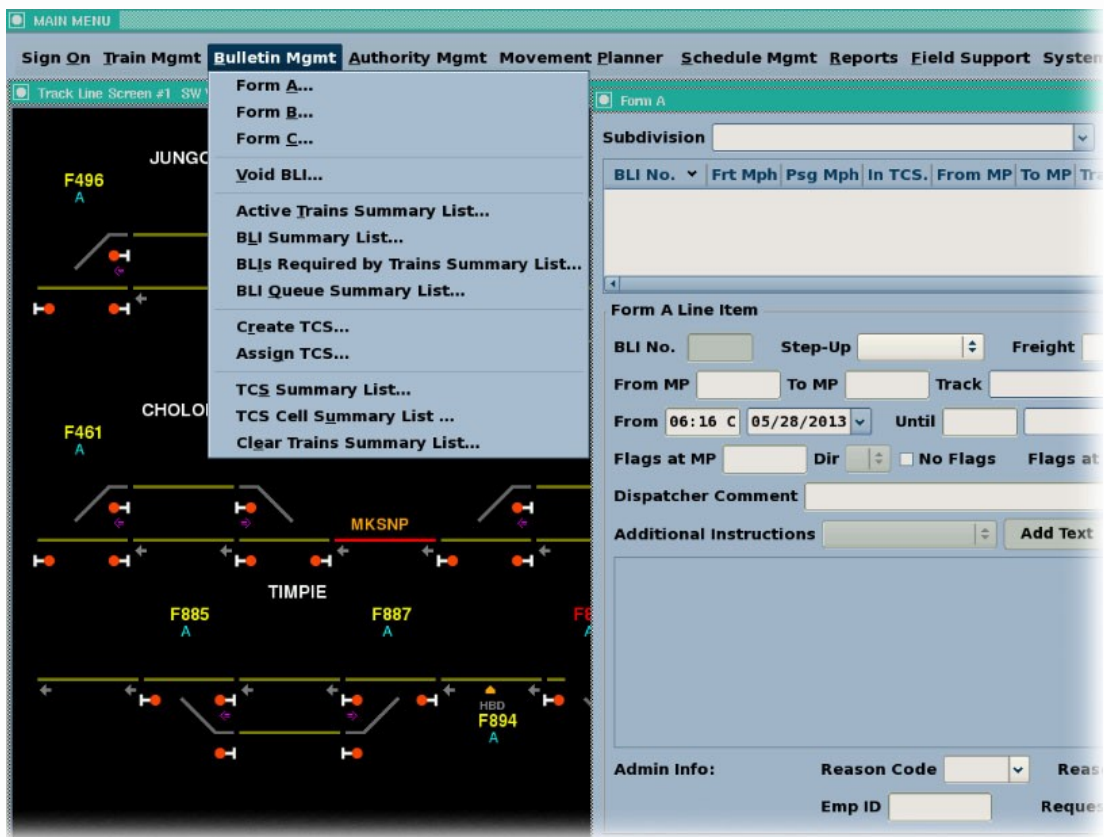


RAIL DISPATCHING SYSTEM FOR HEAVY RAIL NETWORK IN US

Datasheet

Project objective

Develop and upgrade a range of subsystems of a computed-aided dispatching system for the freight rail network to automate planning and enhance control of train movements, as well as improve safety. The upgraded system was expected to help increase throughput of a rail network, and eliminate delays caused by logistic issues, and frequent signaling failures.



Result

The advanced back-office allows for tracking, planning, and controlling of train movement, status checks of tracks and wayside devices, and the implementation of Positive Train Control for 37,000 miles. Optimized Train Planner allows for considering all the significant factors, such as speed, train weight, weather conditions, and throughput to plan the train movements and solve logistic issues in real time.

Scope of work

- ❖ Building up a core of the system based on existing subsystems
- ❖ Implementation of planning and optimization algorithms for Train Planner; preparation profiling and performance reports, and implementation of new functionality for application tasks
- ❖ Signal Maintenance subsystem design to route designated system alarms to the network
- ❖ Development of isolated simulation environment capable of running all of the production software, imitating moving trains, and field equipment
- ❖ Enhancement of Alarm, Bulletin, Authority, and Train Management Subsystems based on requirements drawn up as Software Problem Requests (SPR) for short time fixes and Scope Change Request (SCP) for additional features – predominantly, regarding messaging, communication, and visualization
- ❖ Creation of a color visual representation of a rail section for monitor of tracks, trains, authorities, and signals. GUI updates
- ❖ Implementation of data exchange to the dispatching software that sends data once a predefined event occurs
- ❖ Integration of a new system core into the existing SCADA: reengineering the system protocol of exchange messages, and developing a protocol converter for a tie-in between systems
- ❖ Continues quality assurance of both subsystem and the entire system
- ❖ Development of deployment guides

Activities

- | | |
|--|--|
| ❖ Detailed Design | ❖ Test Cases Definition and Functional Testing |
| ❖ Software Requirements Specification Analysis | ❖ UI design, implementation, and update |
| ❖ Software and Database Development | ❖ Documentation Creation |
| ❖ Unit Testing | ❖ Follow-on Support |

About the project

Technologies

- ✿ C/C++
- ✿ STL
- ✿ Qt
- ✿ Oracle PL/SQL
- ✿ Bash
- ✿ GCC
- ✿ Python
- ✿ C Shell
- ✿ Eclipse IDE
- ✿ VNC
- ✿ Tcl/Tk
- ✿ Boost
- ✿ IBM Rational Clear Case / Clear Quest
- ✿ Gprof
- ✿ Valgrind
- ✿ Insure ++
- ✿ ESX / vSphere / Lab Manager
- ✿ XML
- ✿ X Windows System
- ✿ Linux Red Hat

**Red Hat**

Project size

- ✦ 1 Technical Coordinator
- ✦ 1 Project Manager
- ✦ 13 Software Engineers
- ✦ 2 QA Engineers
- ✦ 1 Graphic Designer
- ✦ 1 Technical Writer
- ✦ 1 Technical Assistant

Duration



32+ months
March 2011 – December
2013

Platform

Linux