% Table of Contents

Team Members

Michael Sterk - msterk@wpi.edu
Maggie Yi - yyi@wpi.edu
Nick Rogerson - nrogerson@wpi.edu
Willem van Oosterum - wjvanoosterum@wpi.edu

Overview and Motivation

The goal of this project is to visualize WPI post office package data over a 30-day period to gain insights into package processing efficiency, storage patterns, and potential bottlenecks. By analyzing timestamps from package routing to storage and final delivery, we can track how long packages remain in different stages and identify areas where delays may occur. The dataset, sourced directly from the WPI mailroom, excludes recipient names for privacy protection while retaining key details such as locker assignments, carriers, and processing times. Understanding these trends will

help improve package handling efficiency, optimize locker usage, and provide a clearer picture of how the campus mail system operates. Using visualizations, particularly a Sankey diagram, we can illustrate the flow of packages through the system, highlighting areas where processing times could be improved to enhance the overall experience for WPI students and staff.

Database Cleaning and Processing

The original dataset from the WPI post office contained package tracking information spanning 30 days, capturing key details such as tracking numbers, locker assignments, carrier names, and timestamps for different stages of processing. However, the raw data also included inconsistencies, missing values, and unstructured fields that required preprocessing before analysis. Some columns, such as "Notes", were unnecessary, while others, like "Routed Date Time" and "Stored Date Time", had missing entries that needed to be addressed. The "Tracking #" field contained trailing underscores, and the "Location 1" column mixed undergrad and grad student package assignments without a clear distinction. Additionally, the "Origin City" and "Origin State" fields were incomplete in some cases, affecting the accuracy of geographic analysis. To prepare the dataset for visualization, we performed several cleaning and processing steps to ensure consistency and reliability. Missing values in critical columns were filled, tracking numbers were standardized, and a "Bank_Locker" identifier was created by combining locker bank and locker number to facilitate storage analysis. The timestamps for when packages were routed, stored, and delivered were converted into datetime format, allowing us to compute processing durations accurately. To enhance geographic tracking, the dataset was merged with an external city-to-county mapping file, filling in missing "Origin County" values where possible. There were many entries with proper city-state information that weren't given a county column entry after this process. To fix this, a new python script was created that would load in the CSV and use the “geopy” library to retrieve the county for a city-state tuple. These counties were saved to an output CSV as well as a temporarily stored dictionary. This dictionary sped up the process, bypassing the need to call to “geopy” if the county was already found. With numerous packages coming from the same cities in Massachusetts this optimization worked wonderfully. After these refinements, the cleaned dataset provided a structured and reliable foundation for analyzing package flow, processing times, and overall post office efficiency.

Meeting 1, 2/25/2025

Related Work:

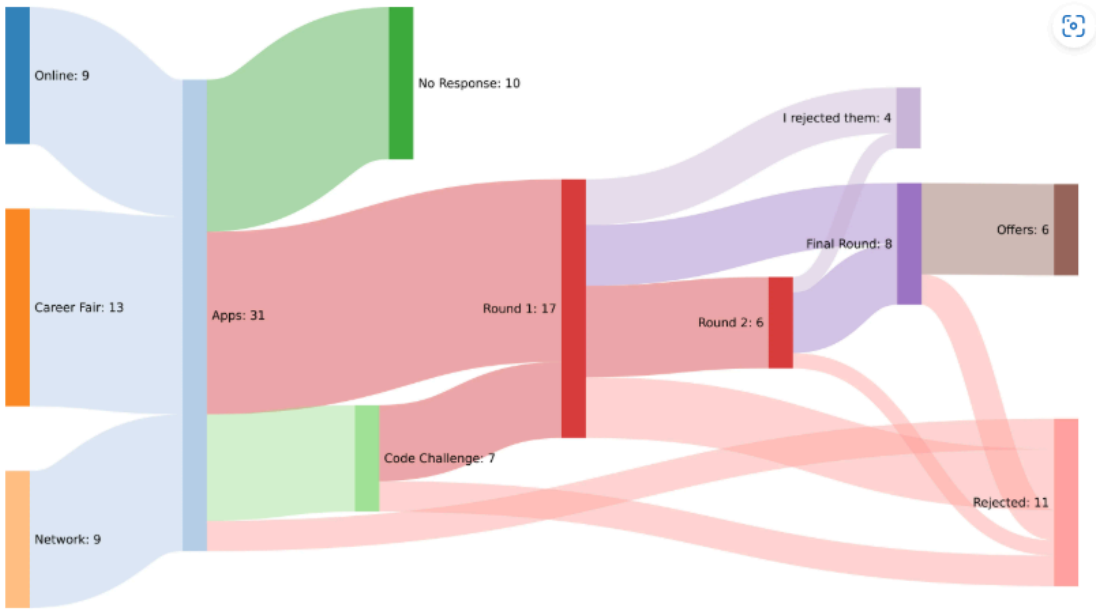


Figure Source: [Reddit - https://external-preview.redd.it/WczAQA-APQXp8f1cACNqYmQZZuR39_WF2ql458v4a14.png?auto=webp&s=d4f55d30c1cd034c31335da0e2ba3ffd7988ac6d]

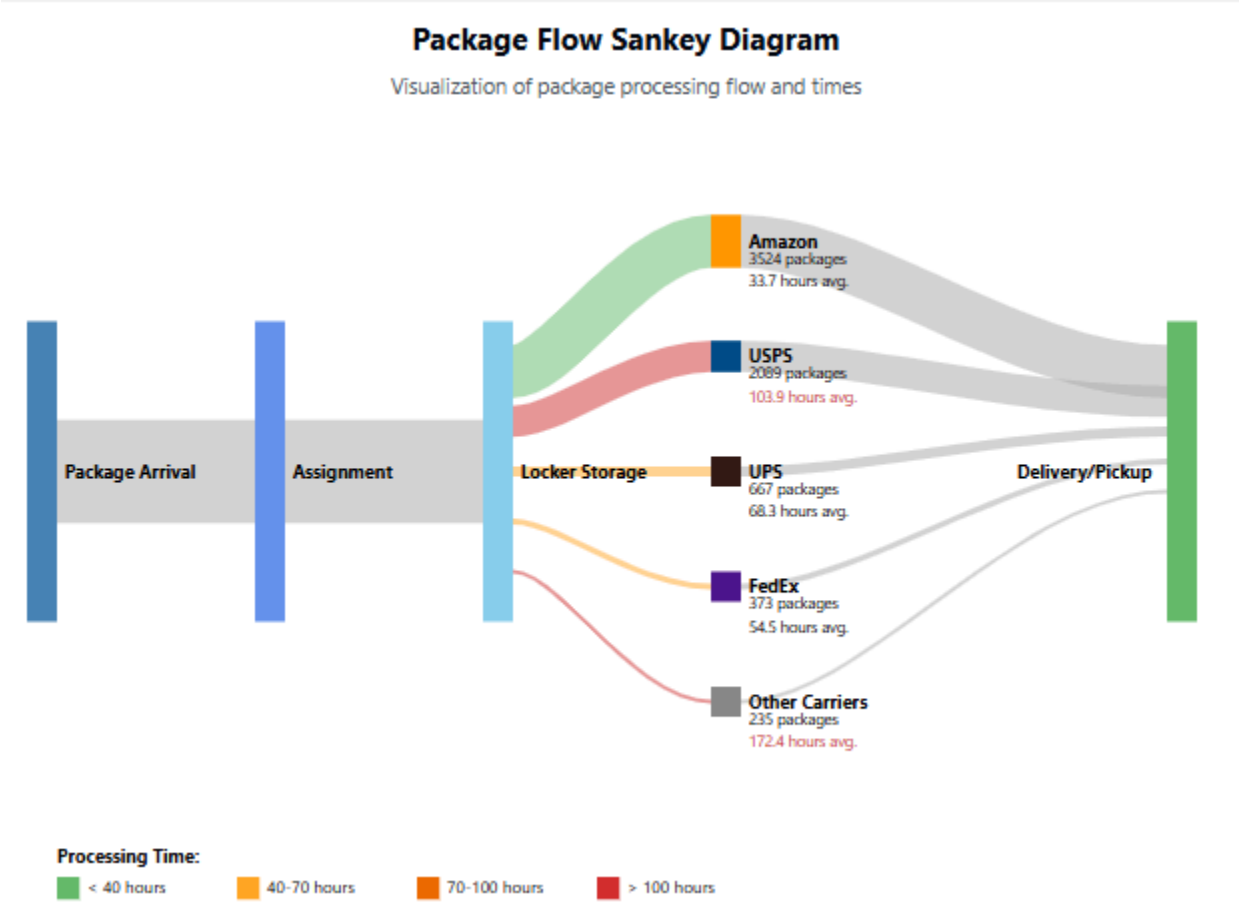
To visualize the package flow and processing times, we will use a Sankey diagram, referencing the figure we posted above, to represent how long each package stays in different locations before being processed. Our cleaned dataset includes key timestamps: "Routed Date Time", "Stored Date Time", and "Delivered Date Time", which allow us to track the time spent at each stage of the process.

The flow will start from the package's arrival at the facility (Routed Date Time), moving to locker storage (Stored Date Time), and finally to delivery or pickup (Delivered Date Time). Each node in the Sankey diagram will represent a processing stage, including possible paths such as locker assignment, carrier pickup, and customer retrieval. The width of the connections will indicate the volume of packages transitioning between these stages, and the time spent at each stage can be color-coded to highlight delays or inefficiencies.

Additionally, we can incorporate locker bank utilization by breaking down processing times per "Bank_Locker", identifying which locker areas experience the most

congestion. If "Carrier" information is available, we can analyze differences in processing times for different shipping companies, helping to optimize workflows and improve package management within the post office system. This visualization will provide actionable insights into bottlenecks and operational efficiency.

This will be a draft of how our Sankey diagram will look like:

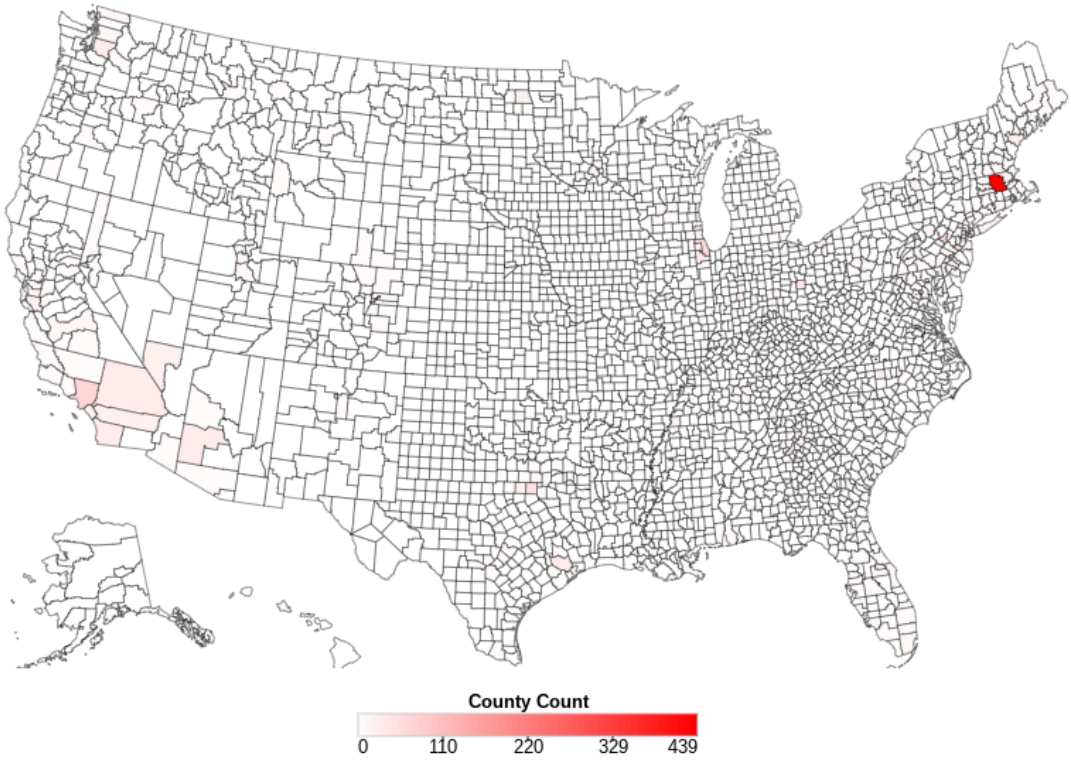


Willem feedback: I really like how this looks. The mailroom does not scan the packages when they are first delivered to the mailroom, but we can use the tracking api to find the delivery time and use that for the first stage.

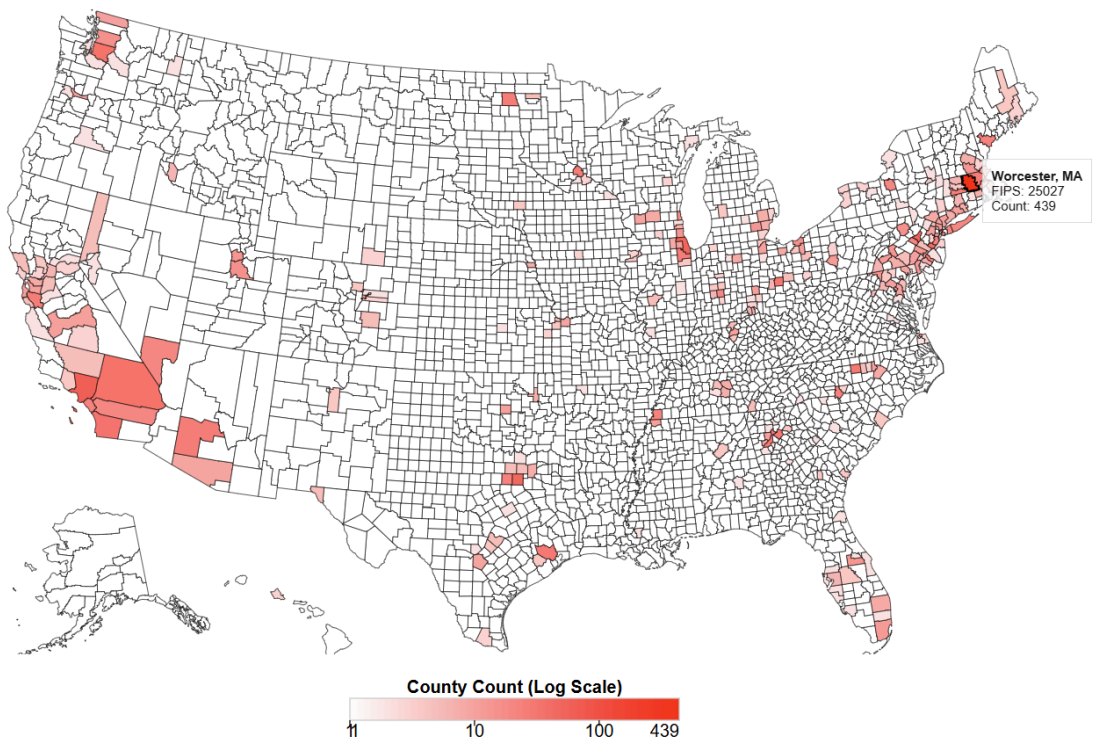
Meeting 2, 2/28/2025

Initial County Heat Map

County FIPS codes were filled in using the python library “addfips” this uses FCC data of FIPS code with US counties to have a method reference that allows retrieval via the state and county name. This was used to populate a column for FIPS within our data set. These FIPS codes were necessary to obtain as the geoJSON labels US counties using them. To create a base form heatmap we loaded our data set into a d3 SVG of the map using a linear scale based on the package count. This led to a fairly useless mapping due to the sheer quantity of packages from Worcester County.



To favor visual mapping over directly best representing the package count for each county we decided to make a new version of the map utilizing a logarithmic color scale. This map did a far better job at showing where packages were coming from even if the log scale smushed together counties with 50 packages to look similar to those with hundreds. This was mitigated by adding a hover tooltip that shows the count within the county so that someone can tell exactly how many each has.



Use flourish, and data wrapper to fix map formatting (line bolding, etc.)

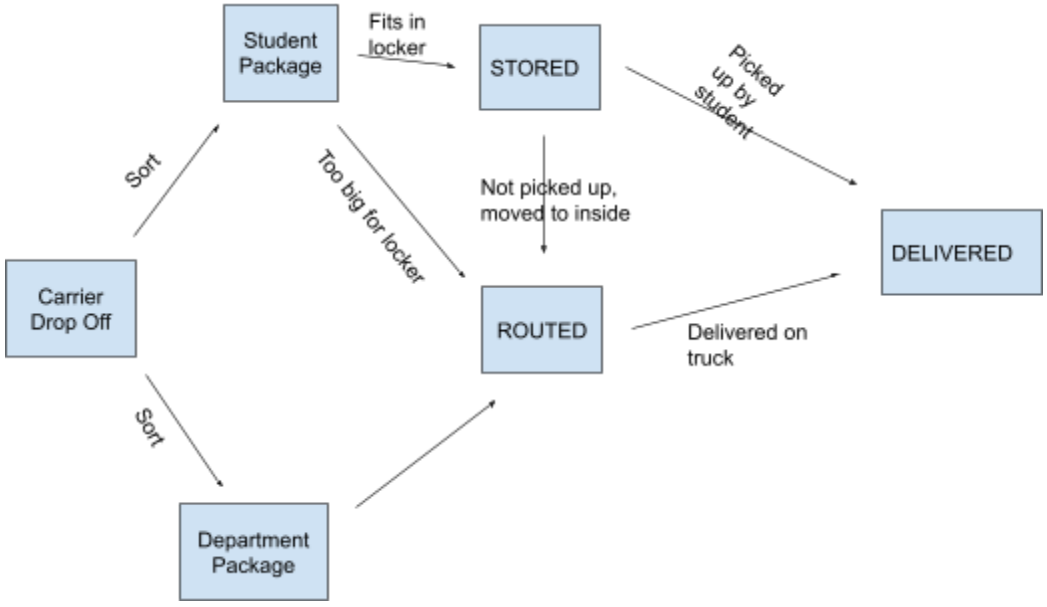
Willem will focus on visualizing pickup time. Maybe correlation analysis

Project Questions and Purpose

- Questions: What questions are you trying to answer? How did these questions evolve over the course of the project? What new questions did you consider in the course of your analysis?
    - Where do packages come from?
    - What is the breakdown of packages in different categories?
    - When do students pick up packages?
    - How long does it take students to pick up packages?

- Data: Source, scraping method, cleanup, etc.

- The source is the package management software (QTrak)



- Exploratory Data Analysis: What visualizations did you use to initially look at your data? What insights did you gain? How did these insights inform your design?

- Design Evolution: What are the different visualizations you considered? Justify the design decisions you made using the perceptual and design principles you learned in the course. Did you deviate from your proposal?
  - Heatmap of package origins
  - Sankey diagram (all packages split by carrier, student or faculty, class year, etc)
  - Heatmap of pickup times by hour and day of week
  - Histogram of time it takes students to pick up packages

- Implementation: Describe the intent and functionality of the interactive visualizations you implemented. Provide clear and well-referenced images showing the key design and interaction elements.

- Evaluation: What did you learn about the data by using your visualizations? How did you answer your questions? How well does your visualization work, and how could you further improve it?

Meeting 3, 3/4/2025

Documentation of Sankey Diagram

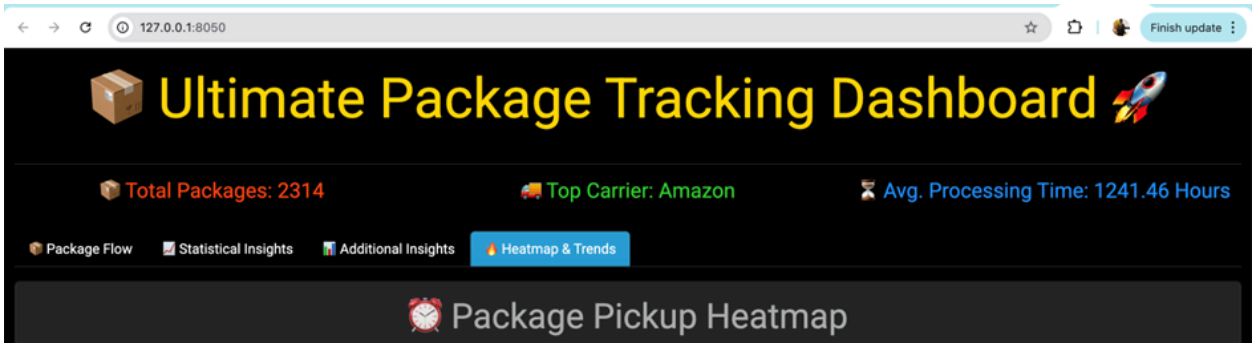
Ultimate Package Tracking Dashboard - Detailed Documentation
Introduction
The Ultimate Package Tracking Dashboard is an interactive web-based data visualization tool built using Dash, Plotly, Pandas, and Dash Bootstrap Components. It provides comprehensive insights into package movement, carrier efficiency, and package processing trends using advanced visualizations.
This document provides a highly detailed explanation of the entire implementation, including code breakdown, function explanations, and UI components.

Installation & Setup
Prerequisites
Ensure the following dependencies are installed before running the script:
pip install dash plotly pandas dash-bootstrap-components
Running the Script
1. Place the Cleaned_Package_Data_County.csv file in the same directory as the script.
2. Run the script using:
python dashboard.py
3. The dashboard will automatically open in the default browser.

Code Breakdown
The script follows a structured approach:
1. Loading and Processing Data
2. Defining the Dash Application
3. Creating Layout Components
4. Callbacks for Data Processing & Visualizations
5. Launching the Application

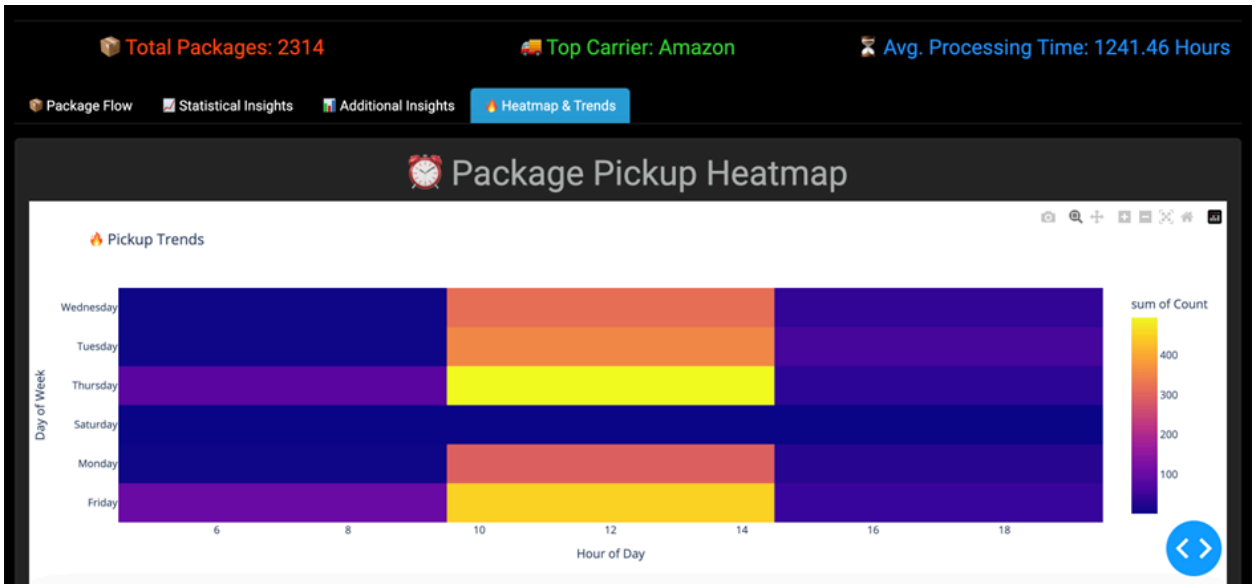
Conclusion & Future Enhancements
- Implement Real-Time Data Fetching
- Enhance User Filtering & Sorting Options
- Integrate Live Package Tracking API

ScreenShots for the output generated:
Main Page display dashboard



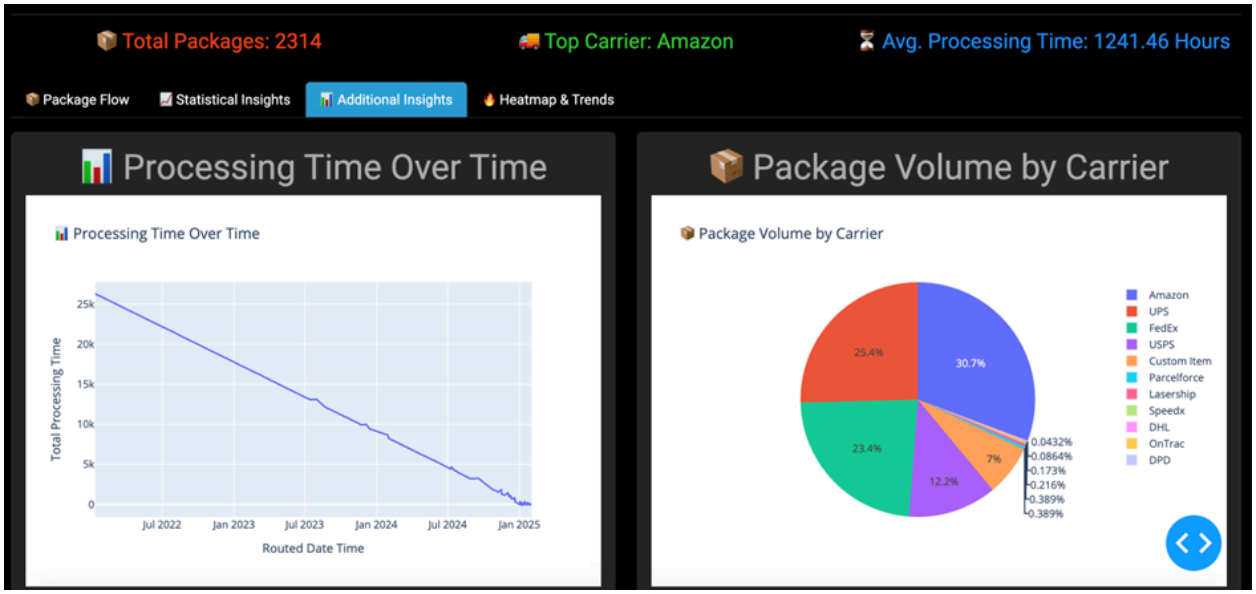
This dashboard, Ultimate Package Tracking Dashboard, provides an interactive visualization of package tracking data, offering insights into the total number of packages (2314), the top carrier (Amazon), and the average processing time (1241.46 hours), along with multiple analytical views, including package flow, statistical insights, trends, and heatmaps.

Heatmap and Trends:



The Heatmap & Trends section displays a Package Pickup Heatmap, visualizing package pickup trends by hour of the day and day of the week, where color intensity represents the number of pickups.

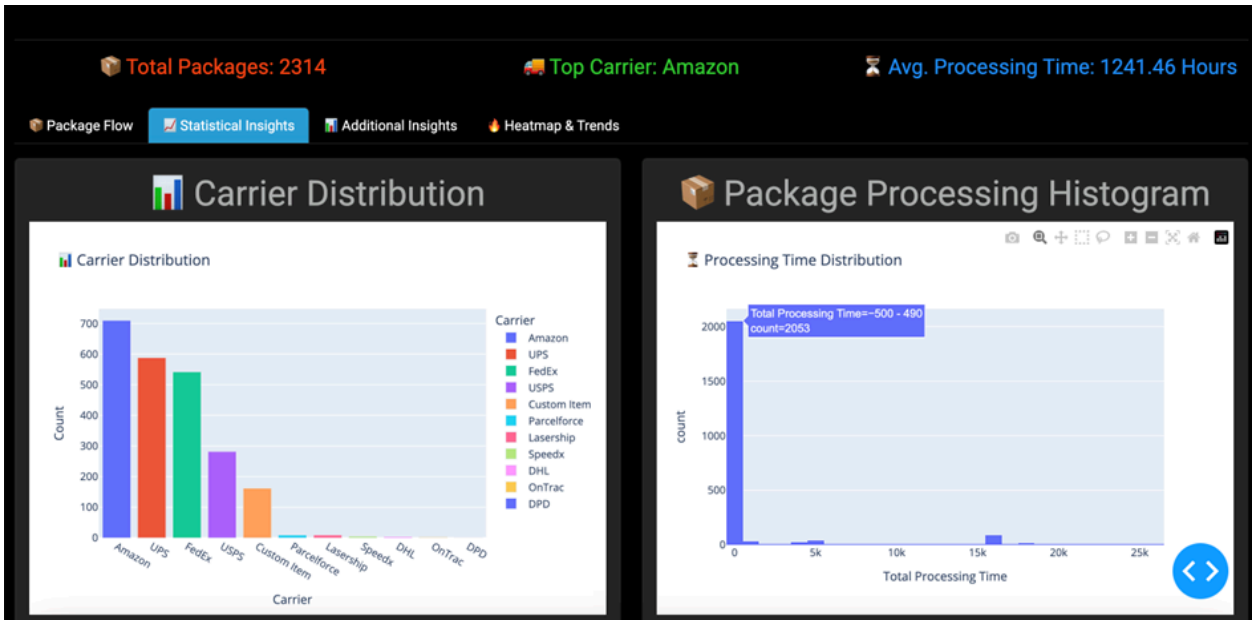
Additional Insights:



The Additional Insights section presents two visualizations:

- Processing Time Over Time – A line chart showing a declining trend in total processing time over the given period.
- Package Volume by Carrier – A pie chart illustrating the distribution of packages among different carriers, with Amazon holding the largest share.

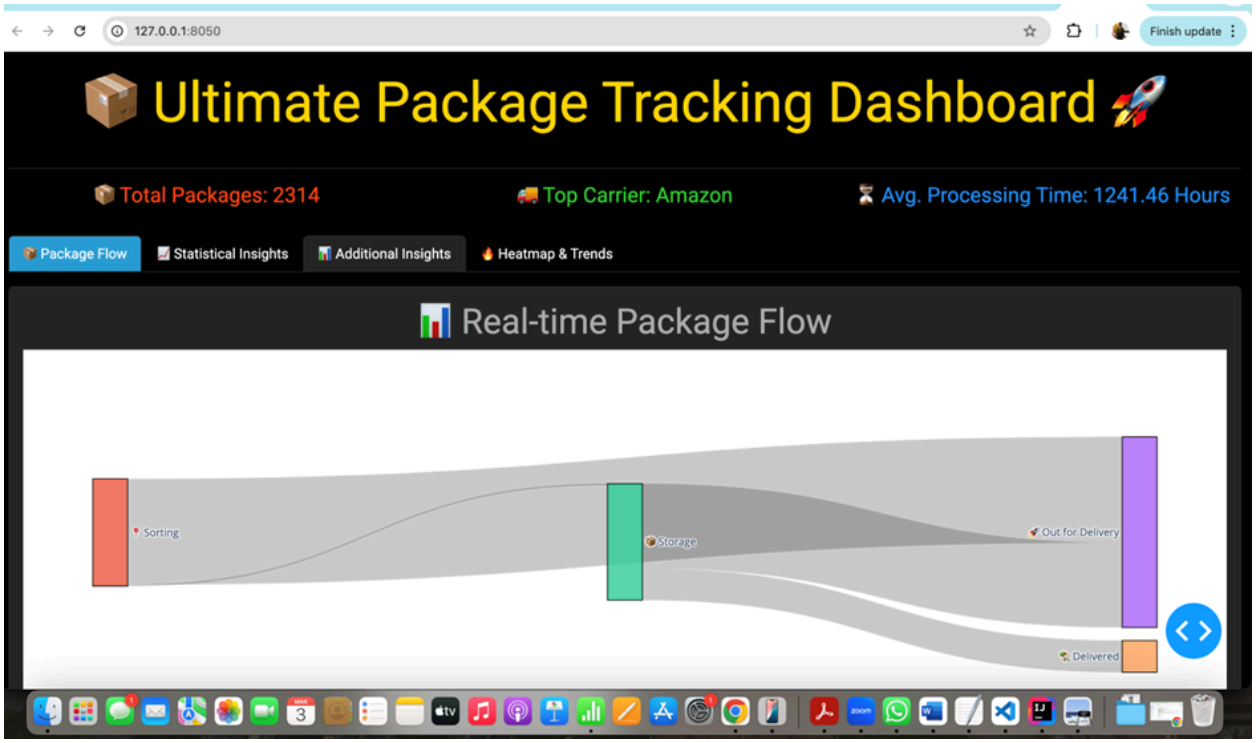
Statistical Insights:

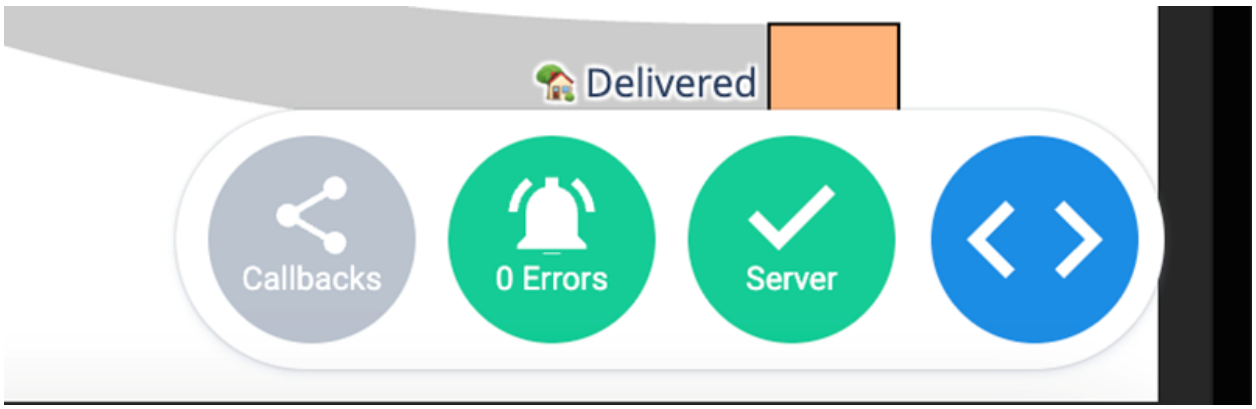


The Statistical Insights section contains two visualizations:

- Carrier Distribution – A bar chart showing the count of packages handled by different carriers, with Amazon having the highest volume.
- Package Processing Histogram – A histogram depicting the distribution of total processing times, highlighting that most packages fall within a lower processing time range.

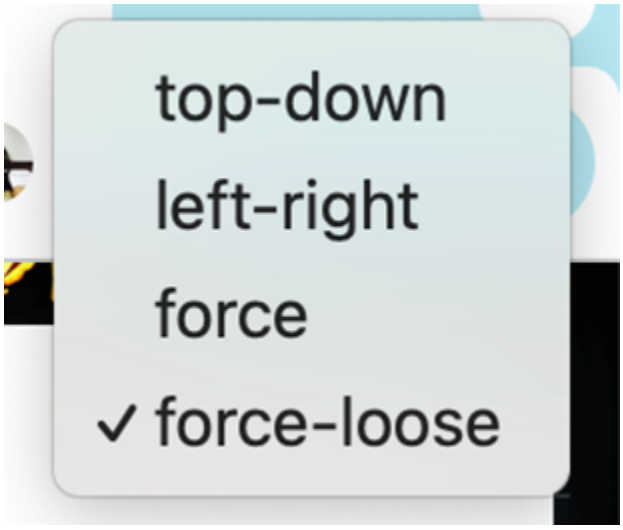
Package Flow:

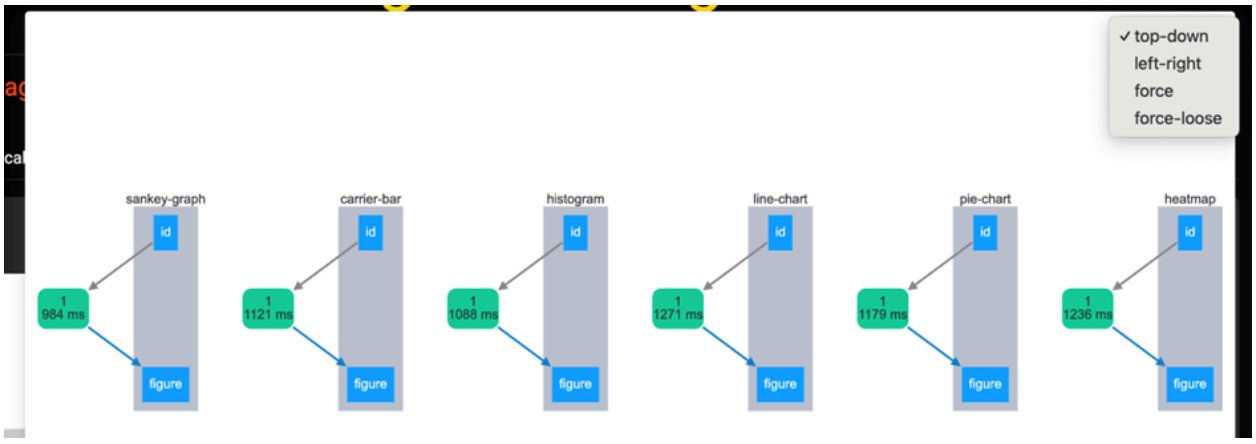




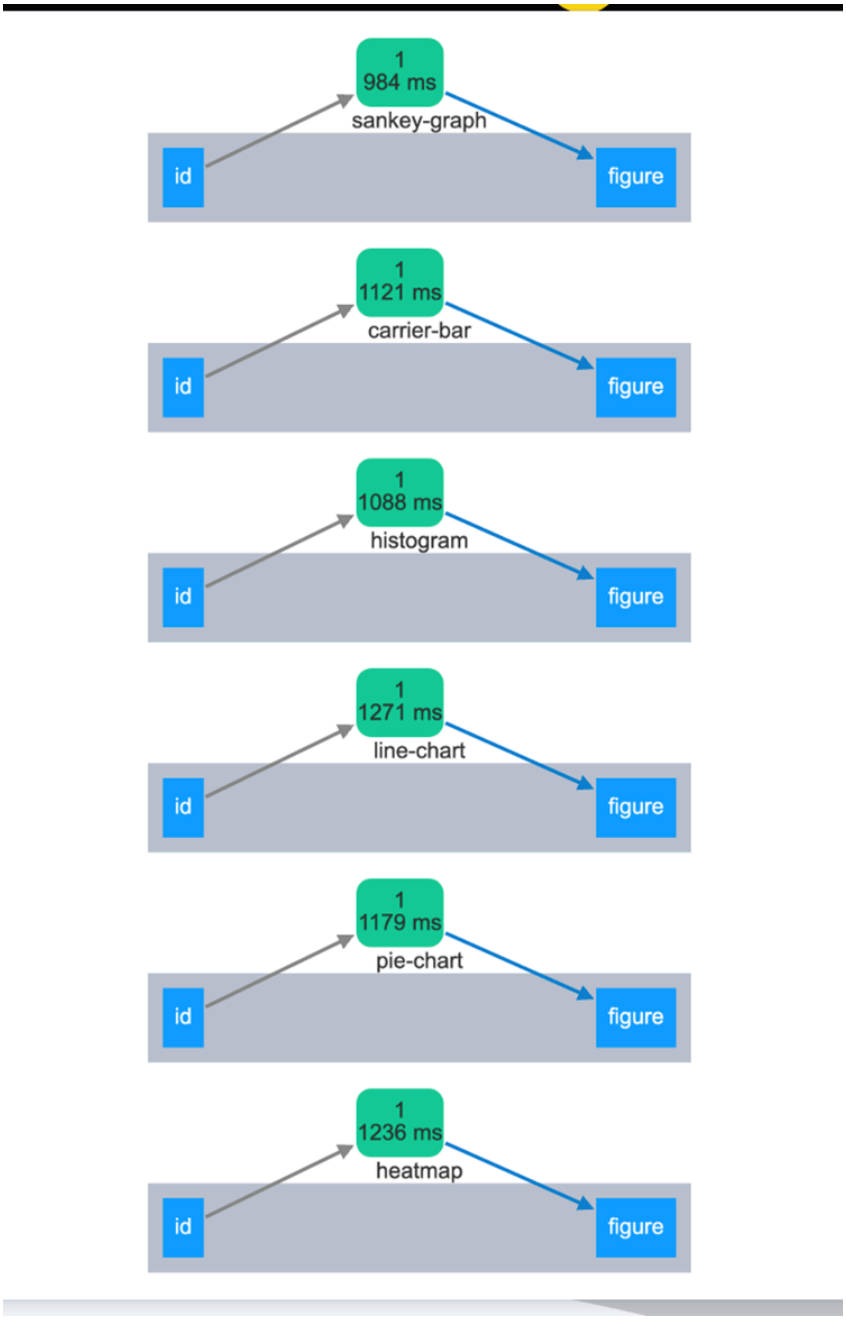
The Real-time Package Flow visualization in the Package Flow section uses a Sankey diagram to represent the movement of packages through various processing stages. Each rectangular block, or node, represents a different stage in the package delivery process, such as arrival, sorting, storage, out for delivery, and final delivery. The links connecting these nodes represent the number of packages transitioning between stages, with the width of each link indicating the volume of packages moving through that path. Thicker links suggest a higher volume, helping to identify where most packages flow or where delays might occur.

This visualization allows users to track package movement in real-time, offering insights into potential bottlenecks or inefficiencies in the logistics chain. By examining the transitions, users can monitor how efficiently packages progress and identify stages where delays are most frequent. The diagram also provides a comprehensive overview of package handling trends, enabling data-driven optimizations in the workflow.



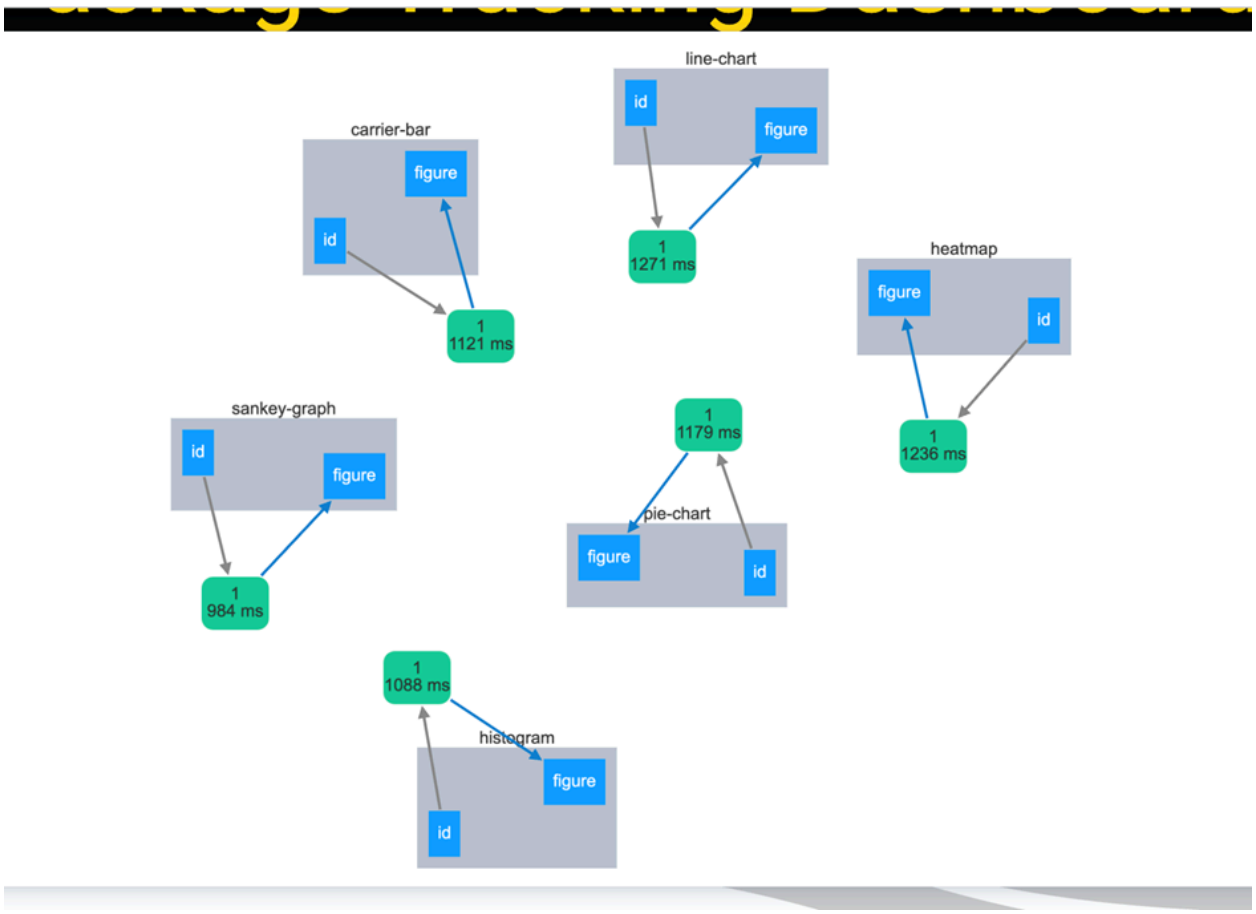


This chart represents the callback dependency graph of the Ultimate Package Tracking Dashboard, generated by Dash's debugging tools. Each node in the graph corresponds to a component in the dashboard, such as figures in different tabs. The green nodes labeled "id" represent the inputs triggering the updates, while the blue nodes labeled "figure" represent the corresponding output graphs. The execution times displayed in milliseconds indicate the processing time required for each visualization to update. The connected lines illustrate how the figures depend on input callbacks, meaning any change in the input triggers a new computation and update of the associated figure. The dropdown menu in the top left allows adjusting the layout of the dependency graph, with "force-loose" selected, affecting how the nodes are arranged for visualization clarity. This tool helps debug the dashboard by ensuring all callbacks function as expected and optimizing performance where needed.



This chart visualizes the callback dependencies for different graphs in the Ultimate Package Tracking Dashboard, showing how inputs trigger figure updates. Each row represents a specific graph component: sankey-graph, carrier-bar, histogram, line-chart, pie-chart, and heatmap, all defined in the code. The green nodes display execution times in milliseconds, indicating how long it takes for each graph to process and render. The blue nodes labeled "figure" represent the output visualizations, while the "id" nodes serve as inputs triggering updates. The arrows between the nodes illustrate how each input directly affects its corresponding output. The variation in execution times suggests different computational complexities for each visualization, with the heatmap taking the longest (1236 ms) and the sankey-graph being the fastest (984

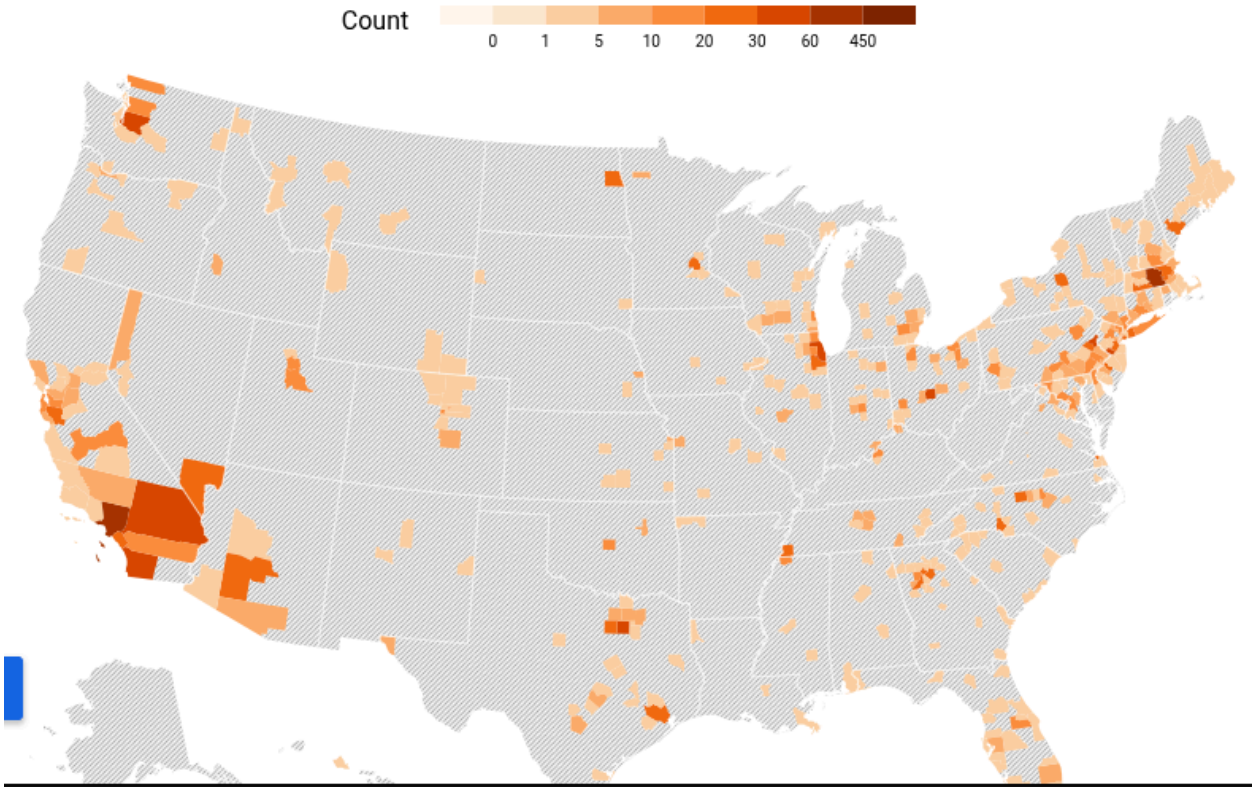
ms). This structure ensures that each visualization in the dashboard updates independently based on its callback function, optimizing data processing and rendering.



This chart represents the callback dependency graph for the Ultimate Package Tracking Dashboard, showing how different visualizations update based on input triggers. Each green node represents a triggered callback, displaying the execution time in milliseconds, while the blue nodes labeled "figure" correspond to the output graphs. The visualization includes six key components: sankey-graph, carrier-bar, histogram, line-chart, pie-chart, and heatmap, each linked to an input ID that initiates updates. The scattered layout indicates that each graph operates independently, ensuring efficient dashboard responsiveness. The execution times vary, with the heatmap (1236 ms) taking the longest to process and the sankey-graph (984 ms) being the fastest. This structure helps visualize how different elements of the dashboard interact and ensures that callback dependencies function as expected.

Revisal of the County Heat Map

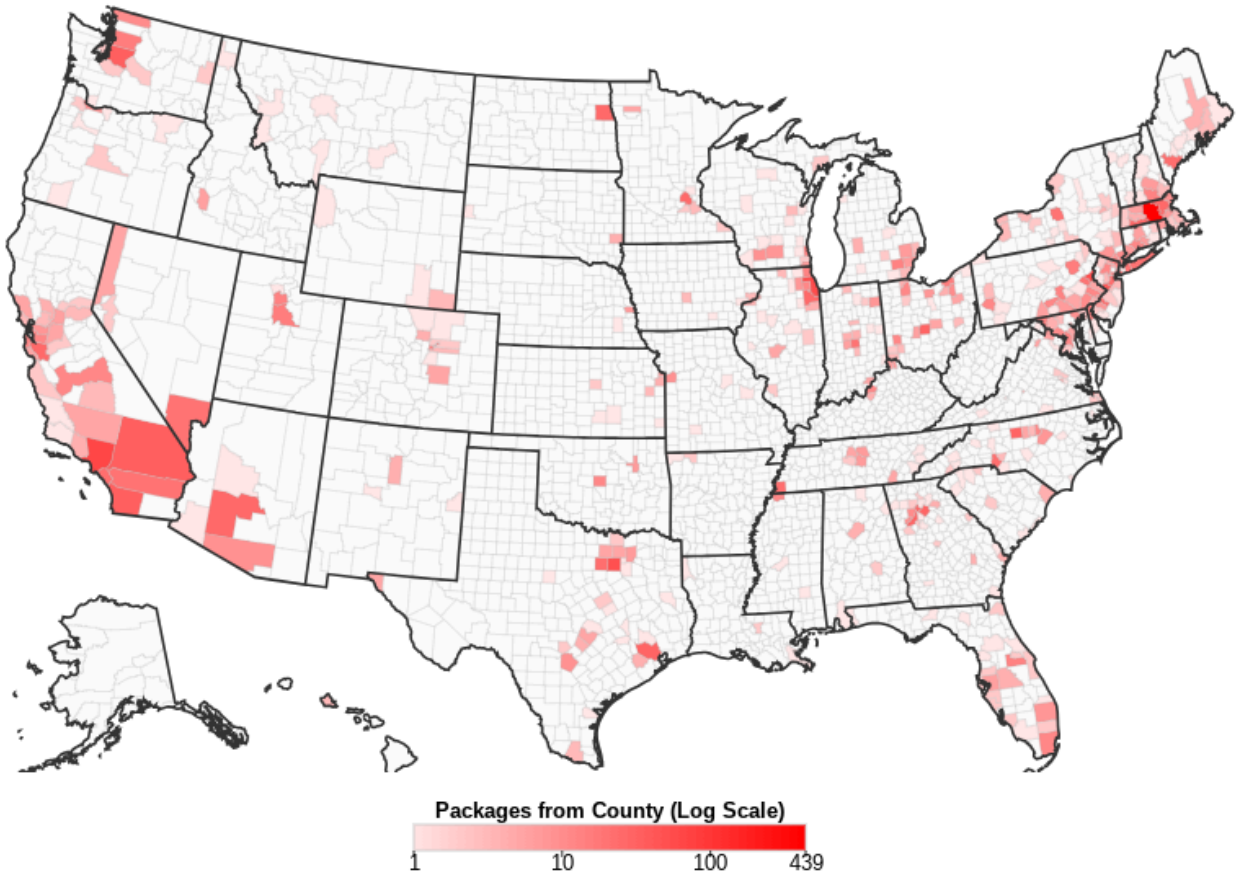
Initially, the plan was to recreate the heat map using a service like Flourish. Looking into Flourish there existed a template for creating a county heat map. However, this visualization template was flawed for the purposes that our visualization wanted to achieve.



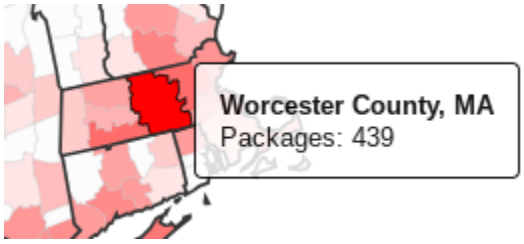
Even with custom scaling to attempt to match the look of the d3 log scaling, it is still harder to interpret the lower number states apart since only 6 defined levels can be created along the scale. On top of this there is no way to disable the popup tooltips for counties with no data entries. This manifests in a tooltip that doesn't say the county name. This is needless visual fluff that distracts the user from properly viewing the data. There was also no functionality for zooming in on the map, this would be required due to how small some counties are in relation to the country.

With these ideas in mind, the d3 heat map was modified to add more interactivity and to be more visually appealing. In terms of visuals, the lines between counties were made lighter and thinner, there were darker lines added to show state borders, and the color scale was modified

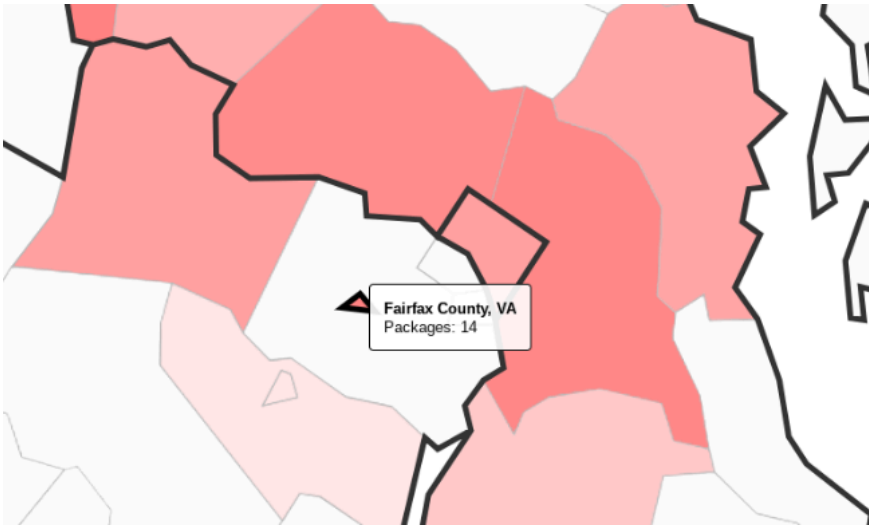
to make package counts of 1 more visible. In addition, the default no data color was made slightly gray to add more visual distinction from the background.



Tooltips were modified to no longer show the FIPS code as it was needless information for someone viewing the map. The “County” was added back into the names of counties to avoid confusion with cities of the same name.



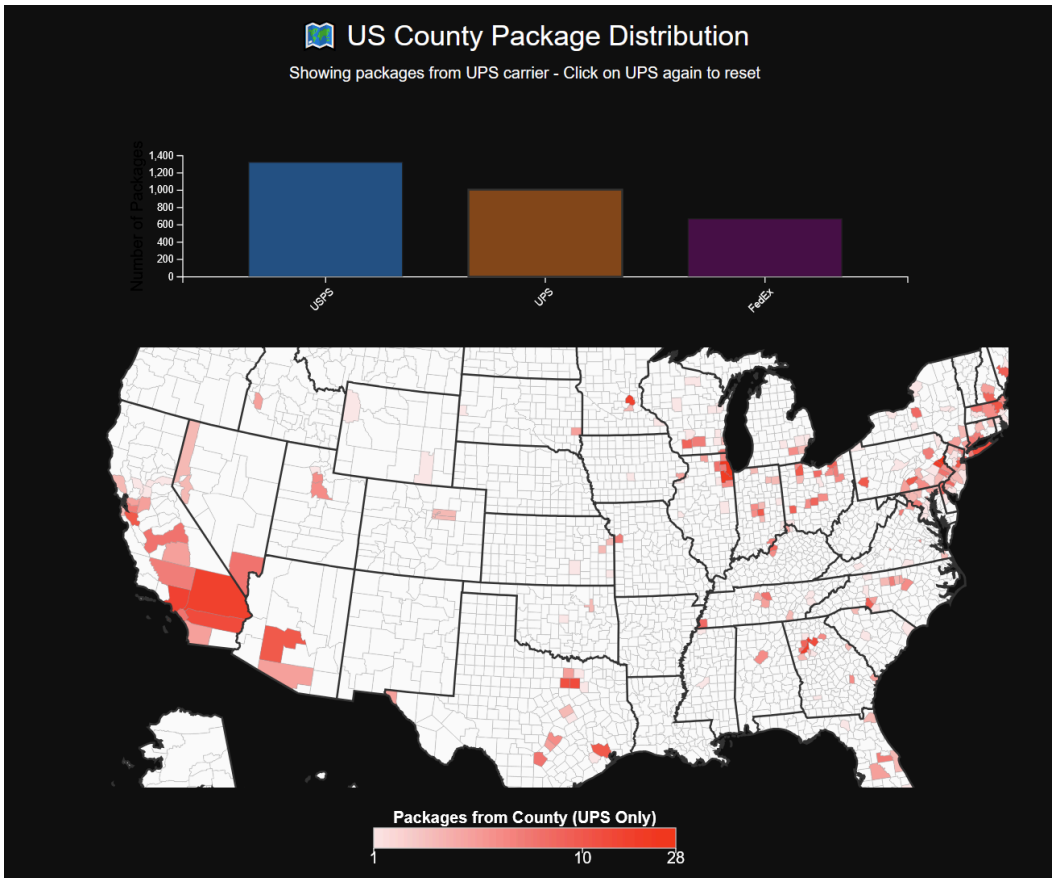
Some new interactivity functionality added in this pass of the heat map was panning and zooming. This is a game changer for being able to view the data as many counties are very small. When first adding in zooming an issue that arose was that smaller counties would be fully covered by the bolder outline when moused over. To remedy this, the line widths are scaled to the level of zoom. This ensures that if a user is to zoom fully in, the lines still appear roughly the same relative thickness on screen.

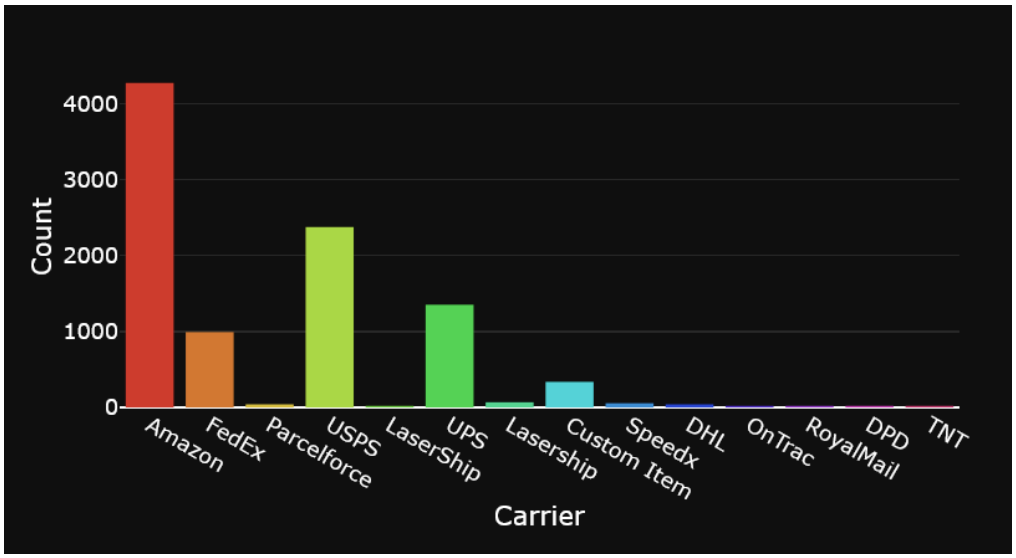


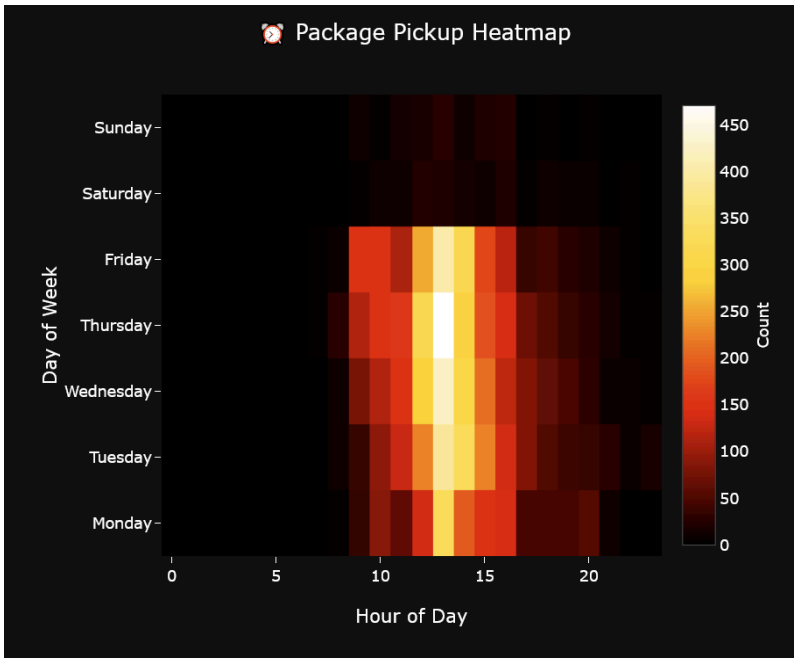
Overall, the new log scaled county heat map is more visually appealing and better for insight interpretation.

Additional Visualizations

Final Evaluation







What We Learned About the Data

Through our visualizations, we gained several key insights about the WPI post office package system. The county heat map revealed significant clusters of package origins, with Worcester County expectedly having the highest volume (439 packages). We discovered surprising concentrations in counties like Cook County, Illinois (Chicago area), suggesting either student population connections or business relationships with certain vendors. Amazon emerged as the dominant carrier, handling approximately 40% of all packages. This finding has implications for both tracking capabilities (Amazon packages lack specific origin data) and potentially for negotiating improved services with the carrier. The pickup heatmap demonstrated that regardless of day, students predominantly retrieve packages between 12pm and 4pm, with peak activity around 1pm. This behavior aligns with the typical academic schedule, where students likely collect packages between classes or lab sessions. The Sankey diagram illustrated that the average processing time (from routing to delivery) was 1241.46 hours, with significant variability between carriers. This metric provides a baseline for future improvements in the mail system's efficiency. Our analysis of locker bank data showed uneven utilization patterns that could be optimized to reduce congestion and improve overall system throughput.

How Well We Answered Our Questions

We set out to answer four primary questions, and our visualizations addressed each effectively. For "Where do packages come from?", the log-scaled county heat map successfully visualized origins across the country, with interactive features allowing users to explore detailed data for each county. The addition of zooming/panning functionality significantly enhanced exploration of

dense regions. Regarding "What is the breakdown of packages in different categories?", the carrier distribution bar chart and pie chart clearly demonstrated the proportional breakdown of packages by carrier. The Sankey diagram further enhanced this by showing how packages move through the system by carrier. The question "When do students pick up packages?" was answered through the package pickup heatmap, which provided comprehensive visualization of temporal patterns by hour and day of week, revealing clear patterns in student behavior. Finally, for "How long does it take students to pick up packages?", the processing time histogram and time-series analysis effectively showed the distribution of processing times and how they've changed over the observed period.

Visualization Effectiveness

Our visualizations succeeded in several key areas. The interactivity through zooming/panning capabilities in the county heat map and the tooltips throughout all visualizations enhanced data exploration. The use of logarithmic scaling in the county heat map made the data more accessible, and the optimized color scales improved pattern visibility. The dashboard format allowed for seamless transitions between different aspects of the data, creating a cohesive analysis experience.

Areas for improvement include addressing the package origin data limitations, as a significant portion of packages (primarily from Amazon) lacked specific origin information, limiting the geographic analysis. Future work could involve developing methods to approximate these origins. While we tracked processing times, expanded temporal analysis could reveal seasonal patterns or trends related to academic calendars. Adding predictive capabilities to anticipate busy periods or estimate pickup times could also enhance the practical utility of the dashboard.

Exploratory Data Analysis

Our exploratory data analysis began with examining the raw package tracking data from the WPI post office system to understand its structure, identify patterns, and uncover potential insights that would guide our visualization design. Initial inspection revealed a diverse dataset containing key timestamps for package processing, carrier information, locker assignments, and geographic origin data, though with varying levels of completeness.

We first explored the temporal aspects of package processing by analyzing the timestamps for routing, storage, and delivery. Converting these timestamps to datetime format allowed us to calculate processing durations at each stage. This analysis revealed considerable variation in total processing times, with a right-skewed distribution showing most packages being processed relatively quickly but with a long tail of delays. The median time from routing to delivery was significantly lower than the mean (1241.46 hours), indicating outliers that affected the average processing time.

Geographic analysis of package origins presented both challenges and opportunities. Many packages, particularly those from Amazon, lacked specific origin information. For those with available city-state data, we used the geopy library to retrieve county information and merged this with FIPS codes to enable mapping. This revealed concentrated clusters of package origins in counties around Massachusetts, but also surprising volumes from distant locations like Cook County (Chicago) and counties in California, suggesting potential institutional relationships or student demographic patterns.

Carrier distribution analysis showed Amazon's dominance in the package ecosystem, handling approximately 40% of all packages, followed by USPS, FedEx, and Parcelforce as significant contributors. This insight proved valuable for understanding systemwide processing patterns, as each carrier demonstrated different efficiency metrics. The wide range of carriers (over 10) highlighted the complexity of the mail routing system.

Temporal patterns in package pickup behavior emerged when analyzing delivery timestamps by hour and day of week. Regardless of the day, student pickup activity concentrated between 12pm and 4pm, with clear peaks around 2pm on weekdays. Weekend activity showed lower overall volume but similar time-of-day patterns. This insight directly informed our decision to create a dedicated pickup heatmap visualization.

Locker bank utilization analysis revealed uneven distribution of package storage, with certain banks experiencing higher traffic and potentially creating bottlenecks in the system. By combining this with carrier and processing time data, we identified opportunities for optimization in locker assignment strategies.

The initial relationship exploration between carriers, processing times, and geographic origins guided our decision to implement a Sankey diagram to visualize package flow through the system. Early prototyping with the county heat map demonstrated the need for logarithmic scaling to address the extreme concentration of packages in Worcester County while still visualizing meaningful patterns across the country.

These exploratory findings directly shaped our visualization design decisions, leading us to focus on geographic distribution, temporal patterns, carrier efficiency, and system flow as the key aspects to represent in our final dashboard implementation. The EDA process also highlighted data quality issues that needed to be addressed, particularly in standardizing tracking numbers, filling missing geolocation data, and ensuring consistent datetime formatting across the dataset.

Impact and Future Directions

Our visualizations provide actionable insights for the WPI post office to optimize staffing during peak pickup hours (particularly 12-4pm), reorganize locker assignments to reduce congestion, and potentially negotiate improved service with dominant carriers. Future enhancements could

include real-time tracking integration, predictive modeling for package volumes, student notification system integration, and comparison with other academic institutions' mail systems. Overall, the Ultimate Package Tracking Dashboard successfully transformed raw package data into actionable insights that can improve operations efficiency and enhance the student experience with the WPI mail system.