ITE 2015 Fall Conference, Tucson, Arizona

ATC Application Programming Interface Reference Implementation Demonstration and Training Workshop
APIRI Demonstration and Training Workshop

- Advanced Transportation Controller (ATC) Standard v06 (ATC 5201 v06)
- Application Programming Interface (API) Standard v02 (ATC 5401 v02)
Basic Transportation Field Cabinet System (TFCS) Components

- Housing
- Inputs
- Controller
- Outputs
- Monitoring
- Power Supply
- Internal Bus

Graphics: Ralph W. Boaz
Basic TFCS Operation

Inputs

Controller

Outputs

Monitoring

Field Sensors

Field Displays

Controller/Monitor Communications Used in NEMA TS 2 and ITS Cabinets

Graphics: Ralph W. Boaz
Architectures for Arterial Management

TFCSs Under Central Control

TFCSs in a Peer-to-Peer System

TFCSs in a Closed-Loop System and Under Central Control

TFCSs in a Closed-Loop System

* Field Management Station

Graphics: Ralph W. Boaz
Evolution of Transportation Controller Equipment

- **1940s**: Electro-mechanical
- **1976**: NEMA TS 1
- **1980s**: Model 33X
- **1992**: NEMA TS 2
- **1998**: TS 2 with NTCIP
- **2006**: ITS Cabinet v01

Graphics and Photos: Ralph W. Boaz
Traditional Controller Equipment

- Equipment runs one application program which uses all of the resources of the TFCS

- Model 170 and most Model 2070 controllers require particular outdated processors
  - Limits opportunity to use new technology

- Traditional NEMA TS 1 and TS 2 controllers are not open architecture
  - Software available from the original manufacturer only

- Doing anything other than originally intended is “outside the box”
Problem with Traditional Transportation Controller Architectures

INCREASING CAPABILITY

CONNECTED VEHICLE / V2I

ADAPTIVE CONTROL


Graphics: Ralph W. Boaz
Arterial Management Applications

- Traffic Signal Control / Traffic Management
  - Center-to-Field, Field Management Stations (FMS)
- Adaptive Control
- Transit/Light Rail Priority
- Emergency Management
  - Emergency Vehicle Priority/Preemption
  - Evacuation Routing
- Data Collection and Distribution
- Lane Use
- Connected Vehicle (CV) Applications
  - Safety, Eco Driving, Platoon Management
Arterial Management Applications (cont.)

- Advanced Traveler Information Systems
- Ramp Metering
- Access Control / Parking
- Others
Purpose of the ATC Family of Standards

- Provide a general purpose field computing platform for transportation applications that is:
  - Open architecture
  - Modular
  - Multi-tasking / Multi-application
  - Can grow with technology
  - Upgrade legacy TFCSs
Key Elements of the ATC 5201 Architecture

- Based on an “Engine Board” concept
- Engine boards have identical pinout – future boards may plug into existing host boards
- Computational capability can grow with technology
- Uses Linux operating system
  - Open source, multi-tasking, multi-application
- Mechanical requirements only for physical interfaces
- Works with all major transportation field cabinet system standards and specifications
ATC Engine Board Concept

Graphics: Ralph W. Boaz
Transportation Controller Comparative Performance

<table>
<thead>
<tr>
<th>Model</th>
<th>Year</th>
<th>Minimum/Available MIPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 170</td>
<td>1982</td>
<td>0.2 MIPS</td>
</tr>
<tr>
<td>Model 2070</td>
<td>1998</td>
<td>4.5 MIPS</td>
</tr>
<tr>
<td>ATC v5/6</td>
<td>2006/2014</td>
<td>500 MIPS</td>
</tr>
</tbody>
</table>

* Millions of Instructions Per Second

Graphics: Ralph W. Boaz
Controllers Conform to Other Standards Using the Engine Board

Host Module for ITS Cabinet

Host Module for NEMA Cabinets

Graphics: Ralph W. Boaz
Examples of ATC Units

Photos: See Photo Credits
Example ATC Units in a TFCS Architecture

NEMA TS2 Type 1

Model 332 Cabinet

Photos: Ralph W. Boaz
ATC 5401 Application Programming Interface

- ATC 5401 Standard specifies software (aka API Software or “the API”)
- API Software operates on ATC units allowing concurrently running application programs to share the resources of the controller and TFCS
- Provides for source code portability, compatibility and interchangeability of application software to any ATC unit
Two Types of Interfaces

- User Interfaces
  - Windowing system for users and application programs to share controller’s front panel
  - Configuration window to view/set system-wide parameters

- C language interface for software developers to create programs that use the API software
  - Front Panel
  - Field I/O
  - Time-of-Day
Front Panel Manager Window

FRONT PANEL MANAGER VER 1.00

SELECT WINDOW: 0 - F
0 Ramp Meter Prgrm
2 Emergency Mngmnt
4 System Checker

SET DEFAULT: *, 0 - F
1 *Signal Program
3 Data Distributor
5
7
9

[MORE - UP/DN ARROW] [CONFIG INFO - NEXT]

Graphics: Ralph W. Boaz
Configuration Window

ATC CONFIGURATION INFORMATION
SELECT ITEM: 0 - F
0 System Time 1 Ethernet Config
2 System Services 3 Linux/API Info
4 Host EEPROM Info 5
6 7
8 9
[UP/DN ARROW] [FRONT PANEL - NEXT]
ATC Controller Units in Different TFCSs

Model 332 Cabinet

NEMA TS 2 Type 1
Bringing It All Together

Example Applications

- Signal Control App w/ASC
- Ramp Meter App w/RMC
- Field Master App w/FMS
- CV SPaT Msg Service

API Software

NTCIP, Other Standards

- ATC 5401 Standard
- ATC 5201 Standard

TFCS Standards & Specifications

Graphics: Ralph W. Boaz
[Demonstration]
ATC API Reference Implementation Project

- Establish and maintain cohesive project management and implementation plan
- Develop an API Reference Implementation (APIRI) which is software representative of the ATC 5401 Standard
- Develop an API Validation Suite (APIVS) which is test software capable of validating the APIRI on an ATC Engine Board
- Perform integrated testing of the APIRI and APIVS
ATC API Reference Implementation Project (cont.)

- Establish an open source software (OSS) environment for the software and documents produced during the project
- Deliver final versions of the software, test documentation and test results
Open Source API Reference Implementation (APIRI)

Software Vendors

Manufacturers

Open Source APIRI

Consultants

Agencies

Photos: See Photo Credits

Graphics: Ralph W. Boaz

Open Source API Reference Implementation (APIRI)
ATC Layered Software Architecture

- User Layer
  - Application SW
    - Application Layer
      - API Software Layer
        - API Interface & Behavior Defined by ATC 5401
          - Hardware & O/S Defined by ATC 5201
            - Linux O/S & Device Drivers

Graphics: Ralph W. Boaz
APIRI Portability Achieved Through Compiling and Linking Software

Compiling and Linking Using Source Code and Board Support Package

ATC Engine Board From Vendor 1

ATC Engine Board From Vendor 2

ATC Engine Board From Vendor 3
Application Software Portability Achieved Through Compiling and Linking Software

Compiling and Linking Using Source Code, BSP and API Software Libraries

ATC Engine Board From Vendor 1

ATC Engine Board From Vendor 2

ATC Engine Board From Vendor 3

Application Software
[Demonstration]
ATC Layered Software Architecture

SOFTWARE DEVELOPER → INTERFACE & BEHAVIOR DEFINED BY ATC 5401

APPLICATION SW ↓ API SOFTWARE LAYER

APPLICATION LAYER

END USER

APPLICATION LAYER

API SOFTWARE LAYER

HARDWARE & O/S DEFINED BY ATC 5201

HARDWARE LAYER

LINUX O/S & DEVICE DRIVERS

ATC BOARD SUPPORT PACKAGE LAYER

Graphics: Ralph W. Boaz
API Validation Suite (APIVS) Software in the ATC Layered Architecture
# IEEE Std 829-1998 Documents Used To Specify Testing

<table>
<thead>
<tr>
<th>Test Document</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Plan</td>
<td>Specifies scope, approach, resources, and schedule for a specified level of testing.</td>
</tr>
<tr>
<td>Test Design Specification</td>
<td>Specifies refinements of the test approach in the Test Plan and identifies the features to be tested by this design and associated tests.</td>
</tr>
<tr>
<td>Test Case Specification</td>
<td>Defines the information needed as it pertains to inputs and outputs from the software or software-based system being tested. May represent a single test case or a group of test cases for a segment of a Test Design.</td>
</tr>
<tr>
<td>Test Procedure Specification</td>
<td>Specifies the steps for executing a set of test cases or, more generally, the steps used to exercise a software product or software-based system item in order to evaluate a set of features.</td>
</tr>
</tbody>
</table>
Documents Used To Specify Testing

- Test Plan
  - Test Design Specification
    - Test Case Specifications
    - Test Procedure Specifications
  - Test Design Specification, etc.
  - Test Design Specification, etc.

Test Execution

Graphics: Ralph W. Boaz
[Demonstration]
Things to Consider

- 170 generally not powerful enough for NTCIP communications or connected vehicle applications
- ATCs can be deployed incrementally into your existing TFCS infrastructure
- ATC units may be used for multiple center-to-field (C2F) systems and for generations of C2F systems
- Cost of not going to ATC equipment
- ATC approach is not just a purchase but also part of a strategy
Changing the way we work
Agency Benefits of ATC Units

1. Significant reduction in potential agency costs due to sharing the controller resource for multiple applications

2. Application portability and interchangeability creates competitive pricing environment for hardware and software

3. Processing power to handle emerging federal initiatives such as the Connected Vehicle program
   - Contemporary processing power now and Engine Board upgradable (if necessary) in future

4. Works with existing agency TFCSs
   - Requires no additional investment other than the ATC unit and signal software
Agency Benefits of ATC Units (cont.)

5. Creates the infrastructure which helps satisfy established agency needs
   - Publicly recognized benefits
   - Provides for the follow-through on regional and strategic plans

6. Availability and integration of new off-the-shelf ITS solutions
   - Potentially reduces costs and provides more solutions
Resources

- ATC API Reference Implementation

- ATC 5201 v06

- ATC 5401 v02

- USDOT ITS Professional Capacity Building (PCB) Program
  - www.pcb.its.dot.gov
Ralph W. Boaz
APIRI Project Manager

Pillar Consulting, Inc.
4511 Jicarillo Avenue San Diego, CA 92117
858-352-6281
rboaz@pillarinc.com
www.pillarinc.com