Building Missoula:

Interactive Permit & Development Data Dashboards

An Initiative for Enhanced Accessibility & Insights
from Permit Data

A Data Engineering Project by Luke Wyman

Table of Contents

Executive Summary

The City of Missoula Community Planning, Development, and Implementation (CPDI) department is focused on improving the efficiency, transparency, and responsiveness of its development review and permitting processes. Currently, permit data is collected at varying intervals using semi-standardized methods. This inconsistency in how and when data is gathered creates challenges in ensuring the data is consistently accessible, reliable, and useful for analysis or informed decision-making. For instance, if data reveals a seasonal spike in permit applications, CPDI can work proactively with the public to manage expectations and adjust internal workflows to reduce delays. This data, which includes information on residential and commercial new construction and remodels, helps the city understand development trends, market fluctuations, and the overall health of the community's growth around commercial and residential developments.

Currently, there is no unified, consistent system for tracking and analyzing historical building permit metrics data, which limits the ability of both internal and external stakeholders to understand what's happening around building permits for the city. While CPDI staff possess a solid working knowledge of permit volumes and timelines, there are few opportunities for both the CPDI team and external stakeholders–such as builders, contractors, and developers–to interact with the data in a comprehensive and meaningful way.

In municipalities across the country, access to permit data is presented in dashboard formats that can provide users with valuable insights into community development trends, permit processing volumes, and opportunities for operational improvements. Without such systems in place, there is no straightforward way to identify emerging patterns, forecast future development needs, or optimize the permit review process, leaving key opportunities for strategic decision-making and operational efficiency largely unexplored.

This project aims to develop interactive dashboards that will provide both CPDI staff and the public with better access to insights derived from permit data. By centralizing and organizing historical building permit data, these dashboards will allow users to explore trends, monitor permit volumes, and gain insights into the efficiency of the permit review process. The goal is to empower CPDI staff, builders, contractors, and developers with the tools they need to make informed decisions around building permits. The CPDI staff can use the dashboards to help identify any bottlenecks and develop KPIs against the dashboard. Finally, this as a great opportunity to provide transparency in the development review and permitting process.

Introduction

The rapid growth of large municipalities has elevated the importance of having development review and permitting processes that are both transparent and efficient. Economic pressures such as rising living costs, housing shortages, and increased development demands make the monitoring and analysis of building permits essential to maintaining growth. These permits serve as key indicator of community's development trends, market dynamics, and growth patterns.

In Missoula specifically, development activity continues to accelerate, yet the city lacks a unified system to organize and analyze permit data effectively. This absence creates a significant gap in the CPDI team's ability to generate accessible insights and respond to changing conditions. By addressing this deficiency, Missoula

officials would gain the tools needed to make data-driven decisions, streamline operations, and respond more effectively to community needs as they evolve over time. Without such systematic analysis, the city risks missing important signals about development patterns, potentially resulting in misaligned policies and resources. A comprehensive approach to permit data not only benefits city planners and officials but also provides valuable transparency to residents, developers, and businesses seeking to understand and participate in Missoula's growth trajectory.

This project, done in coordination between the University of Montana and the City of Missoula CPDI department, seeks to address these challenges by developing a small suite of interactive dashboards that centralize and streamline access to building permit data. These dashboards will provide both internal and external stakeholders with valuable insights into permit volumes, processing timelines, and community development trends.

By leveraging these tools, the CPDI team will be better positioned to improve efficiency, transparency, and responsiveness in the development review and permitting process. Additionally, the public will benefit from access to real-time permit information, allowing residents to track current developments. This democratization of permit data represents a significant step toward fostering a more informed and engaged citizenry, ultimately strengthening the collaborative relationship between Missoula's government and the community it serves.

Solution Overview

The solution I propose centers on the development of several interactive data dashboards designed to transform how City of Missoula CPDI processes and visualizes building permit information. The data exists in their database platform called Acella, which is a Microsoft Azure database, and in Excel spreadsheets derived from Acella reports following an audit. The dashboards I create in this project draw inspiration from municipalities that have spent resources to develop highly informative permit dashboards. Drawing from these data sources using carefully designed SQL and Python scripts, I create data sets that are effective for dashboard filtering and development.

The underlying data comes from building permit records, which reflect a wide range of construction activity in Missoula. When a property owner or developer decides to build a new home, remodel an apartment

complex, install a new roof, or construct a garage, they must apply for a building permit through the city. That application initiates a complex, collaborative review process that often involves multiple city departments such as engineering, environmental health, fire safety, zoning, and planning.

As the permit moves through these phases, it may pass through multiple cycles. A cycle begins when the applicant addresses a specific action item, such as paying a fee or providing additional details. The permit will continue circulating between the applicant and various city departments until all required approvals are obtained. Once the review process is complete and all reviewers have signed off, the permit is issued and followed by final inspections. For new dwellings, the post-review process also includes approvals for temporary certificates of occupancy (TCO) and certificates of occupancy (CO).

These cycles can vary widely in number and duration, depending on the complexity of the project. As the permit passes through each stage, whether being reviewed by the city or the applicant, this data is recorded by both manual and automated processes in Accela. The system generates a detailed history but does not have details about how long the city or applicant is responsible at each touchpoint.

Extracting this information is far from straightforward. The tables in Accela are complex and somewhat abstract to the outside observer, requiring multiple lookups and a deep understanding of the underlying data structure. In some cases, the permit process itself can become mixed around, with multiple departments and action items interacting in ways that skew the data.

So, using carefully designed SQL queries and Python scripts, I extract and transform raw process and permit detail data into structured datasets that support dynamic filtering and visualizations in dashboards. allow users to identify patterns in permit review timelines, understand which departments are involved, measure the number of cycles a permit goes through, and analyze the distribution of responsibility between the city and the applicant. The dashboards will also analyze shed some light on the shared responsibility of pushing a permit to issuance.

Make a better Vis here for permit process
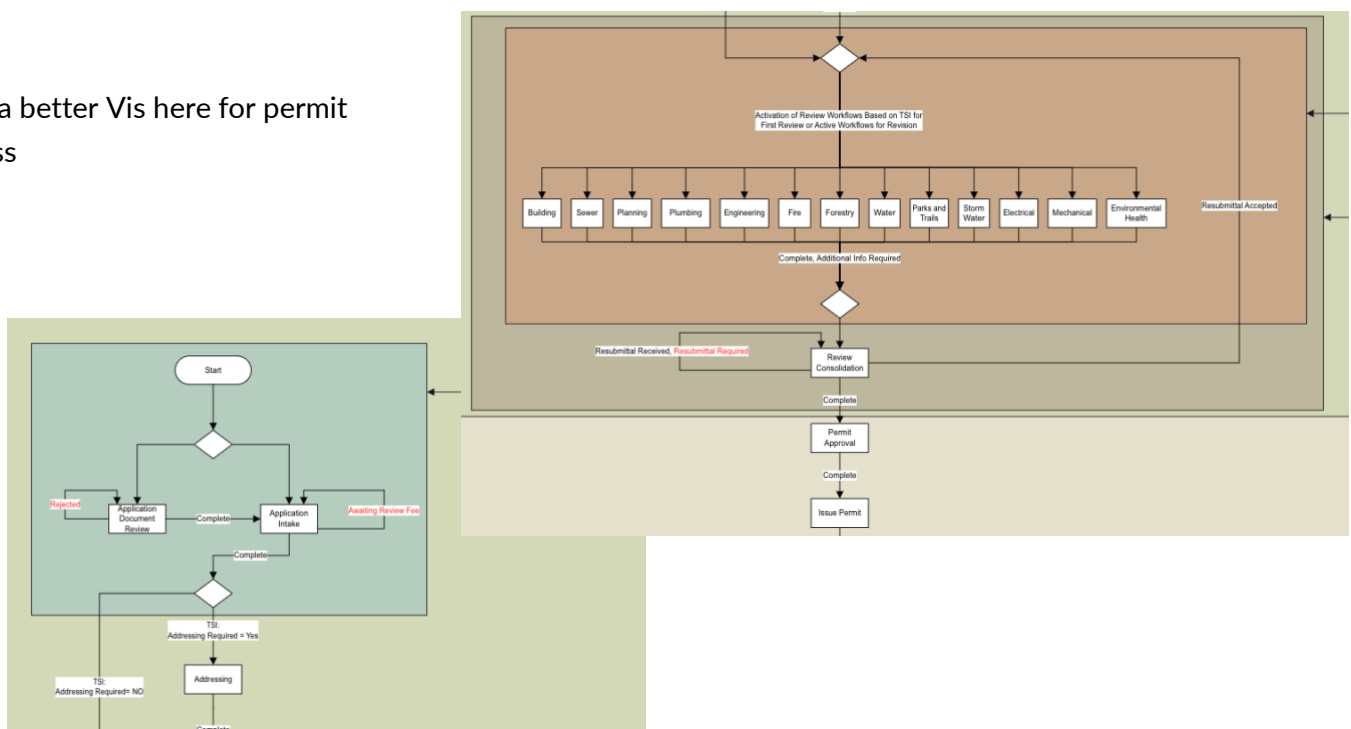
Technology Selection

While collaborating with the CPDI team as an external resource, I navigated specific technology constraints that shaped the project implementation. The city's data environment presented two primary sources: the Acella Enhanced Reporting Database and various Excel Workbooks maintained by department staff. This hybrid data landscape required two data engineering approaches—SQL scripts for database queries and Python Jupyter Notebooks for processing Excel-based information.

For visualizations and user interface development, the CPDI team determined that ArcGIS dashboards would serve as the optimal platform. This selection aligned strategically with the city's existing technology ecosystem, where ArcGIS already powers their GIS data management, city mapping, parcel information, and related spatial services. Though initially unfamiliar with this platform, I was able to use my previous experience in PowerBI and Domo to work through the logic to develop high-impact dashboards.

The ArcGIS implementation ultimately delivered highly functional, visually compelling dashboards that the city staff can maintain independently after project completion. Equally important, was that I learned a new skill. Once the data was hosted on the platform, I had to use ArcGIS's data expression language, Arcade, to shape the data for my data visualization needs.

Data Sources & Integration

To stay consistent with the team's resources and general workflows for data management, many technology decisions supporting the dashboards were determined by the CPDI team. Given accessibility, learning curve, and time constraints to the CPDI team's data environments, my primary focus was developing scripts that

<TBD Flow diagram showing data sources, methods, transformations, dashboards, experiences TBD>

automated data extraction and transformations. I aimed to maximize automation within my accessibility constraints, while attempting to be forward-thinking about dashboard maintenance throughout future lifecycles.

When the project is completed, I will conduct a session with the CPDI team's technical stakeholders to ensure they are equipped with the knowledge needed to maintain and update the dashboards moving forward. During this session, we will review the standard operating procedures for managing the data pipelines, ensuring that data updates are seamless, and the dashboards remain accurate and reliable over time. This will include discussions on data acquisition, processing, and visualization, as well as troubleshooting steps to address potential issues. By providing clear and structured procedures, the CPDI team will be empowered to manage the dashboards independently.

Sub-Projects/Dashboards

To better manage my time and resources, I chose to break this initiative into several sub-projects for enhanced accessibility and insights from permit data. This strategy allowed me to focus on each component more effectively, ensuring that each piece had clear objectives and deliverables. By dividing the initiative into smaller, manageable sub-projects, each resulting in its own dashboard, I was able to work more independently. This approach helped me avoid potential delays, such as waiting on process diagrams or access to the database, while still advancing the overall project efficiently.

01 Building Permit Metrics: Create automated visualizations of key building permit process metrics like processing times, resubmittals, and volumes.

02 Building Permit Operational Insights: Design a dashboard to enhance internal operations by providing metrics on team-specific workflow activity related to building permits.

03 Community Development Snapshots: Provide long-range, filterable reporting on building and other permits to help the community analyze growth patterns in city development.

01 Building Permit Metrics

In this sub-project, I built a data pipeline that transformed raw back-end SQL database tables from the Acella Enhanced Reporting Database into dashboard-ready datasets. In the dashboard, users can find fully automated visualizations of metrics such as permit processing times, volumes, and more. These visualizations are designed to update daily as new data populates within the system, ensuring people always have access to current information. Public-facing metrics were researched and decided upon with the goal of improving overall transparency and helping the public better understand the permit process and lifecycle. The result of this solution is a fully automated, filterable dashboard that requires minimal oversight that the city can release for the public.

<WIP>

Requirements Gathering

Before developing solutions for this project, I took the time to thoroughly understand the building permit processes and the needs of stakeholders to ensure my approach aligned with the team's vision for the new dashboard. As I was informed, the team lacked automated reporting for permit processing and cycle review timelines. Given that one of the primary goals of the dashboard was to streamline the permit application workflow, it was essential to gain a clear understanding of each stage in the process.

Since I was unfamiliar with the building permit process, I requested a diagram, which I then refined to highlight key touchpoints and the flow of information essential for the dashboard. This critical step formed the foundation for developing the logic needed to structure the data to meet the dashboard's needs. The visual representation helped clarify the project's goals and allowed me to define a permit cycle in the data, specifically identifying key hand-offs between the city and the applicant throughout the process.

Since there was not an existing dashboard to reference, I studied dashboards from other municipalities to better understand best practices and key features. I worked with the CPDI team to develop and understand the visualizations that were to be shown in the dashboard. This research and preparation helped shape my approach to structuring the data effectively, since I was not familiar with the native database tables that contained this information.

Data Acquisition and Preparation

Proper data structuring required balancing granularity with usability. The challenge was to track permits throughout their entire lifecycle while ensuring the data format supported intuitive filtering, aggregation, and meaningful insights for stakeholders. To achieve this, I created visualization requirements list that allowed me to theoretically plan the dataset by predefining key features. I then worked closely with the CPDI team to identify the relevant native database tables and fields needed to build the logic for querying the data effectively.

With our requirements in place, I built a test dataset that included the fields needed to display the requested information in the dashboard. From there, it was a matter of locating these fields in the database and structuring the data to fully support filtering and drill-down functionality in the visualizations.

Once I had access to the database, I developed an initial query using a few test building permits. This helped guide my decisions on how to merge tables and perform calculations in SQL. Initially, the query produced a satisfactory table, but I encountered significant performance issues, with processing times reaching about a minute and a half per permit. Given that over 6,000 permits were relevant for this dashboard, this was not feasible. I revisited the query, leveraging my improved understanding of the relevant data tables. By carefully scripting and

<TBD Simplified Building Permit Process Diagram TBD>

optimizing the SQL, I was able to reduce the processing time from 90 seconds per permit to just 12 seconds for the entire table.

To access the data I needed, the CPDI team provisioned the dataset by connecting the SQL query script to the ArcGIS backend using a REST API for scheduled updates. This connection was crucial because it allowed for real-time data synchronization between the backend database and the ArcGIS platform. Using the REST API, we automated the process of retrieving the data at defined intervals. The data from Accela would be pulled and then updated in ArcGIS without manual intervention. This scheduled update system ensured that as new permit data entered the system, it was automatically reflected in the dashboards, providing stakeholders with the most current insights available.

Dashboard Development

<TBD Extensively prepared for dashboard development. I have a list of requirements that will be built with attributes/filters already scoped. TBD>

Results & Capabilities

<TBD Extensively prepared for dashboard development. I have a list of requirements that will be built with attributes/filters already scoped. TBD>

Data Updates

The CPDI team implemented an automated integration that links SQL query scripts directly to the ArcGIS platform through a REST API. This integration enables seamless synchronization of data in real time, automatically pulling and updating permit information from Accela according to a predetermined schedule. When new permit data is entered, it automatically appears in Accela and will surface as data in the dashboard whenever the data is refreshed, without requiring manual updates, ensuring stakeholders always have access to the most current information.

Risks

<TBD TBD>

02 Building Permit Operational Insights

In this sub-project, I built a data pipeline that transformed raw back-end SQL database tables from the Acella Enhanced Reporting Database into dashboard-ready datasets specifically designed for internal permit processing teams. This dashboard focuses on team-specific metrics, providing detailed visualizations of workflow efficiency, processing bottlenecks, and workload distribution across departments. Internal staff can track how long permits stay in review stage and identify where delays commonly occur within their specific team's process. These visualizations update daily as new data populates the system, ensuring the CPDI team has current information for operational decision-making. The result of this solution is a fully automated, filterable dashboard that requires minimal oversight that the city can release for internal teams.

<WIP>

Requirements Gathering

<TBD TBD>

Data Acquisition and Preparation

<TBD Comes from Acella ERD. Needs to be developed, but is highly like 01 data, only with internal team reference points for permit process. More than likely will duplicate scripts from 01 and modify. TBD>

Dashboard Development

<TBD Needs to be developed, but is highly like 01 dashboard, only with internal team reference points for permits points for permit process. More than likely will duplicate dashboard from 01 and modify. TBD>

Results & Capabilities

<TBD TBD>

Data Updates

<TBD Automatically updated daily. Query/ies will be given to GIS team to configure data feed. TBD>

Risks

<TBD TBD>

03 Community Development Snapshots

In this sub-project, I developed scripts that transformed audited data from Acella, which had been compiled into monthly reports in Excel workbooks, into dashboard-ready datasets. These scripts automated the extraction and standardization of historical permit information from multiple years, creating a unified data repository. The resulting dashboard consolidates long-term data, empowering users to discover new insights about city development patterns.

Users can now visualize and analyze development trends, seasonal fluctuations, and growth patterns that were previously difficult to identify when the data existed only in separate monthly reports without on-demand visualizations and navigation across time.

Requirements Gathering

Before developing solutions for this project, I assessed the existing data, processes, and stakeholder needs to ensure what I was working with aligned with the team's vision for a new dashboard. The main inspiration for this dashboard comes from a consolidation of the monthly, quarterly, and yearly analyses for the City of Missoula's development data on their website. Upon reviewing these reports, I found that the monthly development report was the primary tool used to communicate development data to the public. However, it lacked interactive visualizations and other engaging features. It was hard to see from this report any trends happening without diving in and looking at the numbers. Conducting long-term analysis or generating filterable reports required manually consolidating data from multiple monthly reports.

After becoming familiar with the data, I met with the CPDI team to determine the key functionalities they needed in a dashboard. They wanted to analyze the same development data while seamlessly navigating between different months. Additionally, they were concerned about public complaints regarding the lack of calendar year-to-date consolidations, as only fiscal year-to-date data was provided. The dashboard needed to maintain the same reporting format while enabling the team to more efficiently access and analyze long-term trends.

Data Acquisition and Preparation

The team and I first considered a new set of queries for the data report coming out of Accela to be uploaded to the GIS platform. However, given the scope and timeline for this project, we wanted to ensure that the data pull wasn't an issue. To acquire the data, we used the Fiscal Year Development Report Workbooks from 2021 through 2025. Since there were multiple departmental changes and changes that happened to the monthly report around 2021, we decided that this would be a good range of data to use for the dashboard.

There are three primary components to consider in these workbooks and each component has its own sheet for any given month. The components include:

1. **Data Dumps**: This is monthly building permit data that comes out of Acella and is audited by the team. There are two tables on this sheet, one is the main table, and the other is the set of permit revisions that were made to previously reported permits. Each row represents a single building permit completed that month.
2. **Five Reports**: This is monthly all-type permit data that is pulled from Acella. There are five tables on this sheet, and each contains varying information about all consolidated permits, ranging from permit volumes to revenue collected from permits.
3. **Report**: This is the monthly report as seen on the website. Specific cells reference values from the Data Dumps or the Five Reports, depending on a wide variety of conditions. The top of the report is mainly sourced from the Data Dump while the bottom of the report is mainly sourced from the Five Reports.

It should be noted that the YTD consolidations take the current month's values and then consolidate by summing with the previous month's YTD consolidations, which could lead to many downstream reporting errors.

Given the time I had to understand this data and report this data effectively, I decided that two datasets were necessary to support the dashboard. At a high level, here are the data sets that I created:

1. **BuildingPermitDataAll.csv**: This data set supports visualizations inspired by the top of the monthly Report. It's important to note the data is sourced by directly pulling the permit data from the Data Dumps. In my script, there is a wide range of functions developed to pull the data appropriately, clean it, define the building permit type, and pull it all together across multiple years.
2. **PermitDataAll.csv**: This data set supports visualizations inspired by the bottom of the monthly Report. It's important to note the data is sourced by directly pulling data from the Reports sheet, not the Five Reports sheets. While the permit type categorization rules were relatively clear for building permits, the data was in too raw of form in the Five Reports for me to confidently report financials and quantities for all permits. As such, the data comes directly from the already-compiled monthly Report, which has already been reviewed and released by the CPDI team.

The team should be highly aware that I am using the Data Dumps data for the BuildingPermitDataAll.csv and the Reports data for PermitDataAll.csv. The CPDI team should continue performing their monthly audits and report building as they have for the past few years. When new data is uploaded to the dashboard, they should verify the month's data aligns with what was reported in the monthly report. If specific values are off, the logic

used in the script likely needs to be modified, or the actual rows in the data need to be modified. For example, the script attempts to remove duplicates, but if one is missed, it could be worth deleting the row from the data dump.

It is important to emphasize that the goal of this sub-project is not to alter the existing data processing workflow but to consolidate historical data into an interactive and accessible dashboard. While the process may be more detailed, it ultimately opens greater opportunities for exploring long-term development data.

Dashboard Development

The two data sets developed above are inputs to the dashboard and have been optimized to best support various dashboard functionalities. Dashboard tables mimic the values found in the monthly report. Some visualizations required further consolidations, like finding YTD cumulative sums (for things like quantities, units, valuations, and revenue) for any given month in both fiscal and calendar years. These consolidations are performed in Arcade data expressions and used in multiple serial charts on the dashboard.

Results & Capabilities

The completed dashboard successfully consolidates Missoula's development data into an interactive and user-friendly interface. By integrating data from monthly, quarterly, and yearly reports, the dashboard eliminates the need for manual data aggregation, significantly improving efficiency. Users can now seamlessly navigate between different months, compare trends over time, and filter data based on specific criteria, allowing for more in-depth analyses.

A key improvement is the inclusion of both fiscal and calendar year-to-date consolidations, addressing a major concern raised by the CPDI team. This addition enhances transparency and aligns reporting with public expectations. Additionally, the dashboard introduces several year-over-year visualizations, making it easier to identify patterns and insights immediately.

By preserving parts of the existing reporting format, the dashboard ensures continuity in how data is presented while providing new tools for exploration and decision-making. This solution improves public access to Missoula's development data, making long-term trends easier to explore and understand.

Data Updates

To update the dashboard with the latest data, the CPDI team should continue following their current process for pulling and formatting the data as consistently as possible. The script used to generate the dashboard data is designed to recognize the common formats from previous monthly reports, so any changes to the data format could prevent the script from processing the data correctly.

Once the monthly reports are audited and finalized, the team can update the development workbook in the folder where all reports are stored. Running the script will generate two new CSV files which can then be uploaded to the GIS platform. The datasets should be uploaded as to overwrite the previous table instances. To ensure the update works smoothly, it Is critical to manage the script carefully and avoid making changes to the schema, as this could cause the upload to fail.

Risks

There are a few risks associated with this project, mainly due to the complex nature of the data ingestion and permit type declaration. At the completion of this project, the following aspects of this solution pose as future risks to the deliverability of the dashboard:

- The workbooks must be maintained in the same structure and format as they have been previously. During the auditing process, the team should be cautious about where data is entered and how calculations are performed. If the data output appears incorrect or inconsistent, it may be worth reviewing the workbook for issues that could affect the script's processing. Problems within the workbook—such as misplaced data or calculation errors—could lead to unexpected results when the script runs.
- The data schema must remain identical with every data upload to the GIS platform. Any changes to the schema between uploads will cause the data upload process to fail. This makes it challenging to add new features to the dashboard, as the addition of new fields or functionality could alter the schema. If new features are necessary, they must be incorporated carefully and in a way that does not disrupt the existing structure of the data, ensuring compatibility with the GIS platform.
- Processing does not end with the Python code used to generate the data; sometimes, Arcade data expressions are needed to transform the data for visualizations or metrics. While powerful, these expressions can be challenging to manage for those unfamiliar with the Arcade language. It is important to have a solid understanding of how these expressions work, as errors in these scripts can affect how data is displayed or calculated within the dashboard. Equally important, is assuring that all filters are applied to the appropriate visualizations to display accurate information.

Discussion

Key Results & Capabilities

<TBD when dashboard work is done TBD>

Key Risks

<TBD when dashboard work is done TBD>

Future Development

<TBD when dashboard work is done TBD>

Key Takeaways from the Lead Developer

<TBD when dashboard work is done TBD>

Closing Remarks

Summary of Achievements: Recap the project's primary goals, solutions, and results.

Final Thoughts: Offer concluding remarks that tie together the importance of the project and its contribution.

Acknowledgements: Thank CPDI for working with me and acknowledge professionalism and giving me platform to develop important project.