ATC API WORKING GROUP MEETING
WEDNESDAY-FRIDAY, 08/12/09-08/14/09, SAN ANTONIO, TX

ATTENDEES
- Douglas Tarico, Siemens *
- George Chen, LADOT *
- Bill Brownlow, AASHTO
- Doug Crawford, Intelight
- Neal Smith, NYSDOT *
- Peter Skweres, Mn/DOT *
- Bob Rausch, TransCore *
- Siva Narla, ITE
- Steve Alonge, Noblis
- Robert Helliar, Econolite *
- Herasmo Iniguez, Caltrans *
- Ralph Boaz, Pillar Consulting *
- Mike Green, Siemens (Wednesday, Thursday)
- Mark Clark, Siemens (Wednesday, Thursday)
- Peter Ragsdale, Consultant * (via telephone Thursday, Friday)
- Kyle Irvin, Telvent * (Wednesday)
- Bruce Winner, Telegra (Wednesday)
- Henry Wickes, TxDOT (Wednesday)

VOTING MEMBERS NOT PRESENT
- Jarrid Gross, E-VIEWS Safety Systems *
- Zhang, Wapiti Microsystems *
- John Thai, City of Anaheim *
- Ron Johnson, Harris County, TX *
- Ken Montgomery, GADOT *

[* indicates a voting member of the Working Group.]
[Action items are assigned using capitalized last names in brackets.]

GENERAL
- Introductions
- Meeting Conduct Guidance
- Application Programming Interface (API) Working Group (WG) Membership
  - Discussed organization of the Advanced Transportation Controller (ATC) standards development.
  - Discussed standards, ATC Joint Committee, SDOs, public and private sector participation.
  - George Chen of LADOT and Douglas Tarico of Siemens are the Co-Chairs of the API WG.
  - API WG currently has 15 voting members. 7 Public Sector and 8 Private Sector. See names with asterisks at the beginning of these notes.
  - Discussed API Quick Response Group (QRG). This group typically has a 2-3 days to review or respond to items as a representation of the entire WG. They do not make final decisions but are here to assist the chairs and the project team to keep the project on track.
  - Members of the API QRG:
    - Douglas Tarico
    - George Chen
    - Herasmo Iniguez
    - Jarrid Gross
    - Neal Smith (new member)
    - Peter Ragsdale
    - Robert Helliar
  - Additions/Modifications to the Agenda
- Time permitting may discuss corrections to standard.

- **API Project Status**
  - API Validation Suite (aka API Val Suite, API VS). Purpose is to provide a program or group of programs that validate whether or not an engine board has API Software that conforms to the API Standard.
  - API Standard Version 2 (aka API V2). The main thrust of this activity is to develop an ATC Configuration Manager capability including API functions and a Configuration Window. The Configuration Window provides ATC users a standardized interface on all ATC units for setting selected system wide parameters and view various system configuration information. Note that system in this context refers to the ATC controller unit.
  - API Standard Maintenance. Corrections to the standard. Small funding support for this.
  - Ralph Boaz is the Project Manager, Systems Engineer, and Technical Consultant for the all of the activities.
  - Software development for the API VS will be performed by NFocal. Specifically, Steve Kapp and Tom Gauger. Tom Gauger worked on the API Standard V1 as part of Vanteon, Corp.

- **API V2 Deliverables:**
  1) API Software Test Plan including
     a) Test Design Specifications (TDS)
     b) Test Case Specifications (TCS)
     c) Test Procedure Specifications (TPS)
  2) Concept of Operations (ConOps) for the API VS Software
  3) Software Requirements Specification (SRS) for the API VS Software
  4) Software Design Descriptions (SDD) for the API VS Software
  5) API Validation Suite Software (API VS or API VS Software)

- **API V2 Deliverables:**
  1) Concept of Operations (ConOps) Content for API Standard V2
  2) Requirements Content for API Standard V2
  3) Design Content for API Standard V2
  4) API Standard V2

- **API Standard Maintenance.** Corrections will be made as part of the API V2 project.

**API VALIDATION SUITE TEST PLAN REVIEW**

- Purpose of the test plan and subsequent TDSs, TCSs and TPSs are 1) capture testing needs and requirements from a testing point of view and 2) to provide the input necessary for the ConOps and SRS for the API VS software.
- Steve Alonge suggested we also look at the new IEEE 829-2008 Test Plan, etc. for additional help on the test documents. [BOAZ]
- Discussed having a formal statement of what the API and Validation Suite does and does not do. Possibly put in recommendations to agencies that their procurements provide versions of their various application programs that are to run on their platform. Alternatively, they may also want to require that a software program run on controllers from multiple vendors. Needs to go both ways. Bob Rausch to provide a draft. [RAUSCH]
- Discussed the various test documents. Ralph Boaz provided summaries from IEEE Std 829-1998. See also Figure 2.

**Test Plan**

To prescribe the scope, approach, resources, and schedule of the testing activities. To identify the items being tested, the features to be tested, the testing tasks to be performed, the personnel responsible for each task, and the risks associated with this plan.

- Test plan identifier;
- Introduction;
- Test items;
- Features to be tested;
- Features not to be tested;
- Approach;
g) Item pass/fail criteria;
h) Suspension criteria and resumption requirements;
i) Test deliverables;
j) Testing tasks;
k) Environmental needs;
l) Responsibilities;
m) Staffing and training needs;
n) Schedule;
o) Risks and contingencies;
p) Approvals.

Link requirement/design item to a test case.

**Test Design Specification**

To specify refinements of the test approach and to identify the features to be tested by this design and its associated tests.

- a) Test design specification identifier;
- b) Features to be tested;
- c) Approach refinements;
- d) Test identification;
- e) Feature pass/fail criteria.

Comment – Design a test linking requirement and or design items to test cases.

**Test Case Specification**

To define a test case identified by a test design specification.

- a) Test case specification identifier;
- b) Test items;
- c) Input specifications;
- d) Output specifications;
- e) Environmental needs;
- f) Special procedural requirements;
- g) Intercase dependencies.

**Test Procedure Specification**

To specify the steps for executing a set of test cases or, more generally, the steps used to analyze a software item in order to evaluate a set of features.

- a) Test procedure specification identifier.
- b) Purpose;
- c) Special requirements;
- d) Procedure steps.

- Discussed the meaning of "test item" in the IEEE Std 829-1998. Test item is simply "an item to be tested." For example, a test item at a high level (Test Plan) may refer to a group such as a library, whereas a test item at the level of a test case may be a particular function.
- Note the different roles of the documents. Although we expect that the test cases will not overlap, it is not necessarily the case in other projects.
- Ralph stepped through the API Test Plan. Additions to be made include:
  - A discussion of what a user may have to do to get the API VS software to run on different controllers in Section 1.2 or 1.3. Possibly add other statements on what the API VS does or does not do per discussion note listed earlier in this section of the notes.
  - Add statement that API WG requests feedback on the API Standard when the API VS is used to transduce data in Section 8.
  - It was suggested that procurement guidance be added to the Test Plan and/or Standard. [BOAZ]
- Section on testing environment led to potential architectures for testing the API. See Figure 1 for the arrangement the group settled upon as a minimum. It was discussed that: 1) the API VS could operate completely on the engine board with a test fixture; 2) some combination of a PC, an engine board, and test fixture; or 3) options 1 and 2 but using a controller. Need to weigh the value of specifying multiple ways vs. a way to cut down on duplication and complexity to the contractors. [BOAZ, API QRG]
• Discussed that the API VS could be pre-loaded and executable on the ATC Engine Board at the time of purchase. It could also be added by the user. In this case the user would be required to compile and link the API VS for the target processor on the ATC Engine Board.

Figure 1. Draft diagram of test environment.

• Doug and Ralph to work up draft architectures. [BOAZ, TARICO]
• Discussed the relationships of the test documentation to the API libraries. It is anticipated that there will be, at a minimum, three TDSs, one for each API library. There could be additional TDSs required. See Figure 2.
• The project plan has the Test Plan, TDSs and TCSs prepared by the API WG. The TPSs will largely be prepared by the contractor.
Figure 2. Relationship of test documents in IEEE 829-1998.

API VALIDATION SUITE USER NEEDS DISCUSSION

- Discussed that there should eventually be training offered on the API and the Validation Suite. Possibly in conjunction with the ATC Controller and the ITS Cabinet V2. Suggest that AASHTO, ITE and NEMA consider a Webinar. [BOAZ/NARLA]
- Discussed copyright concerns on the API VS software. Users want to be able to freely use it. Perhaps incorporate it in other test processes.
- There was consensus that, to be effective, the API WG (likely through a special subgroup) would need to maintain the source code as its purpose is to demonstrate conformance to the API Standard.
- Possible license options need to be explored. Regardless of license model chosen, there needs to be support (Financial? Institutional? Volunteer?) and commitment to carry it out. It was recommended that the SDOs begin looking into this ASAP since it may take a long time to get the details sorted out between all three organizations. [ITE, AASHTO, NEMA]
- Discussed whether the Validation Suite will include API V2 functionality. This is highly desirable but not committed to. The API WG and project team will try to accommodate the API V2 functionality. [API WG, API PROJECT TEAM]
- It is anticipated that at least one vendor will be providing an API library to test although it is not guaranteed that a working version will exist. Whatever functionality is not provided will have to be simulated via stub functions created during the API VS project. An invitation should be put out to any
vendor to provide a library to be a part of the project. The project leadership will need to reserve right to limit number of participants in order to control project costs. [BOAZ, NARLA, CHEN, TARICO]

- There was a discussion of a possible pilot projects with a public agency. The API WG members should solicit state and local agencies for possible pilot project opportunities once this has been decided. [API QRG, API WG]
- It was requested that the prototype API software from the API V1 development be posted on the API website. [BOAZ, NARLA]
- It was requested that an ITS Cabinet Testing capability be started. This was already noted in the ITS Cabinet WG. [BOAZ, NARLA]
- Testing needs to include applications that are not in “focus” (reference to the application displayed on the front panel at the time). Design implication: in order for the API VS to support automated testing, the VS will need to include some form of front panel emulation to build up buffered screen images which can be compared against expected results.
- Discussed performance tests for the API VS. Since the API Standard doesn’t have performance tests for the API software (other than messaging), it was decided not to include them. If performance requirements are added to the API Standard, it would be good to get them into the API VS Software. It was suggested that performance requirements would be a useful addition to future versions of the API Standard.
- User needs identified by the API WG for the API VS ConOps:
  1. The user needs the API VS source code and documentation to be available to anyone. **Users will want to be able to use it in their organizations for testing and they may not want to use an executable version provided by a manufacturer. In this case, they will need to compile and load it themselves for the particular controller under test.**
  2. The user needs the API VS to have a clearly defined license model which allows users to freely use it. **After obtaining the API VS, users should not be required to ask permission to use it.**
  3. The user needs the API VS to be operational in the test environments (configurations) as indicated by the ATC Test Plan. **There are multiple test configurations that are identified in the Test Plan.**
  4. The user needs the API VS to provide a pass/fail indication of conformance of the API Software to the API Standard. **Users must have a repeatable method to validate that they have conforming operational API Software.**
  5. The user needs the API VS to provide a log or trace of the tests performed and the results of each test and/or step. **Users may need additional information to diagnose anomalies in the API Software.**
  6. The user needs the API VS to test the API Software for completeness of the API Software with respect to the API Standard. **This will validate that each API function is present and that its arguments conform to the standard.**
  7. The user needs the API VS to test the API Software for correctness of operation. **Each function is to be a part of at least one integrated test that validates that behavior of the API function.**
  8. The user needs the API VS to verify that the API supports multiple and simultaneous applications. **Rationale TBD…**
  9. The user needs the API VS to test the Front Panel Manager Functions of the API Software. **Rationale TBD…**
  10. The user needs the API VS to test the Front Panel Manager Window of the API Software. **Rationale TBD…**
  11. The user needs the API VS to test the Field I/O Manager Functions of the API Software. **Rationale TBD…**
  12. The user needs the API VS to test the Utility Functions of the API Software. **Rationale TBD…**
  13. The user needs the API VS to test the Configuration Window of the API Software. **Rationale TBD…**

- Some details were discussed for either the requirements or design stage of the API VS project. Would like the trace or log to contain things like:
  - Library, function and arguments on the call and return values
  - Time stamps
The test case being executed

- Want results in ASCII text files.
- Discussed configurable outputs but the consensus was that the user will likely want a yes/no result or all of the details he can get.

API STANDARD VERSION 2 USER NEEDS DISCUSSION

- Discussed goals from the task order.
  - "Use a Systems Engineering Process to establish needs, requirements, and design content of an ATC Configuration Manager for the ATC API Standard Version 2. The function of this capability is to provide a uniform interface for ATC controller units to set/manage/view system wide parameters (analogous to the Control Panel Utility of Windows Operating Systems). This may include Time, Ethernet Ports, Device Driver Information, Operating System Version Information, API Library Information, and Host EEPROM information as described in the ATC Standard 5.2b, Section B.3.3. It should be noted that the design content for this activity covers both a user interface and associated functions in a fashion consistent with the existing content of API Standard."
- Raised questions on whether the concept was practical to have at all. Concluded that it was desirable to have it, otherwise application programs will continue to do these types of things in proprietary fashions. With this capability, new applications programs will not have to include the capability in their programs. Resources can be managed for multiple applications with less chances of stepping on each other.
- Discussed that possibly we should go to the extent of bringing up a shell and/or allowing the order of loading applications to be done by a technician in the field. Consensus was that this level of control requires a level of integration that should not be done in the field. Also, the person doing such a task would have the knowledge and capability to bring up a shell to use Linux commands to do such things.
- Consensus that the Configuration Window should have system level items that are critical even when there is no application software on the box. Ex. The need to set up communications with a controller without application software on so that an application program can be loaded or updated on a controller that does not contain existing application software.
- Consensus was that the API would still support (or not prohibit) individual programs from performing these actions for legacy purposes.
- User needs:
  1) The user needs the ability to set the time from the configuration window with all of the time options of the API library. This is to provide a consistent method for users to set the time on an ATC controller unit without running application program using a Linux shell. This helps to diminish duplication of this function across application programs. Users need to be able provide all of the time options for intersections not connected to a central system.
  2) The user needs the ability to set network parameters IP Address, Gateway, Subnet Mask, Host Name, DHCP, and DNS from the Configuration Window. This is to provide a consistent method for ATC users to set the network parameters. This helps to diminish duplication of this function across application programs. These parameters may need to be set in order to load an application program onto an ATC controller unit.
  3) The user needs the ability to activate and deactivate network services FTP, SSH, SCP and Telnet from the Configuration Window. This is to provide a consistent method for ATC users to use the network services. These parameters may need to be set in order to load an application program onto an ATC controller unit.
  4) The user needs system wide information displayed in the Configuration Window including the API Library Vendor, API Library Version Numbers, memory on the ATC Engine Board, and selected system information provided by the Linux "uname" function. This information is necessary for users to maintain their ATC controller units.
  5) The user needs the complete contents of the Host EEPROM on the ATC Controller Unit displayed in the Configuration Window. This information is necessary for users to maintain their ATC controller units.
6) The user needs the Configuration Window to be easy to use. *The operation of the Configuration Window should be intuitive and have default values set so that the user is not required to set all fields unless he needs to.*

7) The user needs the Configuration Window to be extensible. *This means that there will be a method for application programs to display information in the Configuration Window beyond those items specifically identified in the API Standard. This is so that new capabilities could be developed and added to the Configuration Window.*

- Need to determine if users need configurable restrictions on this Configuration Window capability. [BOAZ, API QRG]
- Need to determine if users need some special restrictions on Telnet. [BOAZ, API QRG]
- Need to validate that the information to be displayed by the Configuration Window is possible through the Board Support Package defined by the ATC Controller Standard. [RAGSDALE, CRAWFORD]

**PROJECT/PROGRAM DISCUSSION ITEMS**

- Discussed schedules of both the API VS and API V2 projects. We are trying to maximize our meetings and running the projects in parallel. It was discussed that we will try to accelerate the API V2 project so that we might be able to capture more capability in the API VS.
  - API VS project has been approved but the schedule needs to be revised based where we are at. [BOAZ]
  - API V2 project is not yet approved but indications are that it should go through. The project schedule needs to be updated also. [BOAZ]
  - The API VS project schedule needs to also have the following modifications:
    - Need to modify the API VS project schedule to show more involvement of the API WG in the implementation phase.
    - Need to add the deliverables of the Test Plan, TDSs, TCSs, TPSs and User Guide to the API VS project schedule.
    - Need to add acceptance test to API VS project. [BOAZ]

**ADDITIONAL ITEMS**

- Postponed discussion on corrections to standard. Need to collect and analyze. Some identified include:
  - Approximately eight corrections from Siemens.
  - Possible correction to the architecture diagrams in the current standard.
  - User needs for date and time should be expanded to support the requirements and design that is currently in the standard.
  - Need to add a conformance statement to standard.
  - Consider adding performance requirements to the API Standard.
  - Consider having a fixed maximum number of application programs the API will support. [BOAZ, API QRG]
- Need to solicit further input from the SDOs on the current API Standard. [BOAZ, NARLA]
- ATC Controller Standard is undergoing maintenance. Need to recommend any corrections the ATC Controller Standard ASAP. [API WG]
  - Verify that the kernel level interface to the serial drivers is put in.
  - Any of the items in the configuration window that may need supporting functions in the ATC Controller Standard.
  - Recommend the inclusion of FTP into the ATC Controller Standard.
- Need to find out why the ATC distribution is sending out duplicate emails. [BOAZ, NARLA]
- Consider adding a task manager type of capability to display active tasks and their memory and CPU utilization.
  - A task manager was suggested as an addition to the API.
- Consider using Google Docs to get feedback from the API WG on documents. Note – Neal Smith said NYSDOT prohibited the use of Google Docs. [BOAZ]
- Peter Ragsdale requested that some of the material sent out to the ATC distribution list be resent to him. [NARLA]
• Need to put out invitation to anyone building an API Library to be a part of the API VS Project. [BOAZ, NARLA]
• Need to make sure the contractors have plenty of support from any manufacturer providing a library to use in development in the API VS software.

NEXT MEETINGS/TELECONFERENCES
• To be determined.

TASK ASSIGNMENTS/ACTION ITEMS
• Steve Alonge suggested we also look at the new IEEE 829-2008 Test Plan, etc. for additional help on the test documents. [BOAZ]
• Discussed having a formal statement of what the API and Validation Suite does and does not do. Possibly put in recommendations to agencies that you their procurements provide versions of their various application programs that are to run on their platform. Alternatively, they also may want to require that a software program run on controllers from multiple vendors. Needs to go both ways. Bob Rausch to provide a draft. [RAUSCH]
• Section on testing environment led to potential architectures for testing the API. Need to weigh the value of specifying multiple ways vs. a way to cut down on duplication and complexity to the contractors. [BOAZ, API QRG]
• Doug and Ralph to work up draft architectures. [BOAZ, TARICO]
• Discussed that there should eventually be training offered on the API and the Validation Suite. Possibly in conjunction with the ATC Controller and the ITS Cabinet V2. Suggest that AASHTO, ITE and NEMA consider a Webinar. [BOAZ/NARLA]
• Possible license options need to be explored. Regardless of license model chosen, there needs to be support (Financial? Institutional? Volunteer?) and commitment to carry it out. It was recommended that the SDOs begin looking into this ASAP since it may take a long time to get the details sorted out between all three organizations. [ITE, AASHTO, NEMA]
• Discussed whether the Validation Suite will include API V2 functionality. This is highly desirable but not committed to. The API WG and project team will try to accommodate the API V2 functionality. [API WG, API PROJECT TEAM]
• It is anticipated that at least one vendor will be providing an API library to test. An invitation should be put out to any vendor to provide a library to be a part of the project. The project leadership will need to reserve right to limit number of participants in order to control project costs. [BOAZ, NARLA, CHEN, TARICO]
• There was a discussion of a possible pilot projects with a public agency. The API WG members should solicit state and local agencies for possible pilot project opportunities once this has been decided. [API QRG, API WG]
• It was requested that the prototype API software from the API V1 development be posted on the API website. [BOAZ, NARLA]
• It was requested that an ITS Cabinet Testing capability be started. This was already noted in the ITS Cabinet WG. [BOAZ, NARLA]
• Need to determine if users need configurable restrictions on this Configuration Window capability. [BOAZ, API QRG]
• Need to determine if users need some special restrictions on Telnet. [BOAZ, API QRG]
• Need to validate that the information to be displayed by the Configuration Window is possible through the Board Support Package defined by the ATC Controller Standard. [RAGSDALE, CRAWFORD]
• API VS project has been approved but the schedule needs to be revised based where we are at. [BOAZ]
• API V2 project is not yet approved but indications are that it should go through. The project schedule needs to be updated also. [BOAZ]
• The API VS project schedule needs to also have the following modifications:
  − Need to modify the API VS project schedule to show more involvement of the API WG in the implementation phase.
  − Need to add the deliverables of the Test Plan, TDSs, TCSs, TPSs and User Guide to the API VS project schedule.
- Need to add acceptance test to API VS project. [BOAZ]
- Postponed discussion on corrections to standard. Need to collect and analyze. [BOAZ, API QRG]
- Need to solicit further input from the SDOs on the current API Standard. [BOAZ, NARLA]
- ATC Controller Standard is undergoing maintenance. Need to recommend any corrections the ATC Controller Standard ASAP. [API WG]
- Need to find out why the ATC distribution is sending out duplicate emails. [BOAZ, NARLA]
- Consider using Google Docs to get feedback from the API WG on documents. [BOAZ]
- Peter Ragsdale requested that some of the material sent out to the ATC distribution list be resent to him. [NARLA]
- Need to put out invitation to anyone building an API Library to be a part of the API VS Project. [BOAZ, NARLA] RALPH