AS t1 FROM starships WHERE  
    ( (speed = 5) ) AND ( (class = 'NX') ) ) AS t  
  WHERE t1 > 0 GROUP BY t1 ORDER BY t1 LIMIT 100;
"
normalized <- data$ct/sum(data$ct)

...do we really want this?
Enter monet.frame

The virtual data object for R

```r
> data <- monet.frame(conn, "starships")
> nxw5 <- subset(data, class == "NX" & speed == 5)$flux
> t <- tabulate(nxw5, 100)
> normalized <- t/sum(t)
```

R-style data manipulation & aggregation
Meanwhile

Behind the scenes:

```r
> data <- monet.frame(conn,"starships")
SELECT * FROM starships;

> nxw5 <- subset(data,class=="NX" & speed==5)$flux
SELECT * FROM starships WHERE class = 'NX' AND speed = 5;
SELECT flux FROM starships WHERE class = 'NX' AND speed = 5;

> t <- tabulate(nxw5,100)
SELECT t1,COUNT(t1) AS ct FROM (SELECT CAST(flux as integer) AS t1 FROM starships WHERE class = 'NX' AND speed = 5) AS t WHERE t1 > 0 GROUP BY t1 ORDER BY t1 LIMIT 100;
```

Actually executed
Implementation

# R core
subset <- function(x, ...) UseMethod("subset")

# MonetDB.R
unique.monet.frame <- function (x, subset, ...) {
    # some code here
}

> nxw5 <- subset(data, class=="NX" & speed==5)$flux
> str(nxw5)
MonetDB-backed data.frame surrogate
1 column, 1799991 rows
Query: SELECT flux FROM starships
    WHERE ( ((class = 'NX') AND (speed = 5)) )
Columns: flux (numeric)
Optimization

- Result Set Structure Inference
  - Columns, Types
  - # Rows

- Process embedding
  - Run DB inside R process
Performance

~1GB CSV, ~70M Rows
Thank You!

Questions?

CRAN: MonetDB.R