

Standard Development Report
for the
Advanced Transportation Controller
Application Programming Interface
Standard

December 30, 2006

The following Standard Development Report (SDR) is made in accordance with the Institute of Transportation Engineers (ITE) procedures for the Advanced Transportation Controller Application Programming Interface Standard.

1 Latest Version of the Draft Proposed Standard

Appendix I contains the user comment draft (UCD) standard “ATC API Standard v02.02, Application Programming Interface (API) Standard for the Advanced Transportation Controller (ATC).” This is the first complete version of this standard approved for user comment. There are no previously approved versions of this standard.

2 Summary Status

The API Standard is being distributed to the members of ITE and other standard development organizations (SDOs) for review. Comments from potential users of the standard are solicited. The standard is complete with user needs, software requirements and a detailed specification of the interface. Following this comment period, the comments received will be adjudicated by the API Working Group (WG), the API Standard will be updated accordingly, and the standard will be moved through the Recommended and Approved stages to Publication.

3 Status Report

The API Standard began as a work item for the ATC Joint Committee (JC) in 1998. The API Working Group, as subcommittee of the ATC JC, was formed to develop the standard. The first approach of the API WG was to create a very broad interface for ATC controller units that included functionality normally associated with a computer operating system (O/S). In 1999, the development of the API Standard was stopped so that the approach could be evaluated. An independent evaluation of the API Standard was funded by the United States Department of Transportation (USDOT) Joint Program Office (JPO). The results of the evaluation, published in May 2003, made the following recommendations:

- The transportation community might be better served if the general operating system functionality of the API Standard was built upon existing computing industry standards as opposed to attempting to define this functionality within API Standard.
- The transportation specific functionality of the API Standard should be layered “on top of” the general O/S functionality.

In accordance with these recommendations the API WG evaluated the existing standards for providing general O/S functionality and presented that information to the ATC JC. The ATC JC selected the Linux operating system as the best available solution. To provide a base level of software portability for ATC controllers, the ATC Controller Standard was modified to include the Linux operating system. The objective of the API Standard was then changed to focus on the transportation industry-specific interfaces and devices not sufficiently handled by the Linux O/S.

An important step in the development of the API Standard was the preparation of a Software Requirements Specification (SRS) in 2005. The SRS did not contain the details of the software interface to be included in the API Standard in order to focus on user needs and software requirements. In early 2006, “ATC API Software Requirements Specification (SRS) v02.04” was distributed for user comment. The comments received on the SRS were adjudicated by the API WG. The SRS and adjudicated comments became the basis for the first three major sections of the API Standard. An additional activity taking place in the 2005-2006 was the prototyping of the software defined by the API Standard. This was done to the extent project funding would allow to validate the standard prior to distribution.

On December 4, 2006, the API WG submitted API Standard v02.02 to the ATC JC for review. The ATC JC voted to approve the API Standard for distribution as a User Comment Draft on December 19, 2006.

4 Comments Listing

User comments for the 2006 distribution of “ATC API Software Requirements Specification (SRS) v02.04” are found in Appendix II.

5 Committee Objectives

The objectives of the API WG are to define software facilities and a programmatic interface that, when combined with the operating system (O/S) defined in the ATC Controller Standard, forms a universal interface for application programs designed to run on ATC controllers. This interface allows application programs to be written so that they may run on any ATC controller unit regardless of the manufacturer. It also defines a software environment that allows multiple application programs to be interoperable on a single controller unit by sharing the fixed resources of the controller. The sharable fixed resources supported include the controller’s front panel and field input/output (I/O) devices.

6 Committee Members

This standard has been developed under the oversight of the ATC Joint Committee (JC) which is made up of representatives from the American Association of State Highway and Transportation Officials (AASHTO), the Institute of Transportation Engineers (ITE), and the National Electrical Manufacturers Association (NEMA). The work in developing this standard was performed by the ATC API Working Group (WG), a technical subcommittee of the ATC JC.

ATC API Working Group

Ralph W. Boaz, Pillar Consulting
George Chen, Los Angeles Department of Transportation
Jarrid Gross, E-VIEWS Safety Systems
Herasmo Iniguez, California Department of Transportation
Kyle Irvin, Telvent Farradyne
Ron Johnson, Harris County, Texas
Kenneth Montgomery, Georgia Department of Transportation
Eric Raamot, Econolite Control Systems
Peter Ragsdale, US Traffic
Robert Rausch, TransCore
Douglas Tarico, Siemens ITS
John Thai, City of Anaheim
Andy Zhang, Wapiti Micro Systems

ATC Joint Committee

Douglas Acker, McCain Traffic
Gerry de Camp, Gerry de Camp and Associates
Gary Duncan, Econolite Control Products
Scott Evans, Eberle Design
Dave Holstein, Ohio Department of Transportation
Al Kosik, Texas Department of Transportation
Andrew Mao, Harris County, Texas
Dave Miller, Siemens ITS
Kenneth Montgomery, Georgia Department of Transportation
Tim O'Leary, Peek Traffic
Guillermo Ramos, New York State Department of Transportation
Robert Rausch, TransCore
Robert Russo, Safetran
Ed Seymour, Texas Transportation Institute
Martha Styer, California Department of Transportation
Mohamed Talas, New York City Department of Transportation
John Thai, City of Anaheim

7 Other Material of Interest

The documents listed below are consistent with those listed within the API Standard. Newer versions with minor revisions may be available.

“ATC Controller Standard Revision 5.2a”, ATC JC, January 24, 2006. Available from the Institute of Transportation Engineers.

“ATC Standard for the Type 2070 Controller v02.03”, ATC JC, March 12, 2004. Available from the Institute of Transportation Engineers.

“Intelligent Transportation System (ITS) Standard Specification for Roadside Cabinets v01.02.17a”, ATC JC, March 24, 2006. Available from the Institute of Transportation Engineers.

“ATC API Software Requirements Specification (SRS) v02.04”, December 9, 2005. Available from the Institute of Transportation Engineers.

“Proof-of-Concept Evaluation: Application Programming Interface Standard for the Advanced Transportation Controller, Phase 1, ATC API Evaluation Report”, Vanteon, May 9, 2003. Available from the Institute of Transportation Engineers.

8 Declaration Regarding Other Known National and International Standards

This statement confirms that other known national and international standards have been examined with regard to harmonization and duplication of content, and no significant conflicts with another known standard have been identified.

9 Abstract of the Standard

Purpose

The Advanced Transportation Controller (ATC) Standards are intended to provide an open architecture hardware and software platform that can support a wide variety of Intelligent Transportation Systems (ITS) applications including traffic management, safety, security and other applications. The ATC Standards are being developed under the direction of the ATC Joint Committee (JC) which is made up of representatives from the American Association of State Highway and Transportation Officials (AASHTO), the Institute of Transportation Engineers (ITE), and the National Electrical Manufacturers Association (NEMA).

This document defines a software interface for application programs intended to operate on ATC controller units. It has been prepared by the ATC Application Programming Interface (API) Working Group (WG), a technical subcommittee of the ATC JC. It establishes a common understanding of the user needs, requirements, and specification of the interface for:

- a) The local, state, and federal transportation agencies who specify ATC equipment;
- b) The software developers, consultants, and manufacturers who develop application programs for ATC equipment;
- c) The public who benefits in the application programs that run on ATC equipment and directly or indirectly pays for these products.

Overview

The ATC Controller Standard defines a controller that can grow with technology. It is made up of a central processing unit (CPU), an operating system (O/S), memory, external and internal interfaces, and other associated hardware necessary to create an embedded transportation computing platform. The goal of the interface described in this standard is to define a software platform that, when combined with the ATC O/S, forms a universal interface for application programs. This interface allows application programs to be written so that they may run on any ATC controller unit regardless of the manufacturer. It also defines a software environment that allows multiple application programs to be interoperable on a single controller unit by sharing the fixed resources of the controller. The sharable fixed resources supported include the controller’s front panel and field input/output (I/O) devices. The API Standard specifies the interface. Software developed in compliance to the API Standard is known as the ATC Application Programming Interface (API). Figure 1 illustrates the layered architecture of the ATC software.

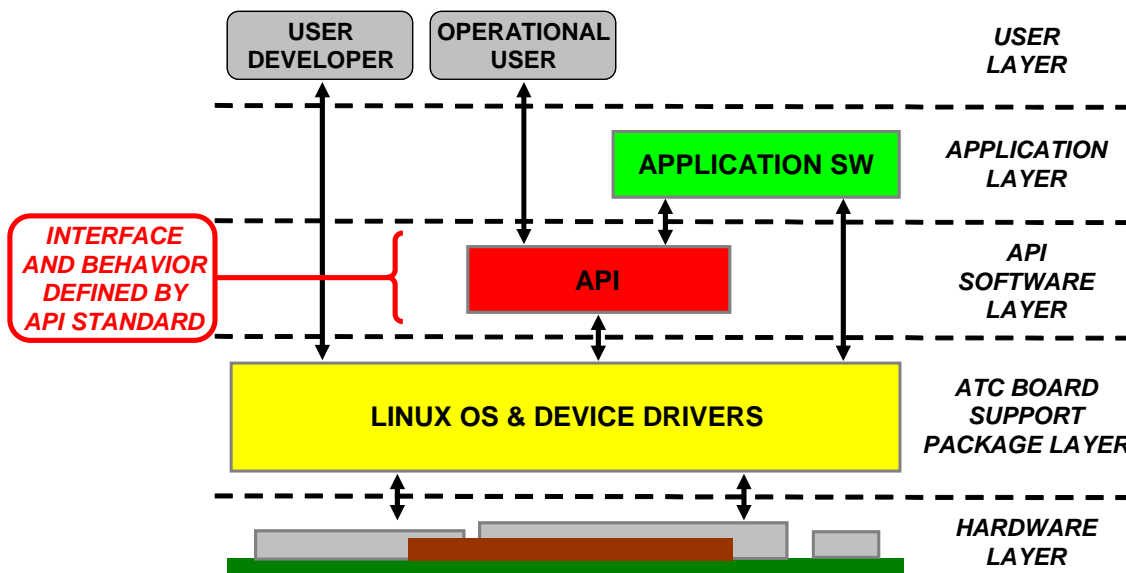


Figure 1. ATC software layered organization.

Document Organization

This standard is made up of four sections, appendixes and an index. Section 1, "Introduction", provides an overview of the entire document. Section 2, "Overall Description", provides background information and the user needs for the requirements defined in the subsequent section. Section 3, "Specific Requirements", defines the requirements that must be satisfied by the ATC API. Section 4, "Application Programming Interface", specifies the ATC API. Appendix A of the standard is a traceability matrix showing the relationship between the user needs, the software requirements and the functions of the ATC API.

APPENDIXES

Appendix I
ATC API Standard v02.02

Appendix II

User Comments for ATC API Software Requirements Specification (SRS) v02.04