Technomics
Better Decisions Faster

Don’t Just Use Your Data… Exploit It!

ICEAA Professional Development & Training Workshop

May 14 – 17, 2019

Presenter Overview

Adam H. James

Technomics, Inc.

Adam James is a Senior Analyst at Technomics, Inc., where he has supported a variety of cost and research studies throughout the DoD and DHS. His areas of focus have included advanced tool and database development and statistical modeling.

His 6 years of experience include supporting OSD CAPE's Cost Analysis Data Enterprise (CADE) solution as well as sharing lead authorship of the Joint Agency CER Development Handbook.

He holds an M.S. in Statistics as well as B.S. degrees in Mathematics and Statistics from Virginia Tech.

He received ICEAA's 2016 Technical Achievement Award.

Acknowledgements

PM-SBCT

The Army's Stryker Armored Combat Vehicle Program Office sponsored the research which served as inspiration for this presentation.

PM-SBCT provided the funding and CSDR data for the underlying analyses.

Jeff Cherwonik

Jeff Cherwonik is currently a Program Manager/Senior Cost Analyst at Technomics, Inc.

Jeff served as the project manager for this research, bringing 28 years of experience providing life cycle cost analysis services in applied cost estimating, cost research, and database development.

Brandon Bryant

Brandon Bryant is a Lead Analyst at Technomics, Inc., and is currently based out of Troy, MI.

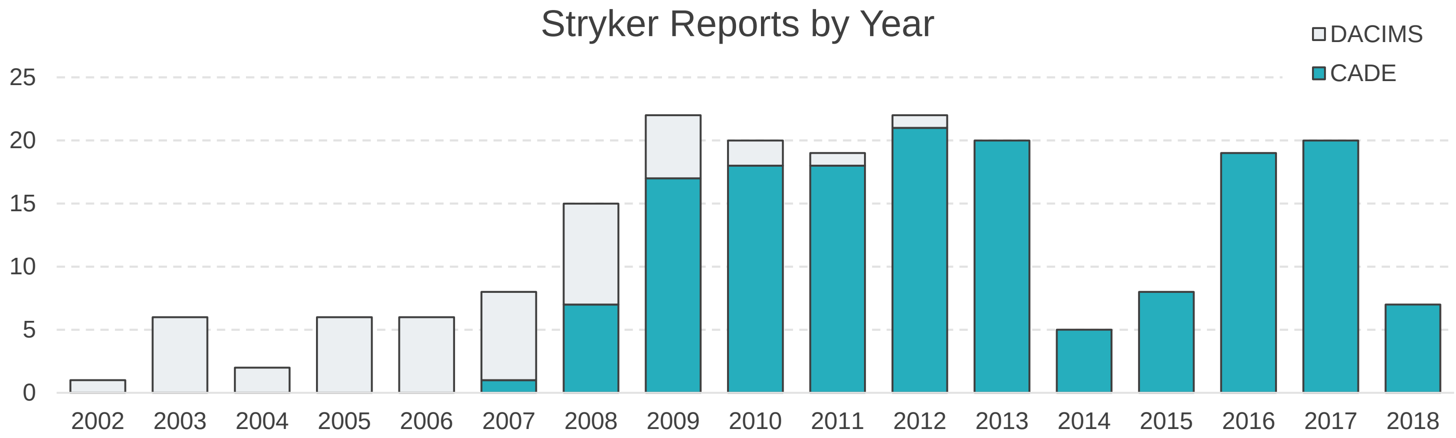
Brandon leveraged his experience implementing cost and software data reporting (CSDR) requirements to support this research.

Presentation Overview

- Introduction
- Working with Data
  - Storing and Structuring Data
  - Understanding a Grammar
- CCDR Data
- Stryker Example
  - Research Questions
  - Project Workflow
  - Analytical Tool
- Closing Thoughts

Introduction

- The Stryker Program Office (PM-SBCT) has a lot of CCDR data
- CCDR data are not utilized to their full potential
  - Too voluminous to handle in spreadsheets
  - Too diverse to quickly understand
  - New data is still being collected, amplifying the problem



207 Reports in the Database (over 50% of the entire surface vehicle commodity!)

How do we increase the utility of the data to help us make informed, data-driven decisions?

Working with Data

best practices, tips, and tricks

Rows and Columns

Rows represent unique data observations, **columns** represent variables

Bad

System	WBS Element	Cost ($)
Truck A		
	Engine Cost	50,000
	Remaining Cost	150,000
	PM Cost	1,500
	Number of Units	10
	Total Cost	2,015,000
Truck B		
	Engine Cost	40,000
	Remaining Cost	120,000
	PM Cost	2,500
	Number of Units	5
	Total Cost	812,500

Better

System	Metric	Element	Value
Truck A	Unit Cost	Engine	50000
Truck A	Unit Cost	Remaining	150000
Truck A	Unit Cost	PM	1500
Truck A	Unit Cost	Surface Vehicle	201500
Truck A	Quantity	Surface Vehicle	10
Truck B	Unit Cost	Engine	40000
Truck B	Unit Cost	Remaining	120000
Truck B	Unit Cost	PM	2500
Truck B	Unit Cost	Surface Vehicle	162500
Truck B	Quantity	Surface Vehicle	5

Single Purpose Variables

A column (or variable) should only contain **one** piece of information

Bad

- 1 Surface Vehicle System
  - 1.1 Variant A
    - 1.1.1 Surface Vehicle
      - 1.1.1.1 Engine
      - 1.1.1.2 Remaining Vehicle
  - 1.2 Variant B
    - 1.2.1 Surface Vehicle
      - 1.2.1.1 Engine
      - 1.2.1.2 Remaining Vehicle

Better

Original WBS	Modified WBS	Element	Model
1	1	Surface Vehicle System	
1.1.1	1.1	Surface Vehicle	Variant A
1.1.1.1	1.1.1	Engine	Variant A
1.1.1.2	1.1.2	Remaining Vehicle	Variant A
1.2.1	1.1	Surface Vehicle	Variant B
1.2.1.1	1.1.1	Engine	Variant B
1.2.1.2	1.1.2	Remaining Vehicle	Variant B

Other Data Tips (1 of 2)

Use **variable names** that any tool can use

Bad Name	Better Name
Work Breakdown Structure	WBS
% Complete	PercComp
Cost (TY $K)	CostTY_K
Unit Cost (FY18)	UnitCost_FY18
1970	Cost_1970

Be explicit in preserving **order**

Month
January
February
…
November
December

→

ID_Month	Month
1	January
2	February
…	…
11	November
12	December

Store data **values** and report data **formats**

Value	Formatted
100000	100,000
2018-10-05	October 5, 2018
2695.255648	$2,695.26

Take advantage of **sparse data** when possible

wbs	rec	non_rec
1.1	0	0
1.2	0	0
1.3	1000	0
1.4	2000	0
1.5	0	0
1.6	0	0

→

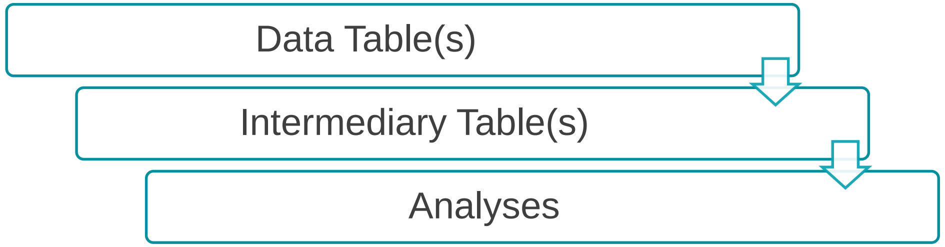
wbs	r_nr	value
1.3	rec	1000
1.4	non_rec	2000

Other Data Tips (2 of 2)

Avoid storing redundant information

- ✓ Only store child elements
- ✗ Do not store subtotals / totals
- ✗ Do not store calculated variables

Use intermediary tables for calculations

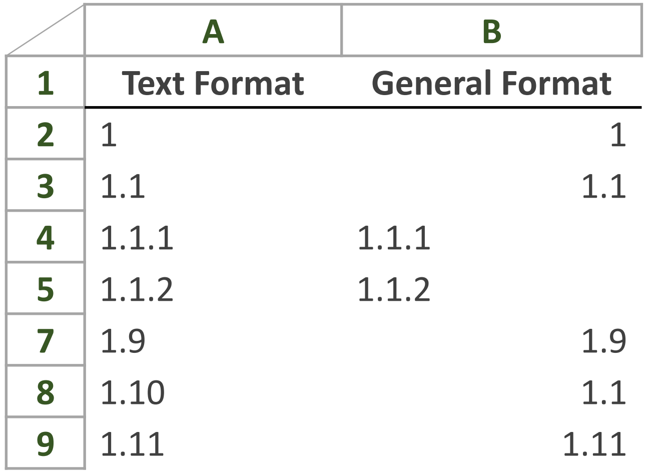

Data Table(s)

Intermediary Table(s)

Analyses

Be mindful of data types

- Numeric, date, and text
- Excel will make (sometimes wrong) assumptions



	A	B
1	Text Format	General Format
2	1	1
3	1.1	1.1
4	1.1.1	1.1.1
5	1.1.2	1.1.2
7	1.9	1.9
8	1.10	1.1
9	1.11	1.11

Grammar of Data Manipulations

“dplyr is a grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges” [1]

- A grammar is the rules of a language – or a set of instructions
  - Data forms a collection of **nouns**
  - A small set of **verbs** operate on the data
- *dplyr* is a package in R which proposes and implements the grammar
- Only 5 (+1) primary verbs create a powerful, flexible framework

The grammar outlines a thought process transferable to any tool (even Excel)!

[1] https://dplyr.tidyverse.org/

Primary Verbs[1]

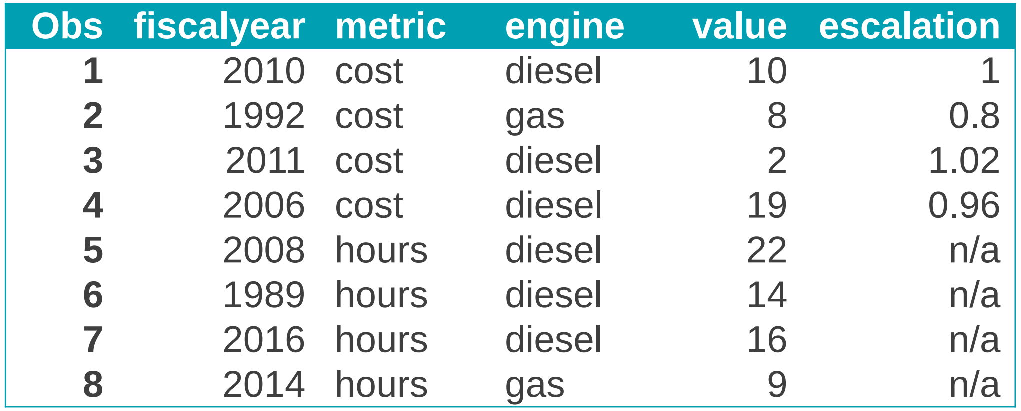
1) **mutate** – adds new variables that are functions of existing variables
2) **select** – picks variables based on their name
3) **filter** – picks cases based on their values
4) **summarize** – reduces multiple values down to a single summary
5) **arrange** – changes the ordering of rows

And the bonus verb **group_by** performs any operation “by group”

[1] https://dplyr.tidyverse.org/

A Simple Example

Goal: to view average dollars (applying escalation) and hours



Obs	fiscalyear	metric	engine	value	escalation
1	2010	cost	diesel	10	1
2	1992	cost	gas	8	0.8
3	2011	cost	diesel	2	1.02
4	2006	cost	diesel	19	0.96
5	2008	hours	diesel	22	n/a
6	1989	hours	diesel	14	n/a
7	2016	hours	diesel	16	n/a
8	2014	hours	gas	9	n/a

Steps:

1) Divide the dataset into cost and hours
2) For each of the cost values, adjust for escalation
3) Calculate the average for the escalation adjusted costs
4) Calculate the average for the hours

A Simple Example – Applying the Grammar

Goal: to view average dollars (applying escalation) and hours

Obs	fiscalyear	metric	engine	value	escalation
1	2010	cost	diesel	10	1
2	1992	cost	gas	8	0.8
3	2011	cost	diesel	2	1.02
4	2006	cost	diesel	19	0.96
5	2008	hours	diesel	22	n/a
6	1989	hours	diesel	14	n/a
7	2016	hours	diesel	16	n/a
8	2014	hours	gas	9	n/a

```
Vehicle data
  mutate value_adj =
    if (metric = cost) then (value / escalation)
    else value
  group_by metric
  summarize avg =
    mean of value_adj
```

CCDR Data

raw data and normalizations

Introduction

Working with Data

CCDR Data

Stryker Example

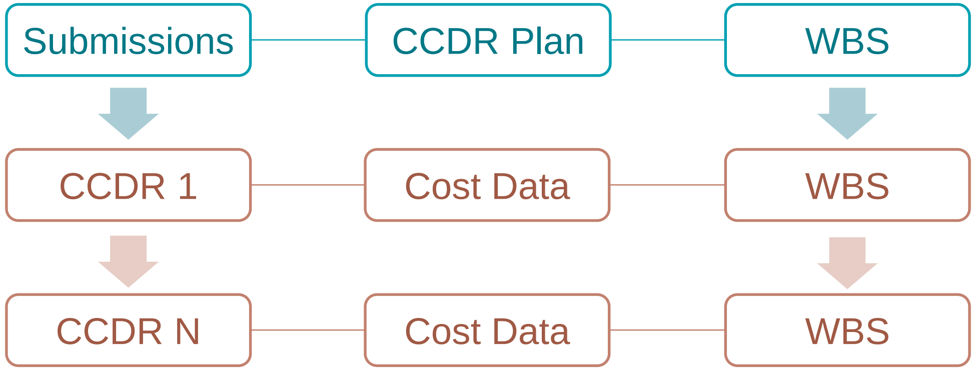
Closing Thoughts

Overview of CCDR Data

- The Contractor Cost Data Report (**CCDR**) collects cost (i.e., dollars & hours) actuals for major programs[1]
- Relates back to a “CSDR Plan”
  - Defines the work breakdown structure
  - Defines the submission frequency

CCDR Report Formats

Feature	1921	1921-1
report metadata (e.g., contractor, contract, dates)	✓	✓
dollars data	✓	✓
hours data		✓
work breakdown structure breakout	✓	✓
recurring/nonrecurring breakout	✓	✓
contract summary (i.e., G&A, UB, MR, FCCM, fee)	✓	
functional category breakout		✓



CSDR Plan Relationship

[1] https://cade.osd.mil/

Report Level Normalization Strategy

- Reports are tagged on a general strategy applicable to any program:
  - **Phase** (i.e., Development, Production, Sustainment)
  - **Sub-phase**
    - Name (e.g., FRP)
    - Start (e.g., 1)
    - End (e.g., 3)
    - Description (e.g., Full Rate Production (FRP) Lots 1 – 3)
  - **EndItem** (e.g., Ground Vehicle, Vehicle Kits)
  - **Process** (e.g., Manufacturing, Maintenance, Systems Engineering)
  - **Report Sequence**
    - ID (i.e., arbitrary key)
    - Description (e.g., Program Management)
    - Type (i.e., Cumulative, Discrete)
  - **Date Event** (e.g., Contract Award)

WBS Level Normalization Strategy

- Normalize to a standard WBS structure
- Map the CSDR Plan WBS (40 total plans) to the standard WBS
- Additional tags are assigned at the WBS level:
  - **Variant/Model** (e.g., ICV, NBCRV, MGS)
  - **Hull Design** (i.e., Flat Bottom, Double-V, ECP, or ICV-D)
  - **Kit Name**
  - **Funding Source** (e.g., RDTE, Procurement)
  - **Destination** (e.g., SBCT, Other Strykers)

Stryker Example

applying what we have learned

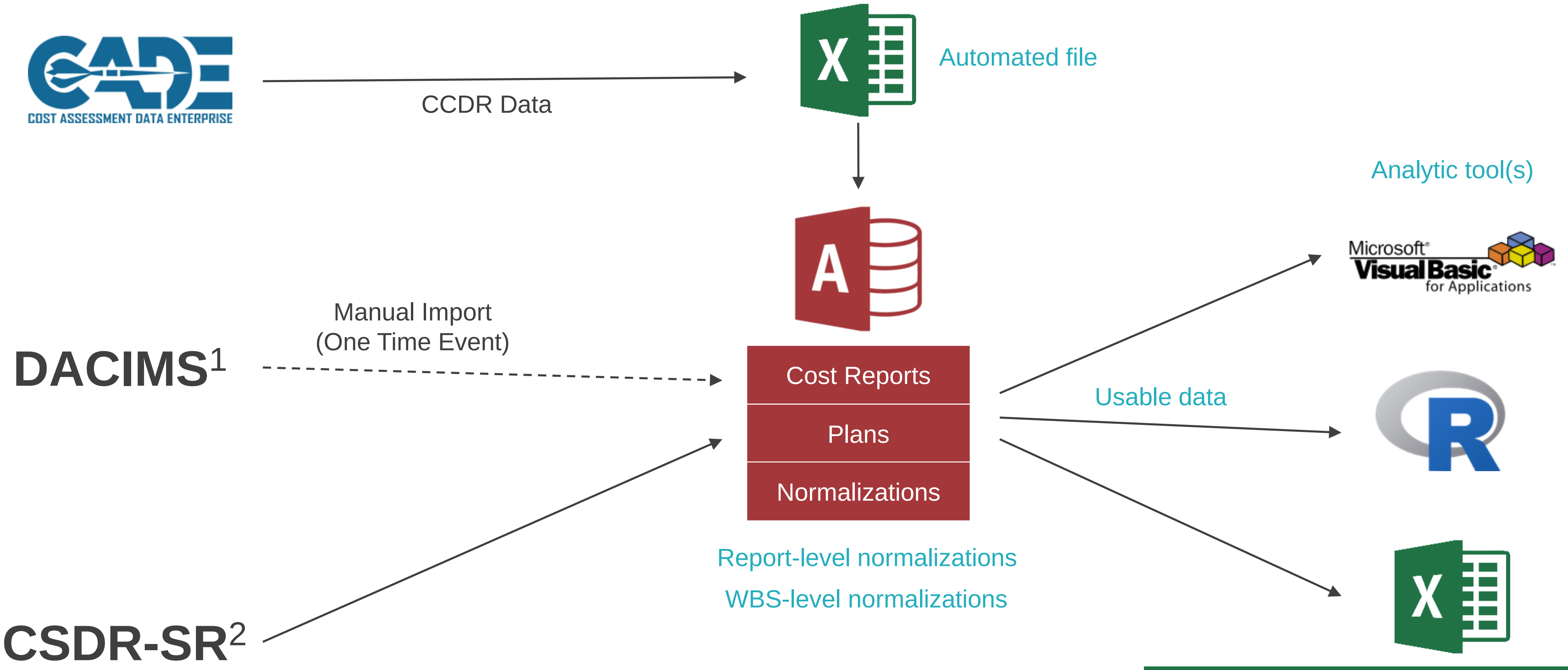
Introduction ○ Working with Data ○ CCDR Data ○ **Stryker Example** ● Closing Thoughts ○

Data Analysis Tool Introduction

- **Goal**: Create an Excel tool framework capable of answering a variety of research questions and problems

- Initial Research Questions:
  - Profit rate across delivery orders
  - Overhead rates (including trending over time)
  - Cost/hours growth by WBS from initial to final submissions
  - Unit cost/hours over time for hardware WBS elements
  - Level of effort over time for contractor support WBS elements

Workflow Overall View

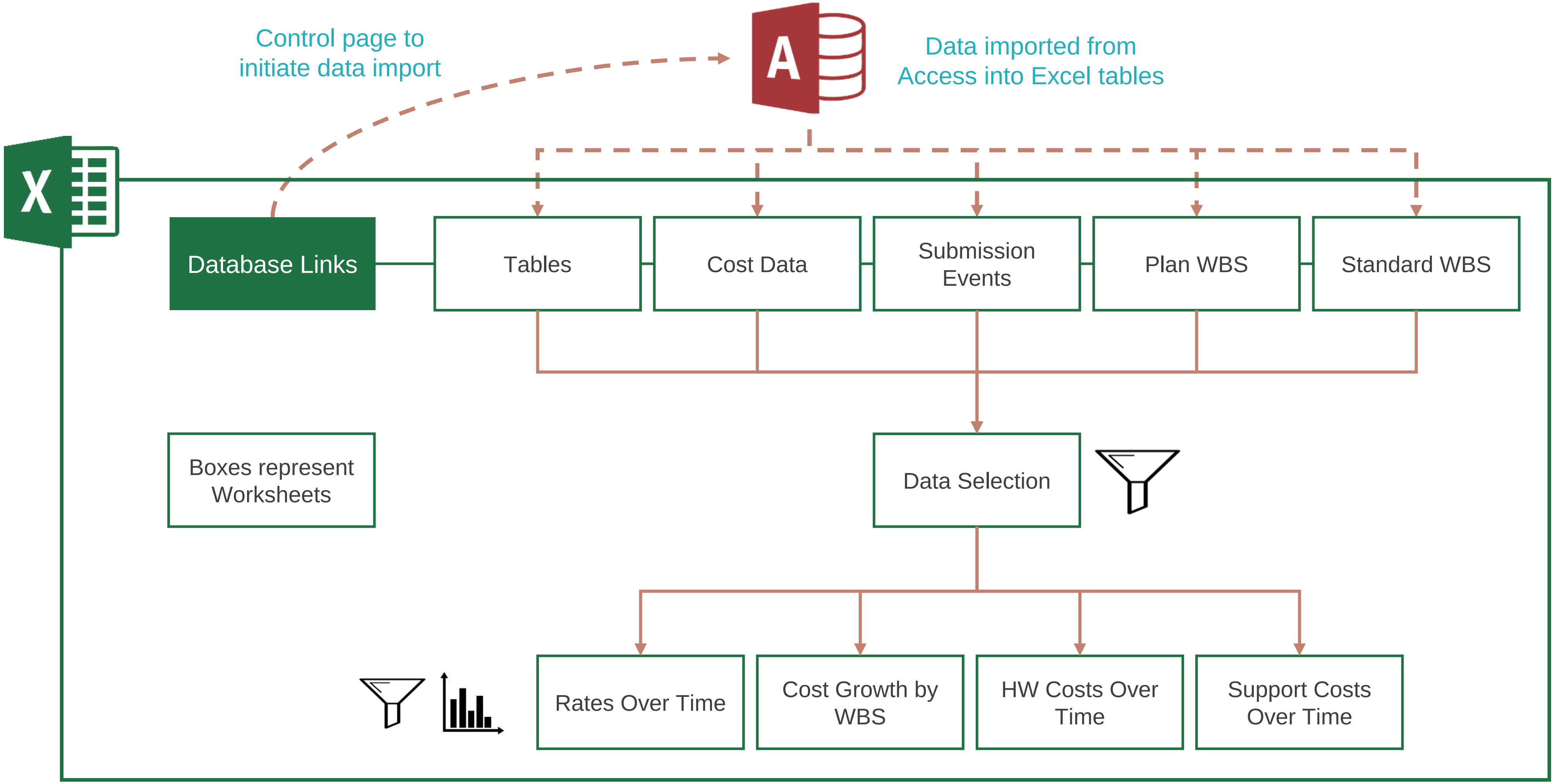


- Profit Rate
- Overhead Rates
- Cost/ Hours Growth
- Unit Cost/Hours
- Level of Effort by WBS
- Data Slices over Time
- Labor Rate Changes

[1] Defense Automated Cost Information Management System

[2] CSDR Submit/Review

Analytical Tool: Overview



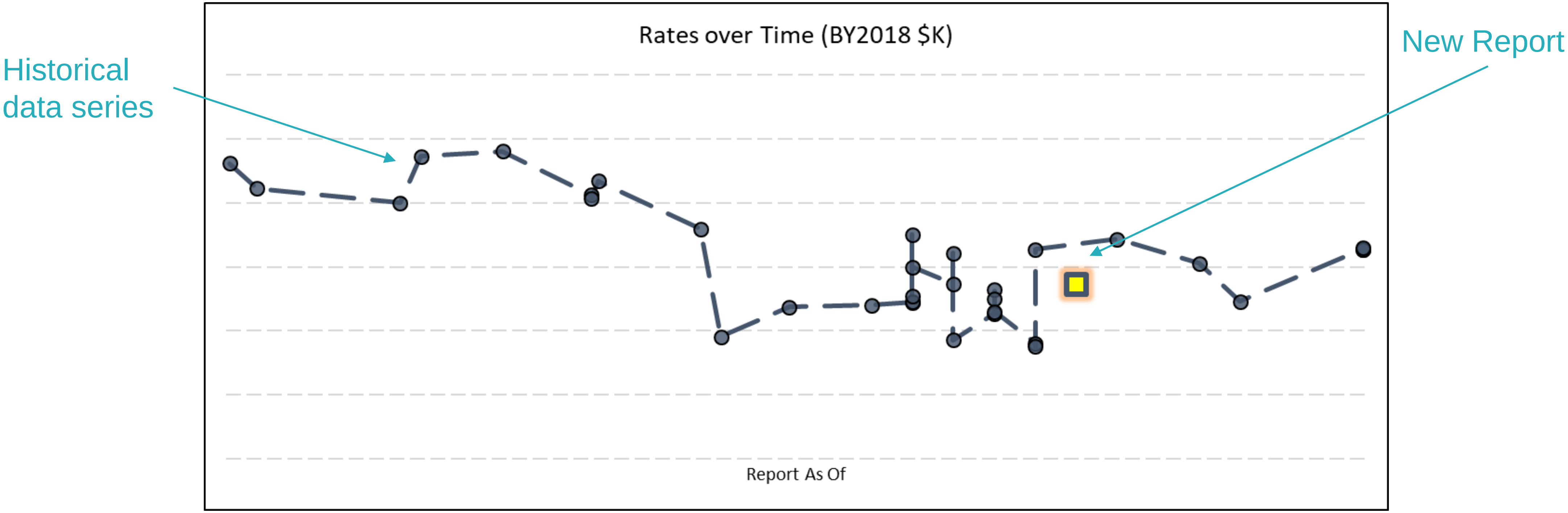
Control page to initiate data import

Data imported from Access into Excel tables

Boxes represent Worksheets

Data Benchmarking

- Import a “temporary” CCDR report
  - Automatically **map and tag** the report by reference back to the Plan
  - Inform validations of “new” data prior to acceptance by OSD CAPE Defense Cost and Resource Center (DCARC)



The Tool: Benchmark Report Rates

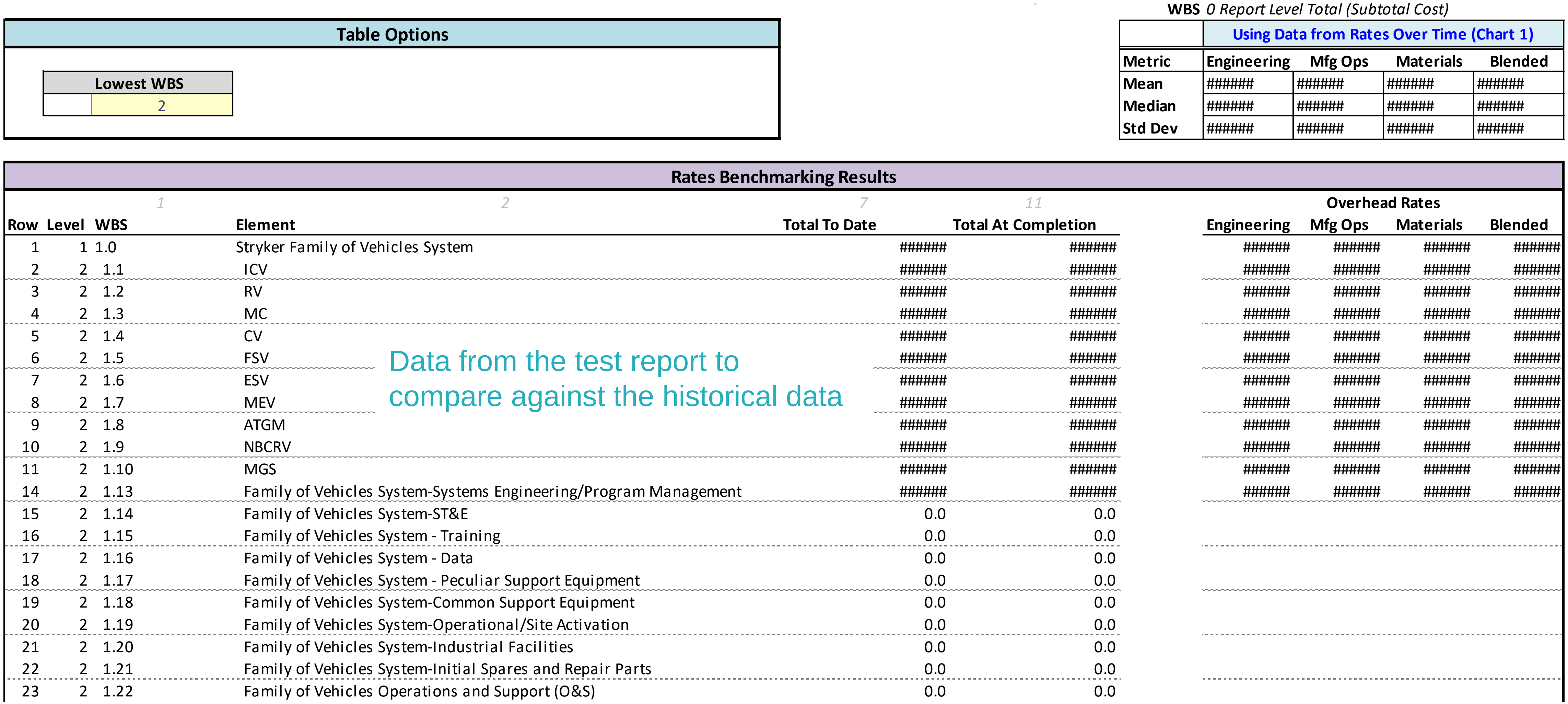


Table Options

Lowest WBS	
	2

WBS *0 Report Level Total (Subtotal Cost)*

	Using Data from Rates Over Time (Chart 1)			
Metric	Engineering	Mfg Ops	Materials	Blended
Mean	######	######	######	######
Median	######	######	######	######
Std Dev	######	######	######	######

Rates Benchmarking Results

		1	2	7	11	Overhead Rates			
Row	Level	WBS	Element	Total To Date	Total At Completion	Engineering	Mfg Ops	Materials	Blended
1	1	1.0	Stryker Family of Vehicles System	######	######	######	######	######	######
2	2	1.1	ICV	######	######	######	######	######	######
3	2	1.2	RV	######	######	######	######	######	######
4	2	1.3	MC	######	######	######	######	######	######
5	2	1.4	CV	######	######	######	######	######	######
6	2	1.5	FSV	######	######	######	######	######	######
7	2	1.6	ESV	######	######	######	######	######	######
8	2	1.7	MEV	######	######	######	######	######	######
9	2	1.8	ATGM	######	######	######	######	######	######
10	2	1.9	NBCRV	######	######	######	######	######	######
11	2	1.10	MGS	######	######	######	######	######	######
14	2	1.13	Family of Vehicles System-Systems Engineering/Program Management	######	######	######	######	######	######
15	2	1.14	Family of Vehicles System-ST&E	0.0	0.0				
16	2	1.15	Family of Vehicles System - Training	0.0	0.0				
17	2	1.16	Family of Vehicles System - Data	0.0	0.0				
18	2	1.17	Family of Vehicles System - Peculiar Support Equipment	0.0	0.0				
19	2	1.18	Family of Vehicles System-Common Support Equipment	0.0	0.0				
20	2	1.19	Family of Vehicles System-Operational/Site Activation	0.0	0.0				
21	2	1.20	Family of Vehicles System-Industrial Facilities	0.0	0.0				
22	2	1.21	Family of Vehicles System-Initial Spares and Repair Parts	0.0	0.0				
23	2	1.22	Family of Vehicles Operations and Support (O&S)	0.0	0.0				

Data from the test report to compare against the historical data

The Tool: Rates over Time

Research Questions:

- Profit rate across delivery orders
- Overhead rates (including trending over time)

```
filter(rows based on desired categories (e.g., WBS Element, Process))
filter(rows for relevant 1921-1 Functional Categories)
spread(Functional Categories into columns)
mutate(Rates = Dollars / Hours)
arrange(by Date)
```

The Tool: Cost Growth by WBS

Research Question:

- Cost/hours growth by WBS from initial to final submissions

```
filter(rows based on desired categories (e.g., WBS Element, Process))
filter(rows for Total Dollars At Completion)
group_by(Report Sequence and WBS Element)
mutate(Cost Growth = Total Dollars - lag(Total Dollars))
arrange(by Date)
```

The Tool: HW Costs Over Time

Research Question:

- Unit cost/hours over time for hardware WBS elements

```
filter(rows based on desired categories (e.g., WBS Element, Process))
filter(rows for Total Dollars and Units At Completion)
mutate(Unit Cost = Total Dollars / Units at Completion)
group_by(WBS Element and Variant)
arrange(by Date)
```

The Tool: Support Costs Over Time

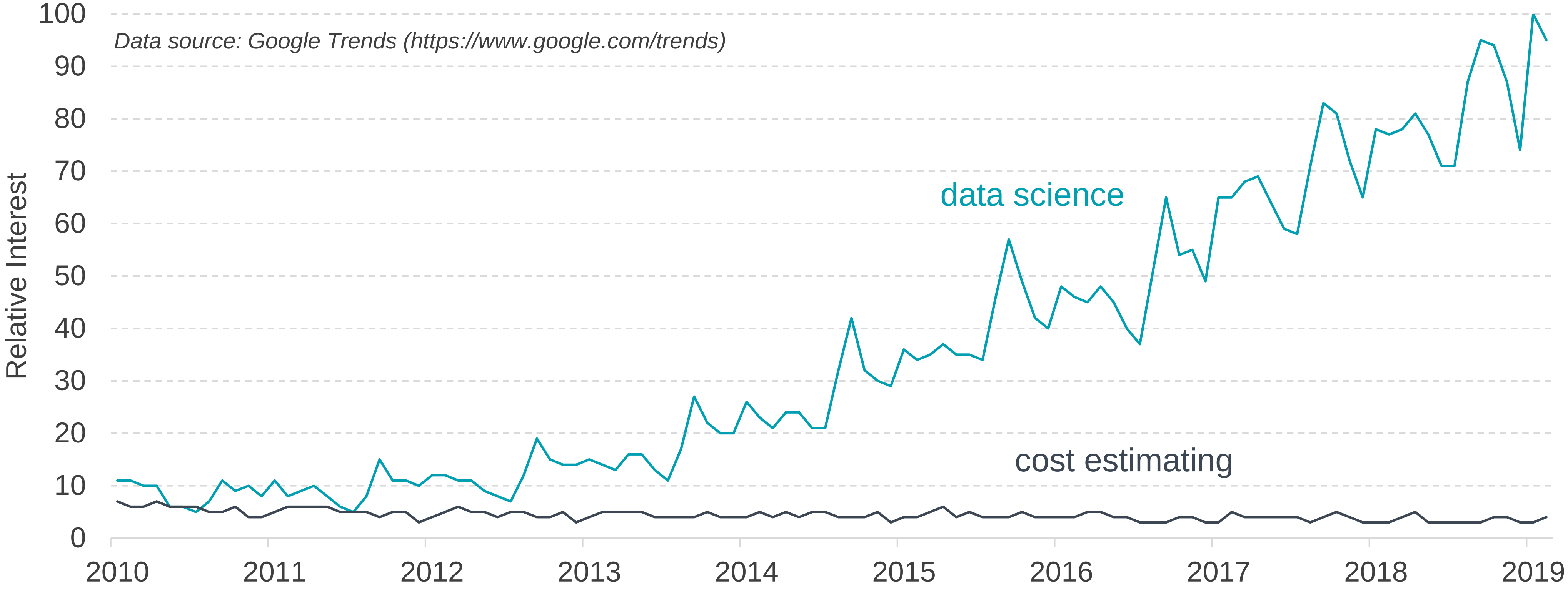
Research Question:

- Level of effort over time for contractor support WBS elements

```
filter(rows based on desired categories (e.g., WBS Element, Process))
filter(rows for Total Dollars At Completion)
group_by(WBS Element and Process)
arrange(by Date)
```

Closing Thoughts

“Until the community as a whole commits to better data management, the true power and value of data science will never be realized.”



Data source: Google Trends (https://www.google.com/trends)