ATSAC LADOT Arterial Network Traffic Count Visualization

Project Management Plan

prepared by

Berkeley Transportation Systems, Inc.

Author: Eric Shieh
Created: July 8, 2009
Revised: Aug 6, 2009
Version: 1.0

Table of Contents

1. Introduction

1.1. System Goals

The goals of the system are as follows:

- Collect arterial detector and signal data from LADOT and warehouse it
- Associate data with configuration (GIS, network, intersection) information
- Create a web based system that presents the data in a format conducive to analyses

1.2. Project Goals

The goals of the project are as follows:

- Capture the requirements with a high degree of accuracy
- Design a system that meets the requirements
- Implement the system on time
- Discover any remaining flaws
- Launch a system that enjoys wide user adoption

2. Work Task Detail

2.1. Task 1: Develop Methodology

This will be an ongoing task throughout the project. Initial work will be to develop a project management plan (PMP, this document) that identifies key issues and their proposed solutions. The continuous portions of this task will be to regularly keep ATSAC staff appraised of progress and new issues as they arise.

- Staffing: This task will be led by Eric Shieh and staffed by Berkeley Transportation Systems (BTS).
- Hours: The following staff will be assigned to this task:
    - Eric Shieh
    - Karl Petty
- Schedule: This task will run for the duration of the project with the project plan delivered by 9/7/2009 and the functional design document delivered by 10/1/2009.

- Deliverables:
  - Project Management Plan including Key Issues (this document).
  - High Level Functional Specification in the form:
    - Introduction
      - Purpose
      - Definitions, Acronyms, and Abbreviations
      - References and Related Documents
      - Overview
    - Data Feed Integration
      - Network Diagram
    - GIS Configuration
      - Mapping LADOT detectors to GIS segments
    - Data Processing
      - Temporal and Spatial Aggregation
      - Travel Time Computation
      - Detector Health Computation
      - Imputation
    - Web based GIS System
      - Home Page/Launch Point
      - Navigational pages
      - Reporting pages
    - Test Plans
      - Methodology

2.2. Task 2: Data Retrieval

This task entails the collection of detector and signal data from ATSAC's data system and the storage of the data into a data warehouse for analysis.

As part of this task, coordination with ATSAC Information Technology staff will be necessary to purchase a suitable data warehousing machine, integrate it with the existing network, and form a plan for its management. Access procedures to the machine by BTS staff from outside of the LADOT network will need to be planned and agreed upon. After purchase, the machine hardware will need to be housed in a sustainable controlled environment (e.g., power, temperature, and network) and prerequisite data warehousing software (e.g., Oracle) will also need to be installed and configured.

Once the data warehousing machine is set up, proper sources for both the detector and signal information will need to be identified and their formats inspected for processing suitability for Task 3.

Data feed applications will then be written to reside on the data warehousing to ensure the smooth flow into the database.

Monitoring functions will also be written on top of the feed application to alert operators of data disruptions.

This task will be broken into pilot and full deployment subtasks. The pilot will take place for the South Park area which is understood to consist of approximately 1,100 detectors. The experience of the pilot will be used to ensure success during the full deployment.

- Staffing: This task will be led by Eric Shieh and be staffed by BTS.
- Hours: The following staff will be assigned to this task:
    - Eric Shieh
    - Karl Petty
    - David Parsons
- Schedule: This task will begin September 7 with a target of machine installation and data feeds running by January 1. This is largely contingent on purchase scheduling and physical installation.
- Work Steps:
    - Identify data access points for detectors and signals.
    - Gather machine requirements
    - Generate Sizing Plan
    - Generate Security/Access Plan
    - Assemble Machine Specifications
    - Assemble Software Specifications
    - Generate purchasing plan for hardware/software
    - Install hardware/software
    - Test hardware/software
    - Develop and install feed processing
    - Develop and install monitoring
- Deliverables:
    - Data Warehousing Machine running within ATSAC data system
    - Data feed applications for data archiving
    - Feed monitoring applications
    - Purchasing Plan document in the form of:
        - Introduction
            - Purpose
            - Definitions, Acronyms, and Abbreviations
            - References and Related Documents
            - Overview
        - Data sources and required connection points
            - Signal Data
            - Detector Data
        - Sizing of data
            - Detector Data
            - Signal Data
            - Temporal and Spatial Aggregates
        - Software Requirements
            - Hardware Ramifications
            - Security Ramifications
                - Open network ports
        - Security and access requirements
            - Restrictions of the LADOT intranet
            - Required accessibility for setup, deployment, and maintenance by BTS staff
            - Agreed upon access procedures

- Purchasing Recommendations
  - Cost
- Installation Plan
  - Schedule Breakdown
- Test Plan
  - Contingency plan
- Data Warehouse Management and Monitoring Plan
  - Contact list

2.3. Task 3: Data Processing and Analysis

The raw data archived in Task 2 will then be funneled into a series of data processing steps to transform it into a format that is more readily analyzed. BTS has formulated and will adapt several algorithms that will be used at the detector level, including:

- Detector Health
- Imputation
  - Local neighbor
  - Temporal medians
  - Global medians
  - Cluster medians

Additionally, configuration meta data based on GIS sources will be applied to provide analytical contexts such as viewing by entire system, single intersection, or single link. An initial set of detector location data provided by ATSAC will be mapped onto a GIS data set to bootstrap the data processing process. Afterwards, a web based system will be put into place for operators to verify and update the configuration in the event updates are required.

Once this configuration is in place, additional processing will be performed on the detector data to produce geographically summarized results.

- Staffing: This task will be led by Eric Shieh and be staffed by BTS.
- Hours: The following staff will be assigned to this task:
  - Eric Shieh
  - Karl Petty
  - David Parsons
- Schedule: This task will run from approximately October 1 through February 15, 2010, starting with GIS configuration and ending with feed data processing. Task 2 is a prerequisite for the latter step.
- Work Steps:
  - Create processing algorithms to be applied to detector data
  - Create processing algorithms to be applied to detector configuration
  - Create verification methodology for detector configuration
  - Apply detector configuration algorithms
  - Create verification tools

    - Apply verification methodology with help from LADOT staff.
    - Apply data processing algorithms

- Deliverables:

    - Detector Data Processing scripts for Detector Health and Imputation
    - Initial set of detectors configuration processed and mapped using GIS sources.
    - Web based GUI for updating and correcting configuration information.
    - GIS based Data Processing scripts
    - Document for algorithms used in the form of:
        - Introduction
            - Purpose
            - Definitions, Acronyms, and Abbreviations
            - References and Related Documents
            - Overview
        - Detector Data Processing
            - Input Data Format
            - Output Data Format
            - Input/Output mapping
        - Signal Data Processing
            - Input Data Format
            - Output Data Format
            - Input/Output mapping
        - Configuration data Processing
            - Input Data Format
            - Output Data Format
            - Input/Output mapping
    - User guide document for LADOT staff detailing the verification and update process:
        - Introduction
            - Purpose
            - Definitions, Acronyms, and Abbreviations
            - References and Related Documents
            - Overview
        - Verifying Configuration Integrity
            - Checking flagged detector configuration
                - Background on confidence heuristic
            - Checking detectors by intersection
        - Updating Configuration
            - Locating existing detectors
            - Update process
        - Entering new configuration
            - Prerequisite information
            - Creation Process
            - Validation process

2.4. Task 4: Speed Calculation and Calibration

Following Task 3 and portions of Task 5, ATSAC's data will be analyzed by BTS staff and recommendations for corridor management of LADOT's arterial network will be provided.

- Staffing: This task will be led by Karl Petty
- Hours: The following staff will be assigned to this task:
    - Karl Petty
- Schedule: This task will run from approximately March 15 through April 15, with Task 3 and portions of Task 5 as prerequisites for completion.
- Deliverables:
    - Report on recommendations for arterial corridor management.

2.5. Task 5: Develop a GIS Based Traffic Volume Display

To help visualize the results of data processing, BTS's Arterial Performance Measurement System (APeMS) will be adapted to provide a web server based application that allows for simple navigation through various processing levels (spatial and temporal) and types.

Types of reporting will include:

- Flows for intersections, links, and turning movements
- Intersection bubble diagram display of peak hour traffic volumes
- Intersection bubble diagram of 24 hour volumes
- Detector health
- Signal timings per intersection
- HCM based travel times per link
- Real Time speed map of the LADOT network

- Staffing: This task will be led by Eric Shieh and be staffed by BTS.
- Hours: The following staff will be assigned to this task:
    - Eric Shieh
    - Tim Wang
- Schedule: This task will run from approximately February 15 through May 15. Portions will overlap with Task 3 because visualization aids in the refinement of data processing tasks.
- Work Steps:
    - Finalize reporting and visualization types
    - Implement wireframes/samples
    - Refine design.
    - Implement finalized interface and reports.
- Deliverables:
    - Web based application deployed in LADOT
    - Map based navigation
    - Reports as described above

2.6. Task 6: Users Manual

This documentation task will be fulfilled as part of the web system and in hard copy format. On the web system, documentation will take the form of context sensitive help. An indexed hard copy form will also be provided that contains similar information.

- Staffing: This task will be led by Eric Shieh and be staffed by BTS.
- Hours: The following staff will be assigned to this task:
    - Eric Shieh
    - Karl Petty
- Schedule: This task will run from approximately May 15 through June 1, with Task 5 as a prerequisite for completion.
- Deliverables:
    - Web base context sensitive help
    - Hard copy user guide in the form:
        - Introduction
            - Purpose
            - Definitions, Acronyms, Abbreviations
            - References and Related Documents
            - Overview
        - Navigation
            - System
            - Intersections
            - Links
        - Report Types
            - Flow
            - Signal Timing
            - Detector Health
            - Travel Time
        - Algorithm Descriptions
            - Imputation
            - Detector Health
            - Travel Time

2.7. Task 7: Final Report

Upon completion of all preceding tasks, a final report will be created, detailing maintenance plan of the data feed collection, the algorithms used in processing.

- Staffing: This task will be led by Eric Shieh and be staffed by BTS.
- Hours: The following staff will be assigned to this task:
    - Eric Shieh
    - Karl Petty

- Schedule: This task will run from approximately May 15 through June 1 with all tasks as a prerequisite for completion.
- Deliverables:
    - Final Development Methodology from Task 1
    - Data Feed Maintenance Plan from Task 2
    - Algorithms used in data processing from Task 3
    - Overall summary of results and recommendations

3. Key Issues

This project faces several hurdles that must be overcome for its successful completion that are both technical and administrative.

- Installation of a machine and full integration into LADOT.
    - There are many requirements imposed by Information Technology (IT) departments that must be met and approved of in order to proceed to the next step. Procedures such as documentation and sign off must be ascertained and followed.
    - Proper sizing and specification of hardware
        - Preliminary requirements
            - 10TB for 17,000+ detectors, 30 second samples for 5 years
            - Signal feed is currently unknown but will need to be taken into consideration
            - Run data feeds
            - Run data processing
                - Detector health
                - Temporal and spatial aggregates
                - Travel time
                - Map tiling
            - Run monitoring
            - Integrated Lights Out Manager (ILOM) or similar console management system
    - Approval of machine hardware
    - Establishing of machine access and security protocols suitable for development and maintenance work
    - Purchase of the machine
    - Approval of machine software including operating system, RDBMS, Web server.
    - Purchase of software licenses
    - Approval of network and security of machine
    - Approval of BTS Staff access to the machine which may require approval of access into the LADOT intranet.
    - Installation of the machine
    - Testing and verification with data sources.
    - There are over 17,000 detectors in the network and the machine will need to be able to process these in real time.
- Pulling of data from the ATSAC MySQL database

  - Once the machine is installed, connected to the network, and access to staff is provided, it is critical for the data feeds to be activated in order for further work to progress. This data will be a prerequisite in test-analyze-refine cycles within tasks 3-5.
  - There are also operational issues that will need to be verified such as impact on the MySQL database

- Modeling of the LADOT arterial network. This network model will form the underlying geographic basis for all other components in the system.
  - An algorithm will be needed to prune the GIS network data into a traffic network model.
  - There are at least 2,500 intersections that will need to be correctly imported and verified.

- Mapping the detectors to the network model
  - There are over 17,000 detectors and over 2,500 intersections that need to be reconciled in the model
  - An algorithm will be needed to take LADOT's detector configuration and map it to the network model.
  - A heuristic will be needed to determine confidence of the matching to allow operators to identify potential errors
  - Tools will need to be developed to update and verify the detector configuration model

- Ascertaining source and format of signal timing data
  - Signal timing is an input used in many APeMS algorithms
  - The source of this data needs to be identified.
  - Once found, a data path to the warehousing machine needs to be established.
  - The format of this data needs to be identified.
  - Feed applications need to be written to integrate the data with APeMS

4. Project Management Approach

4.1. Timeline

Following is a high-level timeline showing the phases of the project and key deliverables during each phase. Note that work does not shut down while the project waits for feedback from the project sponsor.

Activity	Start	Finish
Project Plan *This document represents the project plan.*	8/1/2009	9/7/209
Deliver Project Plan	8/15/2009	
Feedback	8/15/2009	9/7/2009
Status Meeting held by phone	9/2/2009	
Revising Project Plan	9/2/2009	9/7/2009
Functional Specification *The ultimate deliverable from this phase includes: Use Cases, Data Flow Diagram, Data Structure Catalog, and Constraints, and implementation guide.*	9/7/2009	10/1/2009
Deliver First Draft	9/15/2009	
Feedback	9/15/2009	9/22/2009
Status Meeting held in person	9/24/2009	
Revising Specification	9/25/2009	10/1/2009
Implementation *During implementation, engineers are coding. We plan to write test code first and then begin coding modules. During this phase, we plan to hold regular reviews of functionality with the Project Sponsor.*		
Data Feeds *Planning, approval, installation of data warehouse machine and integration of data feeds*	9/7/2009	1/1/2010
Configuration *GIS and detector mapping*	10/1/2009	2/15/2009
Data Processing *Detector health, imputation, travel times, map tile generation*	1/1/2010	3/15/2010
GUI and adjustments *Reports, GIS based navigation, miscellaneous adjustments to the data processing*	2/15/2010	4/15/2010
Code Freeze and bug fixing *Restrict feature changes and only allow bug fixes*	4/15/2010	5/15/2010

Testing and Feedback *Core features should be implemented and further adjustments will be made with each successive iteration.*		
1st Iteration *South Park, ~1100 detectors. Partial GUI only.*	3/15/2010	4/15/2010
2nd	4/15/2010	5/1/2010
3rd	5/1/2010	5/15/2010
Final Review and Acceptance *This final phase is meant to catch any remaining issues that could appear late. Final reports and Users Manual.*	5/15/2010	6/1/2010

4.2. Deadlines

The project contains the following major milestones. We will strive to beat these deadlines if possible.

- Project plan delivered: Sept 7, 2009
- Requirements Specification delivered: Oct 1, 2009
- Data Feeds live: January 1,2010
- Web GUI for adjusting configuration: February 15, 2010
- Pilot Rollout in South Park (partial GUI): March 15, 2010
- Implementation completed: May 15,2010
- Final Deliverable: June 1, 2010

4.3. Roles and Responsibilities

Role	Description	Who
Project Sponsor	• Represents the goals of the system • Provides a single point of contact to the account manager • Arranges meetings • Provides information used to write the formal requirements • Reviews work and provides feedback • Accepts work	TBD
Account Manager	• Provides a single point of contact to the project sponsor • Offers strategic counsel • Reviews deliverables prior to handoff	Karl Petty karl@bt-systems.com
Project Manager	• Manages scope, schedule and budget • Assigns tasks to internal team members • Meets regularly with the project sponsor to communicate project status • Gathers feedback and bug reports • Writes the Requirements Specification	Eric Shieh eric@bt-systems.com
Software Architect	• Analyzes requirements • Writes the Design Specification	Eric Shieh
Software Engineer	• Creates tests according to the test plan • Codes modules according to the functional design • Validates code based on test results • Fixes bugs	Tim Wang David Parsons

4.4. Communication Protocol

Key milestones in the project require feedback and acceptance from the project sponsor. Feedback will be returned to the account manager 10 business days after the milestone is delivered. As a result of feedback, we expect minor revisions to be turned around in two days and a final version delivered by the account manager to the project sponsor. In the event of what we consider major changes based on feedback, we will revise the schedule to accommodate the changes if necessary. We may also revise the schedule if feedback takes longer than 10 days to complete.

While the project sponsor and account manager are the primary contact points, we recognize a need for more flexibility. The project manager can be available to communicate project status on an ad hoc basis. We expect some interaction between the development team and experts on the LADOT side. We will appreciate coordination support from the project sponsor.

Documents generated during development will be provided electronically via email or HTTP links. Feedback returned electronically will be appreciated. A simple OK via email will do for acceptance.

Status meetings will sometimes be in person, but will trend towards being on the phone with the aid of WebEx as the project progresses. Meetings will be tactical in nature with a scope of approximately one month. Meeting agendas will follow the following parts, in order.

Review of major upcoming activities

1. Statement of current status, including objective metrics if available
2. Detailed discussion of identified topics
3. Identify strategic decisions that must be made, if any
4. Restate decisions, action items and key messages agreed on during the meeting

Within 24 hours of the meeting, written notes will be shared with the team.

4.5. Scope Management

Scope change requests come from two sources:

- First, through the process of implementing a specific feature we discover that the results are not what the designers expected. As a result, some additional work needs to be done to investigate the discrepancy and the resulting solution.
- Second, we are in constant contact with project sponsor. From time-to-time they recommend additions, or changes, to the user interface that they think would be helpful.

The methods for dealing with these changes are usually quite different. We evaluate the change requests in two different ways, depending on the nature of the change request:

- If the change request involves algorithmic issues then the change is primarily evaluated on whether or not the resulting solution is accurate and technically feasible.
- If the change request is for a UI feature, the evaluation is focused on usability and effort required to implement it.

In all cases, the change request and the proposed plan is presented to the project sponsor in written form. The change request will include a description of the changes as well as an description of the effect on the schedule.

4.6. Schedule Management

The project plan includes a high-level schedule. A detailed project schedule is kept by the project manager. Adjustments to the detailed schedule likely will occur on a weekly basis. Any adjustment made to the high-level schedule will be communicated to the entire team.

4.7. Cost Management

The cost of the project will be kept at or below the fees described in the contract. BTS has a long history of delivering projects within budget.

4.8. Resource Management

Management of resources at BTS is a shared responsibility of project managers and the VP of Engineering, Leon Atkinson. Staffing levels are reviewed regularly to ensure adequate resources are available to all project. It is expected that all of the tasks of this project will be completed with the available staff.

5. Assumptions

The following are assumptions we made while creating the project plan.

5.1. Assumptions of Scope

- The scope will expand somewhat from that described in the RFP
- The scope will not expand significantly after requirements are accepted.
- Some issues of visual design and usability will be differed until implementation.

5.2. Assumptions of Resources

- BTS will continue to retain talent exceeding the needs of this project.
- We will have full access to servers, networks and data when implementation starts.
- The project sponsor will provide detailed information on a regular basis about defects found during the pilot phase.

5.3. Assumptions of Schedule

- If any phase completes sooner than scheduled, we can adjust following activities to start earlier.
- Feedback from the project sponsor will be returned within the 10-day limit.
- Changes based on feedback will not take longer than two days to address.

6. Development Methodology

Following is a description of how the engineering team develops software.

6.1. Software Development Tools

Developers use a variety of tools according to their personal coding style. We do not have a uniform toolset for editing source code and publishing it to the server. A combination of perl, php, HTML, and javascript and other technologies as deemed appropriate by the software architect will be used to fulfill the requirements of this project.

We utilize the open source package CVS, concurrent versioning system, as our source code control system. We use Bonsai, another open source package, as a web-based CVS query tool to investigate changes.

6.2. Defect Management

We use the defect tracking tool Bugzilla. Bugzilla is a bug tracking system, server software that manages the entire bug tracking process for a software development organization. It allows us to keep track of all of the bugs in the software system efficiently. Bugzilla is a free, open-source software platform from the Mozilla organization.

We anticipate bugs reported by users during the pilot to arrive via email from the project sponsor. These will be entered into Bugzilla by the account manager.

6.3. Quality Assurance

We rely on two tools: Bugzilla, which was described above, and Tinderbox. Tinderbox is an automated software build and test framework. It is comprised of five different components:

1. The software system under development.
2. A number of test cases that need to be applied to the product.
3. A central server that is the repository for the software product. This central server coordinates all of the actions. The central server holds the mapping of which client should do which tests.
4. A number of client machines. These participate in the Tinderbox framework by applying the

tests.

5. A results web page where the results of the tests are automatically posted.

The tinderbox framework is completely automated and it works in a continuous cycle. A typical cycle works as follows: a client machine checks the software source code out of the software repository and builds the software. It runs the tests which have been assigned to this client. The results of the tests are sent back to the central server and displayed on a web site. This cycle is then repeated over and over again. Each client machine that participates in the build and test process is typically given a different set of tests to execute. The results web page gives a graphical view of which tests have passed for all of the client machines.

The developers utilize the Tinderbox framework in two ways. First, they write test cases to cover the functionality that they are including in the product. Once the test case has been written and assigned to a machine it is automatically picked up by the client during the next cycle and applied to the code. In this manner Tinderbox is a test execution platform. Second, when a developer is attempting to modify code that already has tests defined for it then the Tinderbox framework acts as a tripwire for any inadvertent bugs. If the developer changes the code such that the test no longer passes then the status web page indicates that a test has started failing.