

STANDARDIZING LABELS FOR THE OKLAHOMA DEPARTMENT OF TRANSPORTATION 8 YEAR WORKPLAN MAPS



Sam Coldiron, Paige Florie, & Nathan Smith

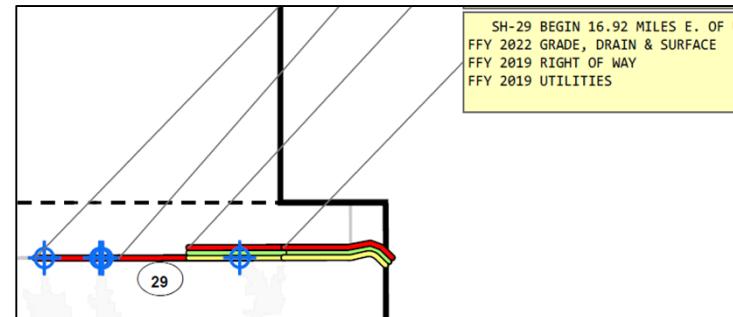
8-Year Workplan Maps



- The Performance Management Department is required to submit a Construction Workplan each year detailing plans for constructions for the next 8 fiscal years.
- The Workplan includes maps for each of the 8 divisions showing all the planned projects and labeled with project and financial information
- Previously these maps were created in GIS but the labels were manually created and edited in Visio.

Requirements

- 36x24 and 11x17 maps
- Projects following the same alignment must be offset
- Some projects will be grouped into a single line based on information provided by the Division Project Managers
- Label must be formatted with title line centered, and projects listed in tabular format and the label boxes must all be of equal width



Label Issues

US-277: BEG 1.45 MIS E. OF CADDO C/L AT E. SIDE OF MIDDLE BILL'S CREEK,
EXT E. 2.7 MIS ON OFFSET ALIGNMENT TO H.E. BAILEY T.P. OVERPASS BR.
FFY 2019 GRADE,DRAINING,BRIDGE & SURFACE \$ 7,692,473

- ESRI labels clips text if there are over 80 characters on a single line. Since the title was in a single field we needed to split the text
- ESRI fonts default to kerning
- Most fonts are proportional instead of monospaced
- Consolas is a monospaced font



LAV LAV

11111111
999999999
111111111
999999999

More Label Issues

US 281:OVER AN UNNAMED CREEK	
FFY 2024 BRIDGE & APPROACHES	\$ 5,667,200
FFY 2020 RIGHT OF WAY	\$ 617,725
FFY 2020 UTILITIES	\$ 926,587
	\$ 7,211,512

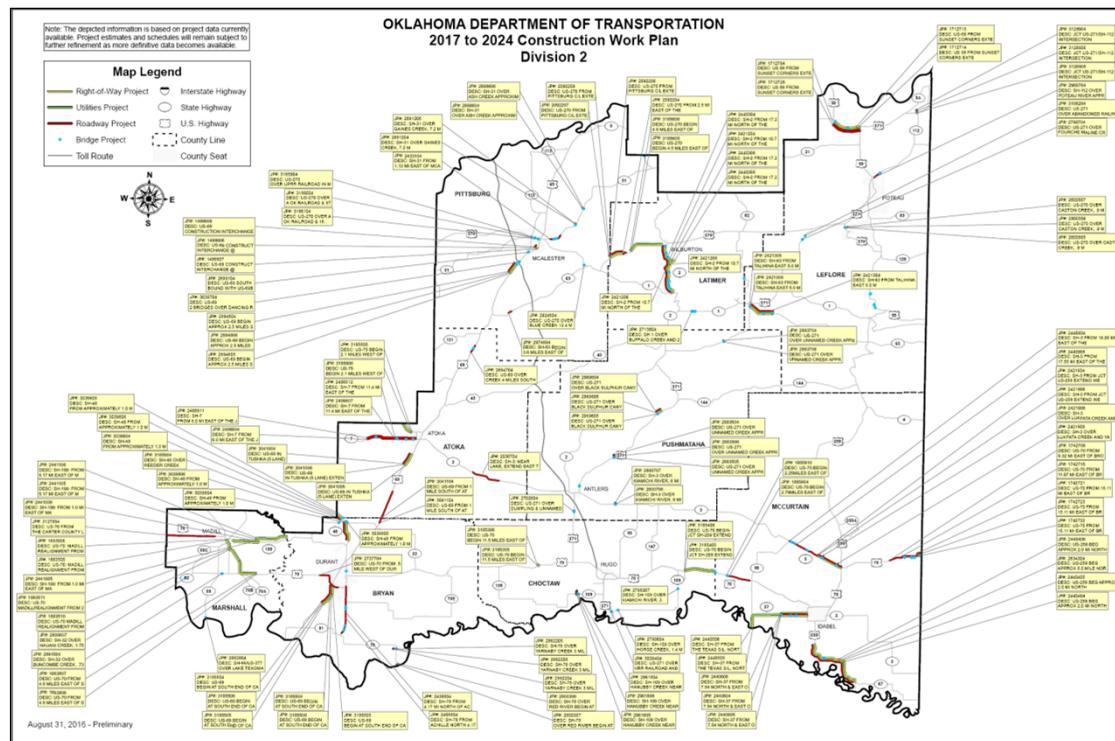
- Standard ESRI labels do not allow you to change font alignments on separate lines
- Total cost needed to be bold if there are more than one project in the label

What we tried...

- Standard ESRI label expression-edited manually.
 - Very slow and need to manually count characters to ensure all labels are the same width and text is aligned
 - Clipped fields with more than 80 characters
- Maplex Label Engine
 - Too slow for 150+ labels
 - Took several hours to generate labels for each division
- Python coding with annotation layer in a geodatabase
 - Must manually make cost total text bold
 - Placement of labels was manual

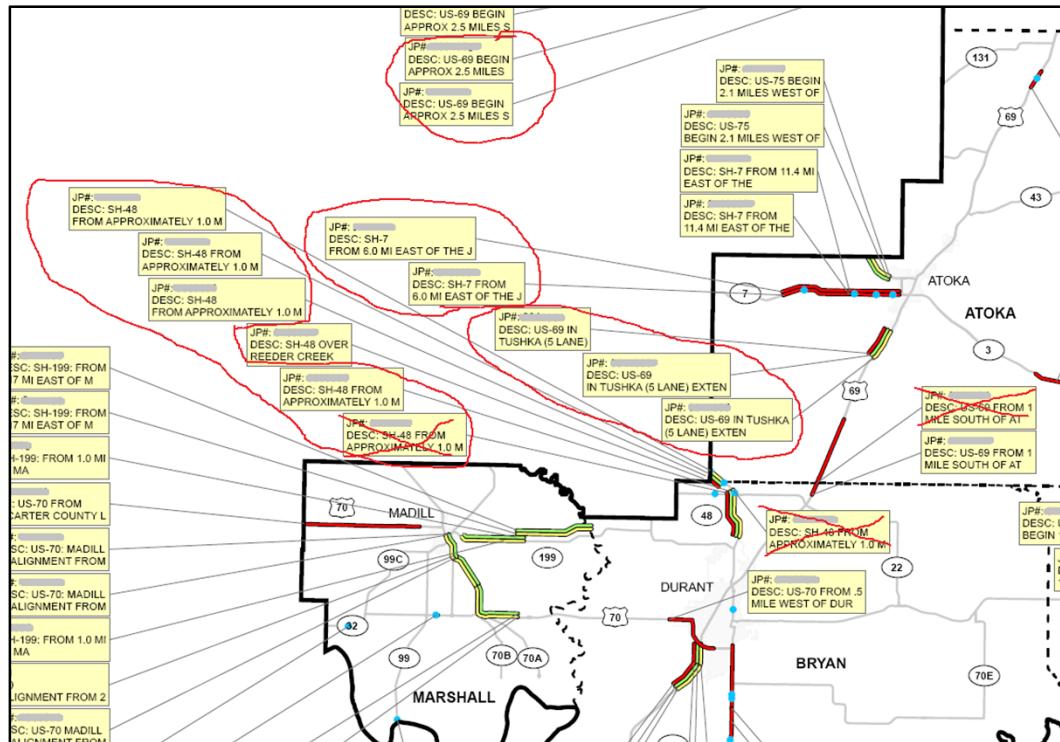
Rough Draft Maps

- Series of rough draft maps with brief labelling
 - Brief descriptions & JP numbers to identify projects
 - Some maps with over 200 labels at this stage



Grouping Projects

- ❑ Projects grouped by hand using rough drafts.
 - ❑ Project groups scattered across the map
 - ❑ Duplicates due to various line breaks in feature class



Building the Data Table

- Organized label detail in a table that could join to FC
 - Description cleanup, minor formatting, totaling project costs
 - Formulas helped retrieve and format text from original attributes

1	A	B	C	D	E	F	G	H	I	J
	OID	GROUP_ID	MA	USE_LBL	DESC	NUM_PROJ	JP_1	JP_1_YEAR	JP_1_TYPE	JP_1_AMT
569	31020	4-47570	4		SH 4: FR SH-66 IN YUKON NORTH TO WILSHIRE, 5 LANE FR S #####			2022	GRADE,DRAINING,BRIDG	7,000,000
570	31021	4-47570	4		SH-4 OVER THE N. CANADIAN RV, & 2 O'FLOWS 1.92, 2.13 & #####			2019	BRIDGE & APPROACHES	11,220,000
571	31022	4-47570	4		SH-4 BEG AT SH-66, IN YUKON, AND EXTENDING NORTH 3.C #####			2019	UTILITIES	1,530,000
572	33487	4-47570	4		SH-4 OVER THE N. CANADIAN RV, & 2 O'FLOWS 1.92, 2.13 & #####			2017	UTILITIES	2,000,000
573	31023	4-47580	4		SH 4 BEG 3.0 MIS. N. OF SH 66 IN YUKON @ WILSHIRE & EXT #####			2022	GRADE, DRAIN & SURFA	11,000,000
574	31024	4-47580	4		SH-4 FR. 3.0 MIS N. OF SH-66 IN YUKON @ WILSHIRE & EXT #####			2019	UTILITIES	663,000
575	31027	4-90320	4		I-35: OVER THE I-240 JCT. RECONSTR INTERCHANGE (PHAS #####			2020	INTERCHANGE	15,965,000
576	31028	4-90320	4		I-35: OVER THE I-240 JCT. (PHASE II) RECONST INTERCHG #####			2021	INTERCHANGE	24,000,000
577	31029	4-90320	4		I-35 @ THE I-240 JCT. (PHASE III) RECONST INTERCHG. #####			2021	INTERCHANGE	16,000,000
578	31030	4-90320	4		I-35 @ THE I-240 JCT. (PHASE IV)RECONST INTERCHG #####			2021	INTERCHANGE	31,000,000
579	31031	4-90321	4		I-35: OVER THE I-240 JCT. (RECONST INTERCHANGE) UT FC #####			2018	UTILITIES	6,060,000
580	31032	4-90321	4		I-35: OVER THE I-240 JCT PHASE 1,2,3, & 4 (RECONSTRUCT II #####			2017	RIGHT OF WAY	5,750,000
581	31033	4-90321	4		I-240: EB TO SB I-35 AT CROSSROADS INTERCHANGE (PHAS #####			2017	GRADE, DRAIN & SURFA	15,000,000
582	31034	4-90331	4		I-235: NB TO WB & EB TO NB FLYOVER BRIDGES I-235/I-44 II #####			2019	GRADE,DRAINING,BRIDG	30,600,000
583	31036	4-90332	4		I-235: MAINLINE THRU I-44 INTERCHANGE (SEGMENT 8) #####			2019	GRADE,DRAINING,BRIDG	40,800,000
584	31035	4-90332	4		I-235: MAINLINE THRU I-44 INTERCHANGE (SEGMENT 8) #####			2019	GRADE,DRAINING,BRIDG	40,800,000
585	31037	4-90332	4		I-44: WESTBOUND TO NORTHBOUND RAMPS AT I-44/I-235 #####			2020	GRADE,DRAINING,BRIDG	24,000,000
586	31070	4-14964	4		SH-74 FROM APPROX. 0.5 MILE NORTH OF NW 164TH ST. N #####			2018	GRADE, DRAIN & SURFA	24,240,000
587	31093	4-17428	4		I-40 CROSSTOWN: BNSF RIVERSIDE CONNECTION TRACKS, #####			2018	RAILROAD REHABILITATI	8,585,000
588	31094	4-17428	4		WP 5.1B: OKCY XTWN FROM WESTERN TO THE WEST SIDE C #####			2018	GRADE,DRAINING,BRIDG	15,655,000
589	31095	4-17428	4		WP 5.1C: OKCY XTWN FROM WEST OF WALKER TO THE WE #####			2017	GRADE, DRAIN & SURFA	6,000,000
590	31154	4-20324	4		I-40: FR MILE MARKER 165 TO 167, LN ADDS & RECONSTRU #####			2017	RECONSTRUCT-ADDED L	43,500,000
591	31155	4-20330	4		I-35: FROM FORT SMITH JCT, EXTEND NORTH TO I-44 #####			2023	RECONSTRUCT-ADDED L	12,000,000
592	31156	4-20331	4		SH-33: FROM 1.0 MILE EAST OF THE LOGAN COUNTY LINE A #####			2023	GRADE,DRAINING,BRIDG	15,800,000
593	31239	4-23310	4		I-40: W.B. AND E.B. BRIDGES OVER CRUTCHO CREEK & SE 15 #####			2019	BRIDGE & APPROACHES	31,620,000
594	31777	4-24163	4		SH-11 FROM 13.0 MI EAST OF THE AI FA FA C/I FAST TO IIS #####			2017	SHOULD'R IMPROVEME	12,000,000

Organizing Label Text

- Sample formulas that helped us out
 - Totaling & formatting cost:

=TEXT((SUM(INT(\$J5),INT(\$N5),INT(\$R5),INT(\$V5),INT(\$Z5),INT(\$AD5),INT(\$AH5))), "###,###,###,###")

- ## □ Extracting text from attribute:

=MID(Workplan_Table!\$AQ569,4,200)

- #### □ Lookup project detail used in labeling:

=VLOOKUP(\$K2, Workplan Table!\$E2:\$AY1578, 23, FALSE)

Creating Label Classes

- Groups project groups with the same number of projects
 - ▣ i.e. Groups with 2 projects go in the “2 Projects” group
- SQL Expression uses “NUM_PROJ” field
- Font and Formatting uniform across all classes
- Label Expression tailored to number of projects in the group

Font and Formatting

US 281:OVER AN UNNAMED CREEK	
FFY 2024 BRIDGE & APPROACHES	\$ 5,667,200
FFY 2020 RIGHT OF WAY	\$ 617,725
FFY 2020 UTILITIES	\$ 926,587
	\$ 7,211,512

- Font: Consolas
- Size: 10 Point
- Horizontal Alignment: Center
- Kerning: OFF
- Text Background: Line Callout

string.ljust(s, width[, fillchar])

- String.py library
- Left or Right justify a string
- Pad with *fillchar* (default space character)
- Returns the string “at least” *width* characters wide
 - Never truncates original string

Label Expression

```
def FindLabel ( RELAVENT INPUTS ):  
    groupDesc = [Grouping_Table$.DESC]  
    jobLine1 = "JP# " + str([Grouping_Table$.JP_1]) + " FFY " + str([Grouping_Table$.JP_1_YEAR]) + " " + [Grouping_Table$.JP_1_TYPE]  
    jobLine1 = jobLine1.ljust(68)  
    jobLine1Amt = [Grouping_Table$.JP_1_AMT].rjust(11)  
    jobLine2 = "JP# " + str([Grouping_Table$.JP_2]) + " FFY " + str([Grouping_Table$.JP_2_YEAR]) + " " + [Grouping_Table$.JP_2_TYPE]  
    jobLine2 = jobLine2.ljust(68)  
    jobLine2Amt = [Grouping_Table$.JP_2_AMT].rjust(11)  
    jobLine3 = "JP# " + str([Grouping_Table$.JP_3]) + " FFY " + str([Grouping_Table$.JP_3_YEAR]) + " " + [Grouping_Table$.JP_3_TYPE]  
    jobLine3 = jobLine3.ljust(68)  
    jobLine3Amt = [Grouping_Table$.JP_3_AMT].rjust(11)  
    groupTotal = "$" + [Grouping_Table$.AMT_TXT].rjust(11)  
    groupTotal = groupTotal.rjust(80)  
    if len(groupDesc) > 80:  
        splitChar = groupDesc.rfind(" ",0,70)  
        groupDescL1 = groupDesc[:splitChar]  
        groupDescL2 = groupDesc[splitChar + 1:]  
        groupDesc = groupDescL1 + '\n' + groupDescL2  
    return groupDesc + "\n" + jobLine1 + "$" + jobLine1Amt + "\n" + jobLine2 + "$" + jobLine2Amt + "\n" + jobLine3 + "$" + jobLine3Amt + "\n" + groupTotal
```

Label Expression

```
def FindLabel ( RELAVENT INPUTS ):  
    groupDesc = [Grouping_Table$.DESC]  
  
    jobLine1 = "JP# " + str([Grouping_Table$.JP_1]) + " FFY " + str([Grouping_Table$.JP_1_YEAR]) + " " + [Grouping_Table$.JP_1_TYPE]  
    jobLine1 = jobLine1.ljust(68)  
    jobLine1Amt = [Grouping_Table$.JP_1_AMT].rjust(11)  
  
    jobLine2 = "JP# " + str([Grouping_Table$.JP_2]) + " FFY " + str([Grouping_Table$.JP_2_YEAR]) + " " + [Grouping_Table$.JP_2_TYPE]  
    jobLine2 = jobLine2.ljust(68)  
    jobLine2Amt = [Grouping_Table$.JP_2_AMT].rjust(11)  
  
    jobLine3 = "JP# " + str([Grouping_Table$.JP_3]) + " FFY " + str([Grouping_Table$.JP_3_YEAR]) + " " + [Grouping_Table$.JP_3_TYPE]  
    jobLine3 = jobLine3.ljust(68)  
    jobLine3Amt = [Grouping_Table$.JP_3_AMT].rjust(11)  
  
    groupTotal = "$" + [Grouping_Table$.AMT_TXT].rjust(11)  
    groupTotal = groupTotal.rjust(80)  
  
    if len(groupDesc) > 80:  
        splitChar = groupDesc.rfind(" ",0,70)  
        groupDescL1 = groupDesc[:splitChar]  
        groupDescL2 = groupDesc[splitChar + 1:]  
        groupDesc = groupDescL1 + '\n' + groupDescL2  
  
    return groupDesc + "\n" + jobLine1 + "$" + jobLine1Amt + "\n" + jobLine2 + "$" + jobLine2Amt + "\n" + jobLine3 + "$" + jobLine3Amt + "\n" + groupTotal
```

Label Expression

```
def FindLabel ( RELAVENT INPUTS ):  
    groupDesc = [Grouping_Table$.DESC]  
  
    jobLine1 = "JP# " + str([Grouping_Table$.JP_1]) + " FFY " + str([Grouping_Table$.JP_1_YEAR]) + " " + [Grouping_Table$.JP_1_TYPE]  
    jobLine1 = jobLine1.ljust(68)  
  
    jobLine1Amt = [Grouping_Table$.JP_1_AMT].rjust(11)  
  
    jobLine2 = "JP# " + str([Grouping_Table$.JP_2]) + " FFY " + str([Grouping_Table$.JP_2_YEAR]) + " " + [Grouping_Table$.JP_2_TYPE]  
    jobLine2 = jobLine2.ljust(68)  
  
    jobLine2Amt = [Grouping_Table$.JP_2_AMT].rjust(11)  
  
    jobLine3 = "JP# " + str([Grouping_Table$.JP_3]) + " FFY " + str([Grouping_Table$.JP_3_YEAR]) + " " + [Grouping_Table$.JP_3_TYPE]  
    jobLine3 = jobLine3.ljust(68)  
  
    jobLine3Amt = [Grouping_Table$.JP_3_AMT].rjust(11)  
  
    groupTotal = "$" + [Grouping_Table$.AMT_TXT].rjust(11)  
    groupTotal = groupTotal.rjust(80)  
  
    if len(groupDesc) > 80:  
        splitChar = groupDesc.rfind(" ",0,70)  
        groupDescL1 = groupDesc[:splitChar]  
        groupDescL2 = groupDesc[splitChar + 1:]  
        groupDesc = groupDescL1 + '\n' + groupDescL2  
  
    return groupDesc + "\n" + jobLine1 + "$" + jobLine1Amt + "\n" + jobLine2 + "$" + jobLine2Amt + "\n" + jobLine3 + "$" + jobLine3Amt + "\n" + groupTotal
```

Label Expression

```
def FindLabel ( RELAVENT INPUTS ):  
    groupDesc = [Grouping_Table$.DESC]  
  
    jobLine1 = "JP# " + str([Grouping_Table$.JP_1]) + " FFY " + str([Grouping_Table$.JP_1_YEAR]) + " " + [Grouping_Table$.JP_1_TYPE]  
    jobLine1 = jobLine1.ljust(68)  
  
    jobLine1Amt = [Grouping_Table$.JP_1_AMT].rjust(11)  
  
    jobLine2 = "JP# " + str([Grouping_Table$.JP_2]) + " FFY " + str([Grouping_Table$.JP_2_YEAR]) + " " + [Grouping_Table$.JP_2_TYPE]  
    jobLine2 = jobLine2.ljust(68)  
  
    jobLine2Amt = [Grouping_Table$.JP_2_AMT].rjust(11)  
  
    jobLine3 = "JP# " + str([Grouping_Table$.JP_3]) + " FFY " + str([Grouping_Table$.JP_3_YEAR]) + " " + [Grouping_Table$.JP_3_TYPE]  
    jobLine3 = jobLine3.ljust(68)  
  
    jobLine3Amt = [Grouping_Table$.JP_3_AMT].rjust(11)  
  
    groupTotal = "$" + [Grouping_Table$.AMT_TXT].rjust(11)  
    groupTotal = groupTotal.rjust(80)  
  
    if len(groupDesc) > 80:  
        splitChar = groupDesc.rfind(",0,70)  
        groupDescL1 = groupDesc[:splitChar]  
        groupDescL2 = groupDesc[splitChar + 1:]  
        groupDesc = groupDescL1 + '\n' + groupDescL2  
  
    return groupDesc + "\n" + jobLine1 + "$" + jobLine1Amt + "\n" + jobLine2 + "$" + jobLine2Amt + "\n" + jobLine3 + "$" + jobLine3Amt + "\n" + groupTotal
```

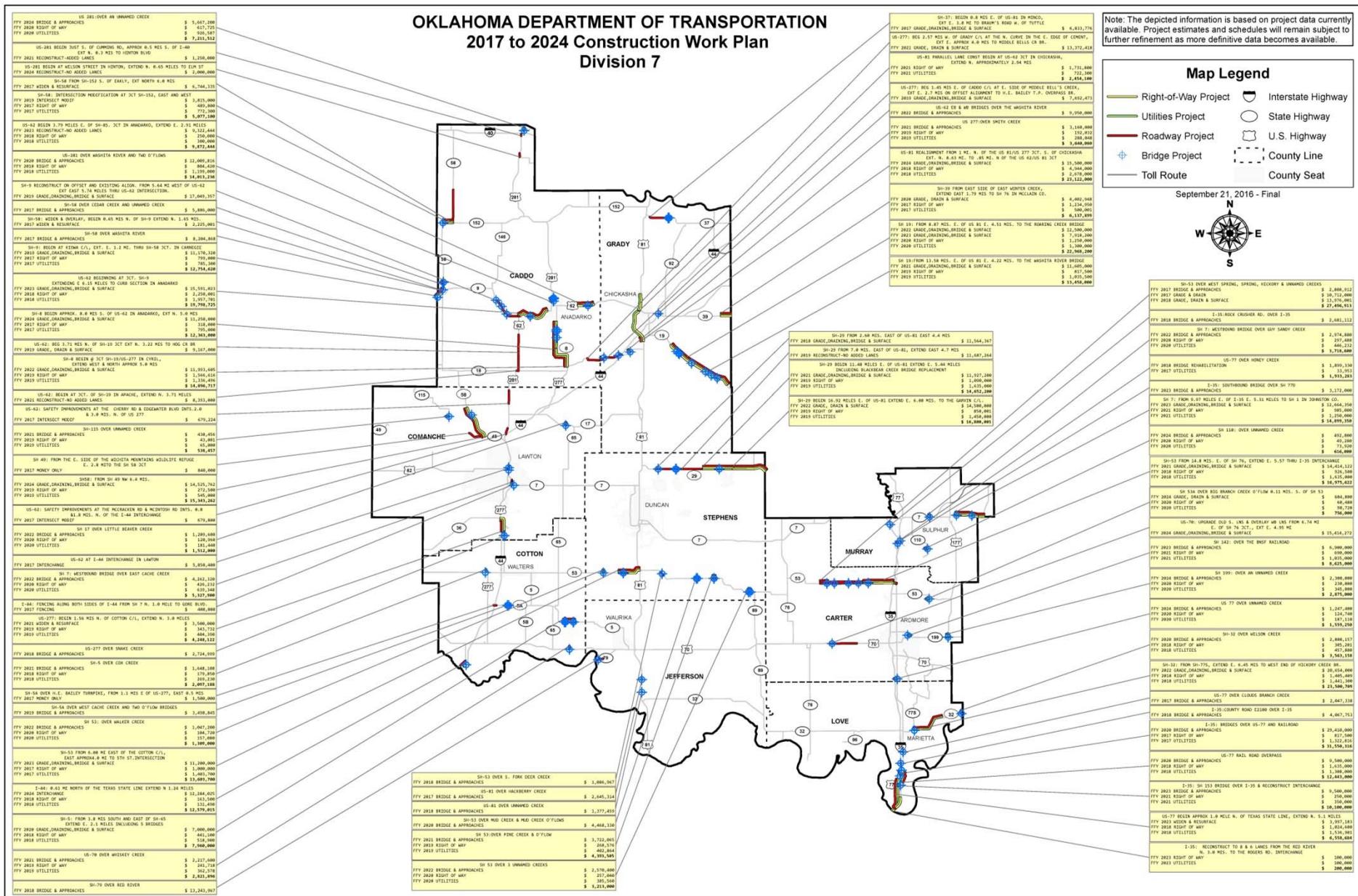
Label Expression

```
def FindLabel ( RELAVENT INPUTS ):  
    groupDesc = [Grouping_Table$.DESC]  
  
    jobLine1 = "JP# " + str([Grouping_Table$.JP_1]) + " FFY " + str([Grouping_Table$.JP_1_YEAR]) + " " + [Grouping_Table$.JP_1_TYPE]  
    jobLine1 = jobLine1.ljust(68)  
  
    jobLine1Amt = [Grouping_Table$.JP_1_AMT].rjust(11)  
  
    jobLine2 = "JP# " + str([Grouping_Table$.JP_2]) + " FFY " + str([Grouping_Table$.JP_2_YEAR]) + " " + [Grouping_Table$.JP_2_TYPE]  
    jobLine2 = jobLine2.ljust(68)  
  
    jobLine2Amt = [Grouping_Table$.JP_2_AMT].rjust(11)  
  
    jobLine3 = "JP# " + str([Grouping_Table$.JP_3]) + " FFY " + str([Grouping_Table$.JP_3_YEAR]) + " " + [Grouping_Table$.JP_3_TYPE]  
    jobLine3 = jobLine3.ljust(68)  
  
    jobLine3Amt = [Grouping_Table$.JP_3_AMT].rjust(11)  
  
    groupTotal = "$" + [Grouping_Table$.AMT_TXT].rjust(11)  
    groupTotal = groupTotal.rjust(80)  
  
    if len(groupDesc) > 80:  
        splitChar = groupDesc.rfind(" ",0,70)  
        groupDescL1 = groupDesc[:splitChar]  
        groupDescL2 = groupDesc[splitChar + 1:]  
        groupDesc = groupDescL1 + '\n' + groupDescL2  
  
    return groupDesc + "\n" + jobLine1 + "$" + jobLine1Amt + "\n" + jobLine2 + "$" + jobLine2Amt + "\n" + jobLine3 + "$" + jobLine3Amt + "\n" + groupTotal
```

Convert to Feature Annotations

- 
- Make sure all labels draw
 - Place Unplaced Labels
 - Convert to Feature Annotations
 - Arrange around border of the map

OKLAHOMA DEPARTMENT OF TRANSPORTATION
2017 to 2024 Construction Work Plan
Division 7





Sam Coldiron
scoldiron@odot.org

Paige Florie
pflorie@odot.org

Nathan Smith
nsmith@odot.org

Manager
Jeremy Planteen
jplanteen@odot.org

Oklahoma Department of Transportation
Strategic Asset and Performance Management Division
GIS Management Branch
200 NE 21st St.
Oklahoma City, OK 73105