



W E L C O M E

**RITA** Intelligent Transportation Systems  
Joint Program Office

# Welcome



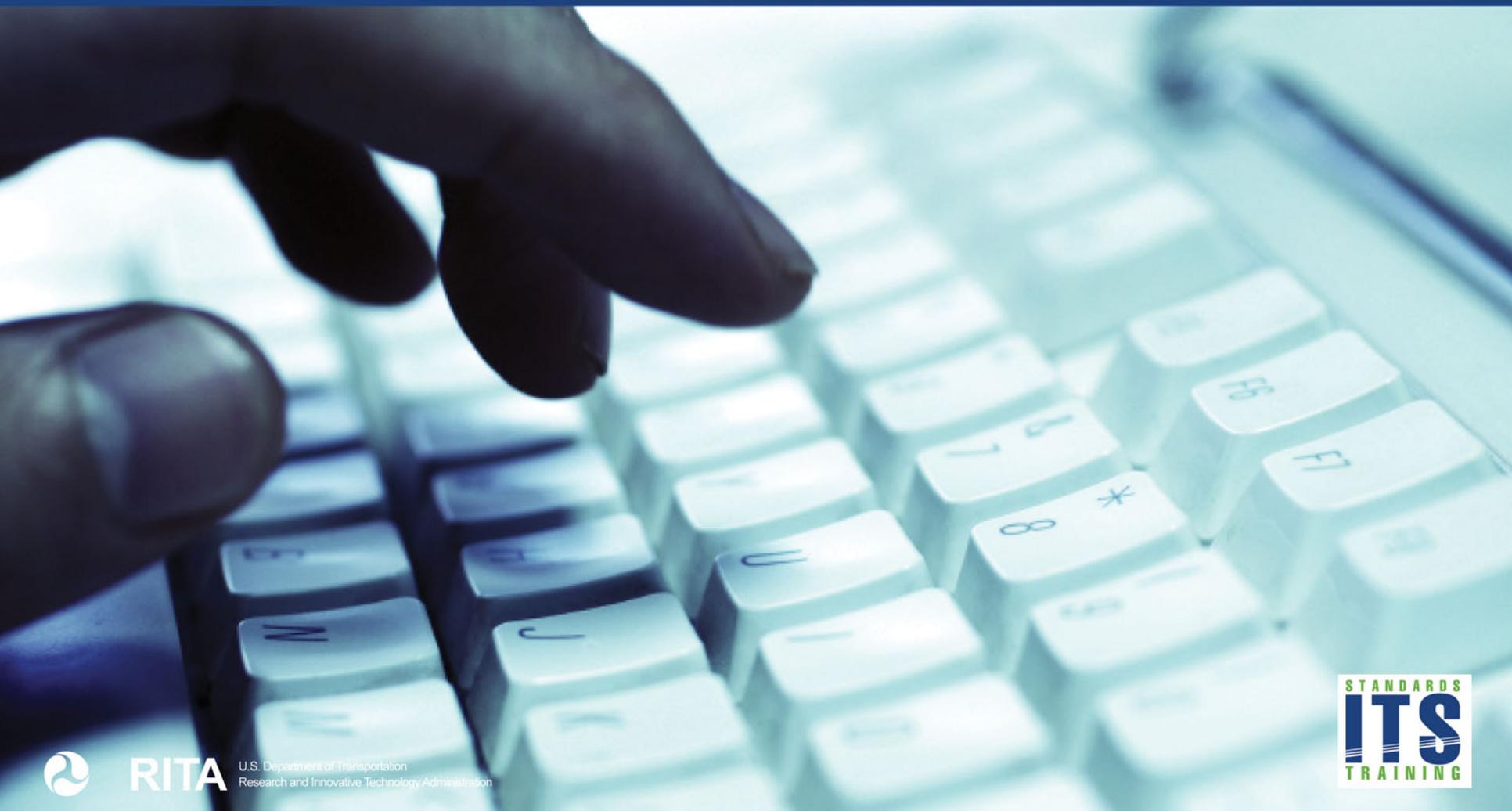
Ken Leonard, Director  
ITS Joint Program Office  
[Ken.Leonard@dot.gov](mailto:Ken.Leonard@dot.gov)

The screenshot shows the homepage of the RITA ITS Professional Capacity Building Program. The header includes the RITA logo and the text "U.S. Department of Transportation Research and Innovative Technology Administration". Below the header, the main navigation menu includes "About", "ITS Training", "Knowledge Exchange", "Technology Transfer", "ITS in Academics", and "Media Library". The main content area features a large banner with the text "Welcome to ITS Professional Capacity Building" and a sub-header "The ITS PCB Program is the U.S. Department of Transportation's leading program for delivering ITS training and learning resources to the nation's ITS workforce." To the right of the banner is a "What's New" section with a list of recent events, including "Upcoming T3 Webinar: Smart Traffic Management: Lessons from New York City's Midtown in Motion Project (4/18/13)" and "Starting February 15, 2013: Several training opportunities are available from the Consortium for ITS Training and Education". Below the banner are three main content blocks: "Available E-Training (free)" listing various web courses and blended training options; "Free ITS Training" with a sub-section for "ITS in Academics" and a list of training opportunities; and "T3 Webinars" with a list of webinar topics and technical assistance information.

[www.pcb.its.dot.gov](http://www.pcb.its.dot.gov)



# ACTIVITY



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration





# **A208: Using the ATC 5401 Application Programming Interface Standard to Leverage ITS Infrastructures**





# Instructor



**Ralph W. Boaz**

President

Pillar Consulting, Inc.

San Diego, CA, USA



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration





# Target Audience

- Engineering Staff
- TMC/Operations Staff
- System Developers
- Software Developers
- Manufacturers
- Private and Public Sector Users





# Recommended Prerequisite(s)

- I101: Using ITS Standards: An Overview
- A101: Introduction to Acquiring Standards-based ITS Systems
- A102: Introduction to User Needs Identification
- A201: Details On Acquiring Standards-based ITS Systems
- A202: Identifying and Writing User Needs When ITS Standards Do Not Have SEP Content
- A103: Introduction to ITS Standards Requirements Development

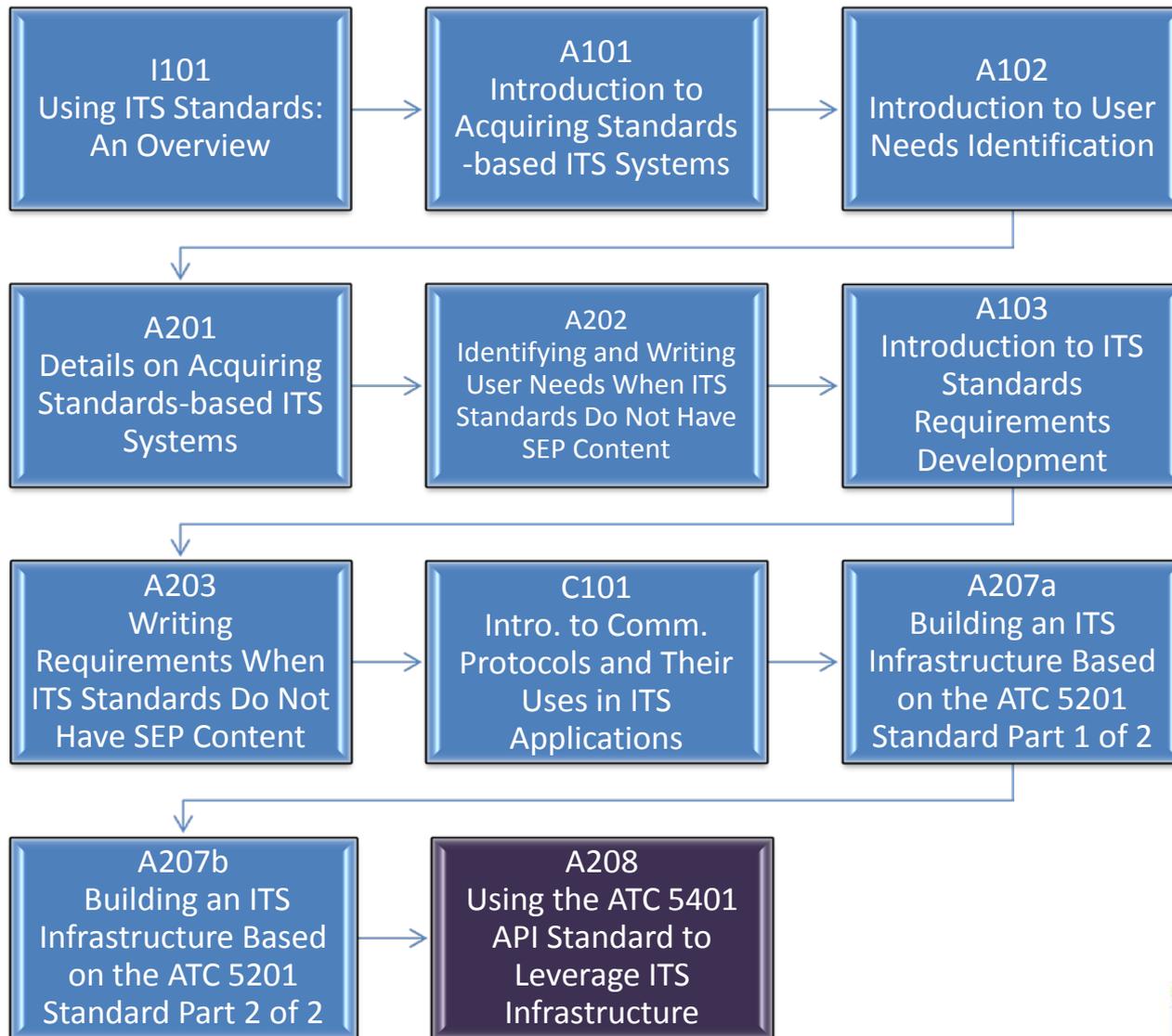




## Recommended Prerequisite(s) (cont.)

- A203: Writing Requirements When ITS Standards Do Not Have SEP Content
- C101: Introduction to the Communications Protocols and Their Uses in ITS Applications
- A207a: Building an ITS infrastructure Based on the ATC 5201 Standard Part 1 of 2
- A207b: Building an ITS infrastructure Based on the ATC 5201 Standard Part 2 of 2

# Curriculum Path (Non-SEP)





# Learning Objectives

1. Identify the features of the ATC 5401 Application Programming Interface (API) Standard
2. Describe the ATC 5401 Architecture
3. Describe how the ATC 5401 Standard works with other ITS Standards
4. Specify API Software for system and equipment procurements



# Learning Objective #1 – Identify the Features of the ATC 5401 Application Programming Interface (API) Standard

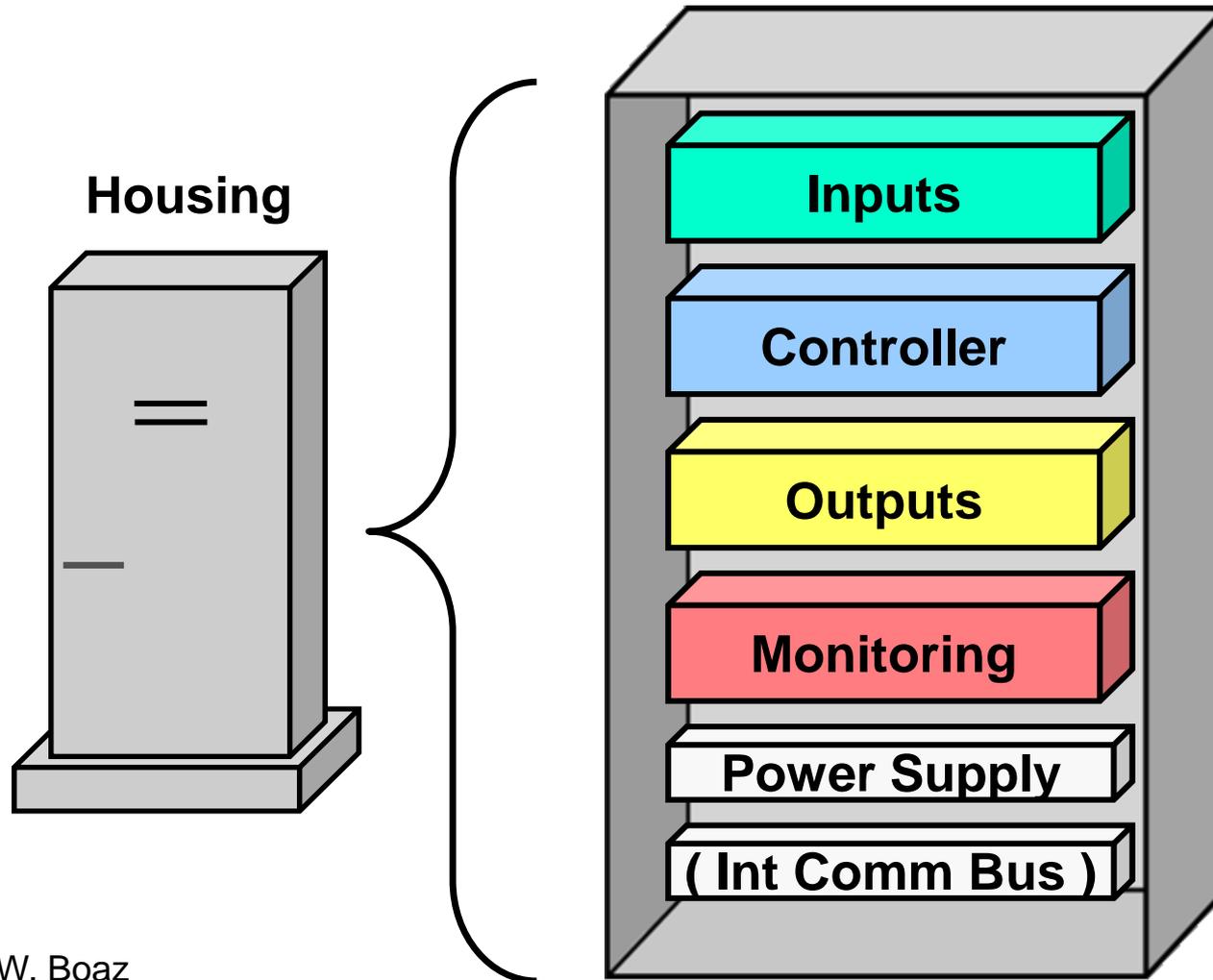
- Sharing resources of the controller and transportation field cabinet system
- Structure of the ATC 5401 Standard
- Features for API Software



# Uses for ATC Controller Units Identified

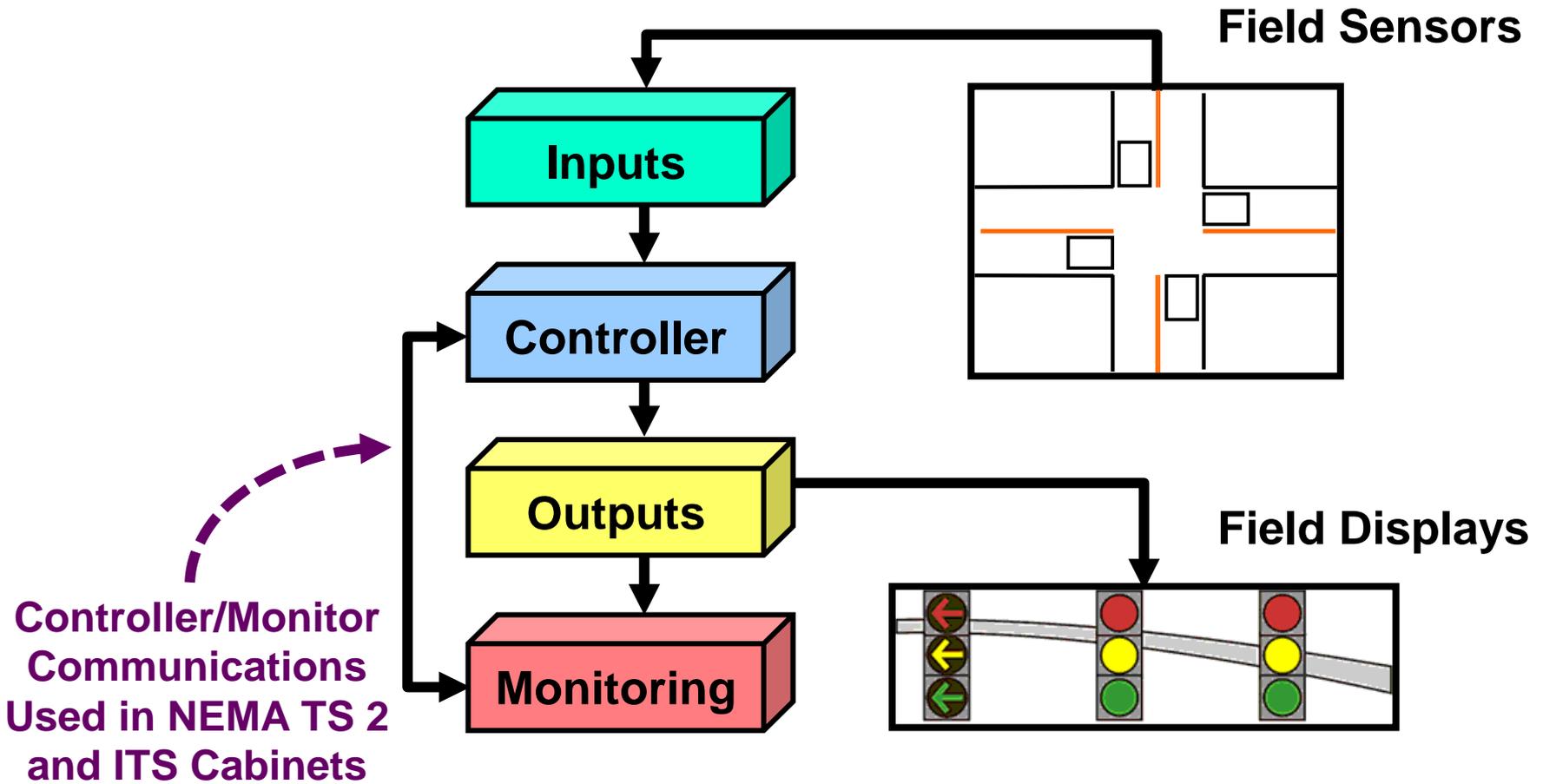
- Traffic Signal Control/Traffic Management
- Transit/Light Rail Priority
- Emergency Management
- Lane Use
- Red Light Enforcement
- Speed Monitoring/Enforcement
- Access Control
- Advanced Traveler Information Systems (ATIS)
- Data Collection Systems / Data Distribution
- Connected Vehicle (CV) Applications
  - Safety
  - Eco Driving

# Sharing Resources of the Controller and Transportation Field Cabinet System (TFCS)



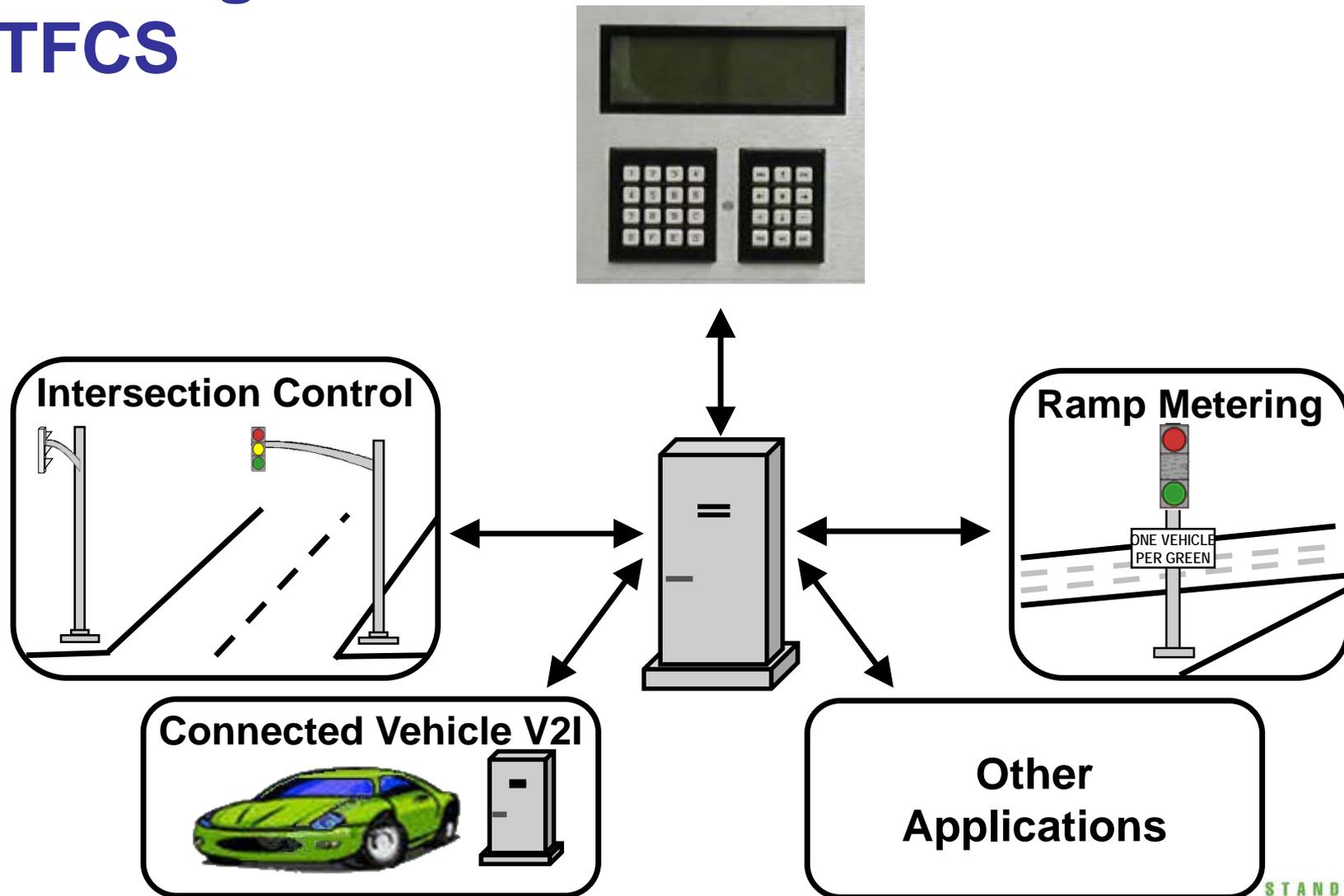
Graphics: Ralph W. Boaz

# Basic TFCS Operation



Graphics: Ralph W. Boaz

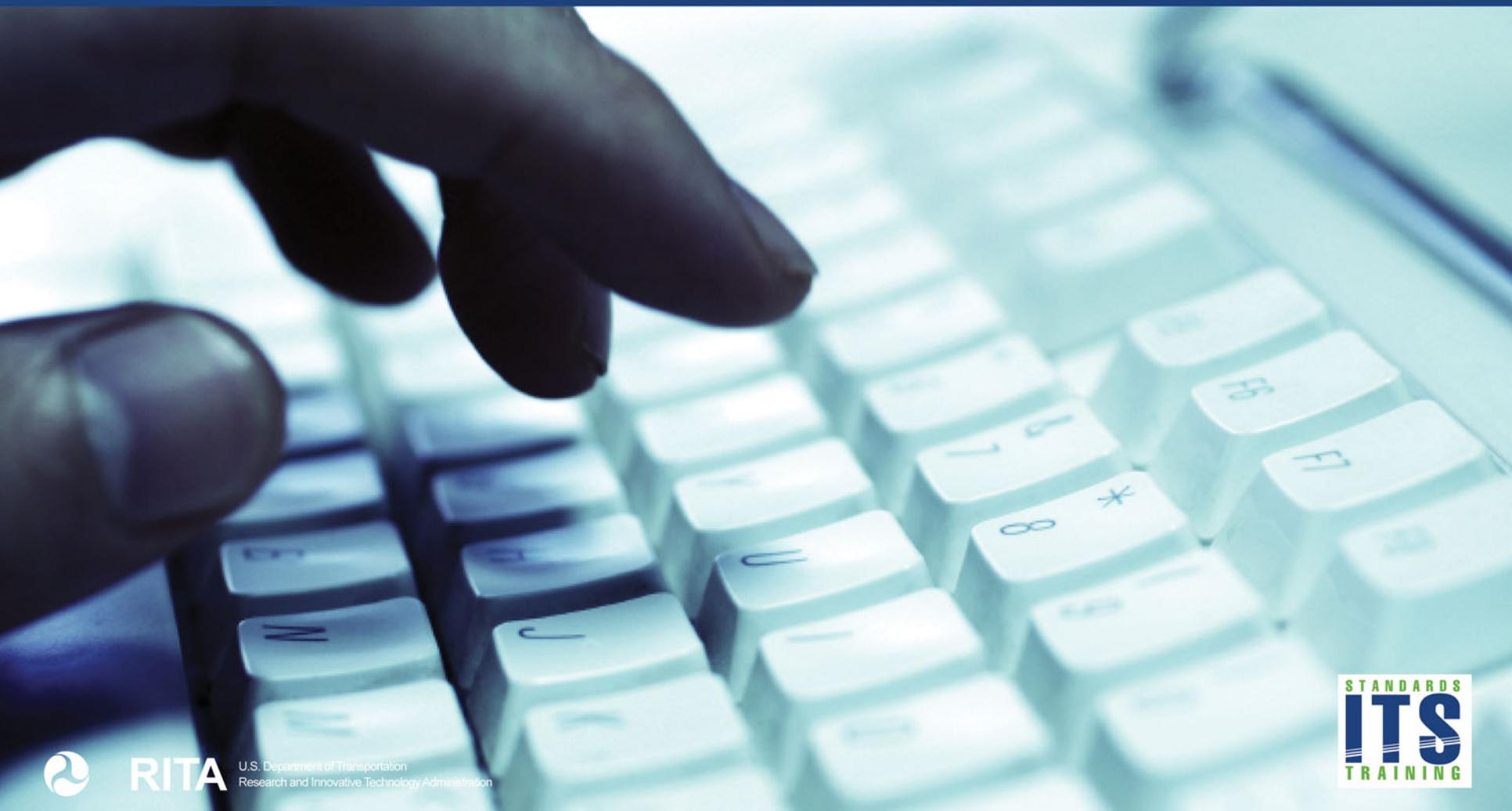
# Sharing Resources of the Controller and TFCS



Graphics: Ralph W. Boaz



# ACTIVITY



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# Which of the following elements of the TFCS is not shared by API Software?

## Answer Choices

- a) Outputs
- b) Inputs
- c) Controller
- d) Power

# Review of answers

-  a) Outputs  
*Incorrect. API Software allows application programs to share the output devices of the TFCS.*
-  b) Inputs  
*Incorrect. API Software allows application programs to share the input devices of the TFCS.*
-  c) Controller  
*Incorrect. API Software allows application programs to share the controller unit of the TFCS.*
-  d) Power  
***Correct. API Software has nothing to do with the distribution of power within the cabinet. The fourth element shared is Monitoring.***

# Structure of the ATC 5401 Standard

- Section 1 Introduction
- Section 2 Overall Description
- Section 3 Specific Requirements
- Section 4 Application Programming Interface
- Appendices
- Index



# Clarification of Terms

- ATC 5401 Standard (aka API Standard) specifies software (aka API Software or “the API”)
- API Software operates on ATC controller units allowing application programs to share the resources of the TFCS

# Two Types of Users of API Software

- Operational Users
  - Technicians or transportation engineers that use application programs that run on controllers to perform their operational tasks
  
- User Developers
  - Software developers that design and develop application programs for ATC equipment

# Two Types of Interfaces Provided by API Software

- User Interfaces
  - Interfaces defined for Operational Users so that multiple programs can run concurrently on a ATC controller unit
  
- Software Interfaces
  - C language function calls that allow User Developers to create application programs that share the resources of a TFCS

# ATC 5401 Sections of Interest to Operational Users

- Section 1 Introduction
  - Section 2 Overall Description
  - Section 3 Specific Requirements
  - Section 4 Application Programming Interface
  - Appendices
  - Index
- Sections of interest to Operational Users**
- 
- ```
graph TD; A[Sections of interest to Operational Users] --> B[Section 3 Specific Requirements]; A --> C[Index];
```

# Features for API Software

- Front Panel Management
- Field Input/Output (I/O) Management
- Real-Time Clock
- API Utilities



# Front Panel Management

- Front Panel Manager Window
- C function interface
- Application programs able to use dedicated window(s)
- Operational Users interact with window/application program in “focus”
- “Default Window” is the window that has focus when the controller unit is powered up
- Works with screen sizes up to 24 lines x 80 characters
- Screen and keypad defined by the ATC 5201 Standard

# ATC Controller Minimum Keypad



Photo: Ralph W. Boaz

# Front Panel Manager Window



```

FRONT PANEL MANAGER VER 1.00
SELECT WINDOW: 0-F      SET DEFAULT: *,0-F
0 Ramp Meter Prgrm     1 * Signal Program
2 Emergency Mngmnt     3 Data Distributor
4 System Checker       5
6                       7
8                       9
[MORE - UP/DN ARROW]  [CONFIG INFO - NEXT]

```

*[Application programs shown are for example purposes only and not included in API Software]*

Graphics: Ralph W. Boaz



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# Example Application Programs



## RAMP METER CONTROL

- |     |                 |             |         |                             |                   |       |
|-----|-----------------|-------------|---------|-----------------------------|-------------------|-------|
| 1 . | M e t e r e d   | L a n e s   | 6 .     | T i m e                     | o f               | D a y |
| 2 . | M a i n l i n e |             | 7 .     | C o m m u n i c a t i o n s |                   |       |
| 3 . | A d d t n l     | D e t e c t | 8 .     | G e n e r a l               |                   |       |
| 4 . | T r a f f       | R e s p     | P l n s | 9 .                         | U t i l i t i e s |       |
| 5 . | C o m m a n d   | S o u r c e | A .     | S t a t u s                 |                   |       |

*[Application programs shown are for example purposes only and not included in API Software]*

Graphics: Ralph W. Boaz

# Example Application Programs (cont.)



## INTERSECTION CONTROL

|                                                                 |               |                   |             |                       |         |           |             |
|-----------------------------------------------------------------|---------------|-------------------|-------------|-----------------------|---------|-----------|-------------|
| 1 .                                                             | T I M I N G   | D A T A           | 6 .         | S T A R T / F L A S H | D A T A |           |             |
| 2 .                                                             | P H           | O V L P           | A S S I G N | 7 .                   | N O     | S E R V E | P H A S E S |
| 3 .                                                             | P E D         | C A R R Y O V E R | 8 .         | D I M M I N G         |         |           |             |
| 4 .                                                             | R E C A L L   | D A T A           | 9 .         | O P T I O N           | D A T A |           |             |
| 5 .                                                             | O V E R L A P | D A T A           |             |                       |         |           |             |
| P R E S S   K E Y S   1 . . 9   T O   S E L E C T   F = H E L P |               |                   |             |                       |         |           |             |

*[Application programs shown are for example purposes only and not included in API Software]*

Graphics: Ralph W. Boaz

# Example Application Programs (cont.)



## EMERGENCY MANAGEMENT OPERATIONS

- 1 . Local Operations
  - 2 . Regional Operations
  - 3 . Evacuation Routing Plan North 1
  - 4 . Evacuation Routing Plan East 1
  - 5 . Evacuation Routing Plan South 1
- [ MORE ]

*[Application programs shown are for example purposes only and not included in API Software]*

Graphics: Ralph W. Boaz



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration

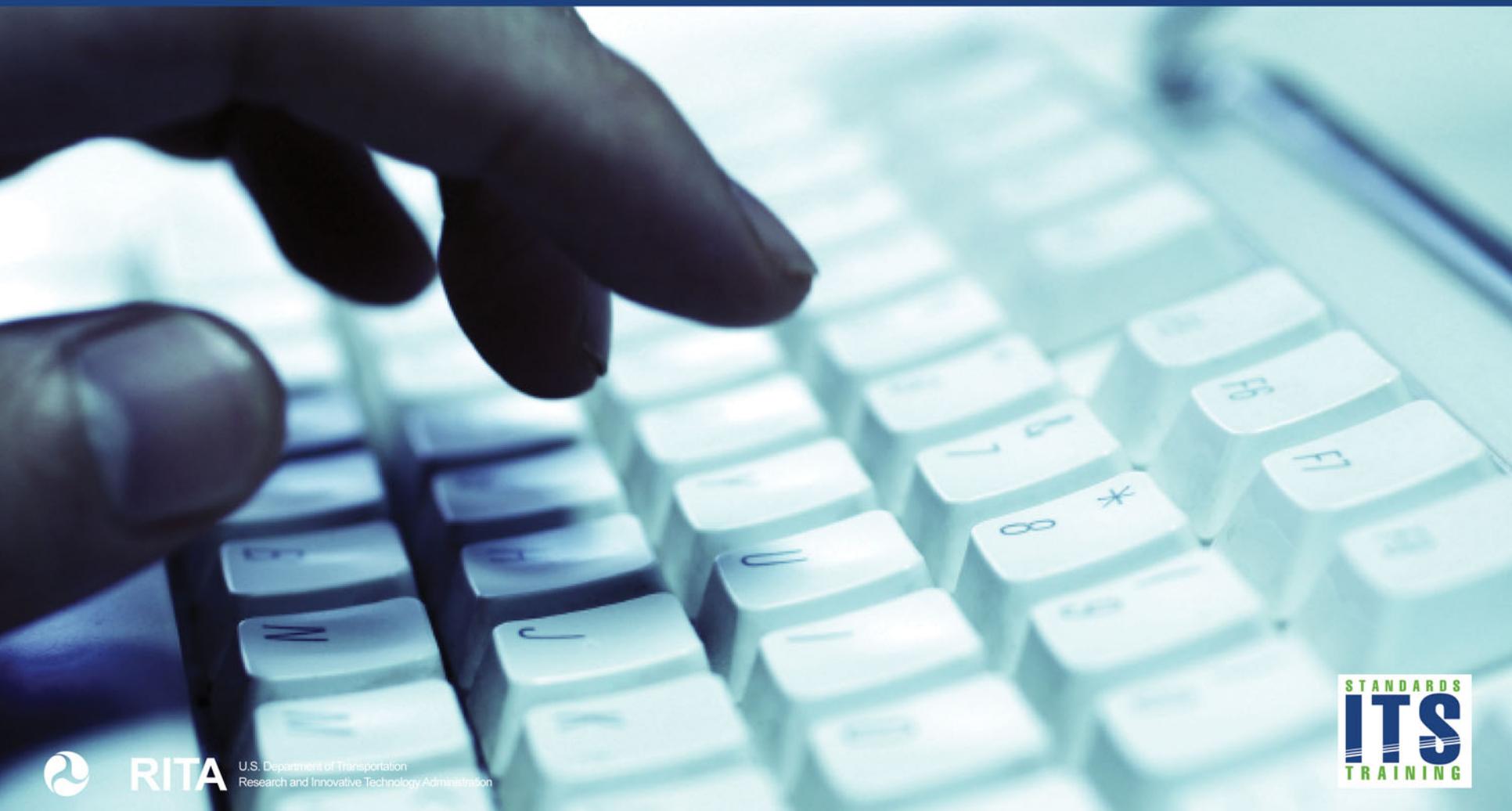


# C Function Interface for Front Panel Management

- General Functions
  - Open, Close, Refresh, Clear Windows
- Window Attribute Functions
  - Character Blink, Backlight, Reverse Video, Underline, Auto Wrap, Scrolling
- Read Functions
  - Poll for Input, Read Character, Read String
- Write Functions
  - Write Character, Write String, Write at Position



# ACTIVITY



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# Which of the following statements is TRUE regarding the operation of the Front Panel Manager Window?

## Answer Choices

- a) The Default Window is always position [1] on the Front Panel Manager Window
- b) An Intersection Control window comes standard with the API Software
- c) Pressing {\*\*, <ESC>} causes the Front Panel Manager Window to be put in focus
- d) The Front Panel Manager is designed for an 8 line display only

# Review of answers



a) The Default Window is always position [0] on the front panel manager window

*Incorrect. The Default Window is user selectable by pressing {\*,[0-F],<ENT>} while in the Front Panel Manager Window.*



b) An Intersection Control Window comes standard with API Software  
*Incorrect. API Software is defined by the ATC 5401 Standard which does not contain specifications for an Intersection Control Window.*



c) Pressing {\*\*,<ESC>} causes the Front Panel Manager Window to be put in focus

***Correct. This action works regardless of the window that is in focus at the time.***



d) The Front Panel Manager is designed for an 8 line display only  
*Incorrect. The Front Panel Manager will work with various size controller screens.*

# Field Input/Output (I/O) Management

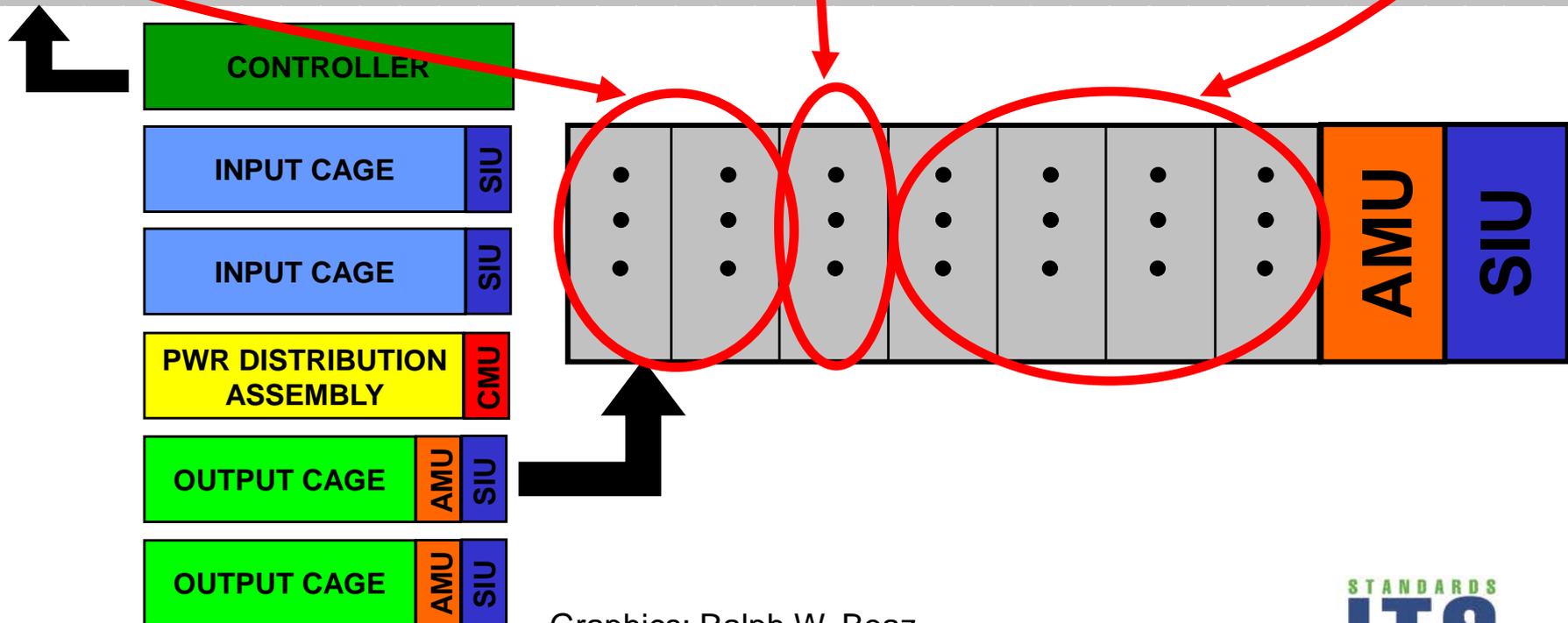
- C function interface only
- Facilitates application programs to be written so they operate in the various TFCS architectures
  - Caltrans 332-Type Cabinets
  - NEMA TS 1, TS 2 Type 1, TS 2 Type 2 Cabinets
  - ITS Cabinets
- Application programs able to query the API to determine what types of devices are used in the TFCS

# Field Input/Output (I/O) Management (cont.)

- Application programs “register” to access a Field I/O device
- Application programs have “read access” to all input and output field devices
- Application programs “reserve” exclusive “write access” to output points of a Field I/O Device

# Application Programs Reserving Outputs

```
FRONT PANEL MANAGER VER 1.00
SELECT WINDOW: 0-F      SET DEFAULT: *,0-F
0 Ramp Meter Prgrm    1 * Signal Program
2 Emergency Mngmnt    3 Data Distributor
4 System Checker      5
6                      7
8                      9
[ MORE - UP / DN ARROW ]   CONFIG INFO - NEXT ]
```



Graphics: Ralph W. Boaz

# C Function Interface for Field I/O Management

- General Functions
  - Register/Deregister, Enable/Disable Field I/O, Get/Reset Field I/O Status
- Input Configuration Functions
  - Get Field Inputs, Get/Set Input Filter
- Output Configuration Functions
  - Set/Get Field Output Reservations, Set/Get Field Outputs
- Frame Functions
  - Set/Get Frame Schedule, Read/Write Frames
- Transition Buffer Functions
  - Set/Get/Read Transition Buffer

# C Function Interface for Field I/O Management (cont.)

- Watchdog/Health Monitor Functions
  - Watchdog Register/Deregister, Watchdog Set/Get Reservation, Health Monitor Register/Deregister, Health Monitor Heartbeat, Health Monitor Reset
  
- Fault/Volt Monitor Functions
  - Set/Get Volt Monitor
  
- CMU/MMU/Channel Functions
  - Set/Get Faults, Channel Mapping

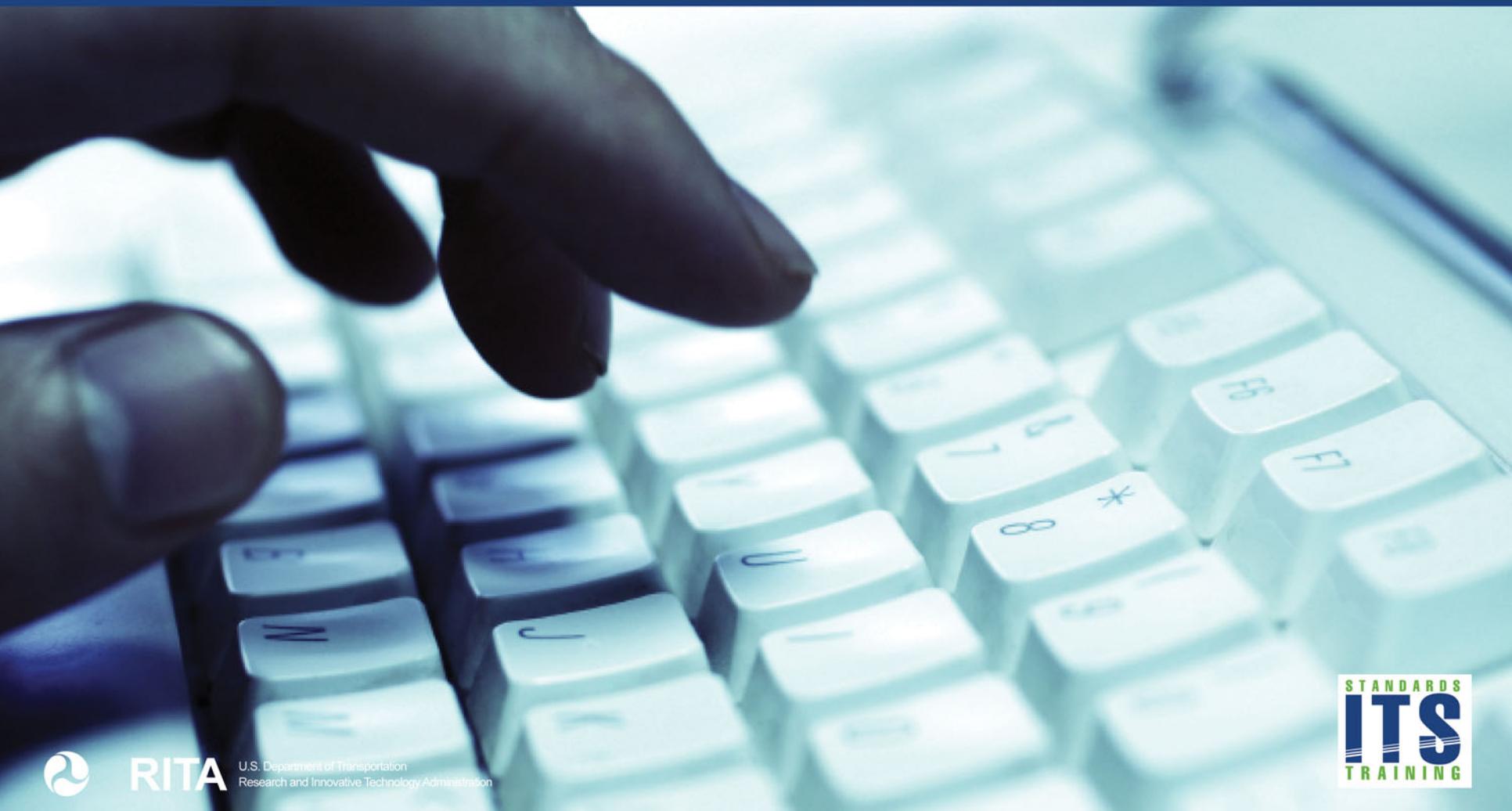
# C Function Interface for Real-Time Clock Management

- Set/Get Time Functions
- Daylight Saving Time (DST) Functions
- Time Source and Signaling Functions





# ACTIVITY



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# Which of the following is TRUE in regards to the Field I/O Management of API Software?

## Answer Choices

- a) All application programs may have read access to all of the input points
- b) All application programs may have write access to all of the input points
- c) All application programs may have write access to all of the output points
- d) Application programs must communicate with each other to avoid writing to the same output points

# Review of Answers



- a) All application programs may have read access to all of the input points

**Correct. There are no restrictions on applications reading input points.**



- b) All application programs may have write access to all of the input points

**Incorrect. No application program has write access to input points.**



- c) All application programs may have write access to all of the output points

**Incorrect. Application programs must reserve exclusive write access to modify an output point.**



- d) Application programs must communicate with each other to avoid writing to the same output points

**Incorrect. The reservation system used by API Software makes communication unnecessary for this purpose.**

# API Utilities

- ATC Configuration Window
- Standard Utilities
  - Setting System Time
  - Setting Ethernet Ports
  - Selecting System Services
  - Getting Linux and API information
  - Viewing Host EEPROM Information
- Extensible so additional utilities may be added
- Utilities differ from application programs
- C function interface

# ATC Configuration Window



```

      A T C   C O N F I G U R A T I O N   I N F O R M A T I O N
S E L E C T   I T E M :   0 - F
0   S y s t e m   T i m e                               1   E t h e r n e t   C o n f i g
2   S y s t e m   S e r v i c e s                       3   L i n u x / A P I   I n f o
4   H o s t   E E P R O M   I n f o                     5
6   7
8   9
[ U P / D N   A R R O W ]                               [ F R O N T   P A N E L -   N E X T ]

```

Graphics: Ralph W. Boaz

# System Time Utility



```

                S Y S T E M   T I M E
DATE           TIME           T - Z O N E   D S T / S t a t u s
0 6 / 1 3 / 2 0 1 3   0 1 : 0 0 : 0 0   - 0 7 : 0 0   E n a b l e / A c t i v e

                C H A N G E
DATE           TIME           T M Z O N E   D S T
M M / D D / Y Y Y Y   h h : m m : s s   s h h : m m   e n a d i s
[ U P / D N   A R R O W ]   [ A P P L Y - E N T ]   [ Q U I T - * * N E X T ]

```

Graphics: Ralph W. Boaz



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# Ethernet Configuration Utility



## ETHERNET CONFIGURATION

### ETHERNET PORT 1

```

Port Enabled: Yes
Packets Sent GD: 120154243 BD: 9898416
Packets Rcvd GD: 782143860 BD: 28328078
IP Address: 192.168.001.065
Subnet Mask: 255.255.255.000
[UP/DN ARROW] [APPLY-ENT] [QUIT-**NEXT]
  
```

Graphics: Ralph W. Boaz



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# Systems Services Utility



```

      ENABLE / DISABLE SYSTEM SERVICES
SERVICE STATUS CHANGE
Bluetooth Enabled ena ord is
Signal Phase & Timing Enabled ena ord is
                                     ena ord is
                                     ena ord is
                                     ena ord is
[UP / DN ARROW] [APPLY - ENT] [QUIT - **NEXT]

```

Graphics: Ralph W. Boaz

# C Function Interface for API Utilities

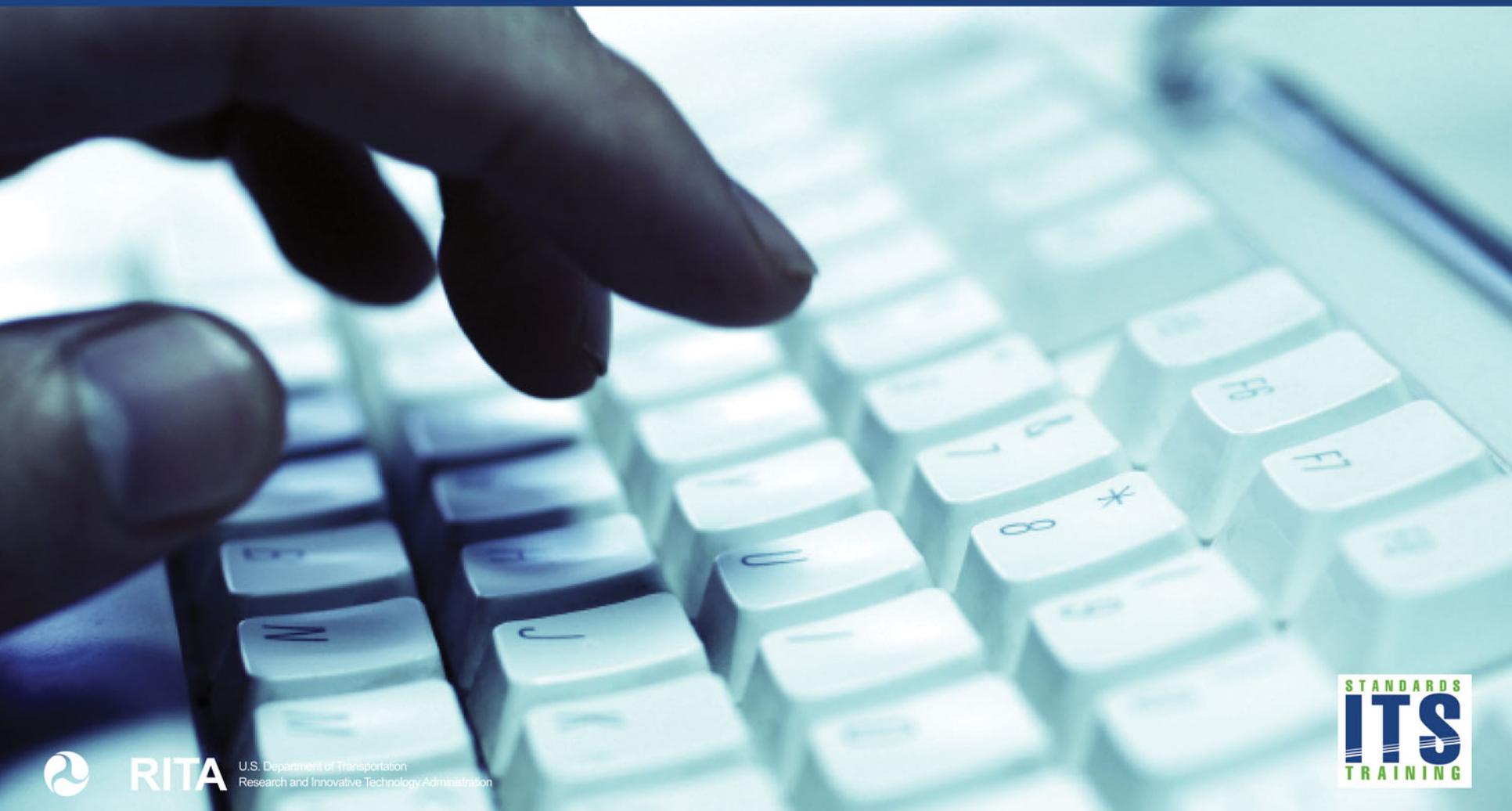
- General Functions – Open/Close Configuration Window, Reserve/Release Resource
- Time of Day Functions – Set/Get Time, Set/Get DST, Time Source Selection
- “ATCConfigurationMenu.txt” file for adding utilities to ATC Configuration Window

```
configitemname1, executablepathname1  
configitemname2, executablepathname2  
configitemname3, executablepathname3  
configitemname4, executablepathname4
```





# ACTIVITY



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# Which of the following is not a standard utility provided in API Software?

## Answer Choices

- a) Ethernet Configuration
- b) Bluetooth Configuration
- c) Linux/API Information
- d) System Time

# Review of answers



a) Ethernet Configuration

*Incorrect. Ethernet is a standard utility allowing Operational Users to set the Ethernet ports of the controller.*



b) Bluetooth Configuration

***Correct. There is no utility for Bluetooth communications currently in the API Software. A utility could be created and added for Bluetooth, however.***



c) Linux and API Information

*Incorrect. Linux/API information is a standard utility in API Software allowing Operational Users to get information on the Linux Board Support Package and the API Software libraries.*



d) System Time

*Incorrect. Setting system time is a standard utility in API Software allowing Operational Users to set the time.*



# Summary of Learning Objective #1

## Identify the features of the ATC 5401 Application Programming Interface (API) Standard

- Sharing Resources of the Controller and Transportation Field Cabinet System
- Structure of the ATC 5401 Standard
- Features for API Software





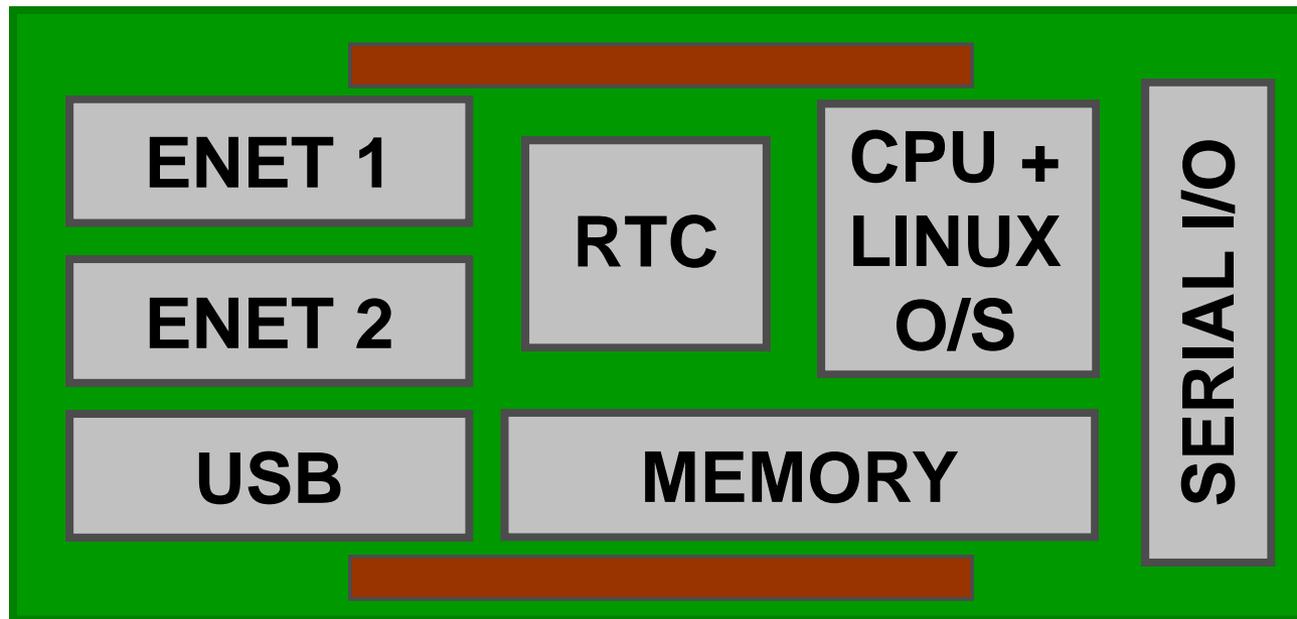
# Learning Objective #2 – Describe the ATC 5401 Architecture

- Key Elements of the ATC 5401 Architecture
- Describe How the ATC 5401 Standard and ATC 5201 Standard Work Together
- ATC Software Portability, Compatibility and Interchangeability

# Key Elements of the ATC 5401 Architecture

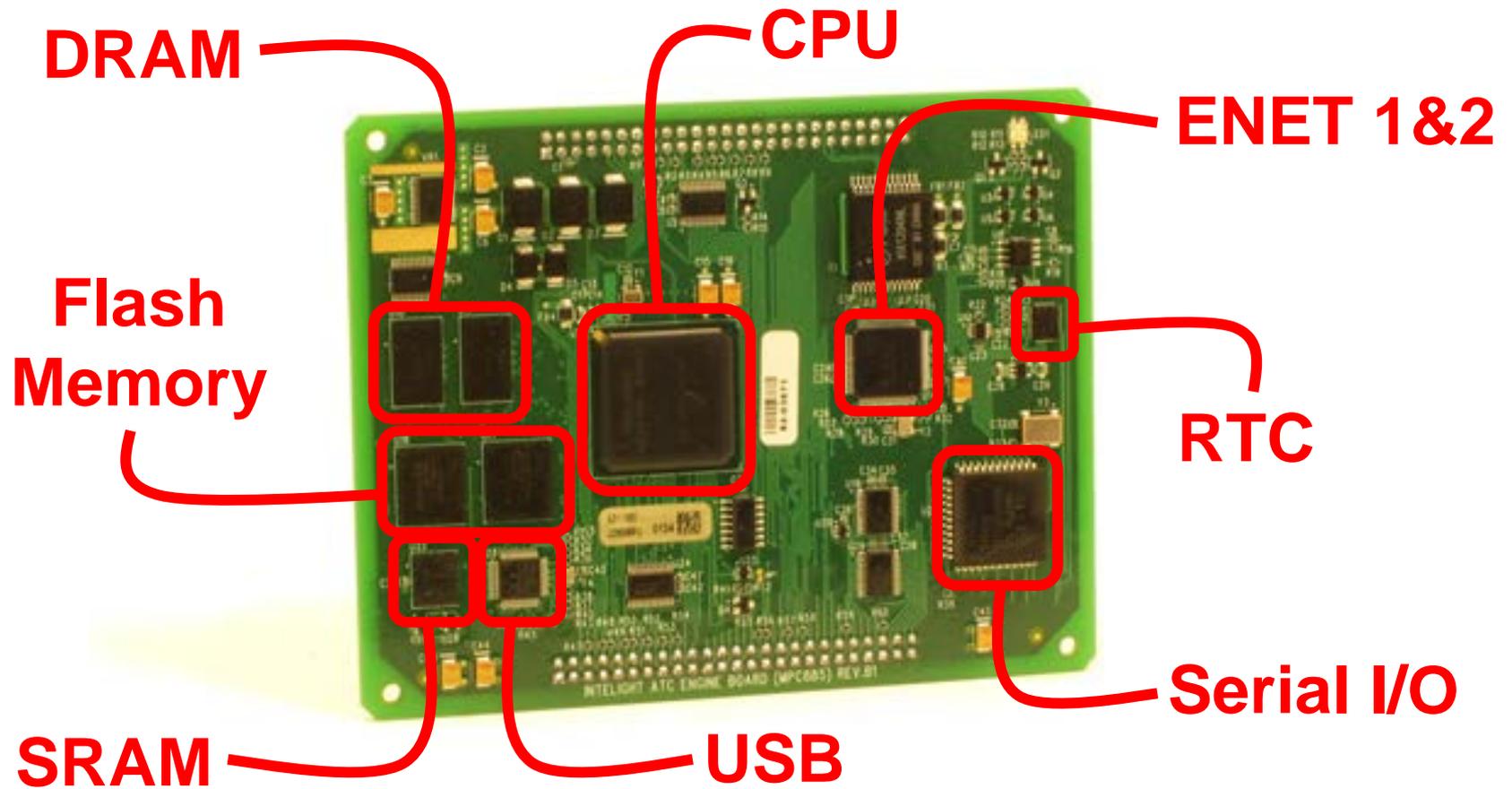
- Based on an “Engine Board” concept specified in ATC 5201
- Uses Linux operating system
  - Open source, multi-process, multi-application
- Allows multiple application programs to operate concurrently ATC controller units
- Shares resources of the front panel, field I/O elements of the TFCS, and the real-time clock
- Source code portability, compatibility and interchangeability for application programs

# ATC Engine Board

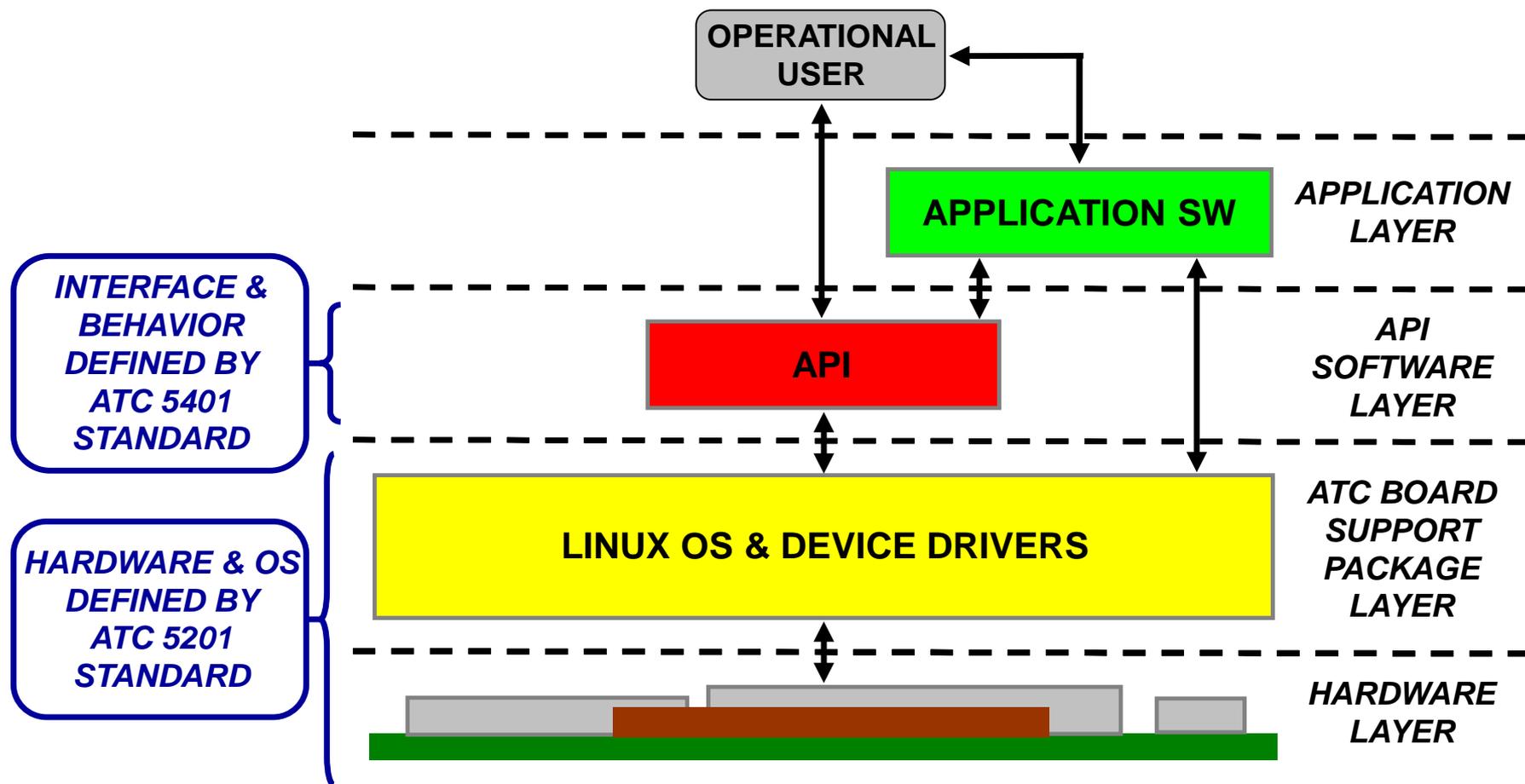


Graphics: Ralph W. Boaz

# Example Engine Board

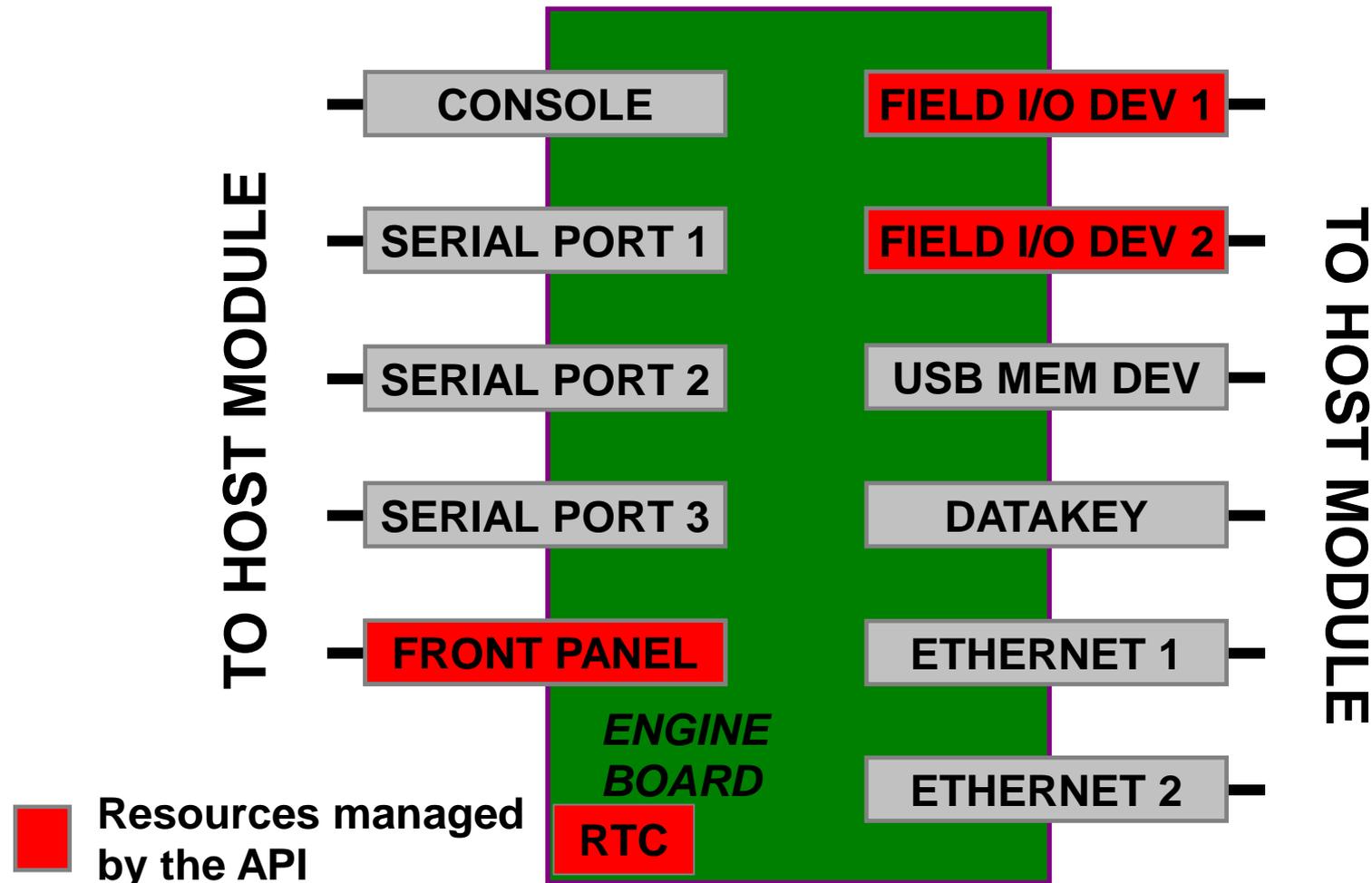


# ATC Layered S/W Architecture



Graphics: Ralph W. Boaz

# API Software manages access to Front Panel, Field I/O Ports, and Real-Time Clock



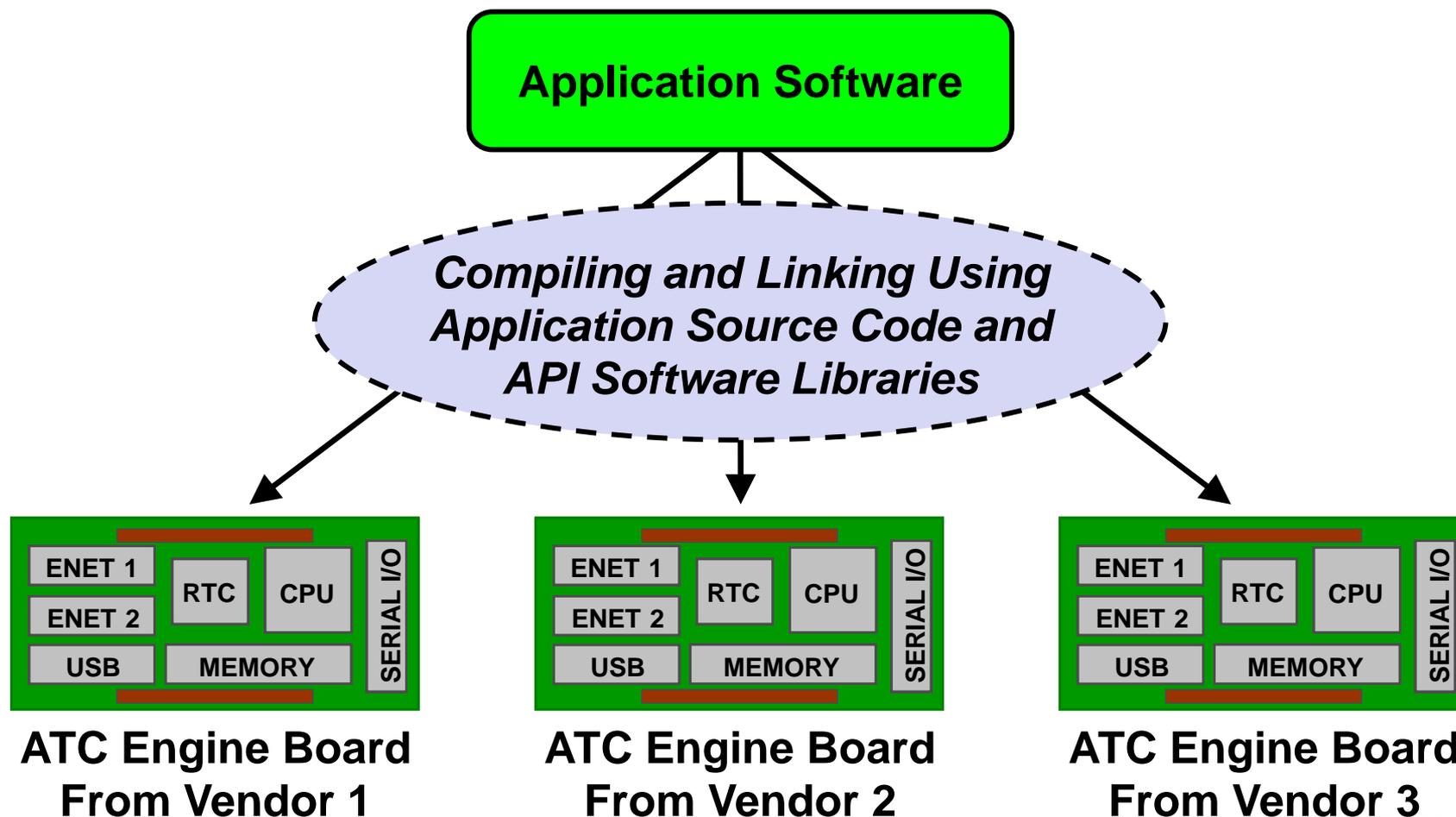
Graphics: Ralph W. Boaz

# ATC Software Portability, Compatibility and Interchangeability

## Portability

- The ease with which applications software and data can be transferred from one application platform to another
- Ability to move, with minimal changes, application software between computers

# Portability Achieved Through Compiling and Linking Software



Graphics: Ralph W. Boaz

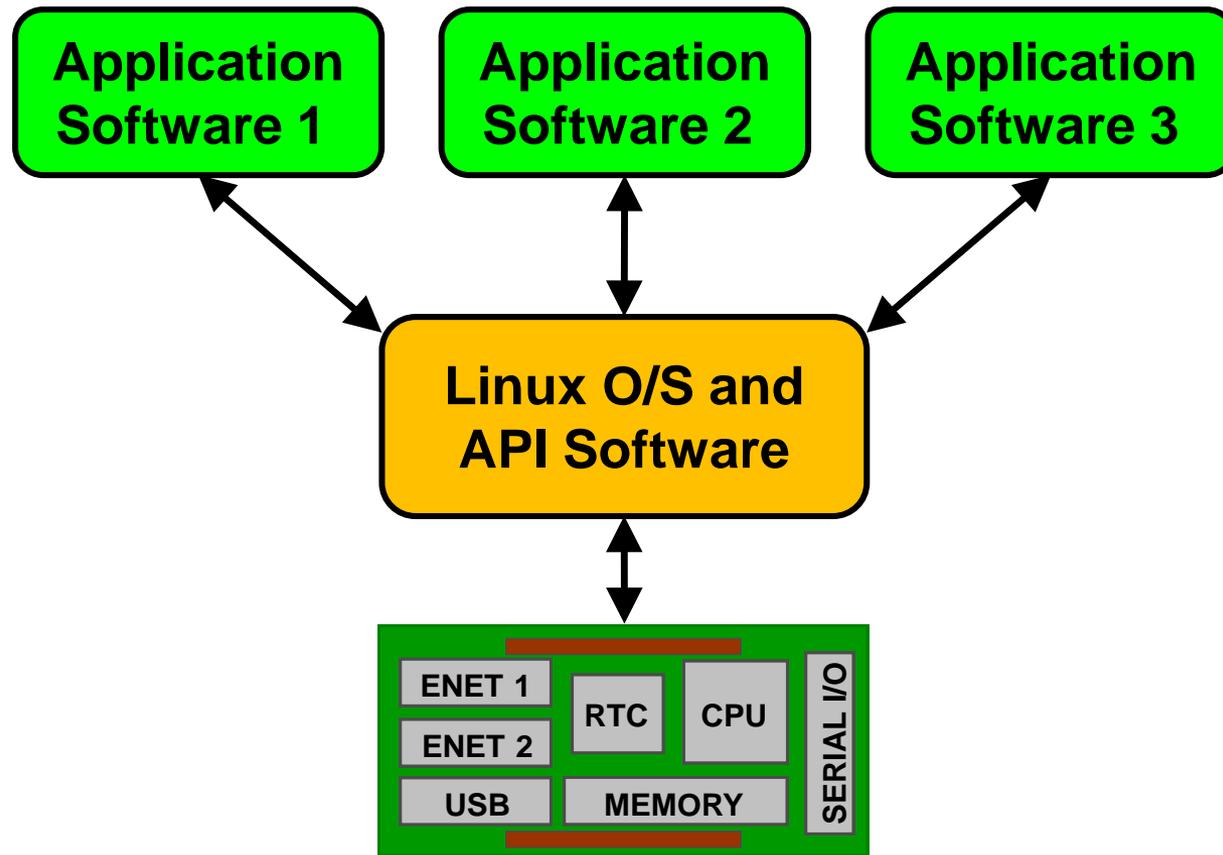
# ATC Software Portability, Compatibility and Interchangeability

## Compatibility

- Two or more systems or components perform their required functions while sharing the same environment
- The two components (or systems) do not need to communicate with each other



# Compatibility Achieved Through Linux O/S and API Software



**ATC Engine Board  
From Vendor**

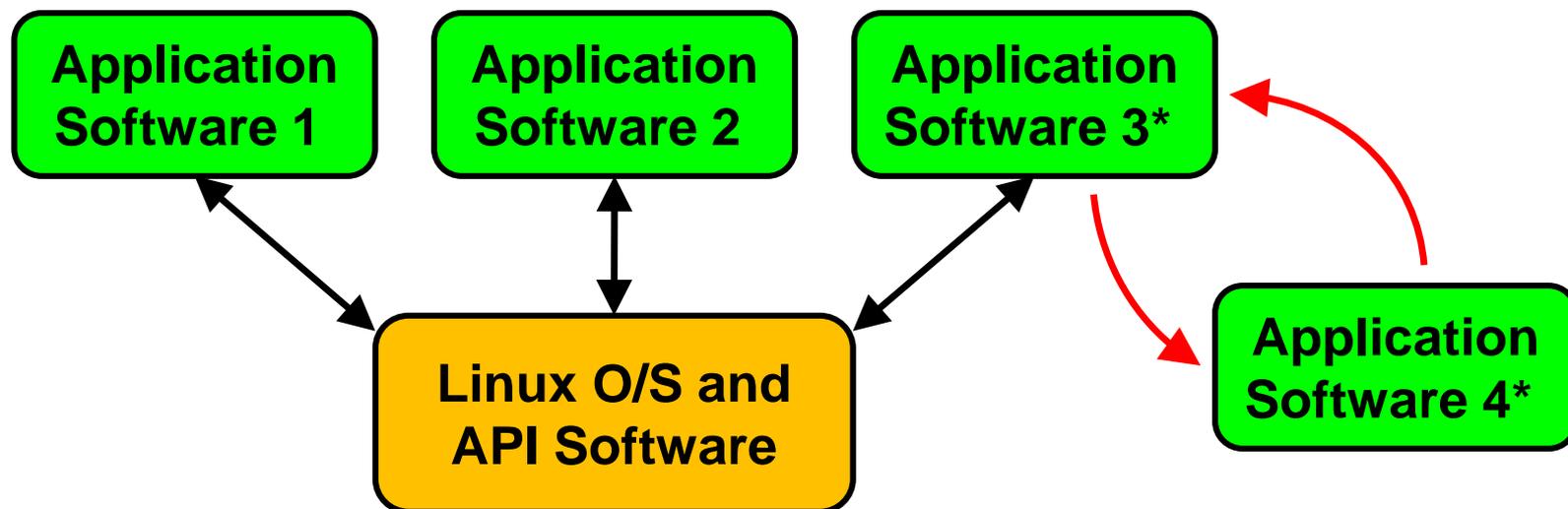
Graphics: Ralph W. Boaz

# ATC Software Portability, Compatibility and Interchangeability

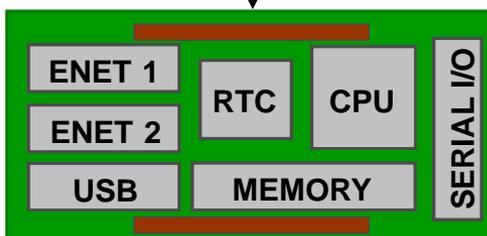
## Interchangeability

- Same functional and physical characteristics so as to be equivalent in performance and durability (subjective)
- Ability to exchange devices of the same type without alteration to the device or adjoining items (adjustments permitted, subjective)

# Interchangeability Possible If Application Features Are Equivalent



*\* Indicates applications are of the same type*

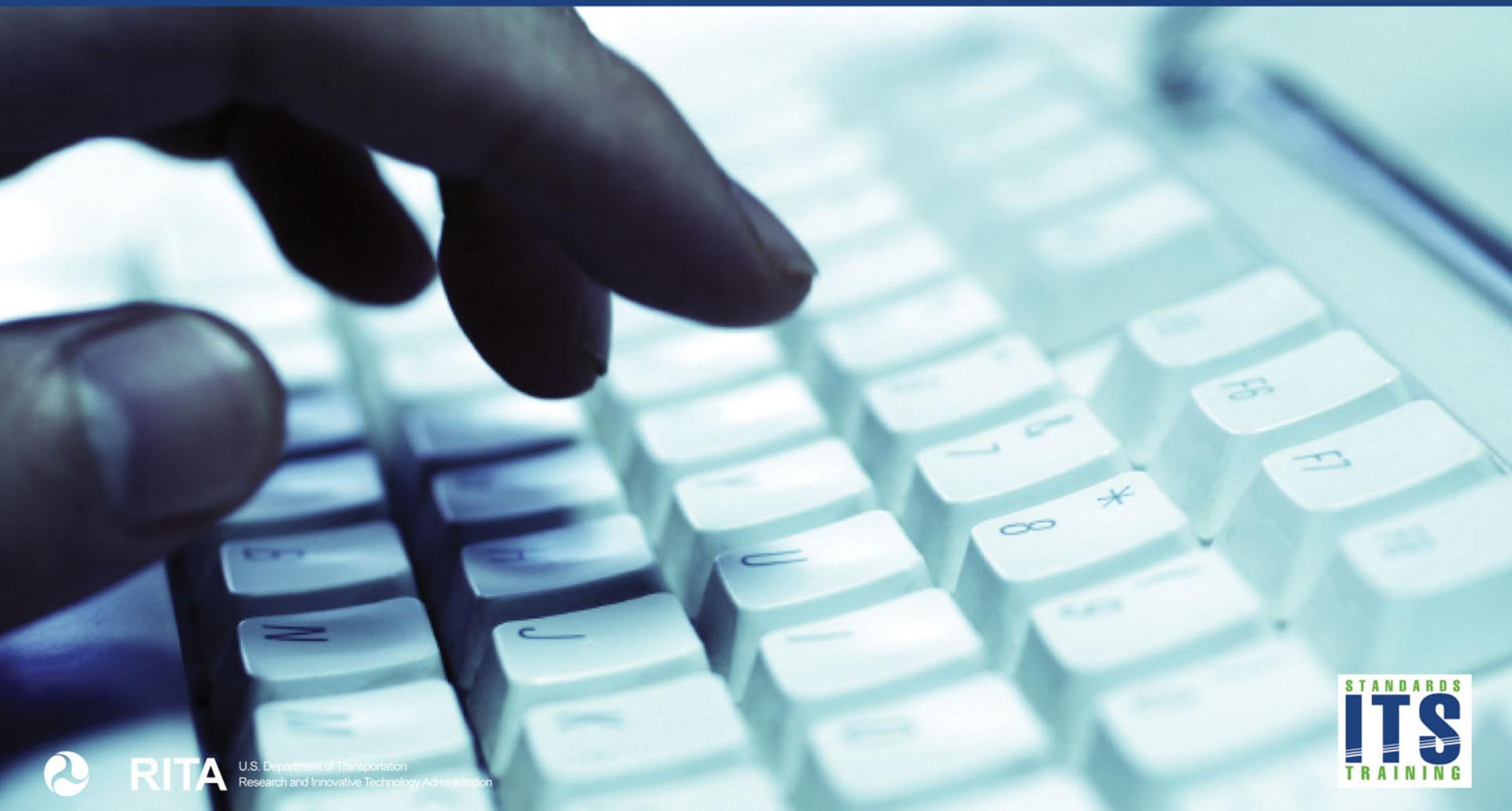


**ATC Engine Board  
From Vendor**

Graphics: Ralph W. Boaz



# ACTIVITY



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# Which of following parts of the ATC Engine Board is not managed by API Software?

## Answer Choices

- a) Front Panel Port
- b) Real-Time Clock
- c) General Purpose Serial I/O Ports
- d) Field I/O Ports

# Review of answers



a) Front Panel Port

*Incorrect. API Software controls the front panel port providing a windowing capability for application programs.*



b) Real-Time Clock

*Incorrect. API Software a System Time Utility and clock functions to allow users and application programs to set the real-time clock.*



c) General Purpose Serial I/O Ports

***Correct. Use of the general purpose serial ports is controlled by Linux not the API Software.***



d) Field I/O Ports

*Incorrect. API Software manages the Field I/O ports of the engine board.*



# Summary of Learning Objective #2

## Describe the ATC 5401 Architecture

- Key Elements of the ATC 5401 Architecture
- Describe How the ATC 5401 Standard and ATC 5201 Standard Work Together
- ATC Software Portability, Compatibility and Interchangeability





# Learning Objective #3 – Describe How the ATC 5401 Standard Works With Other ITS Standards

- Transportation Field Cabinet Systems
- ITS Communications Standards
- Consider New Ways To Solve ITS Issues



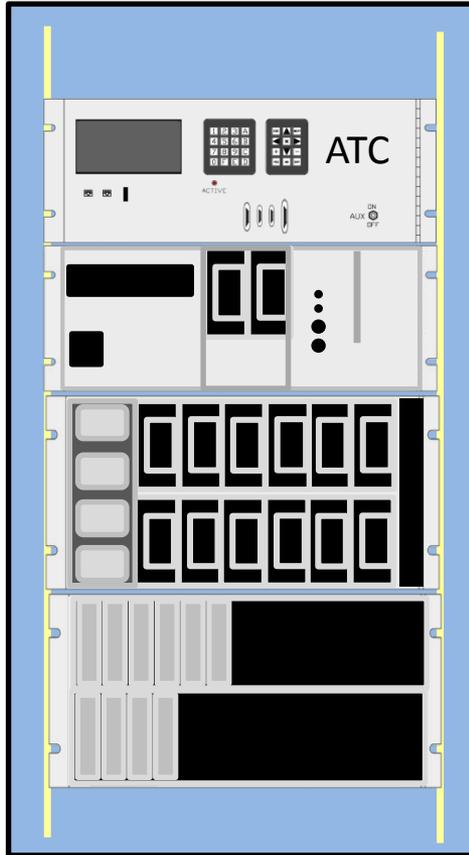
# ATC 5401 Standard Works with all of the Major Transportation Field Cabinet Systems

- ATC 5201 Standard specifies controller units that work with Model 332 Cabinets, NEMA TS 1 and TS 2 Type 2 Cabinets, NEMA TS 2 Type 1 Cabinets and ITS Cabinets
- API Software provides internal cabinet interfaces suitable for all of the major TFCSS
- Application software can be made portable, compatible, and interchangeable to controllers designed for any TFCSS through compilation with API libraries

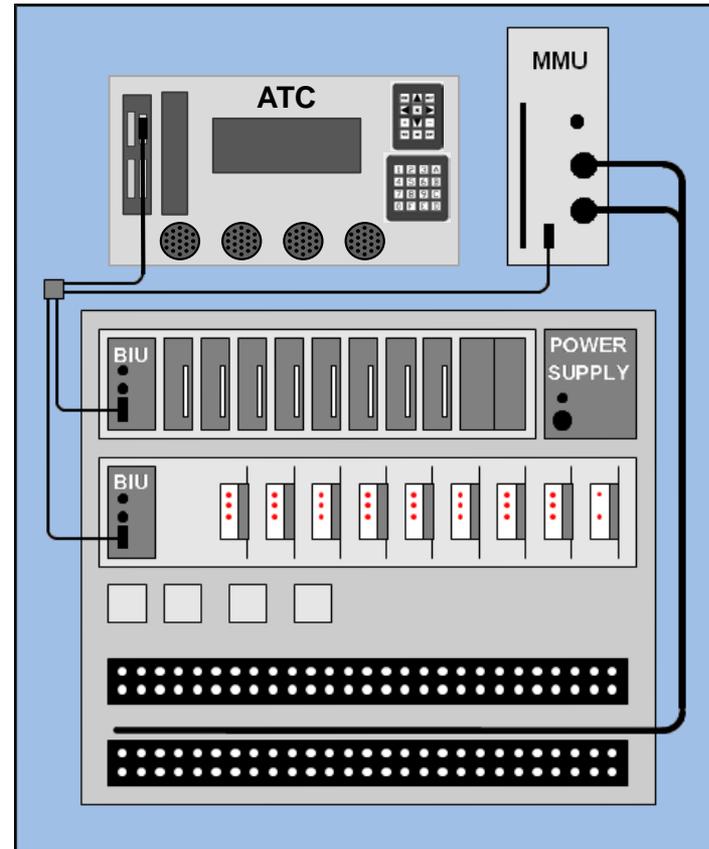
# ATC 5201 Standard and ATC 5401 Standard Provide Computational Power and Interfaces for ITS Communications

- Most TFCs that use 170 controller units will require replacement hardware in order to fully utilize NTCIP Actuated Signal Control (ASC) communications
- ATC standards provide the capability to support multiple applications using different (or same) types of NTCIP communications simultaneously
- ATC standards provide interface and computational power for applications such as Adaptive Control and Vehicle to Infrastructure (V2I) Integration

# ATC Controller Units in Different TFCs



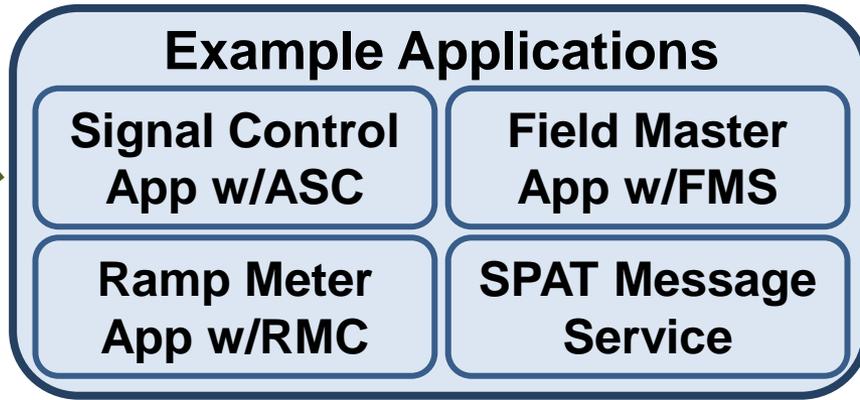
Model 332 Cabinet



NEMA TS 2 Type 1

Graphics: Ralph W. Boaz

# ATC 5401 Working With Other ITS Standards

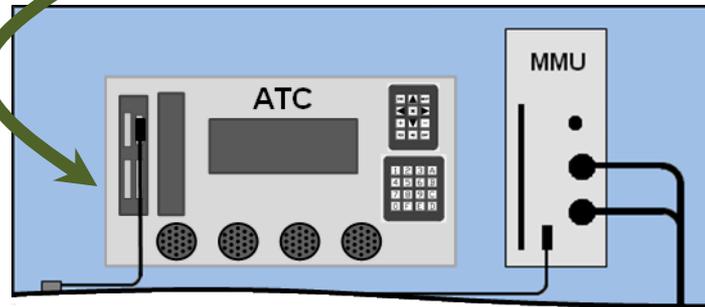
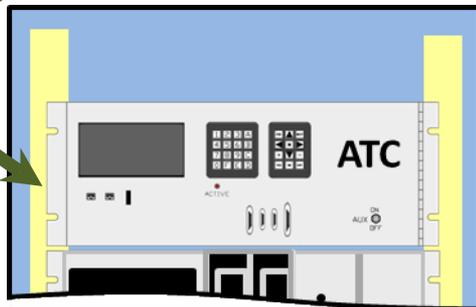
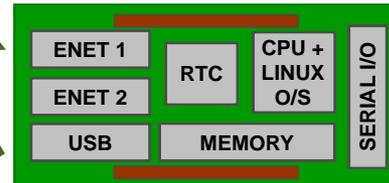
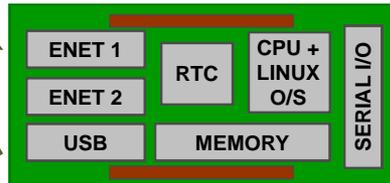


**NTCIP, Other Standards**

**ATC 5401 Standard**

**ATC 5201 Standard**

**TFCS Standards & Specifications**



Graphics: Ralph W. Boaz

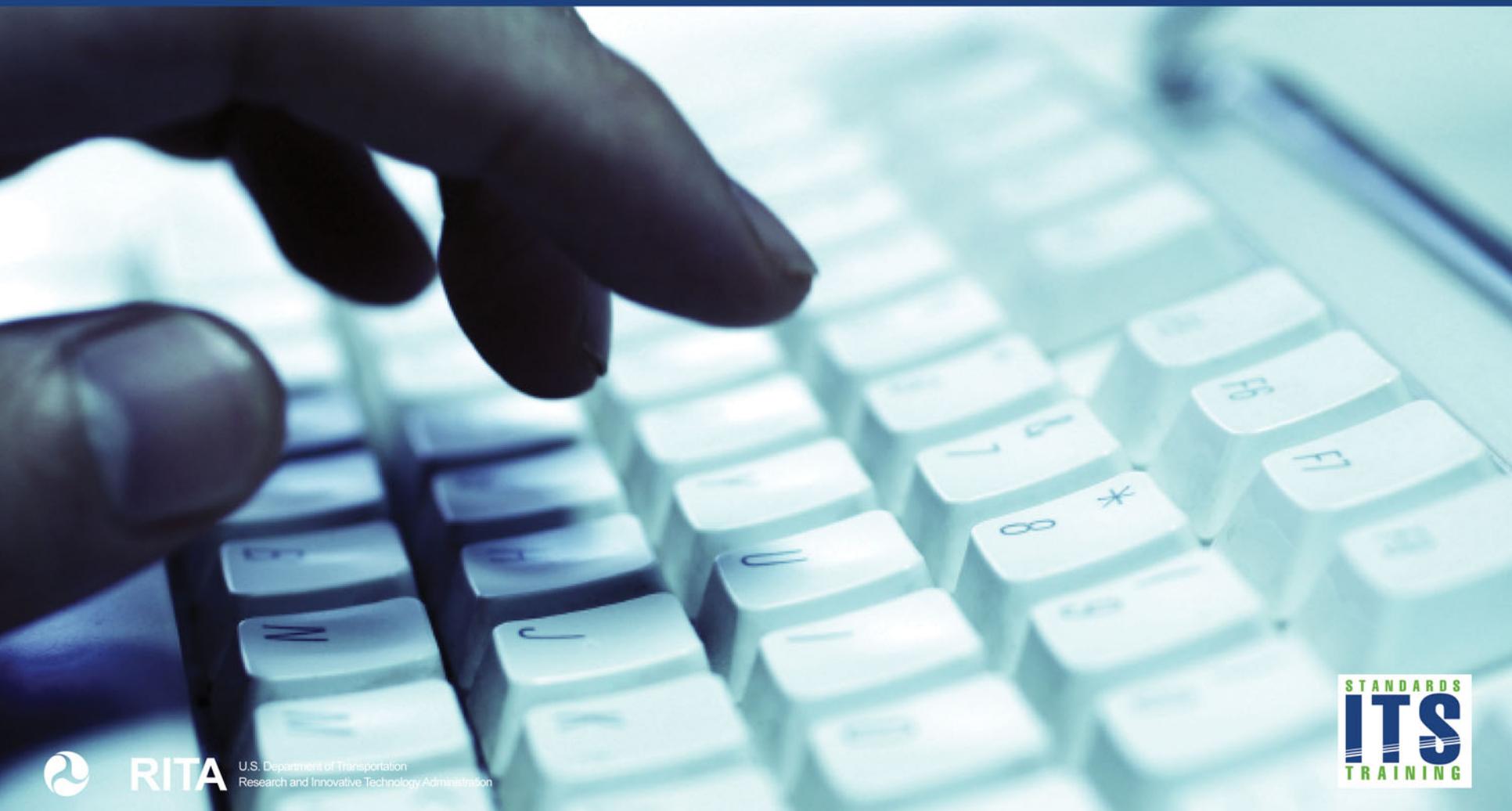
# Consider New Ways To Solve ITS Issues

- Open systems promotes innovation from more sources
- ATC 5401 Standard is opening up new markets for 3rd party software
- New methods to solve ITS problems
  - Software duplicated in field devices may be able to run as applications on an ATC controller
  - Field devices may be more cost effective
  - Custom applications
- Good system engineering practices essential





# ACTIVITY



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



**What must a User Developer do to make an application portable and compatible on two different ATC controllers in different TFCS architectures? Choose the best answer.**

## **Answer Choices**

- a) Write the application software so that it uses the ATC 5401 function calls whenever they are applicable
- b) Compile and link the application source code with API libraries
- c) Compile and link the application source code for the ATC Engine Board used in each controller
- d) All the above

# Review of answers



- a) Write the application software so that it uses the ATC 5401 function calls whenever they are applicable

*Incorrect. This is an essential part of making the software portable and compatible but there are others. Note that software will also need to use function calls appropriate for appropriate TFCS architectures.*



- b) Compile and link the application source code with API libraries

*Incorrect. This is an essential part of making the software portable and compatible between TFCS architectures but there are others.*



- c) Compile and link the application source code for the ATC Engine Board used in each controller

*Incorrect. This is an essential part of making the software portable and compatible between TFCS architectures but there are others.*



- d) All of the above

**Correct. This is the best answer.**



# Summary of Learning Objective #3

## Describe How the ATC 5401 Standard Works With Other ITS Standards

- Transportation Field Cabinet Systems
- ITS Communications Standards
- Consider New Ways To Solve ITS Issues





# Learning Objective #4 – Specify API Software for System and Equipment Procurements

- Discuss how ATC 5401 fits in the Systems Life Cycle
- Develop migration strategies and mitigate deployment issues
- Creating a specification based on the ATC 5401 Standard



# How ATC 5401 Fits in the Systems Life Cycle

- TFCSs (including controllers) can be considered a subsystem of a Center-to-Field (C2F) system
- TFCSs may be in place for multiple C2F systems and generations of C2F systems
- ATC controller units may extend the life of a C2F system
- Deployment of the ATC 5201 and ATC 5401 standards may occur at any stage in the life of a C2F system

# Develop Migration Strategies

- Have a written plan and policy for deploying ATC equipment as part of a strategic plan
  - Procurement specifications and procurement process must be integrated with the strategic plan
  - Factors that may influence the migration strategy:
    - Working with existing TFCS
    - Moving to new TFCS architectures
    - New projects
    - Scheduled maintenance
- (cont.)



# Develop Migration Strategies (cont.)

- Factors that may influence the migration strategy: (cont.)
  - Upgrading to ATC hardware while using an agency's current application software
  - Upgrading to new operational software
  - Compatibility with existing or planned system software
  - Training operational staff

# Mitigating Deployment Issues

- Although the API 5401 v02 has been published, complete API Software meeting the standard is not yet available
- USDOT has started a project to create a reference implementation of API Software (APIRI) for public use in 2015
- APIRI project includes development of a validation suite (APIVS) to test API Software on an ATC Engine Board

# Open Source API Reference Implementation

**Software Vendors**



**Open Source API RI**



**Consultants**



**Manufacturers**



**Agencies**



Photos: Intelight, McCain, Econolite, Peek

Graphics: Ralph W. Boaz

# Benefits of Open Source API Reference Implementation

- Consistent with the Linux O/S open source concept
- Promotes collaboration of developers across transportation industry
- Provides forum for users to express ideas and concerns
- Promotes quick bug fixes and alternative solutions to issues
- Facilitates introduction of new application developers
- Lowers cost to the industry
- Best opportunity for consistent API Software behavior

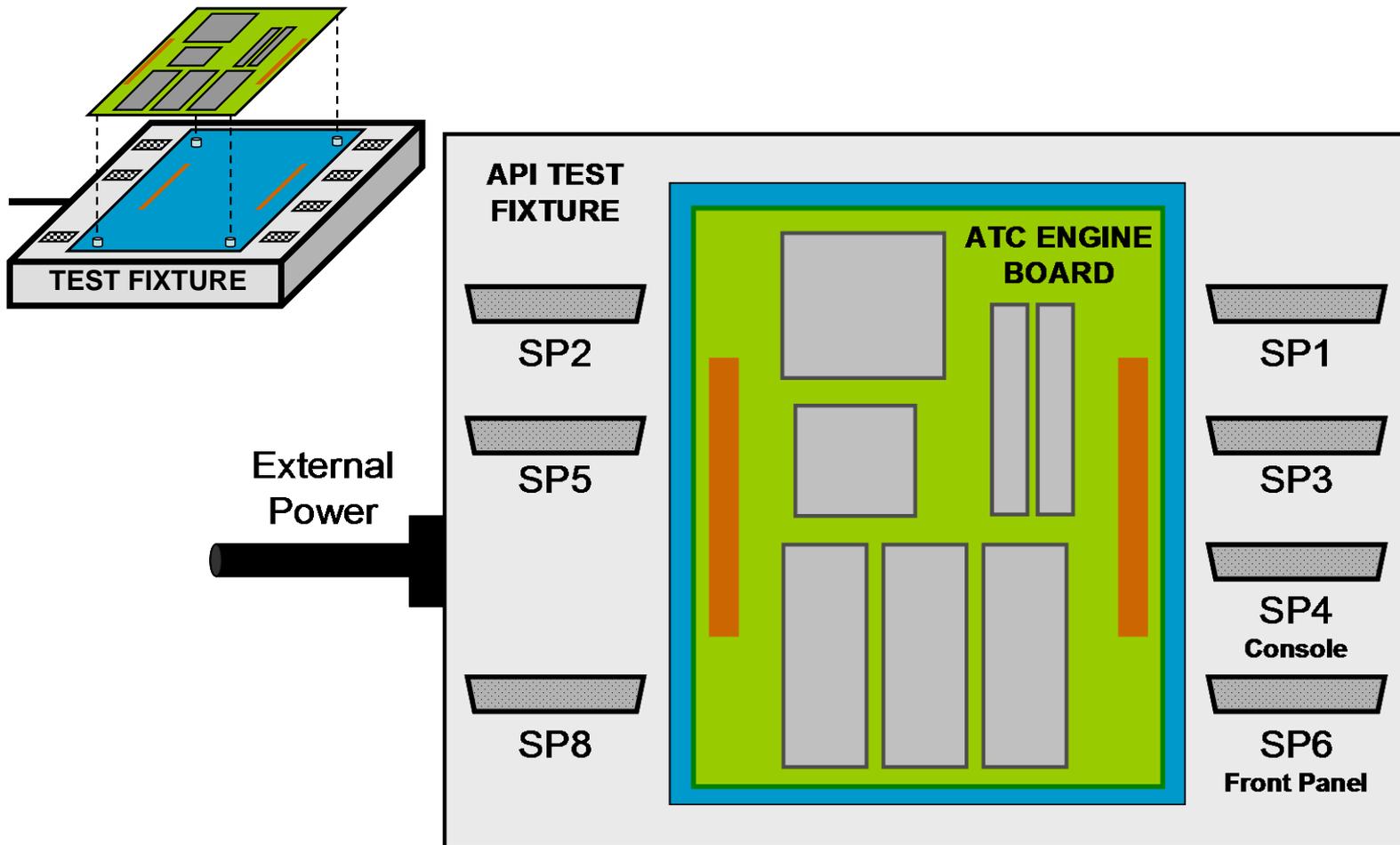


# API Validation Suite (APIVS)

- APIVS software runs on a PC
- Test fixture used to host an ATC Engine Board with API Software
- PC uses test scripts written in XML (Extensible Markup Language) to drive testing of the Engine Board and API Software on the test fixture
- Crossover cables used to connect inputs to outputs of the Engine Board and exercise API Software functionality
- To be made available as open source project

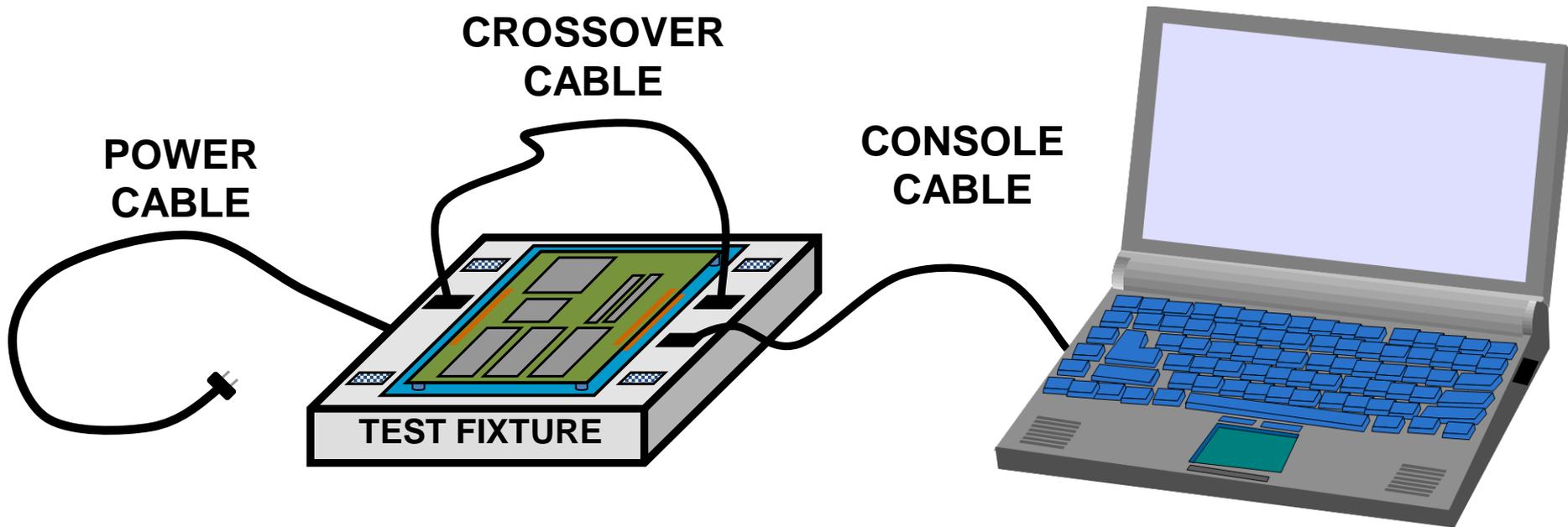


# API Validation Suite Test Fixture



Graphics: Ralph W. Boaz

# API Validation Suite Test Environment



Graphics: Ralph W. Boaz

# Creating a Specification Based on the ATC 5401 Standard

- Specification for API Software a part of ATC controller (or Engine Board) procurement (see PCB Module A207b)
- Require API Software that conforms to ATC 5401 v02.17 when the APIRI is available
- Establish a contractual commitment from Engine Board suppliers to provide operational API Software for their Engine Boards once the APIRI is available
- Require the Tool Chain and Source Code Used to Produce the Linux Environment for the Engine Board



# Creating a Specification Based on the ATC 5401 Standard

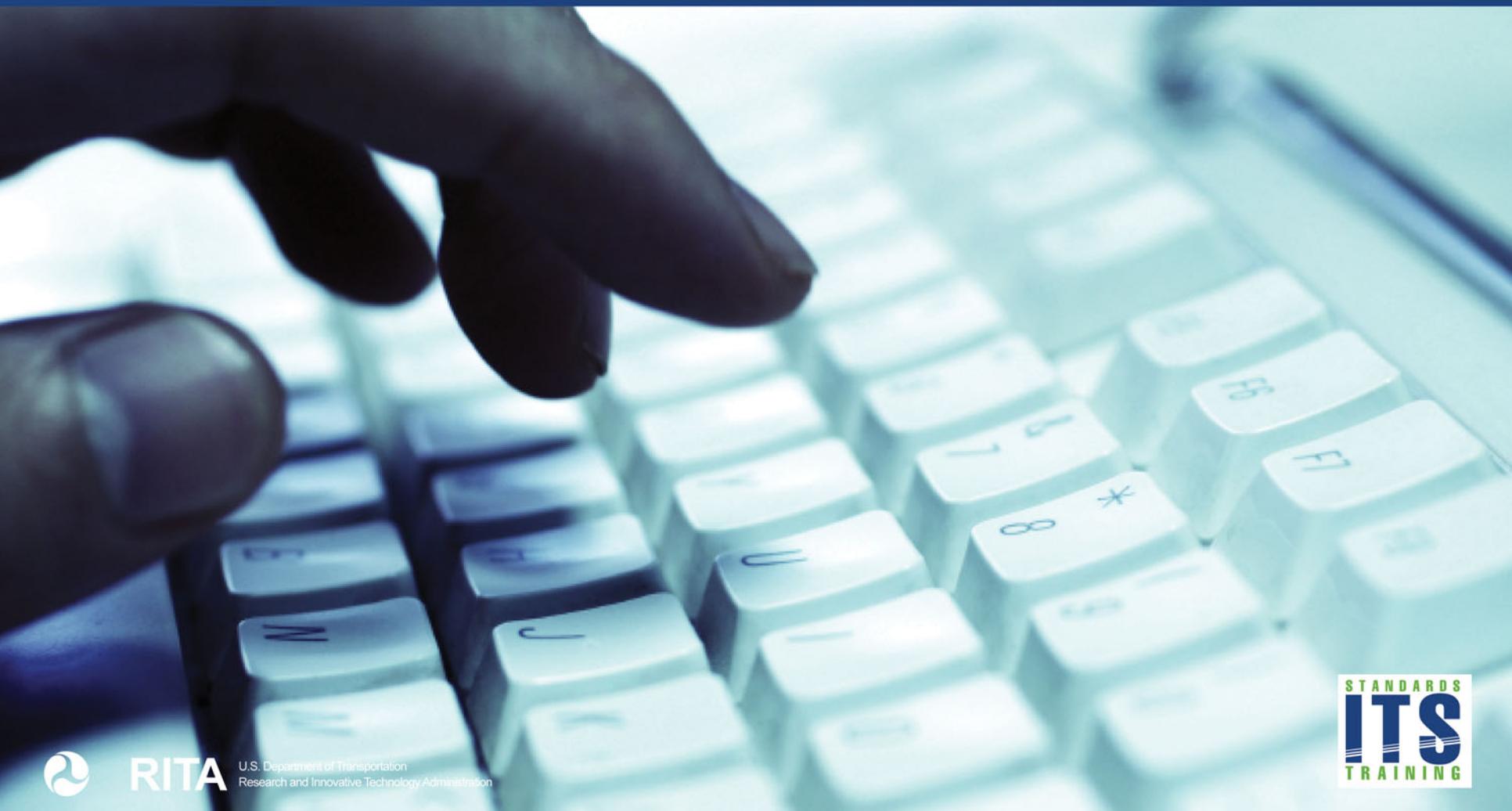
- Require demonstration from manufacturers that each of the API interfaces are operational and multiple programs run concurrently
- Have a certification from manufacturer that the ATC being purchased passed testing using the APIVS (when it is available)

# Additional Guidance When Procuring Application Software

- Prior to purchase, have application providers demonstrate their software on the agency's current ATC units
- Establish a contractual commitment from application software providers to supply software compatible with the API once the APIRI is available
- Purchase support services with application software
- Separate specifications for ATC 5201 and ATC 5401 equipment from that of any application programs



# ACTIVITY



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration



# What is NOT a benefit of the API Reference Implementation?

## Answer Choices

- a) Guarantees bug-free software
- b) Lowers development cost to the industry
- c) Best opportunity for consistent API Software behavior
- d) Promotes collaboration of developers across the transportation industry

# Review of Answers



a) Guarantees bug-free software

**Correct. This is not a benefit of the APIRI. In any non-trivial software, there are bound to be unexpected behaviors.**



b) Lowers development cost to the industry

*Incorrect. This is a benefit of the APIRI. In this joint effort of industry developers, costs are reduced for everyone.*



c) Best opportunity for consistent API Software behavior

*Incorrect. This is a benefit of the APIRI. A common source defining proper software behavior creates the most consistency.*



d) Promotes collaboration of developers across the transportation industry

*Incorrect. This is a benefit of the APIRI. Solutions and improvements by individuals are discussed, assessed and tested by a group of developers prior to being added to the APIRI.*



# Summary of Learning Objective #4

## Specify API Software for System and Equipment Procurements

- Discuss how ATC 5401 fits in the Systems Life Cycle
- Develop migration strategies and mitigate Deployment Issues
- Creating a specification based on the ATC 5401 Standard



# What We Have Learned

1. API Software is grouped into the following feature areas: Front Panel Management, Field I/O Management, Real-Time Clock Management and API Utilities.
2. The ATC 5201 and ATC 5401 define a layered architecture that provides for application software portability, compatibility and interchangeability.
3. ATC 5401 Standard defines API Software which will work with all major TFCS architectures in use today.
4. The API Open Source Reference Implementation will promote collaboration of developers across the transportation industry.

# Resources

- *Caltrans Transportation Electrical Equipment Specifications (TEES), 12 March 2009*
- *Intelligent Transportation System (ITS) Standard Specification for Roadside Cabinets v01.02.17b*
- *NEMA Standards Publication TS 1-1989 Traffic Control Systems*
- *NEMA Standards Publication TS 2-2003 v02.06 Traffic Controller Assemblies with NTCIP Requirements*
- *Institute of Transportation Engineers*  
<http://www.ite.org/standards/>
- *ITS PCB Training*  
<http://www.pcb.its.dot.gov/>



# QUESTIONS?



**RITA**

U.S. Department of Transportation  
Research and Innovative Technology Administration

