Data Acquisition and Management

Data Cleaning Operations

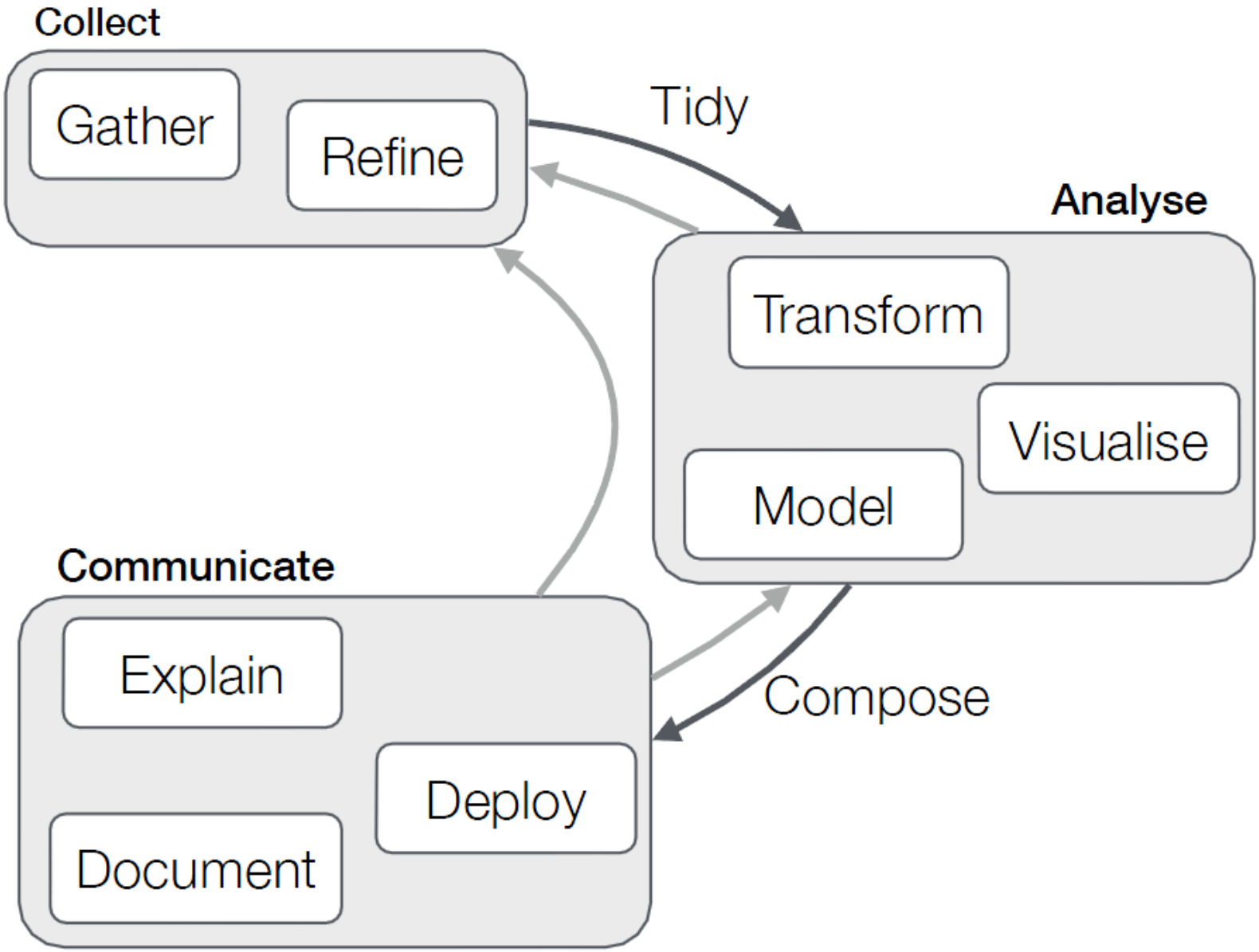
Shaping and Transforming Data

Endgame

What operations do we need to perform on our data to get it ready for use in visualization and modeling?

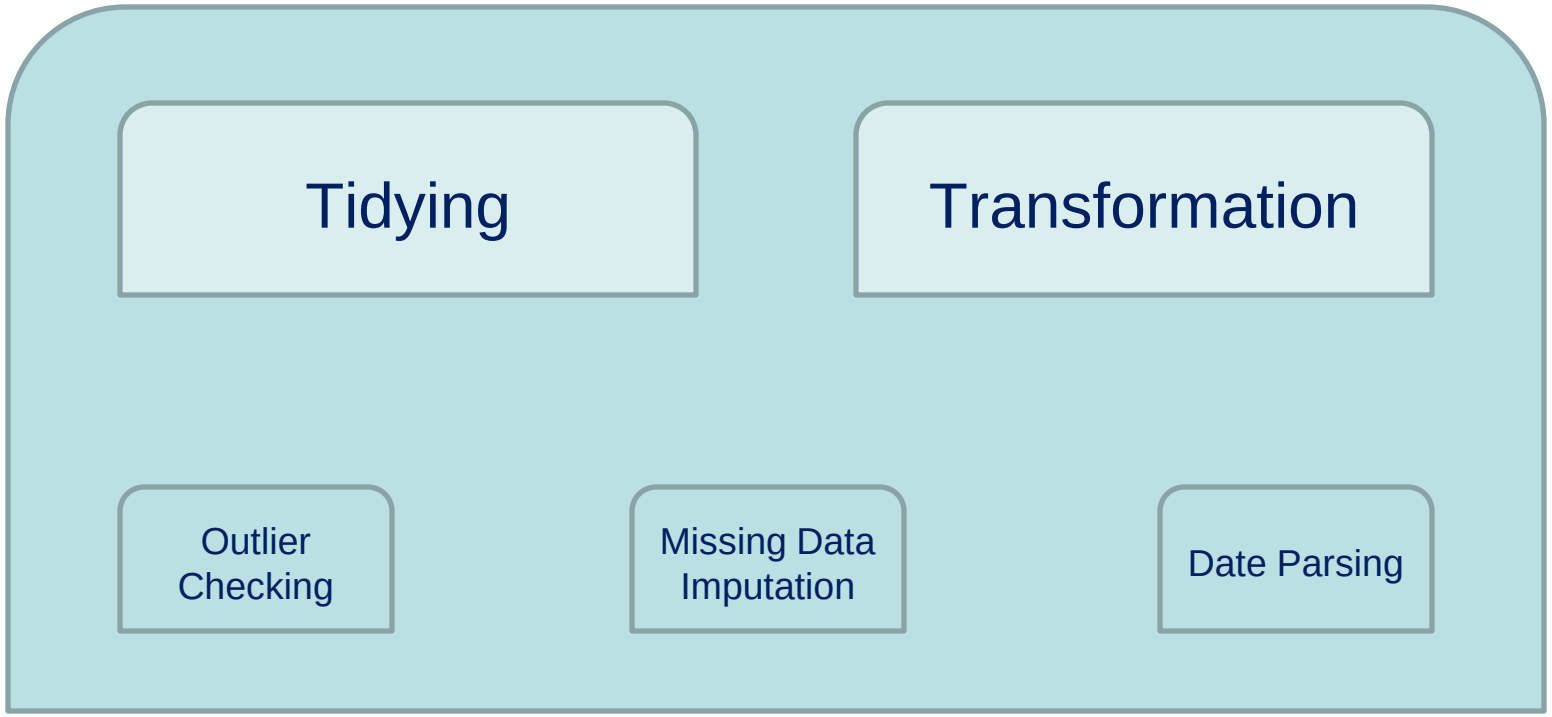
```
plot(Ozone ~ Wind, data = airquality)
```

```
lm( change ~ setting + effort )
```



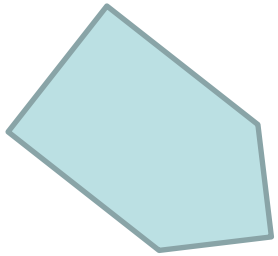
Source: "The Grammar and Graphics of Data Science," RStudio seminar series, Hadley Wickham, July 30, 2014.

Data Cleaning (“Scrubbing”)



Data Tidying (Shaping)

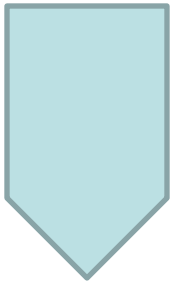
	Ozone	Solar.R	Wind	Temp	Month	Day
1	41	190	7.4	67	5	1
2	36	118	8.0	72	5	2
3	12	149	12.6	74	5	3
4	18	313	11.5	62	5	4
5	NA	NA	14.3	56	5	5
6	28	NA	14.9	66	5	6



	month	day	variable	value
1	5	1	ozone	41
2	5	2	ozone	36
3	5	3	ozone	12
4	5	4	ozone	18
5	5	5	ozone	NA
6	5	6	ozone	28

Data Transformation

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1



cyl	am	avgmpg	avgwt
4	0	22.90000	2.93500
4	1	28.07500	2.04225
6	1	20.56667	2.75500

Which tool(s) are best for tidying (shaping) and transforming our data?

tech stack: **R**, Python, **SQL**, Excel!?, Unix Command Line, many others

tidying: **"base R"** -> reshape -> reshape2 -> **tidyr** -> ?

transforming: **"base R"** -> plyr (or data.table) -> **dplyr** -> ?

Which tool(s) are best for tidying (shaping) and transforming our data?

This week we'll look at Hadley Wickham's `tidyr` and `dplyr` packages.

`tidyr` and `dplyr` packages both implement **data analysis pipelines**, that let us string multiple verbs together.

```
filter(
  summarise(
    select(
      group_by(hflights, Year, Month, DayofMonth),
      Year:DayofMonth, ArrDelay, DepDelay),
    arr = mean(ArrDelay, na.rm = TRUE),
    dep = mean(DepDelay, na.rm = TRUE)
    ),
  arr > 30 | dep > 30
)
```

dplyr code (could also write in a similar manner in base R)

```
hflights %>%
  group_by(Year, Month, DayofMonth) %>%
  select(Year:DayofMonth, ArrDelay, DepDelay) %>%
  summarise(
    arr = mean(ArrDelay, na.rm = TRUE),
    dep = mean(DepDelay, na.rm = TRUE)
  ) %>%
  filter(arr > 30 | dep > 30)
```

same functionality, implemented in a dplyr data analysis pipeline

While the data cleaning **tools** will continue to evolve quickly, the fundamental data cleaning **operations** needed change more slowly.

While you'll need to continually invest in learning best of class tools, you'll find that your expertise in data cleaning operations will grow over time, making you more effective with new data cleaning tool sets.

“The goal is to make it as easy as possible to get your vision of the analysis out of your head and into the computer!”

-- Hadley Wickham