Understanding How Advanced Transportation Controllers Are a Key to Our Future

- **Advanced Transportation Controllers (ATCs)** as defined by ATC 5201 Standard v06
- **Application Programming Interface (API) Software** as defined by ATC 5401 Standard v02

### Basic Transportation Field Cabinet System (TFCS) Components

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

The Problem with Traditional Controller Models

- **Run one application program** and uses all of the resources of the cabinet system
- **Model 170** and most **Model 2070** controllers require particular **obsolete processors**
- **Traditional NEMA TS 1 and TS 2** controllers **not open architecture**
  - Purchase software from the original manufacturer only
- Doing anything other than originally intended is “outside the box”
Numerous Potential Applications

Purpose of the ATC Family of Standards
- Provide a general purpose field computing platform for transportation applications that is:
  - Open architecture
  - Modular
  - Multi-process / Multi-application
  - Can grow with technology
  - Upgrade legacy transportation cabinet systems

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-process, multi-application
- Mechanical requirements only for physical interfaces
- Works with all major transportation field cabinet systems

ATC Engine Board Concept

Graphics: Ralph W. Boaz
ATC 5401 Application Programming Interface

- **ATC 5401 Standard** (aka API Standard) specifies software (aka API Software or “the API”)

- API Software operating on ATC units allows 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
Open Source API Reference Implementation (APIRI)

Software Vendors
Open Source API
Manufacturers
Consultants
Agencies

Portable and Compatible Application Software

Example Applications
- Signal Control App w/ASC
- Transit & LRT Priority
- Train to Wayside Comm
- CV Roadside Processing
- API Software

Other Standards
- ATC 5401 Standard
- ATC 5201 Standard
- TFCs Standards & Specifications

Finding new ways to solve problems

Non-ATC (3 Devices)
- Traffic Controller
- Mgmt Station
- Ramp Meter

ATC (1 Device)
- Traffic App
- Ramp Meter App
- Mgmt Station
- Roadside ATIS App

APIRI Repository
https://github.com/apiriadmin/APIRI

Graphics: Ralph W. Boaz
Photos: See Photo Credits
Finding new ways to solve problems

Connected Vehicle Architecture

- DSRC Radio
- SAE J2735
- NTCIP 1202

Roadside Unit (RSU) - CV Roadside App - Non-ATC Controller - Traffic App - Mgmt Station

DSRC: Dedicated Short Range Communications
SAE J2735: Society of Automotive Engineers DSRC Message Set Dictionary
NTCIP 1202: National Transportation Communications for Object Definitions for Actuated Traffic Signal Controller (ASC) Units

Graphics: Ralph W. Basu

We need your help to get the word out!

- Model 170 controllers must be replaced
- Model 2070 controllers must have 2070-1C modules
- ATCs can be deployed incrementally
- Develop migration plans – lots of options
- Lead your vendors in the way they should go. Specify:
  1) Controllers that conform to ATC 5201 Standard v06
  2) API Software that conform to ATC 5401 Standard v02

If you spec it, they will come!

Resources

- ATC 5201 Standard v06
  www.ite.org/standards/ATCcontroller/ATC5201v06.asp
- ATC 5401 Standard v02
  www.ite.org/standards/index.asp
- ATC API Reference Implementation
  www.ite.org/standards/atcapi/referenceimplementation.asp
- USDOT ITS Professional Capacity Building (PCB) Program
  www.pcb.its.dot.gov
- National Transportation Communications for ITS Protocol
  www.ntcip.org/library/documents