Caltrain CTAMS Final Report

Caltrain Transit Asset Management System Final Report

MARIN CO.
SOLANO CO.
CONTRA COSTA CO.
ALAMEDA CO.
SAN MATEO CO.
SANTA CLARA CO.
SANTA CRUZ CO.
CITY AND COUNTY OF SAN FRANCISCO
SAN RAFAEL
CORTE MADERA
SAUSALITO
HERCULES
PINOLE
MARTINEZ
PITTSBURG
ANTIOCH
CONCORD
SAN PABLO
RICHMOND
EL CERRITO
PLEASANT HILL
WALNUT CREEK
ALBANY
BERKELEY
LAFAYETTE
OAKLAND
SAN LEANDRO
OAK
HAYWARD
PLEASANTON
LIVERMORE
UNION CITY
NEWARK
FREMONT
MILPITAS
SAN JOSE
CUPERTINO
CAMPBELL
SARATOGA
DALY CITY
BRISBANE
COLMA
SOUTH SAN FRANCISCO
PACIFICA
SAN BRUNO
MILLBRAE
BURLINGAME
HILLSBOROUGH
SAN MATEO
BELMONT
SAN CARLOS
REDWOOD CITY
HALF MOON BAY
ATHERTON
MENLO PARK
PALO ALTO
WOODSIDE
PORTOLA VALLEY
LOS ALTOS HILLS
LOS ALTOS
MOUNTAIN VIEW
SUNNYVALE
4th & King
22nd Street
Bayshore
South San Francisco
San Bruno
SFO
Millbrae
Broadway
Burlingame
San Mateo
Hayward Park
Hillsdale
Belmont
San Carlos
Redwood City
Atherton
Menlo Park
Palo Alto
Stanford Stadium
California Avenue
San Antonio
Mountain View
Sunnyvale
Lawrence
Santa Clara
College Park
San Jose Diridon
Tamien
Capitol
Blossom Hill
Morgan Hill
San Martin
Gilroy
N

Revision History

Revision	Date	Revision Description	Author
Rev. 1	12/23/2013	Initial draft	William Wong
Rev. 2	12/28/2013	Added feature changes, schedule, development process, and lessons learned. Revised Introduction.	William Wong
Rev. 3	1/3/2014	Final draft	Zhenlin Guan
Rev. 4	3/3/2014	Revised per comments	Zhenlin Guan

TABLE OF CONTENTS

Page #

1 EXECUTIVE SUMMARY

The Federal Transit Administration (FTA) State of Good Repair (SOGR) pilot grant program gave transit agencies an opportunity to explore and address challenges faced by transit agencies in asset management and SOGR.

In the past, Caltrain used a Microsoft Excel based system to manage its assets such as track, stations, structure, and etc., utilizing the information from the database to plan maintenance and capital programs in order to keep its system in a state of good repair. Over the past 18 months, under the scope of the FTA grant, Caltrain and Rail Surveyors and Engineers (RSE) developed a web-based, flexible, and relational database to grow the functions and improve the efficiency of the existing system. The new SOGR database, Caltrain Asset Management System (CTAMS) is used to determine an asset's state of good repair with the following three major functions:

- Manage asset information
- Rate asset conditions
- Track work orders

This final report will describe the original goals and scope as proposed by Caltrain and RSE under the scope of the grant program, the final product produced the development process and approach, and some valuable lesson learned throughout the implementation of the project.

2 BACKGROUND

2.1 About Caltrain

Caltrain is the oldest commuter rail operation in the San Francisco Bay Area and the only commuter rail service operating on the San Francisco Peninsula. Passenger rail service on the San Francisco Peninsula began in 1863 under the authority of the San Francisco and San Jose Railroad Company, which in 1870, was acquired by the firm that ultimately became the Southern Pacific Railway (SP). The SP operated passenger service on the San Francisco Peninsula through 1980. At that point, the State of California stepped in to continue the operation of the passenger service, which became known as Caltrain. The railroad was ultimately turned over to the Peninsula Corridor Joint Powers Board (JPB) in 1992.

When the JPB began operations of Caltrain in 1992, the railroad had already experienced many decades of deferred maintenance and much of the rail line's infrastructure dated back to the early 1900's. Bringing the railroad into a state of good repair was not only essential to ensuring safe and efficient operation, but was also necessary to increase customer satisfaction and boost ridership. Since that time, the JPB has invested hundreds of millions of dollars in the railroad to bring it into a state of good repair. While much work has been done, the latest 10 year Capital Improvement Program (CIP) for Caltrain identifies a need for over $1 billion worth of projects for its SGR program; yet projected revenue for this work fall far short of the need.

2.2 The Original State of Good Repair Database

Caltrain, with support from RSE, developed the original State of Good Repair Database to make the most effective decisions to prioritize replacement and rehabilitation needs within existing budget constraints. The SOGR database was first established in 2006 as an Excel-based system that tracks the condition of capital assets, identifies trends affecting system assets over time, and helps to prioritize and coordinate replacement and rehabilitation needs for future capital spending. The database included the following modules:

- Rails-Ties-Track Geometry
- Structures and crossings
- Stations
- Signals

The database incorporates factors such as age, FRA standard requirements and Caltrain's SOGR standard criteria to account for asset condition. It also contains links to related documents such as track charts with maps, videos, pictures and aerial photographs, as-built drawings and maintenance inspection reports.

2.3 The Need to Update the SOGR Database

Caltrain's existing SOGR database was established to provide Caltrain with a tool to track the conditions of capital assets, identify trends and problem areas in the system, and also helps to prioritize and coordinate rehabilitation and replacement of assets. The database provided essential information that helps Caltrain to develop the scope of budget of its annual State of Good Repair program, which keeps the railroad in great condition.

While Caltrain's Excel-based SOGR database is an effective asset management tool, it has limitations and short-comings. The Excel-based system lacked the capability for multi-user access; had limitations on the amount of data stored; and most importantly, was inefficient, required a high amount of man hours to upkeep and process data collected from the field.

Caltrain and RSE saw the need to expand and enhance the functionality and capabilities of database beyond Excel to fulfill the growing need of a comprehensive and efficient SOGR database. With the grant opportunity, Caltrain and RSE envisioned a CTAMS database that improves and expands the current system to achieve the agency's SOGR goals and make the information user-friendly for employees throughout the organization.

3 PROJECT IMPLEMENTATION

3.1 Project Implementation Plan Summary

3.1.1 Scope

Over the period of 18 months, Caltrain and RSE worked to develop CTAMS to not only retain and enhance the functions of Caltrain's previous SOGR database, but also to grow new functions and features for a complete product that can be easily applied to others in the transit industry.

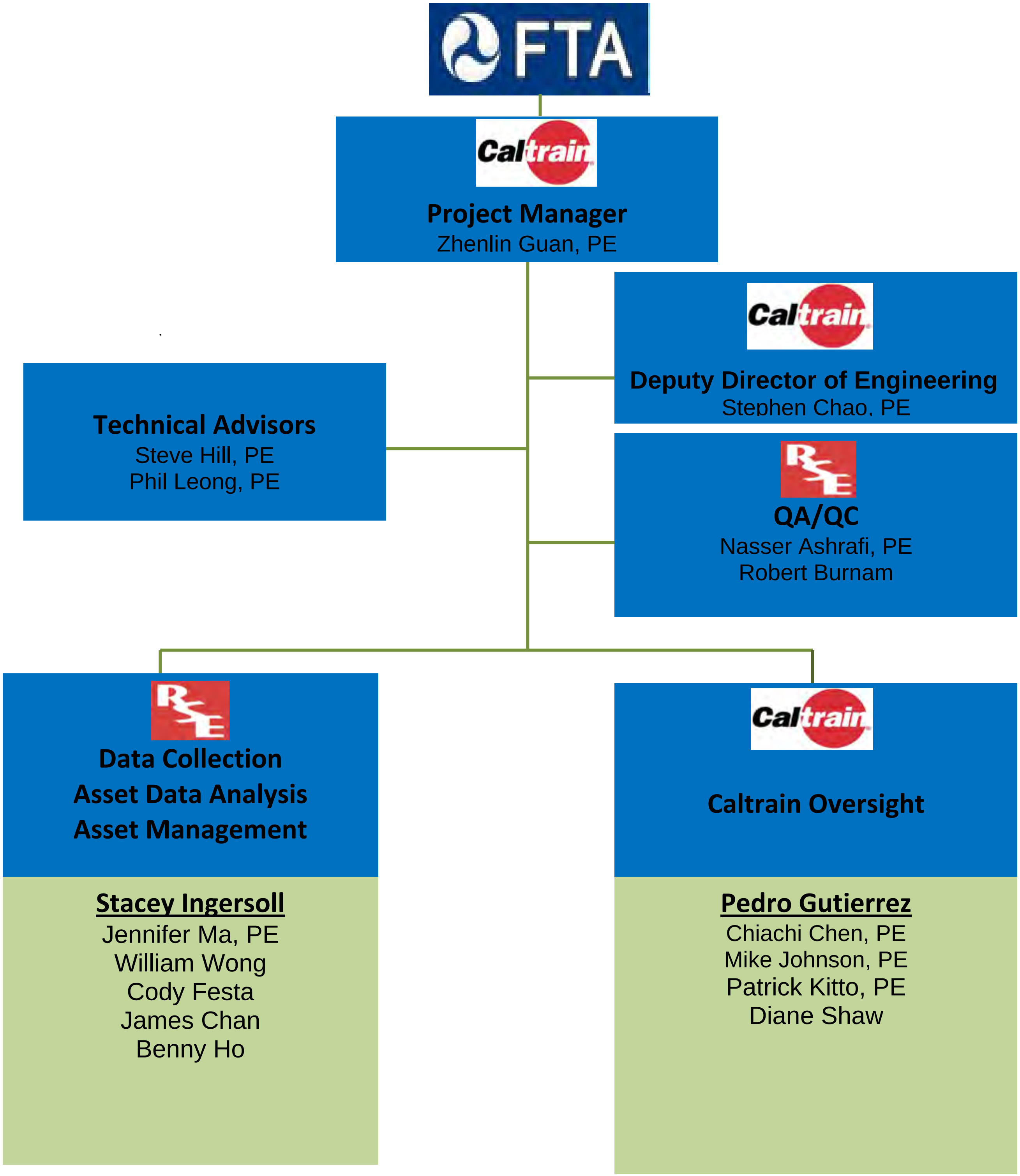
The final product includes modules for the following assets:

- Track
- Structures
- Grade Crossings
- Stations
- Signals
- Communication
- Right of Way
- Equipment

Each of the modules above was developed with the following goals in mind:

- Enhance fiscal responsibilities:
    - A fix it before it's broke principle
    - Maintain facilities at high levels of service
    - Lower the cost to maintain over time
    - Provide the Finance Department with access to information for accounting purposes
- Maintain cost efficiency in development of capital programs
- Maintain proper levels of inventory and maximize surplus
- Prioritize maintenance and capital work
- Meet safety goals and performance measures

3.1.2 Staffing and Organization



FTA

Project Manager
Zhenlin Guan, PE

Deputy Director of Engineering
Stephen Chao, PE

Technical Advisors
Steve Hill, PE
Phil Leong, PE

QA/QC
Nasser Ashrafi, PE
Robert Burnam

Data Collection
Asset Data Analysis
Asset Management

Stacey Ingersoll
Jennifer Ma, PE
William Wong
Cody Festa
James Chan
Benny Ho

Caltrain Oversight

Pedro Gutierrez
Chiachi Chen, PE
Mike Johnson, PE
Patrick Kitto, PE
Diane Shaw

3.1.3 Schedule

Project Date	Milestones
Dec 2011	Proposal
Feb 2012	Project Implementation Plan approved
Mar 2012	Development begin
Aug 2013	Development completion

3.1.4 Budget

Description	Budgeted	Spent
Database Development	$508,959.00	$508,959.00
Project Oversight	$200,999.00	$181,427.00
Project Administration Staff	$15,000.00	$11,441.00
Project Contingency	$25,042.00	$25,042.00
Total	$750,000.00	$701,826.00

*Project budget is as of end of November 2013

3.1.5 Implementation Methodologies

Due to the software development nature of the project, Caltrain and RSE adapted a number of software development practices and processes. In particular, a modified version of the agile development process – exploratory and iterative development process, was used to develop CTAMS.

A core development team determined the basic functions and requirements of CTAMS by gathering the needs of the specific end user groups. The core development team consisted of a project manager, an IT database expert, a database developer, engineering discipline managers for track, signals, structures, etc., and an engineering manager. The needs of the end user groups were discussed and prioritized by the core development team based on the following factors:

1. The safety and operation of Caltrain
2. Rules and regulations (FRA, CPUC, etc.)
3. Essential information for development of maintenance and capital programs
4. Is the function a must have or an add on?

After the needs have been prioritized, they were developed into functions and requirements by the core team.

Each function of the product started with an exploratory process, in which new ideas, functions and additional requirements were discovered by the project team through meetings and discussions as the project progressed.

With the ideas and requirements in place, the database was built through the iterative process. Each component of the final product went through the following steps:

1. Discuss requirements
2. Define features and functions

3. Design
4. Develop
5. Review and test
6. Repeat

These steps were repeated for each component until the project team was satisfied with the end results of the individual component. The iterative approach allowed us to implement and refine each part of the product effectively, not leaving any issues unresolved until the end of the project.

The final step of the implementation of CTAMS is the transition from the existing to the new system. For the transition of CTAMS, there were multiple steps taken to make the transition as smooth as possible:

1. Final testing of the new database – the new database resided on a test server through the development and testing phases. This is ensures that the final product that goes onto the production server is one that is complete and tested
2. Testing of the production server – to ensure the final production server is ready to receive the new database package
3. Migration of the database – the actual move of the new database from the test to the production server with the test data that has been used over the course of development and testing
4. Testing of the new database on the new production server – another round of testing to ensure the move of the database proceeded as planned
5. Burn in period – this is the period where both the existing and new system will be in operation until all existing data has been copied to the new database. The existing database will serve as the backup until that process is completed
6. Training – to ensure the staff is ready to use the new system
7. Data migration – the final move of history data from the existing to the new database

3.2 Changes

3.2.1 Feature Changes and Outcomes

From the outset of the project, there was an understanding that we could not have anticipated all the features and function needed for the end product. Although basic functions and requirements were established early on through discussions with the end users, we adopted the exploratory and iterative development process to add and refine the needed functions and possible capabilities throughout the development process.

Many of the additional functions added throughout the design process were aimed at making the database more efficient and accurate, which was one of the original goals of the project. For example, the automatic track evaluation feature not only allows inspection data to be evaluated faster but also more accurate with custom rule sets that takes the subjectivity out of the evaluation process.

However, other features, such as the work order integration and audit logs are essential functions to the State of Good Repair program. These features were originally not identified in the PIP but were later developed during design based on needs of the various end users. A work order integration function is essential to tracking maintenance work and helps to identify trends in assets as well as capture the maintenance budget spent on a certain asset. Audit logs provide the administrators with a tool to track changes and maintain the integrity of the data.

Below are the original goals as described in the Project Implementation Plan and the additional features incorporated through the exploratory and iterative development process.

Original Project Goals and Status:

Original Requirements	Implementation Status
Web-Based	Interactive online multi-user web application
Asset Information	Asset data management, with customizable types and attributes
Asset Digital Media	Flexible attachment system for any kind of digital media
SOGR Rating	Adaptation of Caltrain's State of Good Repair rating system
Search	Keyword-based and attribute-based search on asset and media content
GIS Map Integration	Google Earth integration for assets and tracks
Asset Inspection	Manual inspection form
Universal Data Format	Import and export using open formats
Access Control	Access and authorization control with users and roles
Security	Password, encryption, centralized server isolation, security hardening
Cost and Budgeting	*Not implemented – lack of concrete requirements*

Additional Feature Set:

Additional Requirements	Implementation Status
Organizing Assets	Flexible hierarchical folder-based asset organization
Deep GIS Integration	Extensive map-based asset introspection and navigation in Google Earth
Automatic Track Evaluation	Rule-based automatic track condition evaluation
Fine Grain Rating	Per foot track evaluation and rating
Rating Aggregation	Rating rollup from sub-asset attributes to asset group
Chart and Graph	Charts/graphs for different aspects and dimensions
Inspect Form Builder	Custom design and build any inspection forms
Rating GIS Integration	KML for track segment rating and asset rating
Milepost GIS Mapping	Map milepost to GIS for automatic lookup
Drag & Drop Upload	Batch attachment upload with drag and drop
Attachment Category	Labeling system to categorize attachments
Audit Log	Detail log for tracking operations done in system
Scope Based Logging	Asset scope or folder scope change logs
Repair Tracking	Work order to track repair of asset defects
Work Order Integration	Custom template system for exporting work orders
Real Time Search Indexing	Index new changes in real time for up to dated search

Media Content Search	Attachment document content are searchable
Custom Asset Type	New asset type can be defined easily
Rich Data Type	Rich set of data type for asset type attributes
Smart Data Validation	Asset data input are validated based on data type
Asset Photo Gallery	Attachments are built into photo gallery automatically
Reference Materials	Asset reference materials are built in every folder
URL Based Attachment	Support URL based attachment to web references
Customizable Home Page	Customizable home page with different folders
Customizable Folder	Customizable folder appearance

3.3 Product Feature Summary

During the course of implementation, many ideas are explored, discussed, and tested. As shown in the previous section, CTAMS incorporated a wide range of features and functions. The highlights, or core functions, are listed below:

3.3.1 Asset Management

- Centralized information – web-based application allows access to the asset data anywhere and anytime.
- Hierarchical asset organization – nested folders to organize different types of asset.
- Asset folder – contains assets of same type, common reference materials and documents.
- Asset page – maintains individual asset information, digital media (photos and videos), conditions, and etc.
- Asset digital media – asset document management, with easy to use drag and drop upload, tag labeling system for document organization, and automatic generation of photo gallery.
- GIS integration – integrated with Google Earth for individual asset and assets in folder, to provide deep map-based asset inspection and navigation.
- Search and Index – keyword search on assets and asset document content.

3.3.2 Automatic Track Condition Assessment

- Evaluation of track condition – custom built rule-based automatic evaluation of track condition on different attributes of the track. Rating can be broken down to per-foot granularity and uses a 1-5 color-coded rating system with 1 = Excellent, 2 = Good, 3 = Fair, 4 = Improvement Needed, 5 = Immediate Attention Required
- Attachments and notes – document attachments and notes for each track segment.
- Graph and charts – graphs for track values with rating colors, pie charts for rating distribution.
- GIS Integration – map based integration of track condition ratings in Google Earth.
- Custom rule definition – user-defined rule set for evaluation

3.3.3 Asset Inspection and Assessment

- Electronic inspection forms – electronic forms for rating asset conditions.
- Asset attributes – evaluation of different aspects of an asset.
- Attachments and notes – digital documents and detail notes added for inspection.
- Condition aggregation – automatically roll up of rating from attributes to sub-categories and overall asset rating. Roll up takes account of worst attribute condition as a conservative fix it before it's broke approach.
- Inspection workflow – workflow for the inspection, review, and approval process.
- Customizable inspection form – easy to create inspection form.

3.3.4 Repair Management

- Work order – work order to track repairs.
- Digital media – documents can be attached to the work order.
- Customizable work orders – customizable fields for user defined work order form.
- Integration with Excel - intelligent customizable Excel form for export and external integration.

3.3.5 System Feature

- User Authentication – user name and password for secured access to the database.
- User Authorization – user roles to grant and restrict access of different user groups.
- Change Log – changes to assets are logged for audit purpose.

4 PRODUCT FEATURE INFORMATION

4.1 Overview

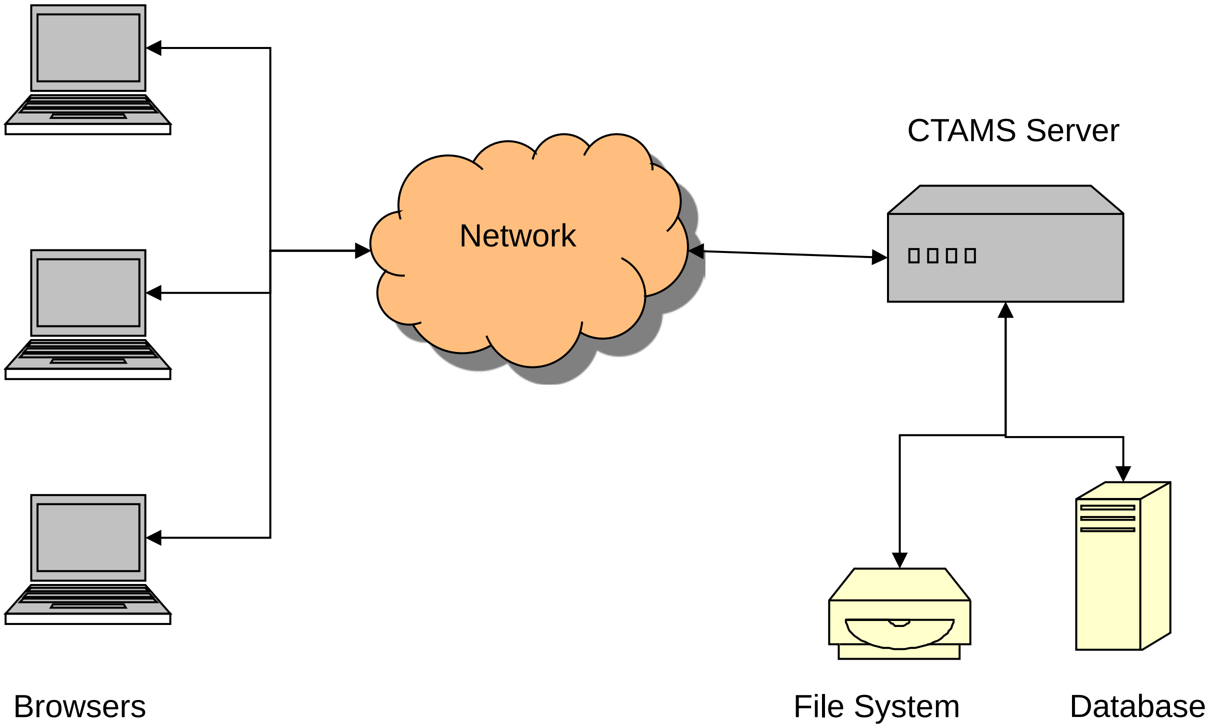
CTAMS is a web-based application for managing transit assets, performing periodic asset assessment, and managing repairs:

- As a web application, it can be accessed through any of modern web browser at any location.
- For asset management, assets can be organized in hierarchical structure with folders. Custom asset types can be defined easily.
- Assets are indexed automatically for quick search.
- Asset documents and digital media (pictures, drawing, docs) are stored along with the asset as attachments.
- Document contents are indexed automatically for easy search.
- CTAMS integrates with KML-aware GIS applications (e.g. Google Earth) to display assets in the map form.

Inspections are done periodically on the assets based on current Caltrain inspection practice to evaluate their condition and health. CTAMS can perform automatic assessment on rail tracks by running through the collected geometry car data to evaluate various aspects of the track according to rule sets. For assets that require manual inspection, CTAMS provides manual assessment via the electronic inspection form. Custom inspection form can be defined easily.

4.2 System Overview

4.2.1 Architectural Diagram



4.2.2 System Components

The CTAMS system consists of four major components:

- The frontend web user interface on the web browser
- The CTAMS application server
- Database to store asset and assessment related data
- File storage to store digital media.

From the user's perspective, all the functionalities are available over the web and can be accessed via the browser.

4.2.3 Open Source Libraries and Components

CTAMS was built with open source libraries and standard off the shelf components as much as possible to reduce software licensing cost and to reduce 3rd party dependency. For example, during technology selection the use of Oracle database or Microsoft SQL Server came up as the choice for the database component per IT recommendation. However, after much of the data usage analysis, the low data volume didn't warrant the use of a heavy weight database. An open source embedded relational database was used instead, which has the additional benefits of simplified setup and simplified IT maintenance.

4.3 Feature Tour

During the course of implementation many ideas and technical capabilities are explored and discussed. The project team ended up with implementing an extensive feature set, highlighted below.

4.3.1 Asset Management

4.3.1.1 Centralized Information

The asset information is stored in a central place accessible online by multiple users. Information updates are centrally and changes can be viewed by the others right the way. E.g. Asset field inspections filed online are available right the way to people in the office.

4.3.1.2 Web Based

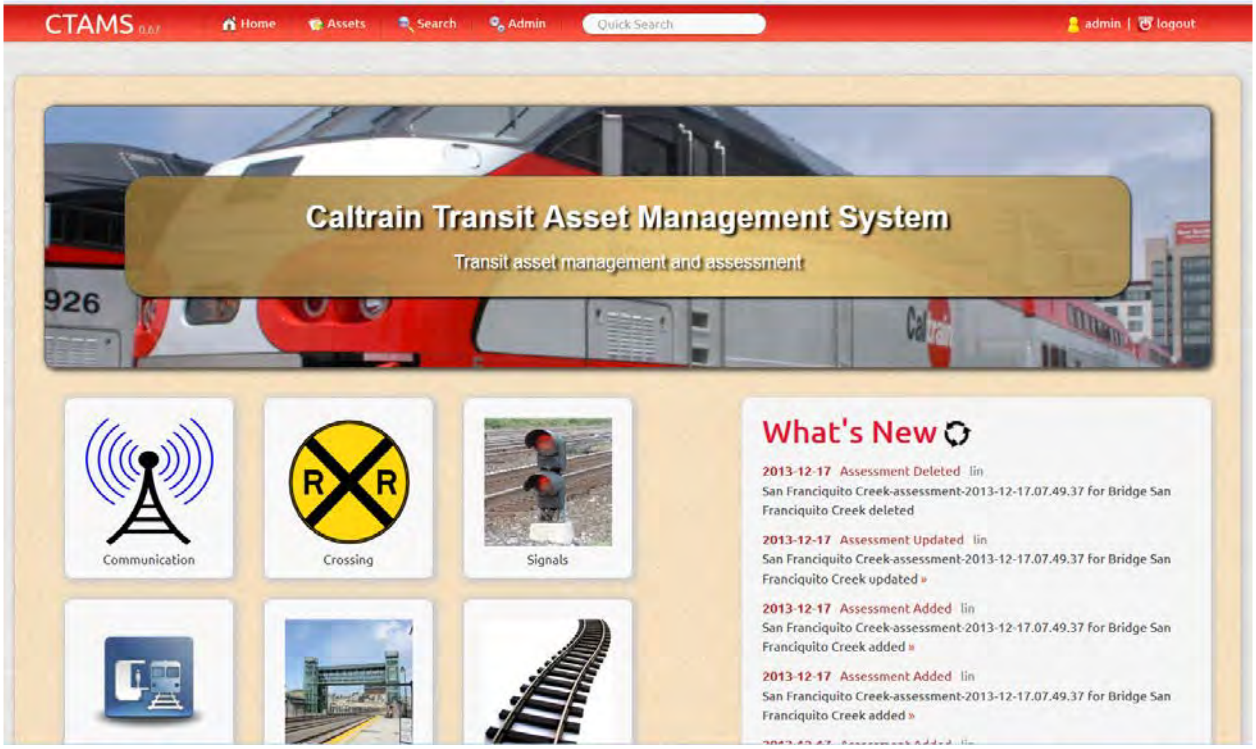
The web-based application let multiple users access the asset data anywhere and anytime using common web browsers. Most modern web browsers are supported – Firefox, Chrome, Safari, and IE 10 and up. IE 8 and 9 are supported with degraded experience. Tablets are supported for mobile users with web connection.

4.3.1.3 Hierarchical Asset Organization

Assets are organized in a hierarchical folder structure, with support of arbitrary level of nested asset folders. Real life assets are modeled as the Asset items in CTAMS and folders are logical structure to organize them.

4.3.1.4 Asset Home Page

The Home Page consists of the frequently accessed asset folders, the major navigation links with the quick search tool, and the What’s New section. It is the first page a user sees after login. Any folder can be configured as icon shortcuts on the Home Page. The What’s New section lists the latest changes to the assets in the system so that users can get on top of the latest changes after login.



2013-12-17 Assessment Deleted lin
San Franciquito Creek-assessment-2013-12-17.07.49.37 for Bridge San Franciquito Creek deleted

2013-12-17 Assessment Updated lin
San Franciquito Creek-assessment-2013-12-17.07.49.37 for Bridge San Franciquito Creek updated »

2013-12-17 Assessment Added lin
San Franciquito Creek-assessment-2013-12-17.07.49.37 for Bridge San Franciquito Creek added »

2013-12-17 Assessment Added lin
San Franciquito Creek-assessment-2013-12-17.07.49.37 for Bridge San Franciquito Creek added »

4.3.1.5 Asset Folder

A folder contains assets or other sub-folders. It is used to organize assets of the same type. It also can have common reference materials related to the assets, such as track chart, design doc, standard procedure doc, and URL links to online reference web pages. Any arbitrary documents can be uploaded as attachments to the folder as its reference materials.

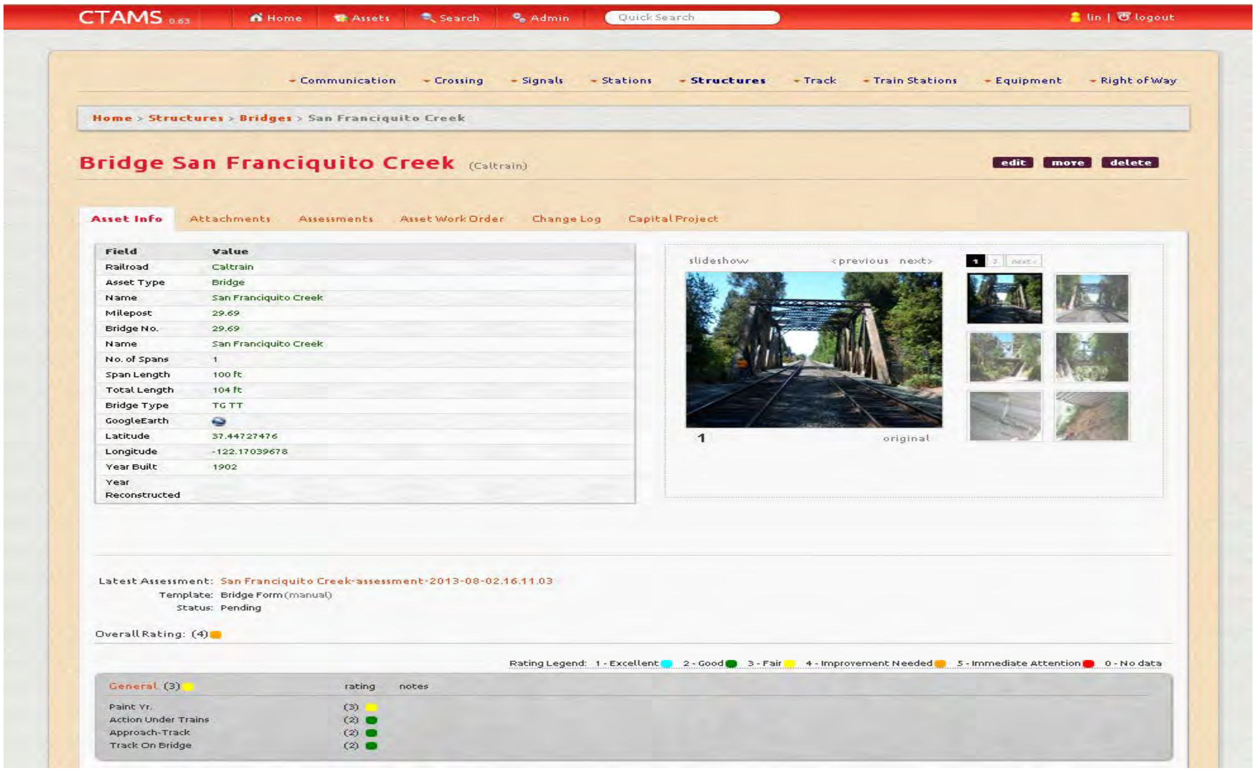
The folder shows its assets in list form, listing the data fields of the assets so that their values can be viewed and compared in a group. E.g. the condition rating of all the assets in the folder can be viewed in one shot. The asset list can be sorted by columns. The GIS integration on the folder includes all the assets in the folder so that all of them can be exported to Google Earth.

Folder can be configured with a user supplied icon easily. Just upload a picture file attachment as the icon for the folder.

4.3.1.6 Asset Page

The asset page maintains individual asset information, its digital media, its conditions, and links to other asset related data. It displays all the asset data fields. It has the photo gallery to show the asset's picture-based attachments. It has GIS integration to display the asset in Google Earth.

Each asset can have arbitrary digital media as file attachments. The picture type attachments are automatically organized in a photo gallery. Any file can be uploaded as attachments to the asset. URL link is also supported as an attachment type, for linking any web pages to the asset.



Home > Structures > Bridges > San Franciquito Creek

Bridge San Franciquito Creek (Caltrain)

Field	Value
Railroad	Caltrain
Asset Type	Bridge
Name	San Franciquito Creek
Milepost	29.69
Bridge No.	29.69
Name	San Franciquito Creek
No. of Spans	1
Span Length	100 ft
Total Length	104 ft
Bridge Type	TG TT
GoogleEarth	●
Latitude	37.44727476
Longitude	-122.17039678
Year Built	1902
Year Reconstructed	

Latest Assessment: San Franciquito Creek-assessment-2013-08-02.16.11.03
Template: Bridge Form (manual)
Status: Pending

Overall Rating: (4) 🟧

Rating Legend: 1 - Excellent 🟦 2 - Good 🟩 3 - Fair 🟨 4 - Improvement Needed 🟧 5 - Immediate Attention 🟥 0 - No data

General (3) 🟨	rating	notes
Paint Yr.	(3) 🟨	
Action Under Trains	(2) 🟩	
Approach-Track	(2) 🟩	
Track On Bridge	(2) 🟩	

4.3.1.7 Asset Input

The input of the asset data is done on the asset edit page, which shows asset specific data fields based on the asset's asset type, which dictates the data fields the asset can have. Data input goes through intelligent data validation based on the field data type. E.g. a number field can only have number digits, a date field must have a date, or an email field must have an email address. CTAMS supports a rich set of data type for asset fields.

Asset input also supports automatic data lookup for milepost to GIS latitude and longitude mapping. The milepost/GIS track mapping is built from track survey data and is maintained separately.

4.3.1.8 Customizable Asset Type

Asset is based on its asset type. Asset type defines the data fields of the asset and the data type of the fields. E.g. a Bridge asset type can have fields like Bridge Number, Bridge Type, Built Date, Span Length, etc. All asset types in CTAMS can be defined by user from scratch. This allows building of completely custom data model for the assets for different installation. The bootstrap procedure of a new CTAMS installation starts with the definition of the asset types by the admin user.

CTAMS supports a rich set of data type for asset data fields to enforce data integrity and validation:

- **Text**, accepts any one-line text data. The Max Length limits the amount of data that can be entered. The input control is a one-line edit line.
- **TextArea**, accepts any multi-line text data. The Max Length limits the amount of data that can be entered. The input control is a multi-line text-area.
- **Integer**, accepts whole integer number (without any decimal).
- **Decimal**, accepts decimal number (decimal point).
- **Date**, accepts date input (mm/dd/yyyy)
- **Email**, accepts email address (name@company.com)
- **PhoneUS**, accepts phone number in U.S. format (xxx-xxx-xxxx)
- **Url**, accepts URL link (http://www.website.com/abc/xyz/123/sample)
- **UrlTitle**, works together with a **URL** attribute and accepts text as its title
- **Boolean**, accepts true or false flag. Display checkbox as UI
- **Choice**, allows selection of a single item out of a list of choices. E.g. *Color* has choice values: *Red,Green,Blue,Yellow,Orange*.
- **Choices**, allows selection of multiple items out of a list of choices. E.g. *Favorite Colors* has choice values: *Red,Green,Blue,Yellow,Orange*. During asset data input, more than one colors can be picked.

4.3.1.9 Asset Digital Media

Managing digital media is central to asset management in CTAMS. Digital media is (not sure what is correct here) added as file attachments in the system. Many types of objects can have file attachments: folder, asset, inspection form, track evaluation, and work order.

File attachments can be easily uploaded by drag and drop. E.g. files dragged from Windows Desktop or Windows Explorer and dropped into the asset web page's upload area are uploaded automatically and attached to the asset. Upload via the standard file picking dialog is also supported.

Digital media files can be organized easily via the tag labeling system. Each file attachment can be tagged with labels. E.g. Engineering, Photo, Design. The file is grouped under the tagged labels.

Photo-based file attachments are automatically placed in a photo gallery for quick viewing.

URL-based link is also supported as a special kind of attachment that can be added to an asset, so that asset-related web reference and web pages can be attached.

4.3.1.10 Search and Index

CTAMS supports powerful free-form search based on keywords on the asset information and the attachment document content. Search is done on the asset data fields, and also on the content of all the file attachments. Popular file formats are supported: Word, Excel, PowerPoint, PDF, and Text.

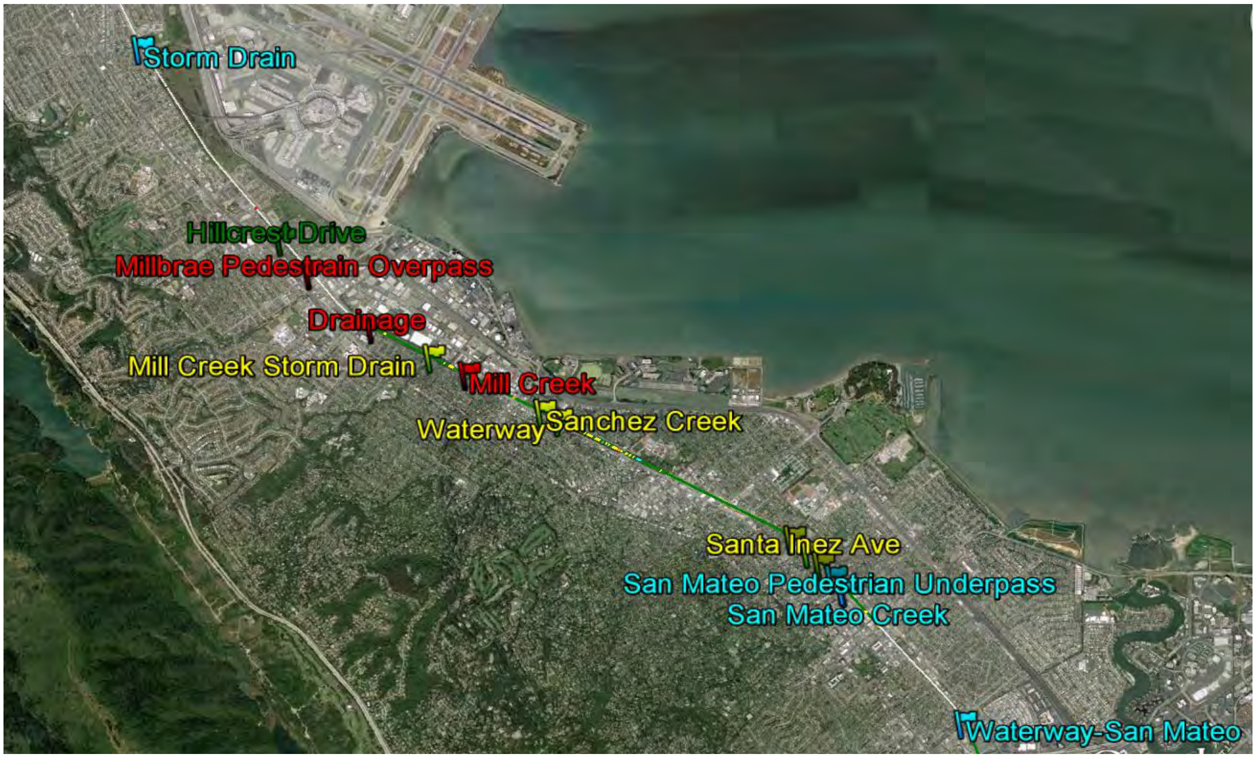
The search is transit aware. Milepost can be used as search parameters to limit the scope of searching. Asset types can be used as search parameter.

The assets and documents are indexed into a search index database for fast search. Indexing is done in real time where the index database is updated as new asset changes are saved and as documents uploaded.

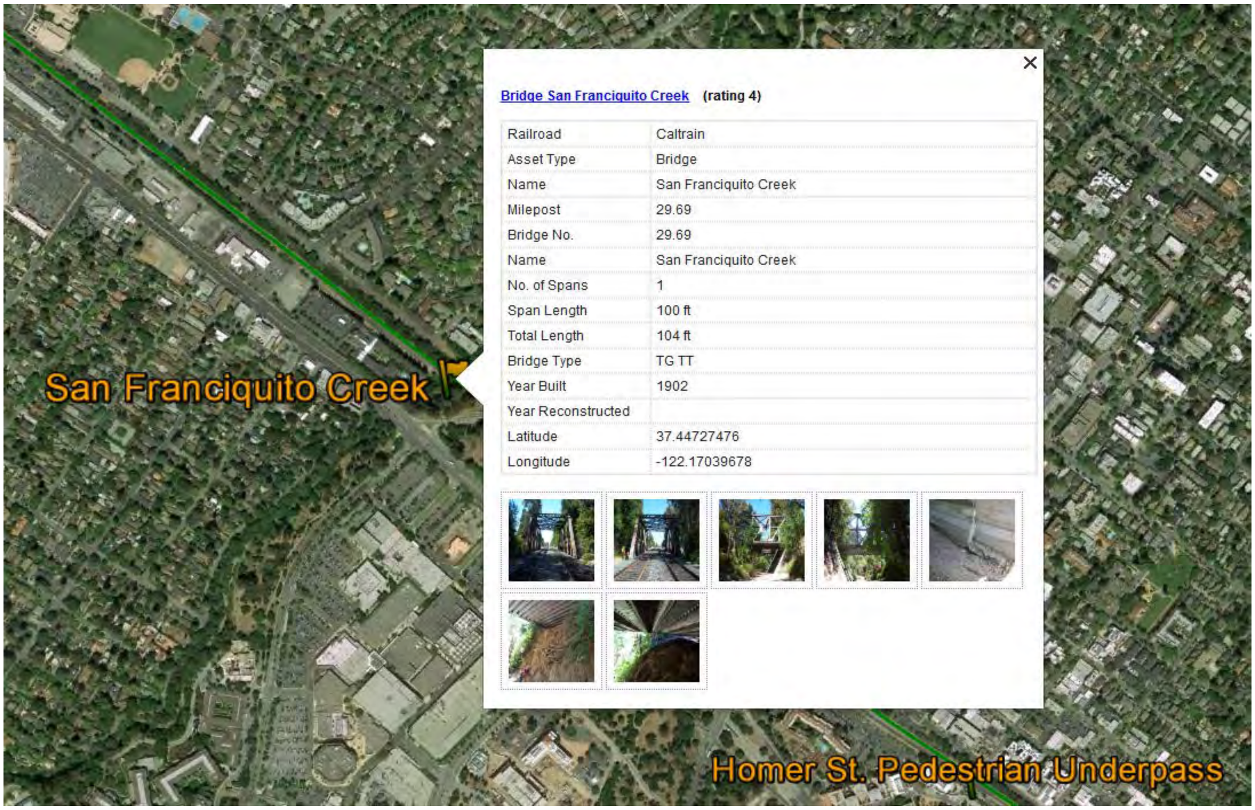
4.3.1.11 Geographic Information System (GIS) Integration

CTAMS provides map-based asset GIS integration via the open standard KML format. Google Earth is a big player in using KML. CTAMS goes beyond the usual displaying of assets in a map and supports deep integration with Google Earth to provide extensive map-based asset introspection and navigation.

GIS integration is done in a number of places: individual asset, asset list, and track evaluation. Multiple layers of map can be overlapped to show different types of asset information at the same time.



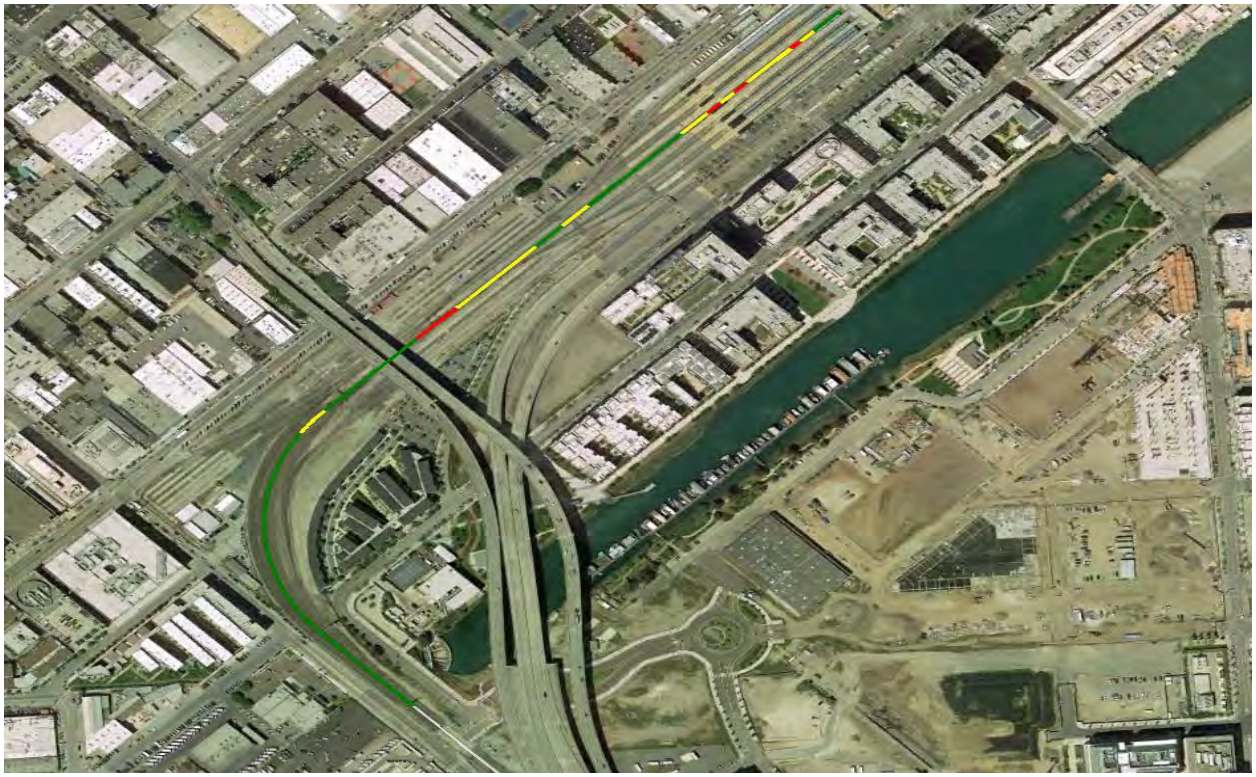
Display of the Bridge assets along the railroad, along with their color coded condition ratings.



Bridge San Francisquito Creek (rating 4)

Railroad	Caltrain
Asset Type	Bridge
Name	San Francisquito Creek
Milepost	29.69
Bridge No.	29.69
Name	San Francisquito Creek
No. of Spans	1
Span Length	100 ft
Total Length	104 ft
Bridge Type	TG TT
Year Built	1902
Year Reconstructed	
Latitude	37.44727476
Longitude	-122.17039678

The detail information of a bridge asset, with clickable links and pictures.



Display of track condition of the geometry car data evaluation in color code ratings.

4.3.2 Automatic Track Condition Assessment

In addition to Asset Management, Asset Condition Assessment is an important feature of CTAMS. A big part of assessment is the automatic track condition evaluation on the geometry car data. The geometry car is run periodically over the tracks to collect detailed data about track conditions such as profile, alignment, gauge, etc. The data are fed into CTAMS for evaluation. Track evaluation is done via user-defined rule set, which provides objective and consistent condition assessment.

4.3.2.1 Rule-based Track Evaluation

Rule-based automatic evaluation of track condition is performed with the track geometry car data. Evaluation is done down to per-foot granularity. Different aspects of the track are evaluated: warp, alignment, gauge, profile, and curve. Every foot of the track is run through the rule-set on all aspects of the track. The result is a per-foot color coded rating.

4.3.2.2 Custom Rule Definition

The evaluation rule set is user-defined. The user can define the rules and their criteria based on their experience or business need, or adopt FTA minimum standards.

Multiple rule set profiles can be defined. This allows running the track data repeatedly against different rule sets to see how the tracks fare under different standards.

4.3.2.3 Root Cause Detection

Each aspect of the per-foot rating has an explanation of why it’s rated that way, the rule failing the segment, and the rule’s criteria failing the segment.

4.3.2.4 Rating Aggregation

The track ratings are rolled up from finer granular level to coarser granular intervals (1/10th mile, per mile, per 10-mile).

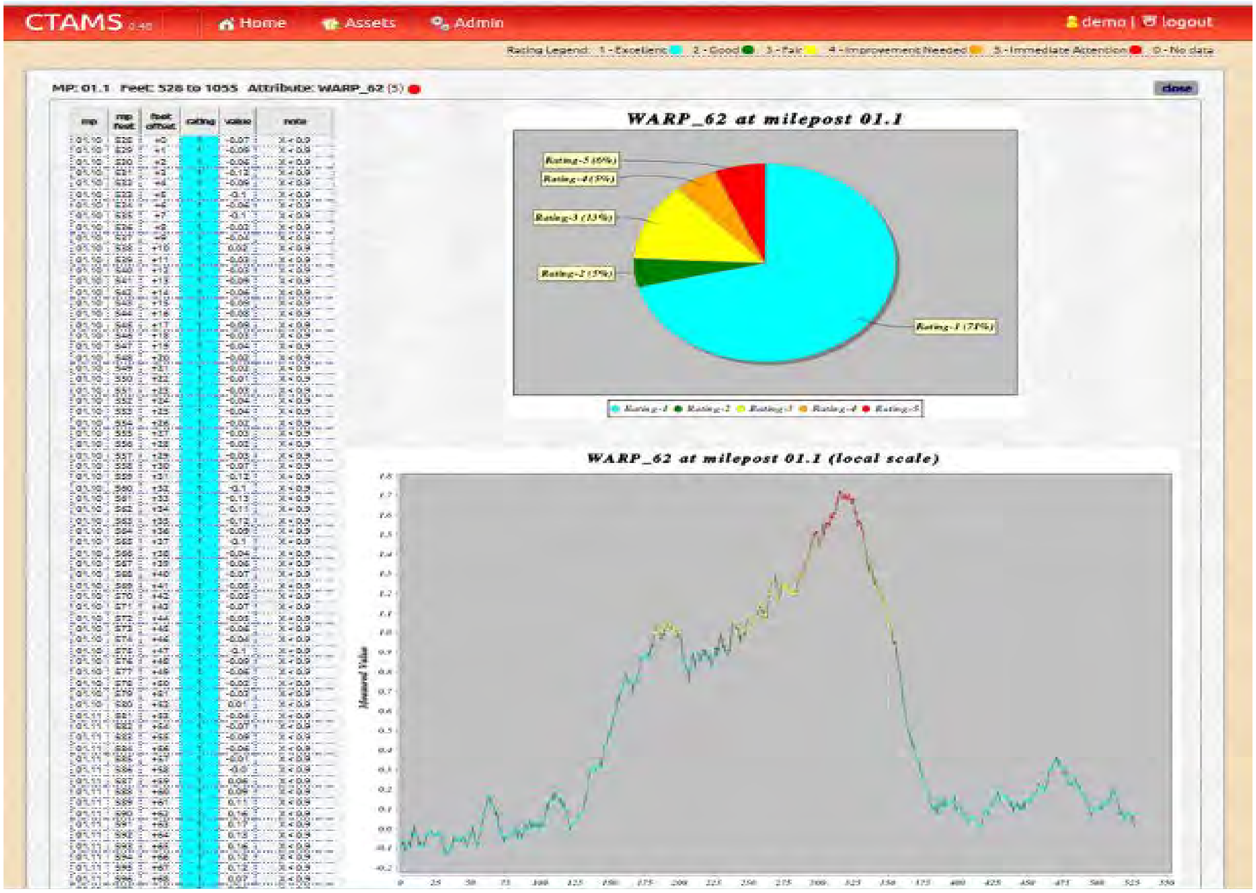
Manual override of the rating is supported to allow the human inspector to have the final say in the evaluation.

4.3.2.5 Attachments and Notes

Each segment (1/10th mile) can have document attachments uploaded to it. E.g. pictures for the inspection, repair diagram, or work documents. Also notes can be annotated to the segment to document the inspection.

4.3.2.6 Graph and Charts

The evaluation results are plotted in graphs and charts. The measured track values with rating colors are plotted along the track length, and pie charts are plotted for rating distribution.



MP: 01.1 Feet: 528 to 1055 Attribute: WARP_62 (5) ●

mp	mp feet	feet offset	rating	value	note
01.10	528	+0	1	-0.07	X < 0.9
01.10	529	+1	1	-0.09	X < 0.9
01.10	530	+2	1	-0.06	X < 0.9
01.10	531	+3	1	-0.12	X < 0.9
01.10	532	+4	1	-0.06	X < 0.9
01.10	533	+5	1	-0.1	X < 0.9
01.10	534	+6	1	-0.06	X < 0.9
01.10	535	+7	1	-0.1	X < 0.9
01.10	536	+8	1	-0.02	X < 0.9
01.10	537	+9	1	-0.04	X < 0.9
01.10	538	+10	1	0.02	X < 0.9
01.10	539	+11	1	-0.03	X < 0.9
01.10	540	+12	1	-0.03	X < 0.9
01.10	541	+13	1	-0.06	X < 0.9
01.10	542	+14	1	-0.06	X < 0.9
01.10	543	+15	1	-0.04	X < 0.9
01.10	544	+16	1	-0.08	X < 0.9
01.10	545	+17	1	-0.08	X < 0.9
01.10	546	+18	1	-0.03	X < 0.9
01.10	547	+19	1	-0.04	X < 0.9
01.10	548	+20	1	-0.09	X < 0.9
01.10	549	+21	1	-0.03	X < 0.9
01.10	550	+22	1	-0.01	X < 0.9
01.10	551	+23	1	-0.02	X < 0.9
01.10	552	+24	1	-0.04	X < 0.9
01.10	553	+25	1	-0.04	X < 0.9
01.10	554	+26	1	-0.02	X < 0.9
01.10	555	+27	1	-0.03	X < 0.9
01.10	556	+28	1	-0.02	X < 0.9
01.10	557	+29	1	-0.03	X < 0.9
01.10	558	+30	1	-0.07	X < 0.9
01.10	559	+31	1	-0.12	X < 0.9
01.10	560	+32	1	-0.1	X < 0.9
01.10	561	+33	1	-0.13	X < 0.9
01.10	562	+34	1	-0.11	X < 0.9
01.10	563	+35	1	-0.12	X < 0.9
01.10	564	+36	1	-0.09	X < 0.9
01.10	565	+37	1	-0.1	X < 0.9
01.10	566	+38	1	-0.04	X < 0.9
01.10	567	+39	1	-0.06	X < 0.9
01.10	568	+40	1	-0.07	X < 0.9
01.10	569	+41	1	-0.05	X < 0.9
01.10	570	+42	1	-0.05	X < 0.9
01.10	571	+43	1	-0.07	X < 0.9
01.10	572	+44	1	-0.05	X < 0.9
01.10	573	+45	1	-0.06	X < 0.9
01.10	574	+46	1	-0.04	X < 0.9
01.10	575	+47	1	-0.1	X < 0.9
01.10	576	+48	1	-0.09	X < 0.9
01.10	577	+49	1	-0.06	X < 0.9
01.10	578	+50	1	-0.02	X < 0.9
01.10	579	+51	1	-0.03	X < 0.9
01.10	580	+52	1	0.01	X < 0.9
01.11	581	+53	1	-0.04	X < 0.9
01.11	582	+54	1	-0.07	X < 0.9
01.11	583	+55	1	-0.08	X < 0.9
01.11	584	+56	1	-0.06	X < 0.9
01.11	585	+57	1	-0.07	X < 0.9
01.11	586	+58	1	-0.0	X < 0.9
01.11	587	+59	1	0.06	X < 0.9
01.11	588	+60	1	0.09	X < 0.9
01.11	589	+61	1	0.11	X < 0.9
01.11	590	+62	1	0.16	X < 0.9
01.11	591	+63	1	0.17	X < 0.9
01.11	592	+64	1	0.13	X < 0.9
01.11	593	+65	1	0.16	X < 0.9
01.11	594	+66	1	0.12	X < 0.9
01.11	595	+67	1	0.12	X < 0.9
01.11	596	+68	1	0.07	X < 0.9

4.3.2.7 GIS Integration

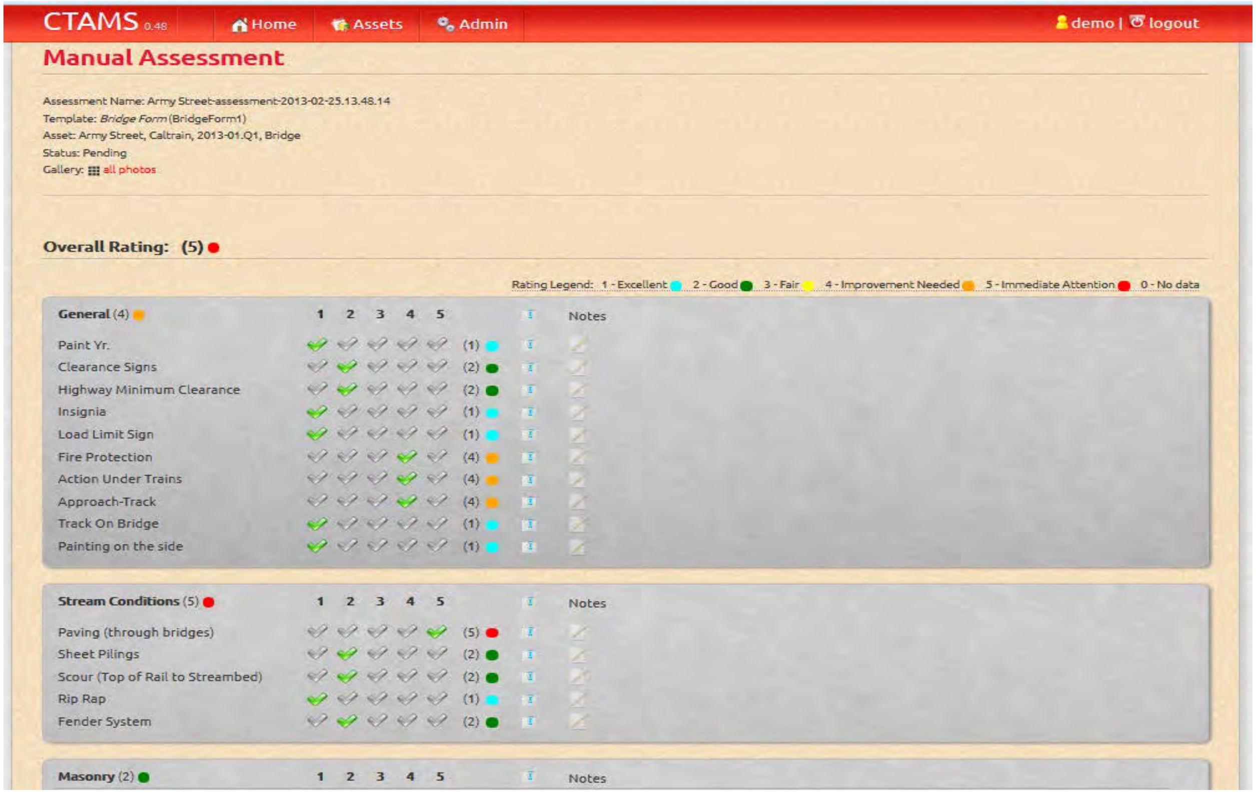
The track condition evaluation ratings can be exported to KML to provide map based viewing of the track condition ratings in Google Earth.

4.3.3 Asset Inspection and Assessment

Another big part of assessment is the manual inspection of assets. E.g. inspection of bridge and signal.

4.3.3.1 Online Inspection Form

Manual assessment is completed via the online electronic inspection forms. Electronic inspection forms are available across all modules and can be customized for each individual asset based on inspection requirements. The form consist sections of different attributes of the asset, allowing an inspector to rate each individual attribute of the asset. Attributes are rated from 1 to 5 according to Caltrain’s existing State of Good Repair rating system.



Manual Assessment

Assessment Name: Army Street-assessment-2013-02-25.13.48.14
Template: *Bridge Form* (BridgeForm1)
Asset: Army Street, Caltrain, 2013-01.Q1, Bridge
Status: Pending
Gallery: all photos

Overall Rating: (5) ●

Rating Legend: 1 - Excellent ● 2 - Good ● 3 - Fair ● 4 - Improvement Needed ● 5 - Immediate Attention ● 0 - No data

General (4) ●	1	2	3	4	5		Notes
Paint Yr.	✓					(1) ●	
Clearance Signs		✓				(2) ●	
Highway Minimum Clearance		✓				(2) ●	
Insignia	✓					(1) ●	
Load Limit Sign	✓					(1) ●	
Fire Protection				✓		(4) ●	
Action Under Trains				✓		(4) ●	
Approach-Track				✓		(4) ●	
Track On Bridge	✓					(1) ●	
Painting on the side	✓					(1) ●	

Stream Conditions (5) ●	1	2	3	4	5		Notes
Paving (through bridges)					✓	(5) ●	
Sheet Pilings		✓				(2) ●	
Scour (Top of Rail to Streambed)		✓				(2) ●	
Rip Rap	✓					(1) ●	
Fender System		✓				(2) ●	

Masonry (2) ●	1	2	3	4	5		Notes

4.3.3.2 *Condition Aggregation*

The evaluation is done interactively on the form and the ratings are rolled up automatically in real time from attributes to sections and up. Currently the worst rating is taken from below and rolled up to higher level. This allows flagging of problem areas during later drill down from high level ratings into finer sections.

4.3.3.3 *Attachments and Notes*

Digital documents can be attached to each rated attribute and to each section, allowing attaching of inspection evidence, visual record of problems, or any documents. Also notes can be added to each rated attribute to provide inspection documentation.

4.3.3.4 *Inspection Workflow*

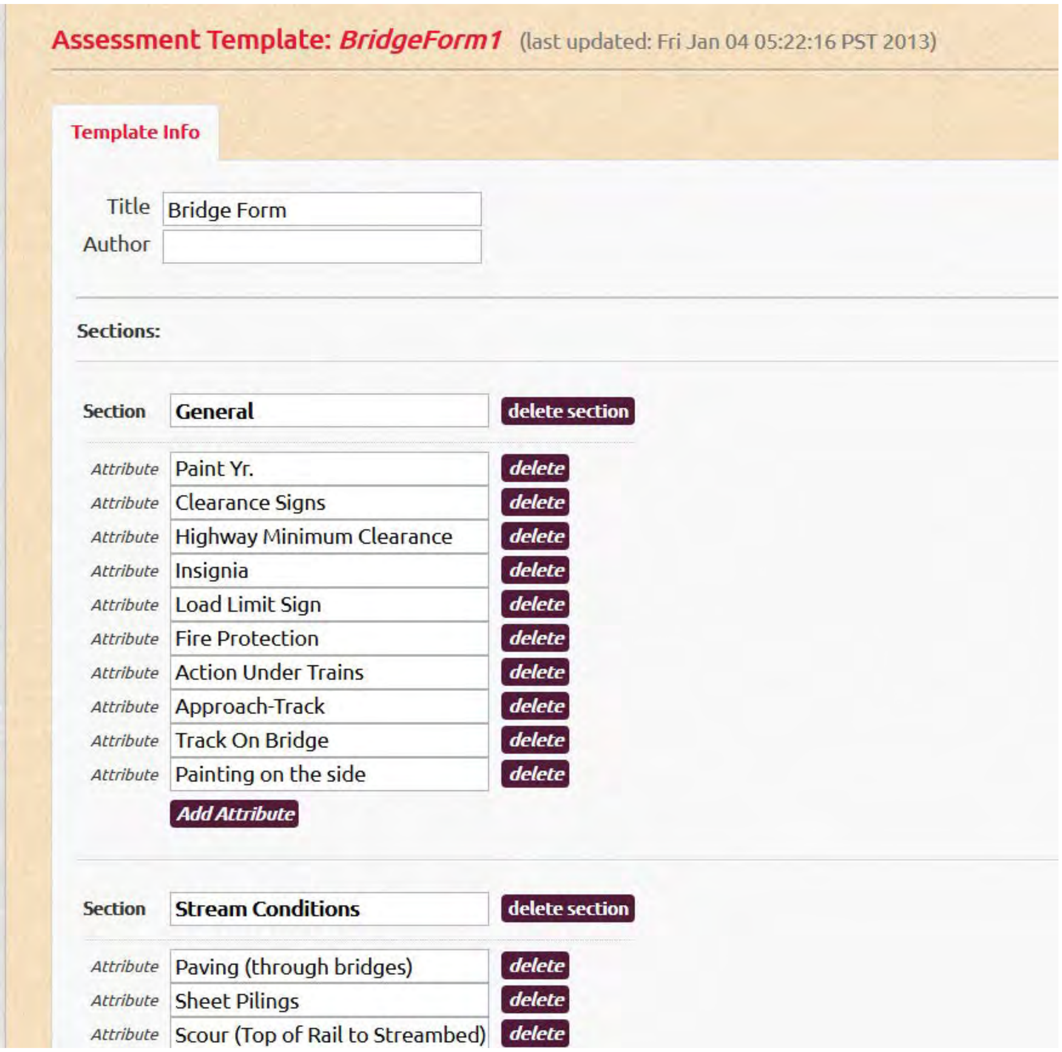
A simple workflow for the inspector, reviewer, and approver is supported. Field inspections are uploaded by the inspectors electronically via tablets or laptop computers available to the Caltrain field staffs. All inspections will starts with the Pending status, allowing the inspector to pause and come back later. When inspection is finalized, the inspection results are saved as Completed. Upon completion of the inspection the reviewers (supervisors) can review the inspection result, either approving or rejecting the inspection results. Once approved, the inspection's rating is published for the asset and becomes the asset's current condition rating.

4.3.3.5 Inspection History

New inspection overrides old inspections. However, old inspections (along with all their digital media) of an asset are saved. This keeps track of inspection records over time. They can trace asset performance over time to discover recurring problems.

4.3.3.6 Customizable Inspection Form

Different asset type has different aspects of relevance and importance for inspection. CTAMS allows user to define custom inspection forms for each asset type. The inspection form editor is an easy to use tool to create and edit inspection form interactively. Form can be test-run beforehand to get it working exactly.



Assessment Template: *BridgeForm1* (last updated: Fri Jan 04 05:22:16 PST 2013)

Template Info

Title: Bridge Form
Author:

Sections:

Section: General

Attribute Paint Yr.
Attribute Clearance Signs
Attribute Highway Minimum Clearance
Attribute Insignia
Attribute Load Limit Sign
Attribute Fire Protection
Attribute Action Under Trains
Attribute Approach-Track
Attribute Track On Bridge
Attribute Painting on the side

Section: Stream Conditions

Attribute Paving (through bridges)
Attribute Sheet Pilings
Attribute Scour (Top of Rail to Streambed)

4.3.4 Repair Management

Beyond asset assessment, CTAMS provides repair tracking via the work order system.

4.3.4.1 Work Order

Work order tracks repair on found defects. Work order can be linked to a track evaluation or a manual inspection form, to address the problems found during the assessment.

4.3.4.2 Customizable Work Order Type

Different assets have different repair details and thus require different work order type. Different type supports different data fields for the work order form.

4.3.4.3 Work Order Attachments

Arbitrary documents can be attached to the work order. This provides a flexible way to track repair detail. E.g. repair cost estimate spreadsheet, repair diagram, repair step document, material document, work schedule, pricing list, pictures, repair evidence, etc. can be attached to the work order.

4.3.4.4 Integration with Excel

The data fields of the work order can be exported with an Excel template to provide intelligent external integration. E.g. the repair cost, crew hours, repair duration can be exported in Excel for integrating with other systems for tracking and for analysis.

4.3.5 System Feature

4.3.5.1 User Authentication

User name and password are used for authenticating user to log in to the web app. Password can be reset by a user or by the administrator.

4.3.5.2 User Authorization

CTAMS utilizes the concept of user roles to grant and restrict access of functionality to a user. A role bundles up a set of access privileges to certain functionality, to make it easier to manage user authorization. To let a user access certain functionality, simply grant the user to have the role. E.g. a user with Reviewer role can read data but not update them. Granting the additional role Updater to the user allows him to perform data update.

4.3.5.3 Change Log

CTAMS implements a system wide change log to tracks changes to all assets, for auditing purpose. Changes to an asset are logged with the information on when, what, and by whom.

The change log is also used for generating the What’s New section in the Home Page.

5 NEW PROCESSES AND PROCEDURES

During the development of the previous SOGR database, Caltrain also developed its own SOGR Standard Operating Procedures (SOP) to outline and define the SOGR processes. The last version of the SOP, from 2008, is still the current SOGR SOP used by Caltrain.

While most of the basic concepts and processes from the previous version of the SOP is still applicable and has proven success for Caltrain's SOGR program to date, changes will be required as the agency adapts to a new system.

The Information Technology group was not heavily involved in the maintenance and upkeep of the previous Excel SOGR database. That maintenance was performed by the Engineering staff. However, a relational database residing on an internal server will require the software and hardware maintenance previously not necessary. The district's IT group will maintain the hardware, i.e. the servers, in accordance with the district's current maintenance schedule.

Because CTAMS will introduce an entirely different data format than the previous Excel Version of the SOGR database, the data management and data backup process will be revamped to meet those changes. Data management and backup will remain the responsibility of the Engineering staff who oversees the State of Good Repair program. The requirements of data management and backup will be defined by the manager of the SOGR program and will be incorporated in a revision of the SOGR SOP.

The other major change will come in the workflow of the inspection, review, and approval process. Although that process is outlined in the current SOGR SOP, a change to an electronic system, where data is available almost instantly, the process will change to accommodate for the new system. The built in feature of multiple user levels is aimed at towards helping to define the process for which documents will be received, reviewed and finalized in the database. This process will also be defined and incorporated in a revision of the SOGR SOP.

Overall, Caltrain's SOGR SOP will remain similar to its existing practice, but taking into account for the changes to the database system. As the agency adapts and adjusts to this new database, further changes to the SOP will likely be necessary to address the introduction of a new system and new functions.

6 TRAINING

Throughout the development process, various Caltrain Engineering staff members were given previews of CTAMS in order for the project team to receive feedback as well as to better understand what the end users ultimately want to use the database for and how.

Those comments and feedback from the Engineering staff members were not only instrumental in the development of the database, but also taken into account in the final live demonstration to ensure the staff received the information they were looking for and what will help them in the everyday operation of CTAMS.

Two levels of training are created for the CTAMS program. The first of which is a basic level training intended for general users without manager or administrator roles in the database. The second level of training is for the users with manager or administrator roles

The live demonstration mentioned above served as the basic training for the general Engineering staff members. The single session to the staff members introduces the basic functions of

CTAMS and its three major functions of Asset Management, Asset Assessment, and Work Order Management. This basic training coupled with the user manual teaches end users the basic skills required to operate CTAMS.

The live demonstration was broken down into the following parts:

1. What is CTAMS?
2. Changes from previous SOGR Database
3. Benefits of CTAMS
4. Introduction to CTAMS Home Page and Update Log
5. Asset Management
    a. Basic Asset Information
    b. Attachments
    c. Inspections
    d. Asset Rating
    e. Export to Google Earth
    f. Change Log
    g. Capital Projects
6. Asset Assessment
    a. Building Custom Inspection Forms
    b. Manual Assessment
    c. Custom Rule Sets
    d. Automatic Track Geometry Assessment
    e. Overlay on Google Earth
7. Work Orders
    a. Open Work Orders
    b. Work Order Status
    c. Attachments
    d. Export to Excel
8. Modules
9. Changes ahead
10. Lessons Learned
11. Q&A Session

In addition to the basic training program above, which is intended for the general staff member, discipline managers and select staff members with administrative roles received additional training on the following subjects:

1. User Setup
2. User Management
3. Create Railroad
4. Add New Assets
5. Edit Asset Attributes
6. Delete Assets
7. Custom Rule Models
8. Track Work Orders

A user manual (Appendix A) was created to summarize the basic concepts and operations of CTAMS and is available to staff members via the agency's internal network. A revision of the user manual will include Frequently Asked Question from the training sessions to provide further guidance and clarity for the end users.

7 LESSONS LEARNED

7.1 Lessons Learned Summary

As Caltrain and RSE implemented CTAMS over the past 18 months, we identified some valuable lessons learned along the way that will not only help our own agency on future projects but may also be of value to other transit agencies as well.

To summarize the valuable experience, we grouped those lessons into two categories: what worked for the project team and what can potentially change in our process.

7.1.1 What Worked

Near the end of the project, as the team reflected on the valuable lessons learned from the process, we realized the need to not only summarize what we need to change, but also summarize what had worked well for this project team so that our experience can potentially help another transit agency.

1. Industry experts work closely with developer

   Having industry experts work closely with the software developer is critical in the development of a successful end product. Past systems projects in the agency have experienced problems on the project when a disconnect is developed between the owners and the developers. That disconnect is often caused by a lack of consistent communication and feedback between the two parties.

   From the onset of the CTAMS project, Caltrain and RSE made the conscious effort to have the subject matter experts work very closely with the primary developer. This includes having regular team meetings, consistent review and feedback, and allowing the developer the flexibility to contact the subject matter experts directly. This close working relationship allowed both parties to be engaged with the project at all times and the result as a smooth process to reach the end product both parties envisioned at the onset of the project.

2. Small, diverse, and stable core project team

   While the project involved input from many parties within the agency, the core project team remained small and stable but also diverse. The core team consisted of the agency's SOGR expert, who led the previous effort of creating a SOGR database; the primary database developer; a project manager; and an engineering manager.

The smaller core project team offered stability as well as continuity as the team gathered the sorted through information collected from the various input sources. It allows the information to flow in a proper manner and not get lost in the shuffle as it will at times when a project team is large. Although, the core team is small, the diversity ensured the project was put through different angles and lenses by the team members so that the end product is complete and well-rounded.

3. Iterative development process

   As mentioned previously in the development process, the iterative approach is one that helped the project team to tackle issues immediately as opposed to leaving a laundry list of action items for the end of the project and the risk of leaving certain items unresolved.

   Tackling one feature and function at a time allowed the team to focus on the specific item and perfect it before moving onto the next. However, it is important to note this approach still included a large integration factor. While each component is built individually, it is still tested on a whole to ensure each function is seamlessly integrated into the database.

4. Weekly working sessions with specific deliverables

   Weekly work sessions throughout the development process kept the project team close and on schedule with targeted deliverables. Based on our experience in the project, this was an essential element to the success of the project. It gave the owners the opportunity to see updates as they're being made to the system and gave the developers immediate feedback they needed to make the necessary changes.

5. Open to explore new ideas

   No idea is a bad idea. That's the mindset the project team took on during brainstorming sessions and discussions. While the team is fully aware certain ideas may not work out or the schedule and budget simply won't allow us to implement, keeping an open mind gave us the opportunity to come up with features that turned out to be essential for a complete Asset Management Program.

   The version of CTAMS that is in operations today is filled with functions that came about as a part of discussions and brainstorming. For example, the idea of a work order management system is absolutely essential to a complete and successful program. Yet, it was not a part of the original concept and was only incorporated as it came up in a brainstorm session. Without the open mind to explore new features and functions, an essential part of the program would not have been a part of the final product.

7.1.2 What Could Be Better

Even though the project overall was a success, there is always room for improvement. What's summarized below hopefully beneficial to other agencies as they go through their own process of creating an Asset Management Program.

1. Aggressive schedule

   The original schedule seemed very reasonable based on the original scope and the basic concepts. However, despite taking into account for some possible changes and adding the necessary contingency in the schedule, the team still did not account for the amount of additional yet essential features and functions that came as a result of the brainstorm sessions, feedback, and working sessions. The final completion of the project was delayed four months due to those changes and some other unpredictable events such as the availability of certain team members due to other projects.

   The nature of a systems project mean there will be unpredictable changes. For future projects, those changes should have been incorporated into the baseline schedule.

2. Involvement of other internal customers

   Getting internal customers involved early on in the project is essential to developing the features and functions of the database. Although the project team engaged staff members early on, there still were other groups that could have been involved earlier.

   The existing SOGR database has resided with the Engineering Department and has been used as a tool for Engineering. Therefore, the focus on the project at the onset was to gather the needs and requirements from Engineering. As a result, other internal stakeholders, such as the Finance Department was either engaged too late or not at all, thus leaving out components that could potentially be incorporated into the database as well. This is one of the reasons the financial aspects of the overall program was not implemented as indicated in Section 3.2.1.

3. Better user interface design

   User interface design turned out to be one of the bigger challenges faced by the project team in the development process. The project team focused on the content and functions of the database to make sure the end user needs were addressed. However, as the development of the database progressed, the team realized making the database application user friendly was a much larger task than assumed.

   Making the application easier to use and navigate was always one of the main goals but took a backseat to the development of the content. The core team simply did not have an expert on user interface and it was not until later in the process when an expert was engaged. While the end product is easy navigate, we all feel there’s always room for more improvement. The team anticipates that getting feedback from users as they transition to the new program will help us improve the visual aspect of the database even more in the next update.

4. Transition of system.

Perhaps the single most difficult thing to do is making the successful transition from system to system.

Learning a system takes time and making effective use of the system takes even longer. The move to a new system means brand new training for everyone who needs to be a part of the process and there will always be a learning curve before the system is working effectively as intended.

The sheer amount of data from the previous database also requires a massive transfer effort in order to populate the new database with essential data. These efforts simple will require time as the agency makes the shift in not only the software, but adapts to the changes in process that comes with this shift.

Early engagement and constant communication is key to making the successful transition, which requires a complete team effort.

8 CONCLUSION

As a transit agency, it is always Caltrain's goal to run a successful and more importantly, safe and reliable service. To do so, keeping the entire system, from railroad spikes and clips to bridges and crossings to the locomotives themselves, in a state of good repair is critical. Caltrain recognized the importance of an asset management system early on and developed the original SOGR database to assist with that effort. This grant from the FTA further provided Caltrain with the opportunity to improve and explore new ways of running an effective and efficient Asset Management Program.

CTAMS will provide Caltrain with an improved tool for asset management, but the end of this pilot project is just a start for the program as a whole. Maintaining the database, incorporating new features, adding modules for new equipment, and further improvements are just some of the steps for the future.

Asset management is an ongoing effort, an effort that has no end. There will always been further improvements and changes as we work to improve the effectiveness of the program. CTAMS is just the first step towards that ultimate goal.

9 APPENDIX A – USER MANUAL

CTAMS User Manual

Caltrain Transit Asset Management System

Revision History

Revision	Date	Revision Description	Author
Rev. 1	07/29/2013	Initial draft	William Wong

USER'S MANUAL

TABLE OF CONTENTS

Page #

1 GENERAL INFORMATION

1.1 Introduction

CTAMS is a web-based application for managing transit assets and performing periodic assessment on them. As a web application, it can be accessed through any of the modern web browsers (see the supported browsers). For asset management, assets can be organized in hierarchical structure with folders. Custom asset types can be defined easily. Assets are indexed automatically for quick search. Asset related documents and media (pictures, drawing, docs) are stored along with the asset as attachments. Attached documents are indexed automatically for easy search. CTAMS integrates with KML-aware GIS applications (e.g. Google Earth) to display assets at their GIS location.

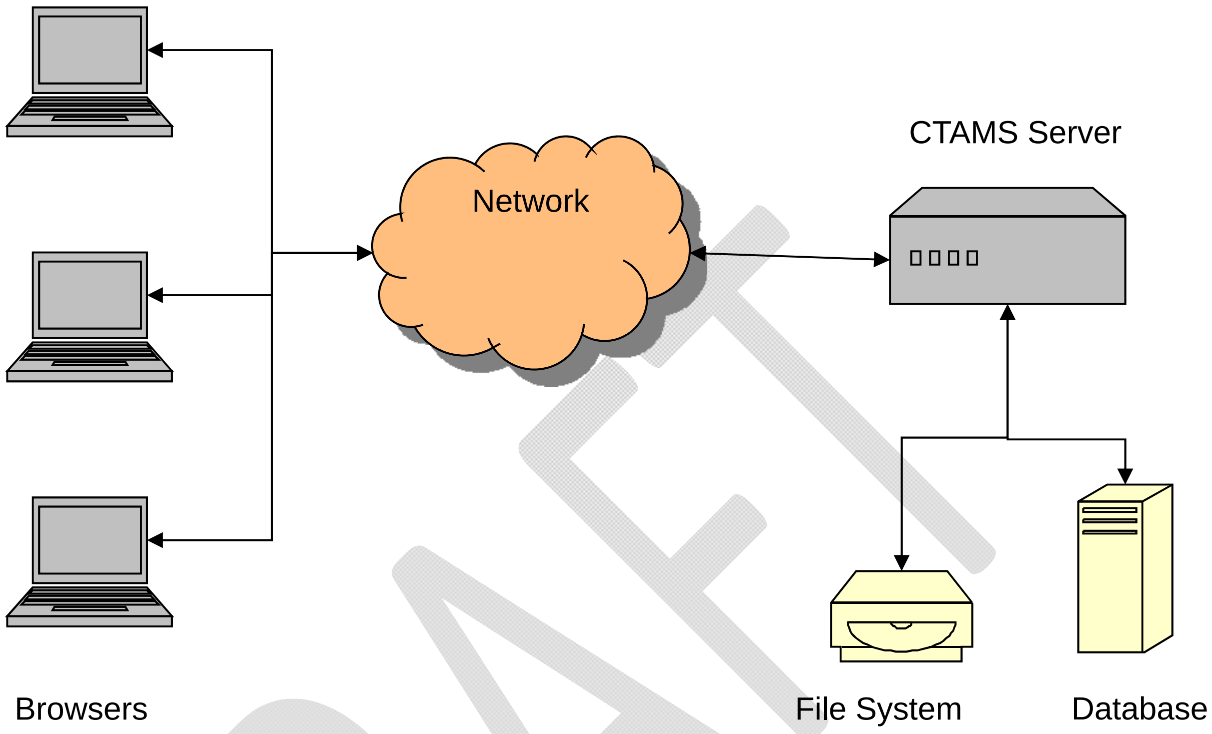
Assessments are done periodically on the assets to evaluate their condition and health. CTAMS can perform automatic assessment on rail tracks by running through the collected geometry cart data to evaluate various aspects of the track according to a rule profile. For assets that cannot be automatically evaluated, CTAMS provides manual assessment via the survey mechanism. Custom survey template can be defined easily.

1.2 Document Overview

This document is divided into three main sections: the general overview, the user guide, and the admin guide. For end user tasks, jump to the User Guide for details. For admin related tasks, setting up the app, or configuring the app, jump to the Admin Guide for details.

1.4 System Overview

1.4.1 Architectural Diagram



1.4.2 System Components

CTAMS consists of four major components:

- The frontend UI on the web browser
- The CTAMS application server
- Database to store any asset and assessment related data
- File storage to store any asset media attachments.

From the user’s perspective, all the functionalities are available over the web and can be accessed via the browser.

1.5 Feature Tour

Here’s a quick tour of the features in CTAMS.

1.5.1 Asset Management

- Hierarchical asset organization
  - Asset folder to hold assets
  - Folders nested within folders
  - Assets and folders can be moved to other folders
- Home page for frequently accessed folders
  - Customizable folder shortcut
  - Folder icon is customizable
  - What’s New section lists all recent changes
- Folder view
  - Show asset list
  - Folder attachments as reference materials
  - Attachment label tags to organize attachemnts
  - Drag and drop to upload attachment (where the browsers support it)
  - URL link as attachment
  - Change log to track the changes done in the folder
- Asset list view
  - List the summary fields of assets
  - Re-arrange order by clicking on field column header
  - Filter by field values
  - GIS integration to display the list of assets in Google Earth
- Asset detail view
  - Display all asset data fields
  - GIS integration to display the asset in Google Earth
  - Photo gallery to show the asset’s picture-based attachments
  - Change log to track the changes done to the asset
- Asset update view
  - Enter or change asset field data
  - Intelligent data validation (e.g. enforce integer, date, email input)
  - Pre-defined field value selection
  - Multi-value selection for field
  - Intelligent GIS data lookup based on milepost
- Custom asset type
  - User-defined asset fields
  - Rich built-in data types
  - Customizable list view and form view
  - Customizable display order and sort order
- Asset attachment
  - Upload any files as attachments for the asset (drawing, photo, design docs)
  - Tag label for attachment to organize attachments into groups
  - Drag and drop to upload (where the browsers support it)
  - Multiple files upload at once
  - Set up URL links as attachments (links to reference materials or other sites)
- Work order

  - Work orders can be created on an asset to track the remediation on the defects
  - Support documents can be filed to a work order as attachments
  - User defined work order type can be created to capture custom fields
  - Integration with Excel via intelligent export to customizable Excel form.
- Capital project
  - Link assets to capital projects to track assets in projects.
- Search and index
  - Search assets by keyword
  - Search attachment by keyword
  - Search by asset type
  - Search by milepost range
  - Index asset data and attributes
  - Index attachment documents of popular file formats (doc, xls, ppt, pdf, etc)
  - Incremental indexing as new assets added or new updates saved

1.5.2 Automatic Track Assessment

- Evaluation of track condition
  - Rule-based automatic evaluation of track condition
  - Color coded rating for easy comprehension of problematic areas.
  - Evaluation of different aspects of a track, e.g. gauge, profile, warp, etc
  - Allow manual override of rating
- Different level of evaluation
  - Precise evaluation down to per-foot level
  - Rating aggregation up to higher granular intervals (1/10th mile, per mile, per 10-mile)
  - Rating explanation shows why the rating is given according to the rules
- Attachments and notes
  - Allow attachments to the assessed track (e.g. pictures of track), for each interval
  - Note taking on track condition, for each interval
- Chart
  - Plot charts for the measured value for all intervals with rating colors
  - Generate rating pie charts to show rating distribution in intervals
- GIS Integration
  - GIS integration to display rating-colored track in Google Earth
- Custom rule definition
  - User-defined rule profile to use for evaluation
  - Rich rule operations to build up rule profile
  - Excel-based custom rules for evaluating complex track curve condition

1.5.3 Manual Assessment

- Assessment Survey
  - Survey to perform manual evaluation on arbitrary asset
  - Easy to use interactive form-based survey
  - Manual evaluation of different aspects of an asset (asset attributes)
  - Support multiple sections to group attributes in survey
  - Automatically roll up of rating from attributes to sections and to overall

- Attachments and notes
  - Allow attachments to asset (e.g. pictures)
  - Note taking on asset condition
- Workflow
  - Simple workflow for the evaluator, reviewer, and approver
- Custom survey definition
  - User-defined survey form for manual assessment
  - Interactive form design

1.6 Supported Browsers

The following list of browsers is supported.

- Modern Firefox (21 and up)
- Modern Chrome
- IE 9 and up

1.7 Concepts and Terms

2 USER GUIDE

3 ADMIN GUIDE

This aims at providing documentation for users that have the Admin role and need to perform administrative tasks on the system.

3.1 First Time Setup

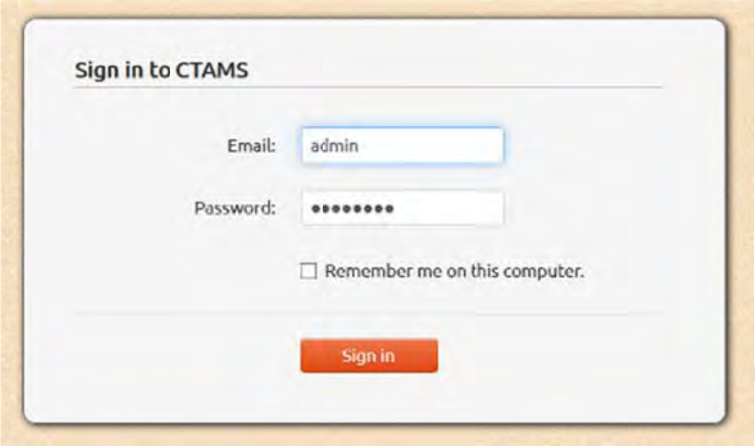
CTAMS is fairly generic and allows extended customization. When first installed, it’s like a blank slate and requires certain setup and configuration to get a working system.

The followings are the typical setup needed on a freshly installed system.

3.1.1 Log on as Built-in Admin

When freshly installed, the app has no user accounts beside the built-in *admin* account. The first thing is to log on with the *admin* account to bootstrap the system.

- Log on as *admin*
- The built-in password for *admin* is *caltrain*



Sign in to CTAMS

Email: admin

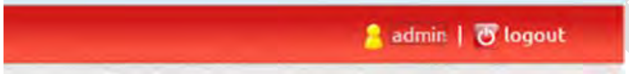
Password: ••••••••

□ Remember me on this computer.

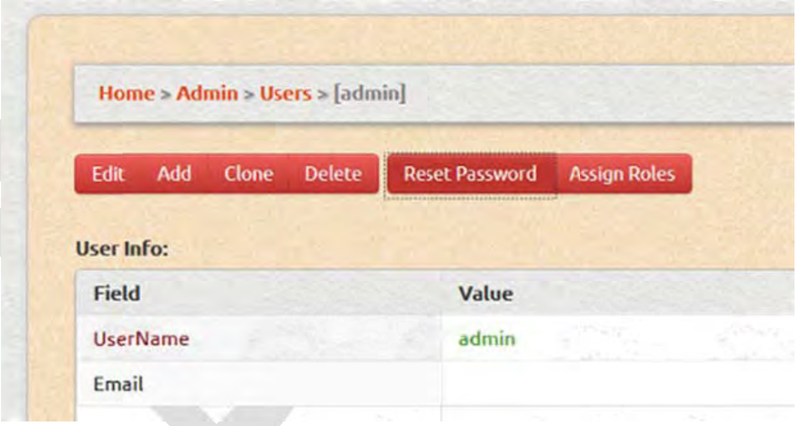
3.1.2 Change Built-in Admin Password

It's a good security practice is to change the password for the built-in *admin* user right the way.

- Click on the **admin** user link on the navigation bar to go to the user info page



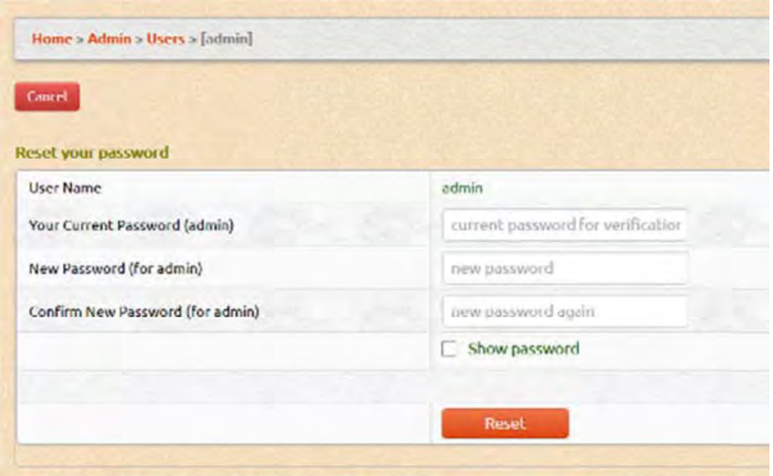
- On the user info page for *admin*, click on the **Reset Password** button



User Info:

Field	Value
UserName	admin
Email	

- Reset the password for *admin*
  - Enter *caltrain* in the **Your Current Password** field
  - Enter the new password for *admin* in the **New Password** field
  - Re-enter the password again in the **Confirm New Password** field
  - Click on the **Reset** button



Home > Admin > Users > [admin]

Reset your password

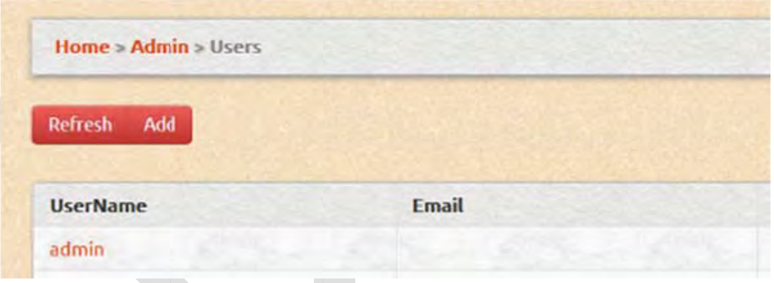
User Name	admin
Your Current Password (admin)	current password for verificatior
New Password (for admin)	new password
Confirm New Password (for admin)	new password again
	□ Show password

3.1.3 User Setup

When freshly installed, the app has no users beside the built-in *admin* user. One of the first tasks is to create some users account.

3.1.3.1 Add New User

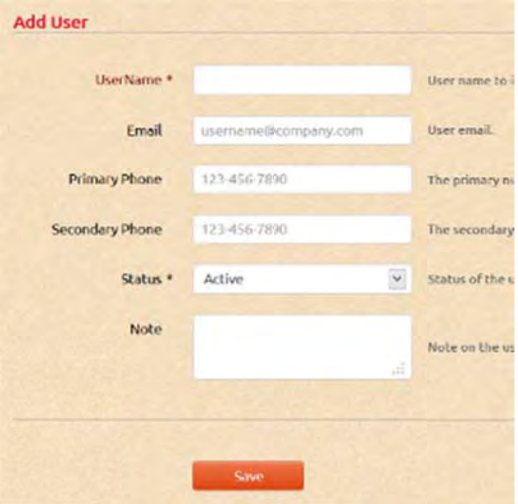
- Go to the **Admin** page
- Click on the **Users** link to go to the **Users** page
- Click on the **Add** command to add new user



Home > Admin > Users

UserName	Email
admin	

- On the **Add User** form, assign a user name for the new user
- Fill out the rest of the field as appropriate and save



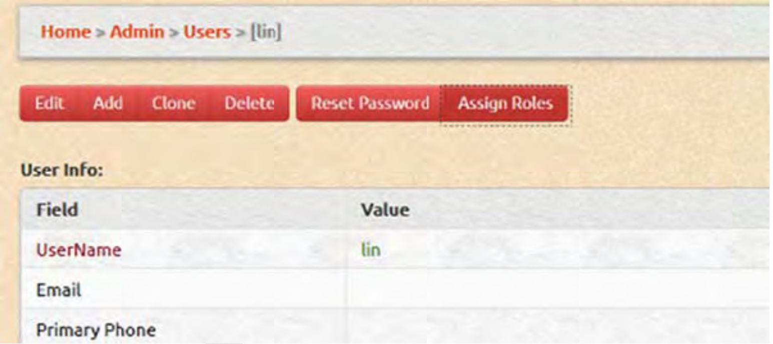
UserName *		User name to i
Email	username@company.com	User email.
Primary Phone	123-456-7890	The primary n
Secondary Phone	123-456-7890	The secondary
Status *	Active	Status of the u
Note		Note on the us

3.1.3.2 Assign Roles

The permissions for a user to perform operations in the app are granted via user roles. A user having the *Administrator* role will have the permissions to access the admin functions and to perform admin tasks. A user with only the *Viewer* role can only read data but not update data.

To assign (or un-assign) roles to a user,

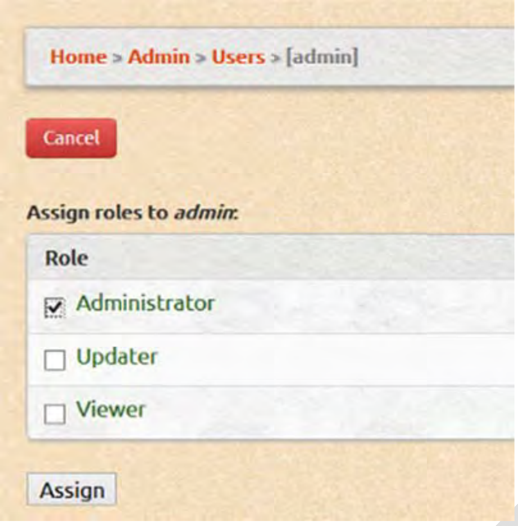
- Go to the **Admin** page
- Click on the **Users** link to go to the **Users** page
- Click on a user to display **User Info** page
- Click on the **Assign Roles** command to assign/change roles for the user



User Info:

Field	Value
UserName	lin
Email	
Primary Phone	

- Select or de-select the checkbox next to the roles to assign or un-assign roles for the user
- Click on **Assign** to save the changes



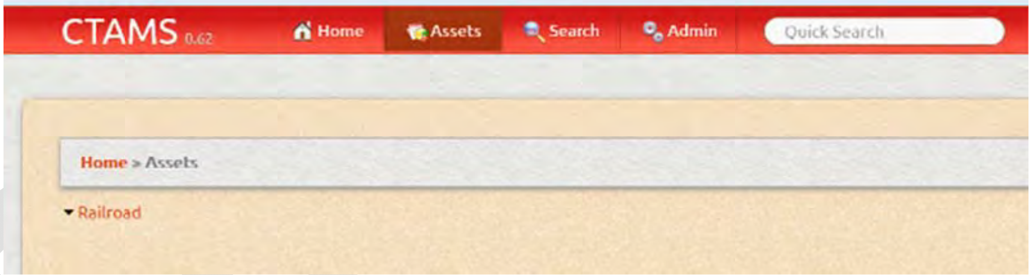
Assign roles to *admin*:

Role
☑ Administrator
□ Updater
□ Viewer

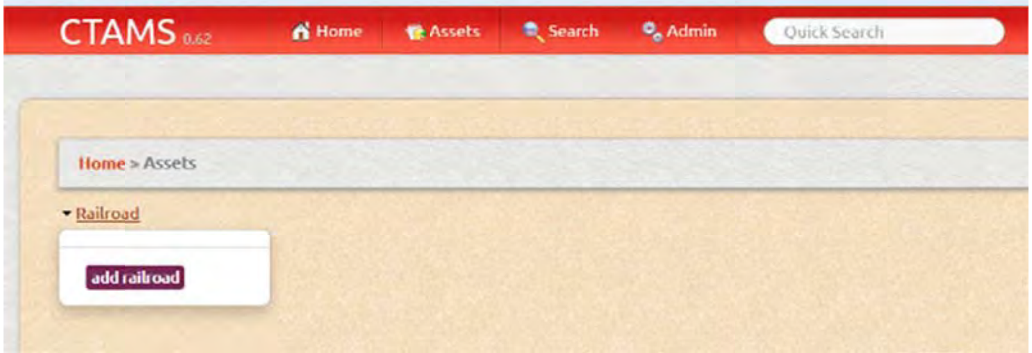
3.1.4 Create Railroad

Asset data in the CTAMS webapp are grouped under the high level entity *Railroad*. Create a railroad for a fresh install before adding other assets.

- Click on the **Assets** link on the navigation bar to go to the **Assets** page

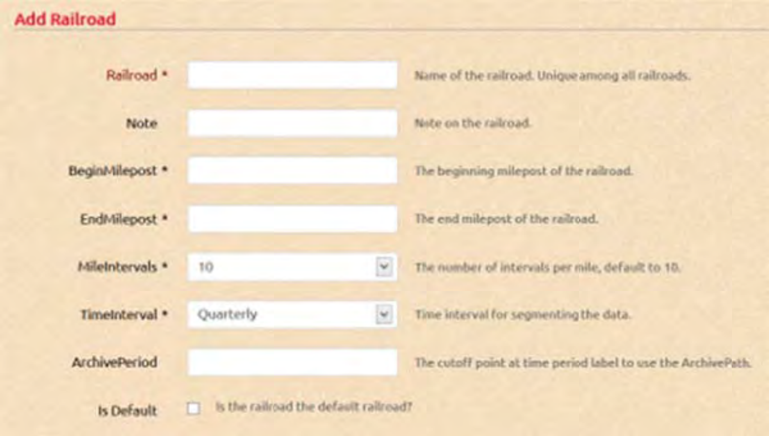


- Click on the **Railroad** dropdown list. Click on the **add railroad** command to add the railroad



- Give the railroad a name and fill in the rest of the fields as needed.

- The **BeginMilepost** and **EndMilepost** fields should span the entire length of the railroad. They are used as beginning and end milepost when processing track data during auto-assessment.
- The **MileIntervals** sets the default interval for giving rating in auto-assessment. It should be set to 10, i.e. 10 intervals per mile, giving 1/10th of length for each interval.
- The **IsDefault** flag indicates whether this railroad is the default one if there are multiple railroads defined. The default railroad is used when no railroad is selected on webapp.



Add Railroad

Railroad *		Name of the railroad. Unique among all railroads.
Note		Note on the railroad.
BeginMilepost *		The beginning milepost of the railroad.
EndMilepost *		The end milepost of the railroad.
MileIntervals *	10	The number of intervals per mile, default to 10.
TimeInterval *	Quarterly	Time interval for segmenting the data.
ArchivePeriod		The cutoff point at time period label to use the ArchivePath.
Is Default	□	Is the railroad the default railroad?

3.2 User Management

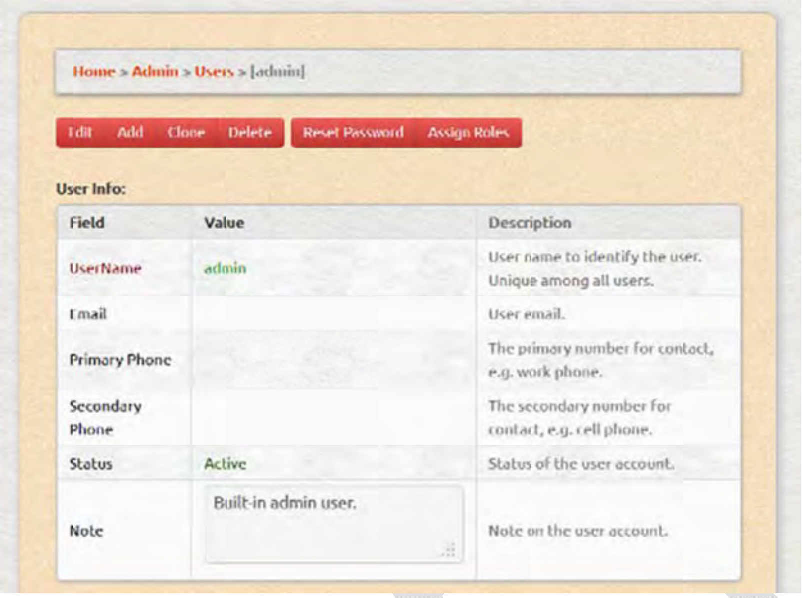
The user list page **Users** can be accessed by clicking on the **Users** link on the **Admin** page.

3.2.1 Add User

See 3.1.3.1 Add New User for detail.

3.2.2 Edit User

Open the user detail page by clicking on the user name in the list of the users. Click on the **Edit** button to start editing the user.



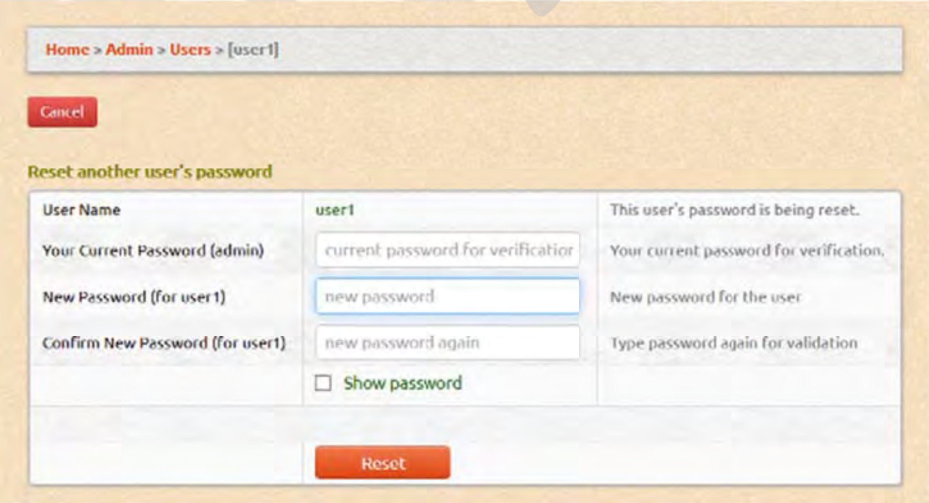
User Info:

Field	Value	Description
UserName	admin	User name to identify the user. Unique among all users.
Email		User email.
Primary Phone		The primary number for contact, e.g. work phone.
Secondary Phone		The secondary number for contact, e.g. cell phone.
Status	Active	Status of the user account.
Note	Built-in admin user.	Note on the user account.

3.2.3 User Management Operations

The user detail page has a number of action buttons.

- **Add**, add a new user account
- **Clone**, add a new user account with cloned data from the current user. Need to use different name for the new account.
- **Delete**, delete the user account
- **Assign Roles**, grant operation permission to user by assigning roles to him
- **Reset Password**, reset the selected user’s password



Reset another user's password

User Name	user1	This user's password is being reset.
Your Current Password (admin)	current password for verification	Your current password for verification.
New Password (for user1)	new password	New password for the user
Confirm New Password (for user1)	new password again	Type password again for validation
	□ Show password	

NOTE: When a user is added, its initial password is *caltrain*. Admin can reset the new user’s password to something else, or inform the user to log on and reset his own password.

3.3 Asset Type Management

Asset type defines the attributes of similar assets, giving uniform structure to assets of the same type. E.g. all bridges have the attribute *Bridge Number, Bridge Type, and Year Built*. The *Bridge* asset type defines these attributes and their data types, so that when we create a new bridge asset, we know to fill in the fields of these attributes.

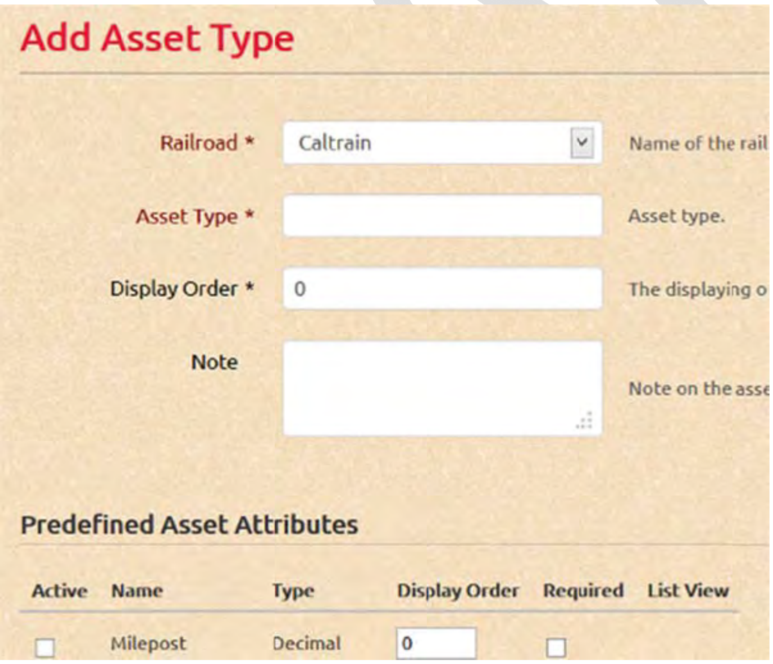
3.3.1 Create Asset Type

Navigate to the **Add Asset Type** page

- Go to the **Admin** page
- Click on the **Asset Types** link to go to the **Asset Types** page
- Click on the **add asset type** button to add an asset type

The **Add Asset Type** page has three sections: general info input, predefined attribute selection, and custom user-defined attribute creation. For general info,

- Select the railroad for the new asset type. Railroad is used to grouped related asset data.
- Name the new asset type
- Fill in the rest of the fields.



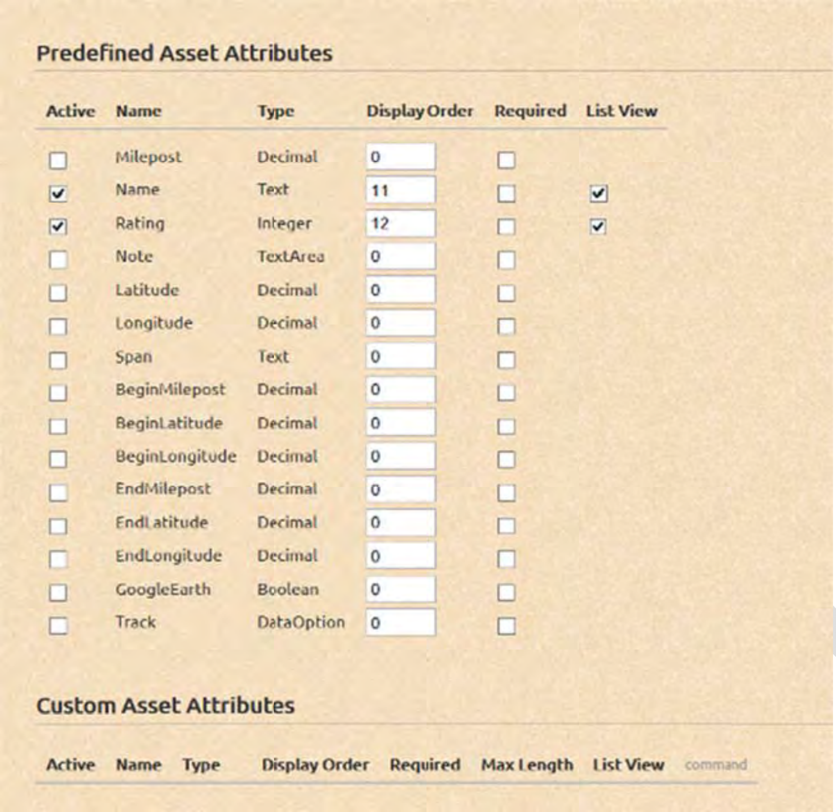
Add Asset Type

Railroad *	Caltrain	Name of the rail
Asset Type *		Asset type.
Display Order *	0	The displaying o
Note		Note on the asse

Predefined Asset Attributes

Active	Name	Type	Display Order	Required	List View
□	Milepost	Decimal	0	□	

Predefined attributes are common attributes that webapp has defined ahead of time so that you can just pick them for use. Select the predefined attributes that are appropriate for the asset type.



Predefined Asset Attributes

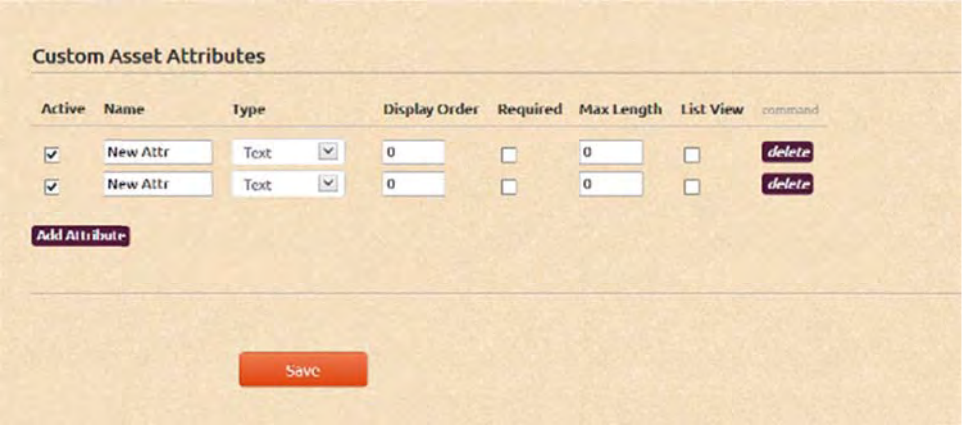
Active	Name	Type	Display Order	Required	List View
□	Milepost	Decimal	0	□	
☑	Name	Text	11	□	☑
☑	Rating	Integer	12	□	☑
□	Note	TextArea	0	□	
□	Latitude	Decimal	0	□	
□	Longitude	Decimal	0	□	
□	Span	Text	0	□	
□	BeginMilepost	Decimal	0	□	
□	BeginLatitude	Decimal	0	□	
□	BeginLongitude	Decimal	0	□	
□	EndMilepost	Decimal	0	□	
□	EndLatitude	Decimal	0	□	
□	EndLongitude	Decimal	0	□	
□	GoogleEarth	Boolean	0	□	
□	Track	DataOption	0	□	

Custom Asset Attributes

Active	Name	Type	Display Order	Required	Max Length	List View	command

- Check the **Active** checkbox to select the predefined attribute for the asset type
- **Name** is the predefined name of the attribute
- **Type** is the data type of the attribute. Data type helps input validation.
- **Display Order** specifies the order of displaying the attribute on the UI
- **Required** marks the attribute as a required field that must have input data
- Check the **List View** checkbox to display the attribute in the asset list view

The last section of the page is for adding user-defined custom attribute. In here you can define your own attributes for the asset type. The custom attribute has similar definition items as the predefined ones. In addition, you can specify the name, the data type, and other input validation criteria.



Custom Asset Attributes

Active	Name	Type	Display Order	Required	Max Length	List View	command
☑	New Attr	Text	0	□	0	□	delete
☑	New Attr	Text	0	□	0	□	delete

3.3.1.1 Data Types of Asset Attribute

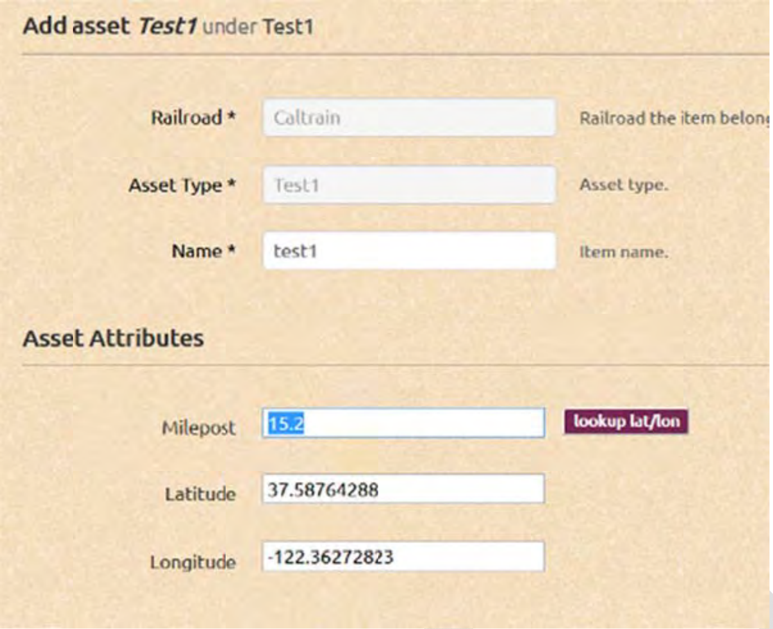
Each attribute has a data type to define what kind of data it can hold. Data type also helps in validate the data during input, to keep the data clean. The webapp supports the following data types for attribute:

- **Text**, accepts any one-line text data. The Max Length limits the amount of data that can be entered. The input control is a one-line edit line.
- **TextArea**, accepts any multi-line text data. The Max Length limits the amount of data that can be entered. The input control is a multi-line text-area.
- **Integer**, accepts whole integer number (without any decimal).
- **Decimal**, accepts decimal number (decimal point).
- **Date**, accepts date input (mm/dd/yyyy)
- **Email**, accepts email address (name@company.com)
- **PhoneUS**, accepts phone number in U.S. format (xxx-xxx-xxxx)
- **Url**, accepts URL link (http://www.website.com/abc/xyz/123/sample)
- **UrlTitle**, works together with a **URL** attribute and accepts text as its title
- **Boolean**, accepts true or false flag. Display checkbox as UI
- **Choice**, allows selection of an item out of a list of choices. When defining the attribute, list all the choices separated by comma. E.g. Attribute *Color* has choice value: *Red,Green,Blue,Yellow,Orange*.
- **Choices**, allows selection of multiple items out of a list of choices. When defining the attribute, list all the choices separated by comma. E.g. Attribute *Favorite Colors* has choice value: *Red,Green,Blue,Yellow,Orange*. When entering asset, more than one colors can be picked.

3.3.1.2 Special Attribute: GIS Aware Attributes

The predefined attribute **Milepost, Latitude, and Longitude** are special in that they can perform automatic conversion from milepost to latitude and longitude during data input of the **Milepost** field.

When after entering a milepost, tabbing out of the field will look up the lat/lon values and fill in the latitude and longitude fields if they are blank. If the latitude and longitude fields already have data, they are left alone. That’s in case the values have been overridden by hand previously. To force a lookup, click on the **lookup lat/lon** button.



Add asset *Test1* under Test1

Railroad * Caltrain Railroad the item belon

Asset Type * Test1 Asset type.

Name * test1 Item name.

Asset Attributes

Milepost 15.2

Latitude 37.58764288

Longitude -122.36272823

Mapping between milepost and lat/lon is maintained under **Milepost Mapping** on the **Admin** page.

3.3.1.3 Special Attribute: GoogleEarth

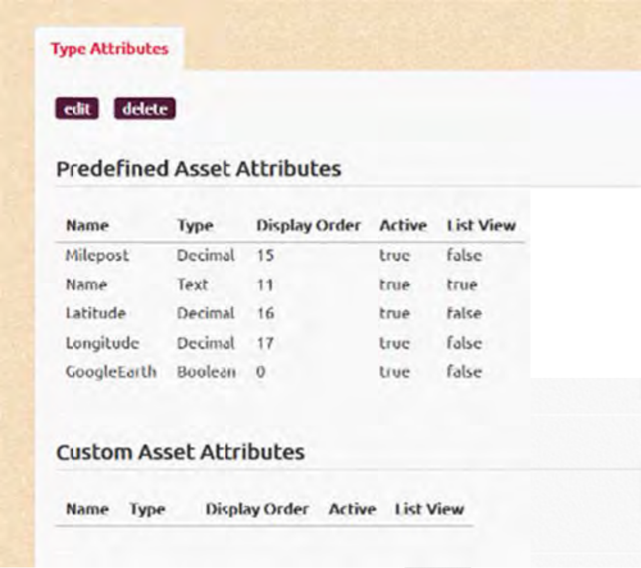
The predefined attribute **GoogleEarth** is a special attribute. When it's added along with the **Latitude** and **Longitude** attributes, the webapp will display a GoogleEarth button that can generate a KML with the lat/lon value to be displayed in Google Earth.

3.3.2 List Asset Type

The list of asset types can be displayed from the **Admin** page. Click on the **Asset Types** link on the **Admin** page to go the asset type list page.

3.3.3 Display Asset Type Definition

From the asset type list page, click on individual asset type link will go to the **Asset Type** info page. Here you can examine the defined attributes of the asset type.



Predefined Asset Attributes

Name	Type	Display Order	Active	List View
Milepost	Decimal	15	true	false
Name	Text	11	true	true
Latitude	Decimal	16	true	false
Longitude	Decimal	17	true	false
GoogleEarth	Boolean	0	true	false

Custom Asset Attributes

Name	Type	Display Order	Active	List View

3.3.4 Edit Asset Type

On the **Asset Type** info page click on the **edit** button to start editing the asset type. The **Asset Type Edit** page is similar to the create asset type page. You can change any aspect of the asset type, except the name, which can't be changed once created.

3.3.5 Delete Asset Type

Before deleting an asset type, you should delete all the assets that are created from it; otherwise, those assets would have no information on how to use its attributes.

Note: for the current version, the webapp does not prevent you from deleting the asset type before the assets, nor does it automatically delete all the assets of the asset type. It's upto to you to enforce those operations for now.

3.5 Rule Profile

Auto assessment on track data uses the rule set defined in the rule profile. This section describes the detail of rule profile and how to define them.

3.5.1 Rule Profile Overview

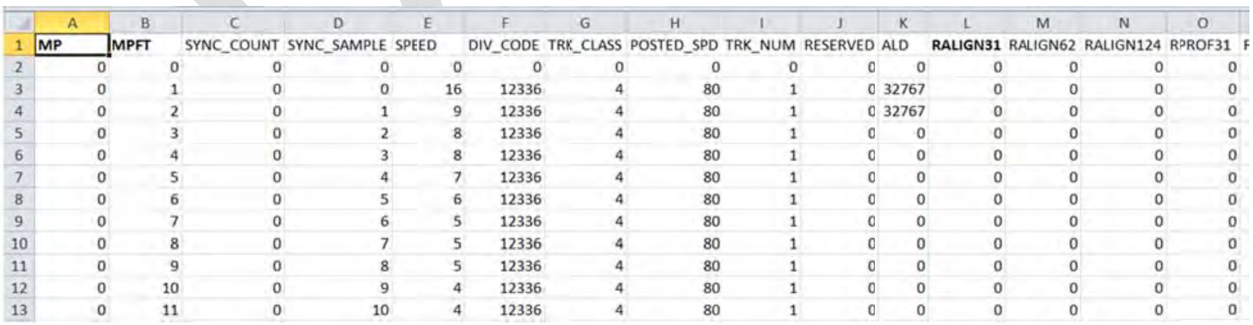
The rule profile consists of a list of rule sets. Each rule set is responsible for evaluating one attribute of the track to give a rating. E.g. there's a rule set for GAUGE, a rule set for RALIGN31 (right alignment 31"), etc. The rule set consist of a list of rules, where each rule has a conditional criteria to check whether the attribute value satisfied for its rating. Each rule is responsible for one rating (1 to 5). Here's an outline depicting the rule profile and its components.

- Rule Profile
    - Rule Set for GAUGE
        - Rule for rating 1
          Condition: pass if gauge value is between 56.0 and 56.5
        - Rule for rating 2
          Condition: pass if gauge value is between 56.5 and 57.0
        - Rule for rating 3
          Condition: pass if gauge value is between 57.0 and 57.25
          …
    - Rule Set for RALIGN31
        - Rule for rating 1
          Condition: pass if ralign31 value is less than 0.4
        - Rule for rating 2
          Condition: pass if ralign31 value is between 0.4 and 0.5
          …

      …

3.5.2 Track Data Format

The rule profile runs against the track's geometry cart data file. The file is a CSV file and should have the following format:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	MP	MPFT	SYNC_COUNT	SYNC_SAMPLE	SPEED	DIV_CODE	TRK_CLASS	POSTED_SPD	TRK_NUM	RESERVED	ALD	RALIGN31	RALIGN62	RALIGN124	RPROF31
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	1	0	0	16	12336	4	80	1	0	32767	0	0	0	0
4	0	2	0	1	9	12336	4	80	1	0	32767	0	0	0	0
5	0	3	0	2	8	12336	4	80	1	0	0	0	0	0	0
6	0	4	0	3	8	12336	4	80	1	0	0	0	0	0	0
7	0	5	0	4	7	12336	4	80	1	0	0	0	0	0	0
8	0	6	0	5	6	12336	4	80	1	0	0	0	0	0	0
9	0	7	0	6	5	12336	4	80	1	0	0	0	0	0	0
10	0	8	0	7	5	12336	4	80	1	0	0	0	0	0	0
11	0	9	0	8	5	12336	4	80	1	0	0	0	0	0	0
12	0	10	0	9	4	12336	4	80	1	0	0	0	0	0	0
13	0	11	0	10	4	12336	4	80	1	0	0	0	0	0	0

- Each row is a set of data collected for the track attributes at the milepost location
- The columns on the right are the measured values of the track attributes
- The first column is **MP**, the milepost marker of the data collect
- The second column is **MPFT**, the feet offset from the beginning of the current mile

- The attribute value column, e.g. **RALIGN31**. The rule set references the attribute column name to look up the attribute value.

3.5.3 Rule Evaluation Model

When assessment is performed on a track, the rule profile runs through the track's geometry cart data file line by line. For each line, it picks up the milepost mark, the feet offset, and extract track attribute values based on the column names. Each rule set defines an attribute column name it would extract from.

Note: It's important that the rule set's attribute name matches the attribute column name in the geometry cart data file.

The extracted attribute value is evaluated against each rule in the rule set and the first one that satisfies the conditional criteria will set its rating for the attribute at that milepost location. Evaluation for other ratings stops for the rest of the rules in the rule set.

This is repeated for all attributes value for all rule sets. And it further repeats for all data lines in the data file for all mileposts.

3.5.4 Rule Definition

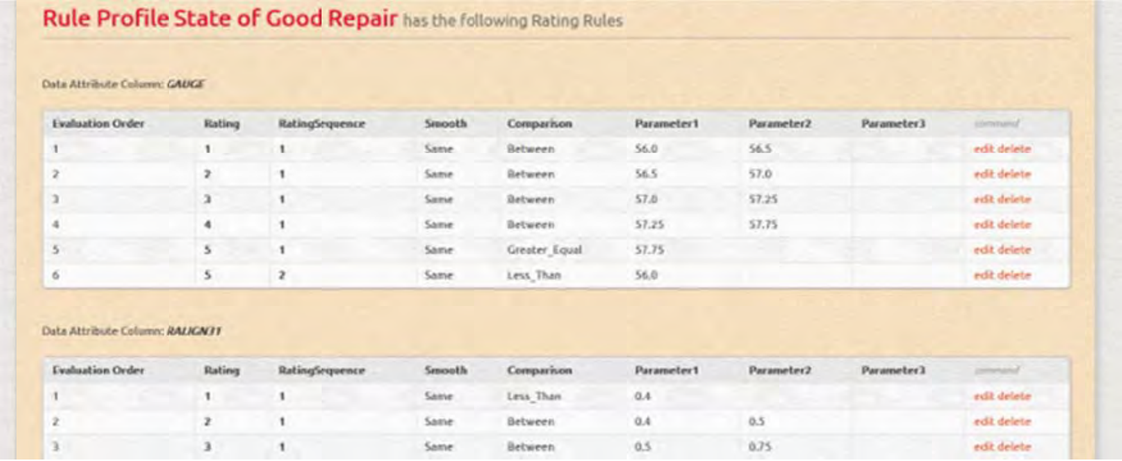
3.5.4.1 Rule Profile Creation

Go to **Admin** page and click on **Rule Profiles**. Click on **Add** to add a new profile. Fill in the name and note, and click on **Save**.

A rule profile when newly created is just an empty shell. Need to fill it with the rule sets and rules.

3.5.4.2 Rating Rule List

The rule profile page shows the list of rating rules, which are grouped into rule sets for each attribute.



Rule Profile State of Good Repair has the following Rating Rules

Data Attribute Column: *GAUGE*

Evaluation Order	Rating	RatingSequence	Smooth	Comparison	Parameter1	Parameter2	Parameter3	command
1	1	1	Same	Between	56.0	56.5		edit delete
2	2	1	Same	Between	56.5	57.0		edit delete
3	3	1	Same	Between	57.0	57.25		edit delete
4	4	1	Same	Between	57.25	57.75		edit delete
5	5	1	Same	Greater_Equal	57.75			edit delete
6	5	2	Same	Less_Than	56.0			edit delete

Data Attribute Column: *RALIGN31*

Evaluation Order	Rating	RatingSequence	Smooth	Comparison	Parameter1	Parameter2	Parameter3	command
1	1	1	Same	Less_Than	0.4			edit delete
2	2	1	Same	Between	0.4	0.5		edit delete
3	3	1	Same	Between	0.5	0.75		edit delete

3.5.4.3 Rating Rule

Add rating rule to build up the rule set. A rule set for an attribute is formed automatically with the added rating rules with the same attribute name.

When adding a rating rule, it has the following fields.

- Profile – this is the rule profile it belongs to.
- Attribute – the attribute name this rule applies to. This should match the attribute column name in the data file.
- Rating – the target rating to assign the attribute when the criteria is satisfied.
- Rating Sequence – a target rating can have multiple rules. This distinguishes one rule from the others. E.g. To get rating 1, the attribute value needs to be between 10.1 and 10.2, OR between 15.3 and 15.4. This would result in two rating rules, one with the criteria between 10.1 and 10.2, and the other with the criteria between 15.3 and 15.4. In order to distinguish two rules, a different RatingSequence number is used.
- Evaluation Order – this sets the order to evaluate the rule within the rule set for the attribute. An attribute rule set has a list of rules for different rating. During evaluation, as soon as one rule's criteria is satisfied, the evaluation for the rule set stops and the rating is given. So if the evaluation order is set up to evaluate from rating 1 to rating 5, the criteria of rating 1 to rating 5 are evaluated in that order. When rating 2's criteria is satisfied, for example, the evaluation is stopped and rating 2 is assigned to the attribute.
- Smooth – this is the smooth function to smooth out the attribute value before evaluating it. Sometimes the rating system work better if the data is damped or amplified. This gives a chance to smooth the data before runs it through the criteria. The following smooth functions are supported.
  - Same, keep the attribute value the same
  - Abs, convert the attribute value to its absolute value, e.g. abs(-5) = 5
  - Log, take the logarithm(2) of the attribute value. This damps the value with Ln2
  - Log10, take the logarithm(10) of the attribute value. This damps the value with Log10
  - Square, take the square of the attribute value, e.g. 5*5 = 25
  - Square Root, take the square root of the attribute value, e.g. sqrt(36) = 6
- Comparison, and Parameter1, Parameter2, and Parameter3. This forms the criteria of the rule. The intent is to compare the attribute value against the parameters using the one of comparison operators. Each operator takes on one, two, or thee parameters. The following comparison operators are supported.
  - Between, uses Parameter1 and Parameter2. Parameter1 <= attribute < Parameter2
  - Equal, uses Parameter1. Attribute == Parameter1
  - Not_Equal, uses Parameter1. Attribute != Parameter1
  - Greater_Than, uses Parameter1. Attribute > Parameter1
  - Greater_Equal, uses Parameter1. Attribute >= Parameter1
  - Less_Than, uses Parameter1. Attribute < Parameter1
  - Less_Equal, uses Parameter1. Attribute <= Parameter1
  - Match_Any, uses no parameter. Wildcard match any attribute value. Always return true.

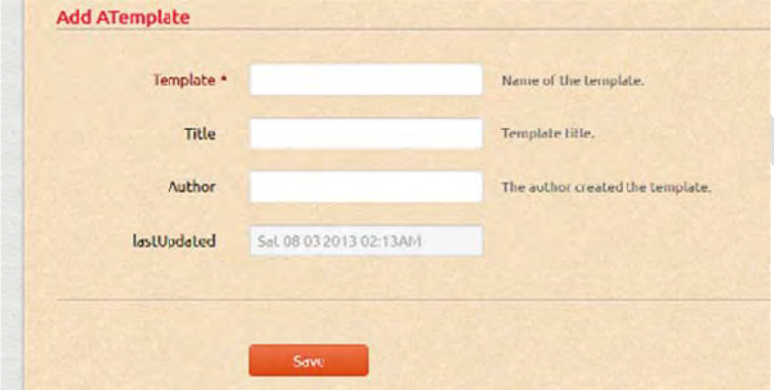
3.6 Template for Manual Assessment

Manual assessment on assets is done via the assessment survey. An asset is evaluated manually on a range of its attributes and given a rating from 1 to 5 for each one. Since each asset type has different attributes, it takes different assessment survey forms to evaluate against different assets.

You can design different assessment survey form templates for different assets in the **Assessment Templates** page. From the **Admin** page, click on the **Assessment Templates** link.

3.6.1 Add Assessment Template

On the **Assessment Templates** page, click on the **add template** button to add a new template.



Add ATemplate

Template * — Name of the template.

Title — Template title.

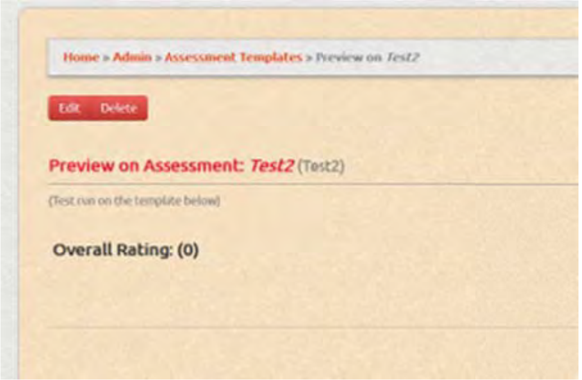
Author — The author created the template.

lastUpdated — Sat 08 03 2013 02:13AM

Type in a name for the template and fill in the rest to create the template. When created, the template is empty. The app will return to the template list. Add the template sections and attribute next when editing the template.

3.6.2 Add and Edit Assessment Template Detail

To add detail to the template, select it first from the **Assessment Templates** page to show its detail page. The template detail page shows a preview of the assessment survey. It's empty when first created, like below.

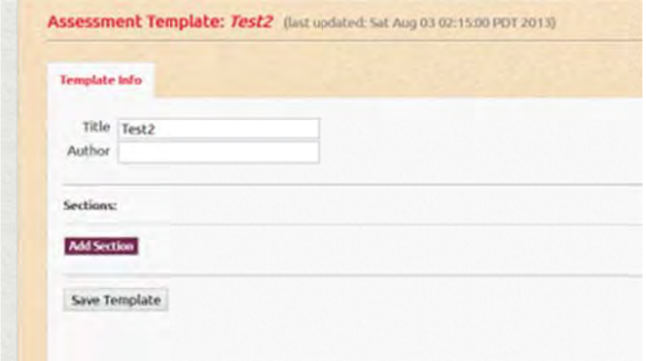


Preview on Assessment: *Test2* (Test2)

(Test run on the template below)

Overall Rating: (0)

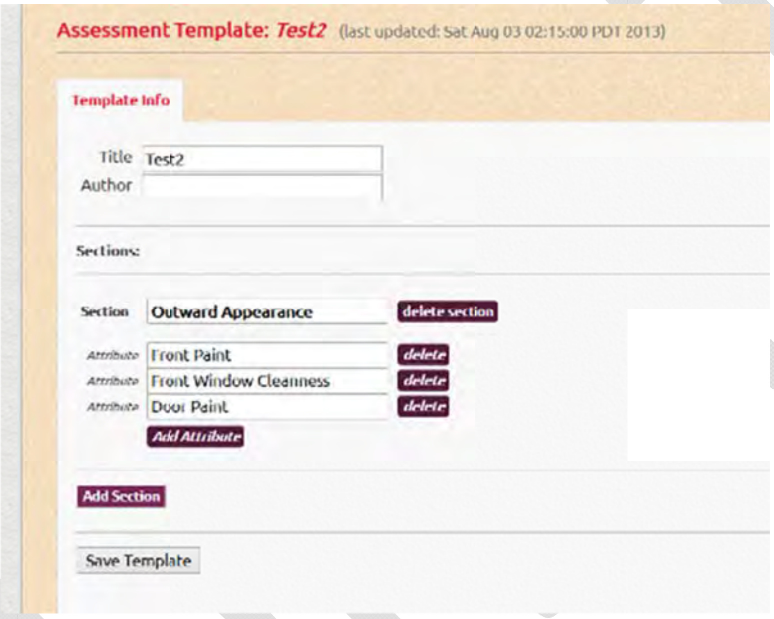
Click on **Edit** to start editing. The edit page is shown as below. It has a single **Add Section** button. The survey form is divided into a number of sections which group similar attributes together under a topic.



Assessment Template: *Test2* (last updated: Sat Aug 03 02:15:00 PDT 2013)

Title: Test2
Author:

Click on the **Add Section** button to add a new section. After a section has been created, you can add attributes under it. Each attribute will be rated from 1 to 5.



Assessment Template: *Test2* (last updated: Sat Aug 03 02:15:00 PDT 2013)

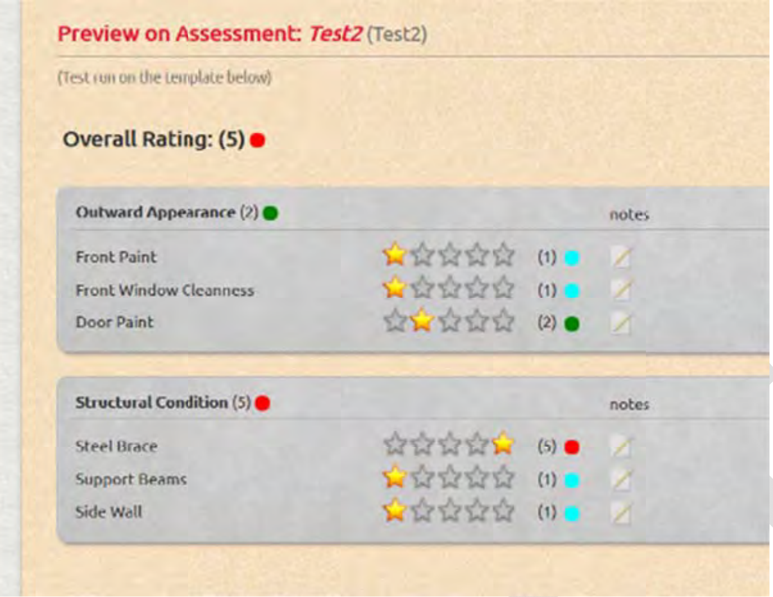
Title: Test2
Author:

Section: Outward Appearance
Attribute: Front Paint
Attribute: Front Window Cleanness
Attribute: Door Paint

3.6.3 Assessment Template Preview

Once an assessment template for survey form has been created, it can be previewed in the detail page. You can try out the survey form with some ratings. The attribute ratings will be rolled up automatically to the section rating, and the section rating will be rolled up to the overall rating.

Note: currently rating rollup is taking the worst rating to rollup, reflecting the desire to expose the worst condition of the asset to call for attention.



Preview on Assessment: *Test2* (Test2)

(Test run on the template below)

Overall Rating: (5) ●

Outward Appearance (2) ●			notes
Front Paint	★☆☆☆☆	(1) ●	
Front Window Cleanness	★☆☆☆☆	(1) ●	
Door Paint	☆★☆☆☆	(2) ●	

Structural Condition (5) ●			notes
Steel Brace	☆☆☆☆★	(5) ●	
Support Beams	★☆☆☆☆	(1) ●	
Side Wall	★☆☆☆☆	(1) ●	

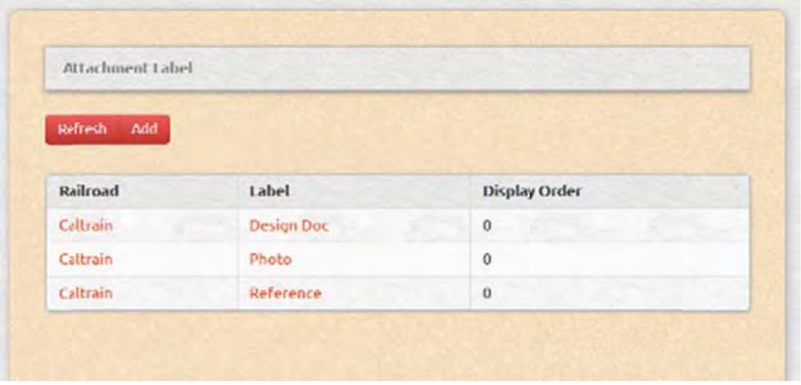
3.7 Attachment Label Management

Uploaded attachments to assets or folders can be tagged with labels for organizing them. E.g. a file can be tagged with two labels: *Photo* and *Design*. Then when the label *Photo* is selected, the file will show up. Also when the label *Design* is selected, the file will show up as well.

The webapp is flexible enough that you can define any attachment label to organize the attachments. Some examples: *Official*, *Draft*, *Work*, *Home*, *Photo*, *Design*, *Reference*, etc.

Attachment labels are defined system wide. Once defined, they will be available to tag all attachments for all assets and folders.

To add or remove attachment label, starts on the **Admin** page. Click on the **Attachment Label** link.



Attachment Label

Railroad	Label	Display Order
Caltrain	Design Doc	0
Caltrain	Photo	0
Caltrain	Reference	0

Click on **Add** to add a new label. Click on the label itself to display the label detail.

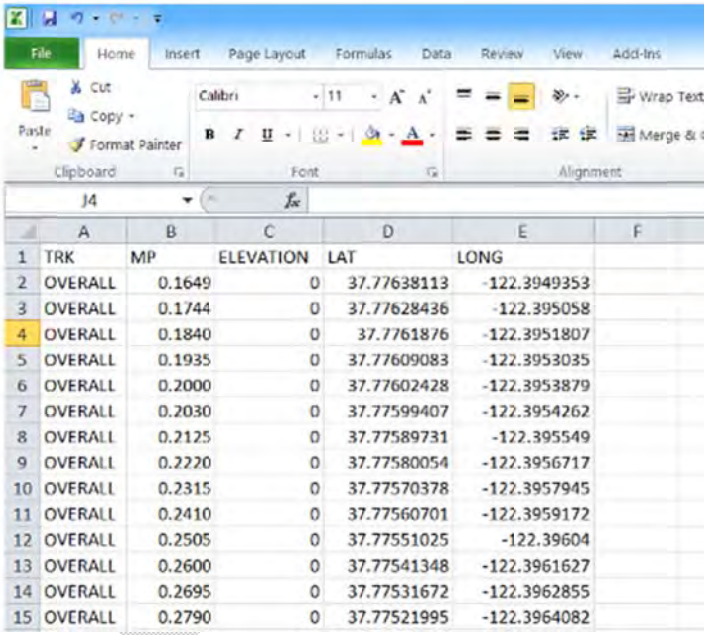
On the label detail page, you can delete it by clicking on **Delete**.

Deleting a label has no impact on the tagged attachments.

3.8 Milepost Mapping

The milepost to GIS latitude and longitude mapping allows conversion and lookup from milepost to lat/lon during asset input. It is maintained in the following steps:

- The raw milepost to lat/lon mapping data is first constructed in a CSV file. The CSV file should have the following columns
    - TRK, the track name, must be named OVERALL
    - MP, the milepost in 1/1000th of mile intervals (3rd decimal). It's rounded up to 1/100th mile intervals during import.
    - ELEVATION
    - LAT, with value up to 8th decimal.
    - LONG, with value up to 8th decimal.

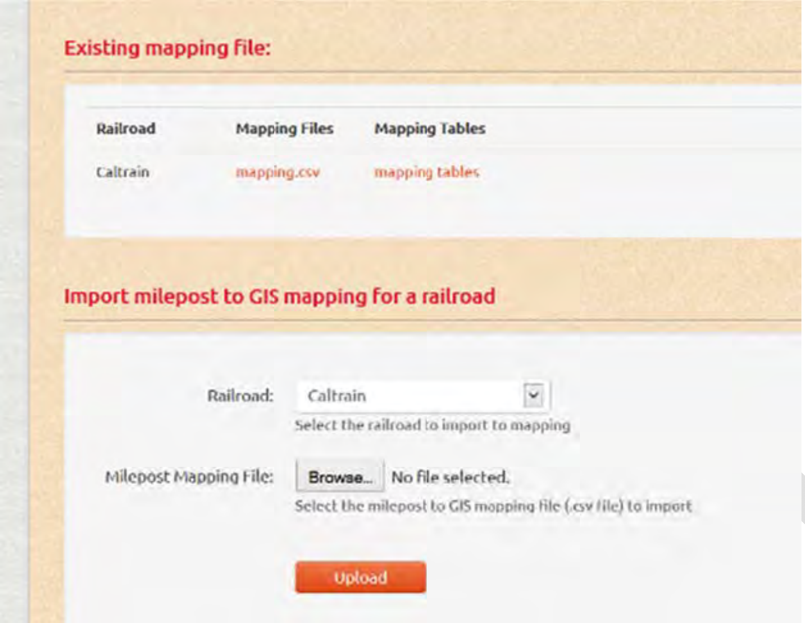


	A	B	C	D	E	F
1	TRK	MP	ELEVATION	LAT	LONG	
2	OVERALL	0.1649	0	37.77638113	-122.3949353	
3	OVERALL	0.1744	0	37.77628436	-122.395058	
4	OVERALL	0.1840	0	37.7761876	-122.3951807	
5	OVERALL	0.1935	0	37.77609083	-122.3953035	
6	OVERALL	0.2000	0	37.77602428	-122.3953879	
7	OVERALL	0.2030	0	37.77599407	-122.3954262	
8	OVERALL	0.2125	0	37.77589731	-122.395549	
9	OVERALL	0.2220	0	37.77580054	-122.3956717	
10	OVERALL	0.2315	0	37.77570378	-122.3957945	
11	OVERALL	0.2410	0	37.77560701	-122.3959172	
12	OVERALL	0.2505	0	37.77551025	-122.39604	
13	OVERALL	0.2600	0	37.77541348	-122.3961627	
14	OVERALL	0.2695	0	37.77531672	-122.3962855	
15	OVERALL	0.2790	0	37.77521995	-122.3964082	

The file should have a mapping point for every 1/1000th of mile, and covers the entire span of the railroad. Note that not just the tracks but the entire railroad. You might need to concatenate several tracks together to get an overall span of the railroad. Duplicate entry for milepost is ok. Duplicates are ignored during import.

- Import the mapping data on the Milepost Mapping page. Go there by clicking on the **Milepost Mapping** link on the **Admin** page.
- Click on **Browse** to select the mapping CSV file to import
- Click on **Upload** to start the import

The mapping data are imported, processed, and saved in a database for fast lookup. The original CSV file and the processed data can be examined. See the table **Existing mapping file**.



Existing mapping file:

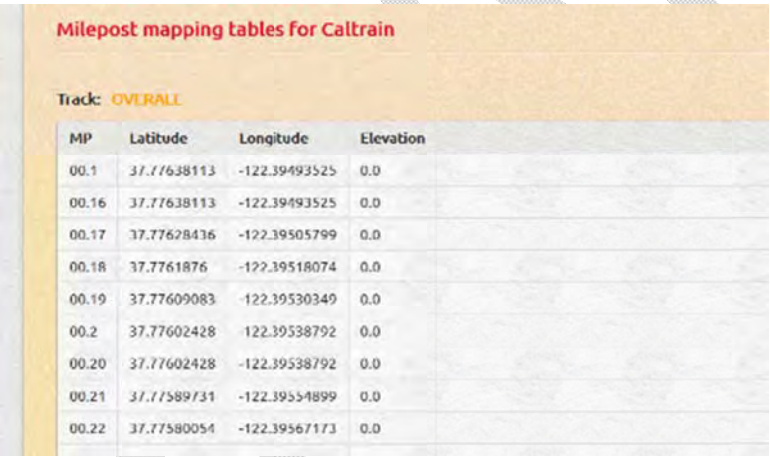
Railroad	Mapping Files	Mapping Tables
Caltrain	mapping.csv	mapping tables

Import milepost to GIS mapping for a railroad

Railroad: Caltrain
Select the railroad to import to mapping

Milepost Mapping File: Browse... No file selected.
Select the milepost to GIS mapping file (.csv file) to import

The processed mapping data are shown in the mapping tables.



Milepost mapping tables for Caltrain

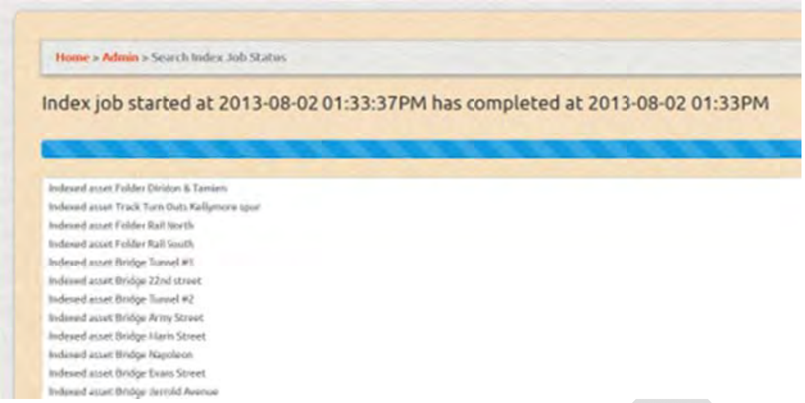
Track: OVERALL

MP	Latitude	Longitude	Elevation
00.1	37.77638113	-122.39493525	0.0
00.16	37.77638113	-122.39493525	0.0
00.17	37.77628436	-122.39505799	0.0
00.18	37.7761876	-122.39518074	0.0
00.19	37.77609083	-122.39530349	0.0
00.2	37.77602428	-122.39538792	0.0
00.20	37.77602428	-122.39538792	0.0
00.21	37.77589731	-122.39554899	0.0
00.22	37.77580054	-122.39567173	0.0

3.9 Search Index

Searches on assets and attachments rely on building a search index to index all the key words from the assets and from the attachment documents. The search index is built automatically and incrementally as new assets are added and existing ones updated. The search index can also be rebuilt from scratch. You might want to do that to clean up old entries, or if there are attachments added via back channel like copying directly to the asset's storage file directory.

To rebuild the search index, go to **Admin** page, click on the **Re-index Search Data** link.



Home > Admin > Search Index Job Status

Index job started at 2013-08-02 01:33:37PM has completed at 2013-08-02 01:33PM

```
Indexed asset Folder Diridon & Tamien
Indexed asset Track Turn Outs Kellymore spur
Indexed asset Folder Rail North
Indexed asset Folder Rail South
Indexed asset Bridge Tunnel #1
Indexed asset Bridge 22nd street
Indexed asset Bridge Tunnel #2
Indexed asset Bridge Army Street
Indexed asset Bridge Harn Street
Indexed asset Bridge Napoleon
Indexed asset Bridge Evans Street
Indexed asset Bridge Jerrold Avenue
```

3.10 Work Order Type

Work order is used for taking remediation on the defects found on an asset. Different types of asset require the tracking of different attributes; therefore, different types of work order are needed. Custom work order type let you specify custom attributes. When a new work order is created, it is created based on a work order type, where the custom attributes defined in the work order type are available to be used for the new work order.

3.10.1 Create Work Order Type

Go to the **Admin** page and click on the **Work Order Types** link. Click on **add work order type** button to create a new work order type. Work order type definition is similar to asset type definition. See asset type definition for adding attribute and attribute data types.

3.10.2 Add Export Excel Template

Work order can be exported to Excel file for external record keeping or integration with external work order systems. Export template is associated with work order type, so that when a work order is created with a work order type, it will use the associated export template automatically.

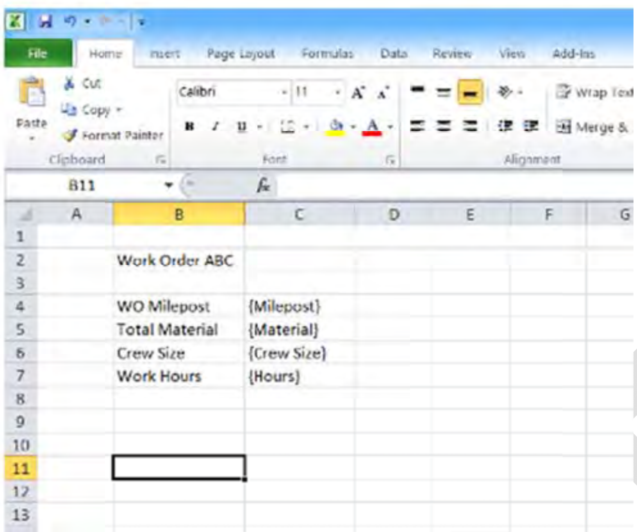
To add an export template, click on a work order type link on the **Work Order Types** page to show its detail. Click on the **upload export template** button. Select an Excel file as template and upload it.

3.10.3 Export Excel Template Format

An export template is a regular Excel file. Only .xlsx file is supported. The template can have placeholder variables that take on the work order's attribute values when exported.

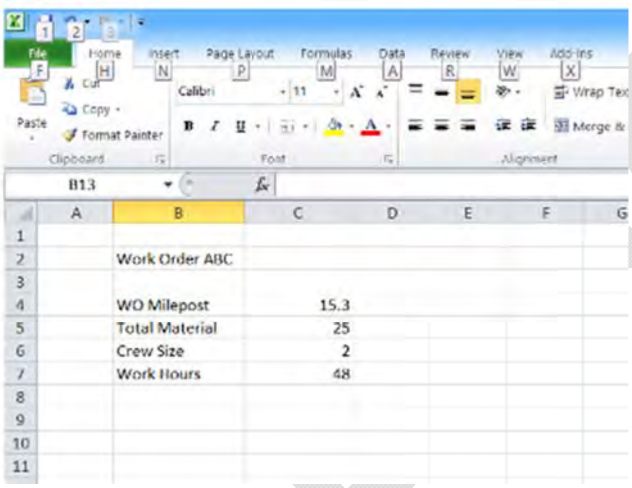
The placeholder variable is in the form of *{attribute}*, where the *attribute* is the attribute name of the work order, surrounded by {} brackets. During work order export, the placeholder variables are replaced with the actual values of the attributes in the work order.

For example, in the Excel template below, the placeholders are {Milepost}, {Material}, {Crew Size}, and {Hours}. The placeholder names are the attribute names defined in the work order type used in the work order.



	A	B	C
2		Work Order ABC	
4		WO Milepost	{Milepost}
5		Total Material	{Material}
6		Crew Size	{Crew Size}
7		Work Hours	{Hours}

When the work order is exported, the template is used. Its placeholders are replaced with the actual attribute values of the work order, as shown in the example below.



	A	B	C
2		Work Order ABC	
4		WO Milepost	15.3
5		Total Material	25
6		Crew Size	2
7		Work Hours	48

3.11 Event Logs

A lot of operations in the system are logged. Some of them are shown in the **What's New** section on the Home Page, the folder pages, and the asset detail page. The event logs displayed in **What's New** is reduced to what are relevant to the subject at hand. To see all the event logs, go to the **Admin** page and click on the **Event Logs** link.

3.12 Track Management

The **Tracks** list on the **Admin** page maintains the list of track names available to Work Order. When an work order is created, it can be assigned to a track. Its Track field gets the track selection from the list maintained in the **Tracks** list.

3.13 Project

Projects are usually related to budgeting or planning. Project let you track assets in a different dimension. An asset can be linked to a project under its Capital Project tab.

The project list itself can be defined in **Admin** page on **Projects**.