

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

### Mac Lister's Introduction

ITS Standards can facilitate the deployment of interoperable ITS systems, and make it easier to develop and deploy regionally integrated transportation systems. However, these benefits can only be realized if you know how to write them into your specifications and test them. This module is one in a series that covers practical applications for promoting multi-modalism and interoperability in acquiring and testing standards-based ITS Transit systems for public transportation providers.

I am Mac Lister, Program Manager, Knowledge, and Technology Transfer in the ITS Joint Program Office of the USDOT and I want to welcome you to our newly redesigned ITS Transit standards training program of which this module is a part. We have worked closely with the Federal Transit Administration and the American Public Transportation Association to develop this material. We are also pleased to be working with our partner, the Institute of Transportation Engineers, to deliver this new approach to training that combines web based modules with instructor interaction to bring the latest in ITS learning to busy professionals like you.

This combined approach allows interested professionals to schedule training at your convenience, without the need to travel. After you complete this training, we hope that you will tell colleagues and customers about the latest ITS standards and encourage them to take advantage of the archived version of these webinars.

ITS Transit Standards training program is one of the offerings of our updated Professional Capacity Building (PCB) Training Program specific to transit industry domain to promote the use of ITS Transit standards such as TCIP, Automated Fare Collection, and Transit Management, to name a few. Through the PCB program, we prepare professionals to adopt proven and emerging ITS technologies that will make surface public transportation systems safer, smarter, and greener, which improves the livability for us all. This series of online courses based on ITS Transit standards is in addition to a 35-module series available for free on ITS Standards for practitioners in the state and the local highway agencies and transit agencies. You can find information on additional modules and training programs on the USDOT website at [www.pcb.its.dot.gov](http://www.pcb.its.dot.gov).

Please help us to continue to make improvements to our training modules through the evaluation process. We look forward to hearing your comments. Thank you again for participating and we hope you find this module helpful.

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

### Jeffrey Spencer's Introduction

ITS Transit Standards help simplify the complexities, overcome procurement challenges, and reduce costs of acquiring ITS systems. I would like to use a simple example to explain how this approach to ITS Transit standards is analogous to our day-to-day life. Imagine that when buying a computer, and you want to buy an upgrade, a printer, or anything else, that you must always buy that same brand at an exorbitant price. Of course, this is not the case because standards have been successfully implemented to allow interoperability. Similarly, transit standards have been developed by transit professionals like you at a national level to encourage competition and limit costs within our industry.

I am Jeffrey Spencer, ITS Team Leader in the Office of Research, Demonstration, and Innovation of FTA within the USDOT and I would also like to welcome you to this ITS Transit standards training. FTA actively supports the development and implementation of transit standards and we hope that you find this series of online courses a useful tool in promoting standardization. We look forward to your participation and input.

**Jerry Lutin:** Okay, good afternoon. This is module 4, Transit Communications Interface Profiles (TCIP), part 2 of 2, structure and elements of TCIP—accessing TCIP via TIRCE and TCIP tools. This is a webinar brought to you by the U.S. Department of Transportation Capacity Building Program. During this presentation, there will be a number of places in which an activity slide, looking like this, pops up. Immediately following that slide will be a quiz, a chance to test your comprehension of the material that went in the previous learning objective. There will be a quiz and you can follow along and answer the questions on your own.

Good afternoon. I'm your instructor for today. I'm Jerry Lutin. I'm retired as senior director of statewide and regional planning from New Jersey Transit, and I've been a long time participant in the transit standards program that has been sponsored by the American Public Transportation Association. I also want to acknowledge my partners in developing the material for this course, Ayers Electronic Systems, LLC and Critical Link, LLC, who are co-instructors with me, and the National Transit Institute, which offers a two day hands on course, and is the source of a lot of the material that you're going to see presented in this module.

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

There is a broad target audience for this. I think we could probably say that this audience is rated TP, transit procurement staff, transit IT staff, metropolitan planning organizations, and if your job description falls in this, well, welcome. And if your job description doesn't fall into this, stay tuned anyway, because you may learn some very interesting things that actually could broaden your experience and your value to your organization. I think it's important to recognize that intelligent transportation system ITS technology, being procured by transit agencies, amounts to a more than \$2 billion a year industry. Maybe you should be a part of that. Let's continue.

There are several prerequisites. This is a pretty technical module, module 4 and as you can see from this chart, it is intended primarily for project managers and project engineers, people who are actually going to be in charge of doing software development and procuring software for a transit agency. There are several curriculum paths here, depending upon whether you're a project manager and you can follow this one. I hope that you have taken some of the previous modules, particularly TCIP 1 of 2. If you're not familiar with TCIP and have not taken that module, you may find that this module is a little bit difficult for you. And if you're a project engineer, the same thing is true and this would be the curriculum path that we have spelled out for you in this. There are a lot of really good modules after this, including traveler information, fare payment, and transit in the connected vehicle. Let's continue.

We have seven learning objectives that we're going to go through today. Learning objective number 1 is going to talk about illustrating the communications stack and show how it relates to TCIP—or actually—TCIP relates to the communication stack. In learning objective number 2, we're going to describe TCIP implementation, requirements, and capabilities editor. That's a mouthful, so it's long and we call it TIRCE, and how it's used the key to TCIP. Number three, we're going to identify and provide examples of data elements, data frames, messages and dialogs, and these are the building blocks of this standard. Learning objective 4, we'll describe how data are organized in TCIP data exchanges, and some of the interesting ways that TCIP allows you to manipulate data. Learning objective number 5 will let us define a profile requirements list, PRL, and explain how it is used to specify TCIP requirements in a transit ITS project. This is a tool that you can use to procure systems that actually have TCIP built into them. Learning objective number 6, we'll review and articulate and describe the uses of each tool in the TCIP suite of tools and learning objective number 7 is going to summarize the range of TCIP applications, implementation tools and how to get additional training.

Let's go into learning objective number 1. We're going to illustrate the communications stack and show how it relates to TCIP. We're going to talk about the communication

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

layers. We're going to give you a little concept of how the TCIP model for data exchange works, and we're going to talk about the differences between file transfers and real time transfer of data.

The open systems interconnection model, the OSI model, is something you may have heard about. The communications between two computers is really a stream of bits, little signals, 1s and 0s and pulses going through wires or over the airwaves. And within that stream of bits, you can conceive of seven different layers. The first layer at the bottom is the physical layer. That includes the electrical properties of the connection, voltage, etcetera, frequency. On top of that is something called the data link layer. This provides access to the physical layer. It also provides error detection. The third layer on top of that is the network layer. That provides information on addressing and routing. The transport layer creates segments for packets of data. Number five on top of that is the session layer. That manages the connection between the computers, kind of like a switchboard. Number six is the presentation layer. It's encryption and decryption, and for TCIP, largely what we're going to talk about is the language XML, which is used in this presentation layer. Finally, the top layer is where you find TCIP. This is how we're formatting messages and data exchanges and this in the top layer is where we find TCIP.

TCIP model for data exchange. We have two rectangles here. On the left, we have transit subsystem A and on the right we have transit subsystem B. Imagine that these are people and they don't know how to talk to each other. In fact, they haven't learned how to talk at all. TCIP is really a system of teaching these business systems, these computers, how to talk to one another. And just from the get go, including how do you shake hands? How do you start a conversation? How do you end a conversation, let alone what's in the conversation? In the middle there, that oval represents what TCIP is trying to do. It's trying to provide the communications link between two entities, two computers that really don't necessarily know how to communicate with one another.

The TCIP standard is composed of building blocks and this graphic is intended to describe the distinction between what data is transferred, which is shown below the blue line, and how the data is transferred, which is above the blue line. Below the blue line, we start at the very bottom with data elements, and we're going to get into this a little more, but data elements are then organized together into data frames and the data frames themselves can actually be organized into higher level data frames. These then are organized into messages, so the what of the exchange is data elements—whoops, let me go back. I don't want to get ahead of myself here. Data elements is at the bottom of it and those are like the words, for example. Data frames are organization, and maybe we could say they're sentences. Messages may be more complex sentences, okay. That is the

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

what that's being transferred. Above it you'll see arrows going in two directions. On the left, there's a vertical arrow that goes up to file based transfers. These are kind of conventional ways of communicating data between computers. We're going to talk about that a little more. On the right is something that shows dialogs and dialog patterns and dialog based transfers. On the right, the dialogs, patterns, and dialog based transfers are interactive, real time kind of data exchange. File transfers are not real time data exchange.

Let's talk about the file based exchange that we had on the left. This is a kind of application in which, okay, your computer generates some data. Maybe it's a file of where all the buses are or where all the bus stops are. The computer generates that, saves it in a file. That file gets transferred to another system and the file is loaded and read by that other computer application. That sometimes involves human interaction. It could be that you actually plug in a USB drive into one computer and load the data onto it, take it out, and physically put it into the other computer. There may be other ways of transferring it as well. But this interaction requires an agreed upon description for the files. The TCIP messages provide that description.

Dialogs may be more interesting because they're real time. Dialogs specify an operational purpose. They actually have a pattern. These are exchanges of messages that we can save and store and use in many different ways. And it also includes the ability to include special conditions and constraints on it. The text that you see on the right of this slide says, publication dialog definition, refer to section 7.1, is an example of the description for a dialog, a definition that you would find in the written portion of the TCIP standard. It shows the business area where it comes from. This particular dialog is called publish fleet locations. It shows a purpose, a type of it, and it also shows some other data conditions, special conditions about this particular dialog pattern. The dialog based transfer again is real time transfer. It's like machines having a conversation. Now you've got to remember that dialogs do define, what they do define—let me go back to that. They do define the purpose, the pattern, which is the sequence of messages. They do not define how the data is stored, translated, and manipulated. They do not define how the data is formatted and presented to human users, how systems trigger or initiate the dialogs, or the details of the interactions with human users. This is strictly a transfer between machines. Dialog patterns defines the sequence of actions for a dialog, and a pattern can be reused. So once you write the basic pattern or steal it from somebody else, you can use it for a number of different kinds of purposes and we'll get into that more.

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

All right, we are now about to embark on our first activity, the first quiz if you will. The question is, which of the following is defined in a TCIP dialog? The answer choices are a) how data is stored and translated, b) how data is formatted, c) how systems present data to human users, and d) how messages are sequenced. Let's review the answers now. Let's look at the correct answer, which is d). Dialogs specify the sequential order of messages. They do not determine how data are stored and translated. Dialogs do not determine how data are formatted. We're not talking about data, we're talking about messages and dialogs. And we don't really deal with the human interaction. Again, this is machine to machine.

So the summary of learning objective number 1, illustrate the communications stack and show how it relates to TCIP. There are seven layers in the OSI model of bits flowing between computers, that communication stack. The TCIP model for data exchange includes the what, which are the data elements, data frames and messages, and the how, which are file transfers, dialogs, and dialog patterns. And TCIP can exchange data in both file transfer and real time.

Let's go into learning objective number 2. Now we're going to describe for you the TCIP implementation, requirements, and capabilities editor, TIRCE, and how it is used as the key to TCIP. TCIP is unusual as a standard, because it really not just describes what you need to do it actually has associated with it tools that you can use to implement. We're going to show you how to specify an interface. Again, the interface is the two subsystems or machines talking to one another. We're going to do it from the agency perspective, defining business applications by creating a profile requirements list, a PRL, and from the vendor perspective, testing product compliance using a profile implementation conformance statement. You see how we're thinking about this, that TCIP is going to be largely used by vendors who are going to provide products to the transit industry and these documents kind of deal with it from both sides. The interface specification, the detailed interface specifications are critical for success. Remember, we started out by saying these computers not only don't even have a language to talk to each other, they don't even know how to start the conversation. So we're going to have to teach them how to do that, and that's what the interface specifications do. They provide the ability to understand and specify exactly what information is to be exchanged across the interface, both the how and the what. We want to minimize opportunities for technical disconnects between the agencies and the vendors. We want to provide verification test criteria and we want to be able to provide for maintenance and future upgrades.

Using TIRCE for interface specification. TIRCE provides a guide that gives you a step by step process for converting an agency's functional requirements into a TCIP compliant

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

specification. It is a top down systems engineering approach and that's something if you've taken earlier PCB modules, you are aware of what systems engineering entails. I hope that if you haven't learned that, that's something that you will look at. We're going to use systems engineering to defining the data exchange interfaces between the components. Again, the information that will be exchanged and the manner in which it's exchanged.

TIRCE is actually a tool that would be loaded on your computer and there are what we call five different tabs to it. It's kind of like a TurboTax process. Tab one is profile information, tab two allows you to select components, tab three defines the interface, tab four allows you to tailor it, and tab five actually creates some output documents. So this actually gives you the opportunity to exactly put down how you want these two systems to talk to one another.

The profile requirements list and the profile implementation conformance statement, these are complementary kinds of documents. They are very similar in structure. The PRL defines the interface from the agency's perspective. What is the agency trying to get at here? What is the conversation that they want these computers to have or these systems to have? The PICS defines it from the vendor's perspective, so you can compare the PRL and the PICS and make sure that you're getting exactly what you want. So both of these define dialogs and messages for each interface in the system. Again, the profile requirements list, it generates a detailed interface requirements. It's top down systems engineering, and the interface requirements are actually embodied in the PRL. This is actually kind of an outline for the code that you're actually going to want included in this. The PRL actually allows you to define a project incrementally over time. You don't have to define everything all at one time. It does give you a document that you can quickly compare the vendor responses to the RFP. This is a really helpful tool in terms of getting exactly what you want through a systems procurement. On the vendor side, a profile implementation conformance statement, it provides a TCIP compliance spec sheet for vendors' products. Vendors may have developed software products for other agencies using TCIP and they can provide you with the PICS for it. It is a structured, precise response to interface requirements in a PRL and it allows the vendors to evaluate RFPs against existing product capabilities. And you see in parens there, there is a diff function that we'll talk about. It is a software development support tool and it actually allows you to tailor XML schemas specifically to meet your needs. The diff function is specifically included in TIRCE to compare the PRL and the PICS.

And we're ready to go to our next activity. The question is, which statement best characterizes a profile requirements list, PRL? The answer choices are: a) a PRL

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

includes a diff function, b) a PRL is developed by the vendor, c) a PRL specifies the size of the files in the data exchange, and d) a PRL is developed by the agency. Okay, now let's look at the correct answer. The correct answer is d) a PRL is developed by the agency, profile requirements list. A) is incorrect. The diff function is used to compare the PRL and PICS but it's not included in the PRL. A) A PRL is developed by the vendor, no. Typically the PICS is developed by the vendor. C) the PRL specifies the size of the files in the data exchange. No, it doesn't really talk to the size of the files.

Let's go on to the review of learning objective number 2. Describe the TCIP implementation requirements and capabilities editor, TIRCE, and how it is used as a key to TCIP. The TIRCE provides detailed specifications and XML schema, actually code for interfaces between machines that are exchanging data. The agency perspective allows the agency to create a PRL that defines the components to be connected, the interfaces needed, and the dialogs to be implemented to facilitate the data exchange. And from the vendor perspective, creates a PICS document that allows a vendor's product to be compared with an agency's requirements.

Let's go on to learning objective number 3, identify and provide examples of data elements, data frames, messages, and dialogs. So we have talked about the file transfer and dialogs. Now we're going to get into the real guts, the building blocks. A data element is an atomic piece of information related to a person, place, thing, or concept. Remember, here we have a couple of examples. This is kind of using the shorthand identifiers that you actually find in TCIP: SCH stands for scheduling system-Timepoint ID. This would be an atomic piece of information that is a timepoint alphanumeric identifier. SCH-TimepointName would be a timepoint name, and in this example, a timepoint, the identifier could be ABC123 and the timepoint name could be Maple Street at Main Street.

The third one there, LRMS.Latitude is an atomic element that specifies the latitude in micro degrees of a particular location. LRMS stands for location referencing system. That is a different standard. TCIP does not want to reinvent the wheel, so if there are other standards that provide useful data that we can incorporate in our messages, we do that. The LRMS is one example of that.

There are a number of different kinds of data element types, and they're based on abstract syntax notation, ASN.1. They're Boolean, that could be a 1 or a 0. It could be enumerated, it could be an integer, it could be a UTF string, a numeric string, octet string. They've also created some subtypes. We won't get into the details but we'll tell you where you can find all this in the TCIP standard.

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

Data frames are groupings of data elements and other data frames to describe more complex concepts. The groupings help organize information to describe or identify objects or concepts in the real world. In this example, we're showing a data frame that's called SCHTimepointiden. It is a data frame that identifies the timepoint and it actually contains two different data elements. It includes a data element with a unique alphanumeric identifier, the SCH-TimepointID, and it can also contain optional data elements including the agency number and the name. And we also can have a data frame that can describe the timepoint, SCHTimepointInfo, and it will contain its identifier and location. Another example of a data element in a data frame is the fact that an operator may have a first name and a last name. The first name might be Jerry. The last name might be Lutin. Each one of those could be a data element. A data frame might be Jerry Lutin, my first name and my last name that carry around. You may not always want to have my first name and my last name. You might want to just use my last name. So that's why we have both data elements and groupings of data frames.

TCIP messages. TCIP messages are aggregations of data elements and data frames into a larger, more complex structure. A message is really a complete understandable one-way communication. Remember, a message is a one-way communication that consists of data elements and data frames that convey metadata and information. For example, CPT, which is StoppointList, is a message that transmits a list of all stop points. So that message, CptStoppointList, could consist of data frames that we showed you before about identifying different stop points and timepoints. This is what that looks like when it's documented in the TCIP standard, and you can actually find all these messages in kind of a library for you that's included in the standard. And it provides the name of the message, it provides the use, how it's used. It has remarks and some contextual information, and it shows in ASN.1 some of the different hyperlinks that will get you the content and data frames and other data elements that could be included within the message. And it actually goes down there in the bottom, you see published timepoint list, it says, the following dialogs use this message. So this actually gives you a pointer to not only the message and its contents and how it's organized, but where it could be used in an exchange between two systems.

TCIP dialogs. This again is the exchange of messages. They specify the operational purpose of the exchange of information, the dialog pattern. The message is included in the dialog, special conditions and relationships with other dialogs. And again, this is another example from the documentation that you would see when you open up TCIP. It shows you a particular dialog here that says publish timepoint list. You can see we've been using consistently starting off with timepoints and data elements, the identifier and then we linked it into data frames, and then we also had a message that has all the timepoints in it. This is a higher level, the dialog that is a publish subscribe. We'll talk

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

about that a little more. It shows down here the dialog contents in terms of the actual messages that get exchanged. We're going to give you a diagram to make this a little clearer.

All right, TCIP includes 11 different types of dialog patterns already in the library within the documentation. We'll go over these a little bit. The first four are actually publication pattern, is the most common and that accounts for about half of all the dialogs that we typically use. There are a couple of types. There is a three message dialog pattern that defines a conversation between a subscriber and a publisher. The three messages are a query, a subscription request that brings back a single response. There is a periodic subscription and there is an event subscription. We'll see a little more when we get into a diagram. A command-response pattern. This is a two message pattern that defines the conversation between a controller and a controlled entity, a device for example. Three, there's a report pattern and this again is a two message pattern that defines the conversation between a reporting entity, such as a field or mobile based employee computer, and receiving entities, such as a supervisor's handheld device, a dispatch system, or a maintenance management system. Number four, the silent alarm pattern. This is a unique pattern specifically intended for use with transit vehicles that have a silent alarm. Very often, unbeknownst to passengers, there is a secret button or a secret way that an operator can trigger an alarm to alert authorities that there is an event going on that needs police attention.

Diagram patterns five to six. There is a load pattern, and this is especially set up to load large files from a fixed business system to a corresponding on board component, or business system. You may want to unload a whole schedule to a bus and that would be transmitted in a dialog pattern according to a load pattern. There's also an unload pattern, which is intended on the receiving end to request an unload, to unload a large file at a destination, usually a fixed business system from a corresponding source. There is a voice radio call pattern, and there are two different ones, operator-initiated and dispatch-initiated. Now the TCIP role in here is when an operator may hit a button on the console saying that the operator wants to talk to the dispatcher. That would be transmitted electronically and digitally to the control center and it would show up on the dispatcher's screen. The dispatcher may be engaged in other conversations or other business, so that's why we use this kind of a pattern, so that we make sure that we're not just interrupting, blurring out a voice message. And the same thing may be true with a dispatch-initiated pattern. Number eight is signal control and prioritization dialog pattern. This actually works to provide to support conversations between a signal priority request generator that initiates the request for preferential treatment, and a priority request server. Then this supplements an existing SAE standard that we'll talk about a little later.

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

The final three types are a blind notification pattern in which a pattern that provides for a notifier to send information to a receiver, but without expecting a response. There is a push pattern that provides for a sender of a message, usually a large file, to initiate a transfer to a receiver. And then finally, there is a special type called a traveler service request pattern, which defines a dialog to allow a transit traveler or passenger to make a request that, for example, a vehicle provide an amenity at a stop point, like a wheelchair lift or transfer connection protection. So these summarize the 11 different dialog patterns. As you can see, they're geared to very specific kinds of uses within the transit industry.

Now this is a diagram that shows you how these dialogs take place. This could be a conversation between two people. Here we have on the left, this represents a subscriber or a subscribing business system, and this represents a publisher. Here in this, the subscriber sends a message to the publisher requesting a periodic subscription and the publisher can send back a response with the actual publications, the information. So this is one example of a two message publish subscribe. There also can be periodic publications. The subscriber here sends a message to the publisher to say, "Hey, I want to get this daily." So the publisher says okay, and every day it sends out another message, or it could do it every minute, depending upon the kind of system. So these diagrams here are intended to show the two parties, the two business systems and the exchange of messages in between, that are conducted in a specific dialog pattern.

We're going to get to our next activity now. The question is, which of the following statements about data elements is correct? We're going way back to the beginning of the learning objective now, to the fundamental basis. And the answer choices are: a) a data element can include a data frame, b) a data element can include only integers, c) a data element cannot be used to identify a person, and d) a data element can be used to represent a concept. Okay, let's look at the correct answer, and the correct answer is d) a data element can be used to represent a concept. A person, place, thing, or concept. A) a data element can include a data frame. No, it's the other way around. A data frame can include a data element, but a data element can't include a data frame. A data element can include only integers. No, remember, we talked about a number of types of ASN.1 representations, Boolean strings, many different kinds. C) a data element cannot be used to identify a person. Incorrect. A data element can be a person's name. In fact, I gave you an example using my name.

Let's go on to the review of learning objective number 3 and the summary. What have we learned in this objective? We identified and provided examples of data element, data frames, messages, and dialogs. We looked at data elements which were the atomic pieces of information related to person, place, thing, or concept. Data frame, groupings of

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

data elements and other data frames to describe more complex concepts. Messages, which are aggregations of data elements and data frames into a complete understandable one way communication, conveying metadata and information. A dialog, which is a real time exchange that specifies a sequence of messages for an operational purpose and it includes special conditions. And d) a dialog pattern, a common form of dialogs that can be used repeatedly for multiple purposes. Remember, we showed two message patterns and three message patterns. There can be a lot of different patterns here that you can store and use.

Now we're going to go on to learning objective number 4. We're going to see how TCIP data are organized in data exchanges. We're going to talk about identifiers, optional items, versionings, row updates, applicability, local extensions, and multi language support. So this is going to cover a lot of the capabilities and kind of additional capabilities that are included in TCIP. There's a lot of stuff to learn here, I know.

Let's start out with a TCIP identifier. A TCIP identifier allows an agency to provide information about items, e.g., a bus stop inventory, timepoint inventory, to an application once and then refer to the item in subsequent messages. For example, if we want to associate the passenger count information with a bus stop, we can associate it with the identifier, and not send the complete description of the bus stop with each set of passenger counts. And we have specific identifiers already included in the library for a number of items that are common to public transportation: employees, intersections, vehicles, fare zones, trips, amenities and so on, routes, runs, timepoints. Any number of things have unique identifiers so we always know that we're talking about that specific item, because it is encoded as a specific identifier. The TCIP identifier and the data frame always contains an alphanumeric identifier. That is, for example here, a vehicle ID. The identification of Iden Data Frames may contain an optional agency number. It's possible that you may have systems that are exchanging information about more than one transit agency, multimodal areas where different bus companies operate, and you may need that. You may need the agency number to identify that yes, this bus is operating for a specific agency. It could also include an optional string identifier that uniquely represents the item within the agency, like a route designator. The Bay Express, for example, might be something there that you want to include. And you might have more optional string names for the identified item. Again, we've talked about the idea of a first name, last name, or vehicle identification number. There are optional items that can be included as well, different fields in our data exchange. These may be used to support a functional requirement that may not be common to all agencies. This could also be text, that's our audio or text and audio for an announcements that's going to be made on the bus. It could be actually the bus stop name in Spanish or something like that. We'll talk about foreign languages in a minute. It can support a capability that's only required in certain

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

contexts, for example, a warning and a message. You may want to say, "Mind the gap." It could contain information that may or may not be available. For example, there might be an optional field in your database that you may or may not have an optional field that actually references an image of a stop point. It may not be available for all of the stop points. And again, for this data frame, I'm showing you at the bottom of this slide an example of what's found in the library of data frames using identifiers, exactly taken out of the TCIP standard documentation. This one talks about a data frame, CPTS, that's Common Public Transit Stop point identification. Its use is to uniquely identify a stop point whether it's in a single or multi agency environment. And below it, in the ASN.1 representation, it shows hyperlinks to the sequence of some of the optional data elements that you can use as part of this data frame that's being represented by a unique identifier.

Versioning. Data items in TCIP need to be updated from time to time. TCIP's primary means of tracking versions of these items is by updating the effective date-time. And some TCIP data items also have optional integer based version numbers to provide backward compatibility with legacy business systems. Remember, for example, a stop pattern for a route could change over time. He may still hold the route number. It might be the number 100, but you may change that and you don't necessarily only hold one version in the system. You may need to have the future system, next Sunday, the route 200 is actually going to change the stops and we're going to have to get that out. So we may be holding in our system the current version that's good until next Sunday, and a next version which will take effect after midnight on Saturday night. Again, we're showing you here how it's represented in the ASN.1 and hyperlinks to the various elements that are included in this particular frame.

Row updates. Very often we are storing information in an array. And there are many TCIP messages provide lists of items that could be lengthy. At my agency, we had 17,000 bus stops. Maybe 3,000 timepoints. Some agencies could have up to 40,000 employees or 6,000 vehicles. You don't necessarily want to re-transmit the entire array of all that stuff, so a row update allows you to make changes to a previously provided list without repeating all of the other things that aren't changed. Here's an example. We have a list of stop points and we have a new stop point that we are going to introduce, stop point 234. So what we're going to want to do—and this is shown in a particular sequence. This may be the sequence of stop points for a particular trip along a particular route. We're going to insert stop point 234 between stop point 211 and 245. So that's exactly what happens here. We provide a pointer, we provide the actual information about the stop point, and we update that particular list. So a row update dialog would include a message requesting row updates that includes a field indicating the date-time when the changes are requested. It will also have a message providing the row updates that

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

contains a field indicating that the changes are provided. Again, this is in a publisher subscription kind of a mode. The row update message may contain the rows to be added, rows that might be replaced, or rows to be deleted, so you can have all three possibilities that could be dealt with within this data exchange.

Applicability. Just like we have row updates and changing lists, we may be sending out messages to a bunch of different buses or something like that. Here's an example. We have six buses in our fleet here. We have a TCIP message that may contain an applicability field indicating the scope of information requested or provided by that message. And the applicability field, if present, usually occurs in the message application header. Here, three of our buses are going to be affected by a brake recall, so we want to send them a message to put on the list that as soon as that bus pulls in the garage, we've got to take it out of service. Three of the other buses, actually four buses here, need to have their maps updated. So it may be that when those buses come into the garage, we need to flag them so that when they get close enough to download the map data wirelessly through the garage server, that that happens. So the message will go out to our fleet and it will have an applicability header saying, okay, the brake recall applies only to bus 212, bus 559, and bus 712. They may have been the ones we bought that particular year. Then the update message for the maps may apply to bus 1232, 834. Oh, 559 is also subject to the brake recall, so it's got two things that are going on, and 933. So this is an example of how TCIP can be used to deal with special applications and send messages out, blanket messages, but they only apply to certain vehicles within the fleet.

Local extensions, this is another one of the flexible ways that we can use TCIP. The type 1 extension is a code extension. It can add a value to a specified enumerated list of values. For example, we may have a service type that describes a special type of transit service and we have a couple here. There are certain codes, we'll show you an enumerated list here. For example, we may have standard codes up to a certain point, but then we might be a special service here, the Bay ferry connector. We're going to give that an enumerated value of 128 in the type of service types. Then we may have a onetime event, Super Bowl, and we're going to give that a code of 129 and we'll just take a look at this. Here is how that gets applied. TCIP comes with a number of enumerated service types, regular, express, circulator, radial feeder, jitney, limited service, non-revenue, charter. Some of these are typical and so they're already provided. You don't have to decide on what to code up yourself. You can use this from the TCIP library. But if you do have a need, like we showed you, for the Bay extension to meet the ferry or the Super Bowl, down here you have the option of using one of the codes where we have placeholders, from 128 to 255, for local use and they would have local content.

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

A type 2 is a data frame message extension. A data frame message extension can add locally specified data to the sequence and here we have a data frame that's called a CCBLOCKWorkRecord. The block is the record of what the bus does in a particular day. It describes the events related to work done by a PTV. It may specify the timepoints it's supposed to hit, the routes that it's supposed to do. And we may want to add something special here. In this case, let's say that this particular bus is equipped with an air quality sampling device. So we may want to add an instruction to that bus, the computer system and the data management system on that bus, to download air quality samples throughout the day. So this kind of provides the ability to add that into the work record.

Multi-language support. I already mentioned to you, we might have the name of a bus stop in Spanish and English. TCIP can support multiple languages, beginning with TCIP version 3.0.4. You may be using an older version of TCIP. You have to update to this version or a newer one. There is a list of all of the languages that are possible to use, using the ISO 639 codes. That's a standard kind of a list. And there is a data frame that can specify the default in additional languages that you want to use for your system. It is an optional item and you can also put in string content in a message, and that would be under CPT additional language content, that data frames actually provide the information that you want in those additional languages. So this can be used for canned messages in multiple languages.

All right, we come to another activity now. It's a chance for you to think about what you've been learning and see how well we've assimilated all this information about TCIP that we have been talking about. Here's the quiz. Which statement best characterizes a TCIP identifier? And the answer choices are: a) uses a shorthand code to represent an item, b) identifies a TCIP generated message, c) identifies a unique data frame, and d) always includes a string designator to identify an item. Hmm. Okay, let's look at the correct answer, the review of answers, and the correct answer is a) correct. A TCIP identifier can be used to represent an item that may have a longer description. So it is a shorthand code for an item. It represents the items in the real world, it doesn't represent TCIP elements. Incorrect, it would not be used to represent a TCIP data frame. An identifier is a data element. Incorrect, string identifiers are optional types of identifiers. Alphanumeric identifiers are the mandatory parts of an identifier.

Okay, and to summarize this module which covered a whole host of different topics. I think the important thing is to remember that we have a lot of flexibility, a lot of capabilities. So data, we can use TCIP to create unique identifiers for information across all transit agencies. We can identify versions of information that are provided. We can incrementally update very large datasets and do it pretty efficiently. We can define and

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

use subgroups of items and different entities, and we can include locally defined data in TCIP messages. What's the bullet not included here? Well, we have a multi-language capability as well. Let's go on.

We're now going to turn to learning objective number 5, defining a profile requirements list and explain how it is used to specify TCIP requirements in a transit ITS project. So we're turning from some of the structural capabilities built into TCIP in learning objective 4 and in this one, we're now going to talk about how we develop an interface and how do we develop the procurement document that we can use in order to specify it for a vendor. Within the PRL, we're going to talk about profile information, component selection, interface definition, interface tailoring, and document creation. Remember way back in one of the earlier modules, we showed you the five tabs of TIRCE and here they are. We're going to go back into them in more detail.

Profile information. What I'm showing you here is actually a screenshot of what it looks like when you open up TIRCE to begin developing a project. It gives you an opportunity to put in your own project name for it. This project might be, we're having a CAD/AVL system that's going to talk to a passenger information system. So CAD/AVL to passenger information, whatever you want. Agency, company name. We actually include in TCIP all of the Federal Transit Administration known transit agencies. There are hundreds, if not thousands, in there. You can probably find your own agency name there or your client agency real easy. There's a corresponding ID. Model is basically whether you're operating in a multi-agency environment, or whether—this is kind of standalone. I don't have any other transit agencies that are working here within the next 100 miles. It's just one city. I don't need to deal with that kind of thing. So this is what it looks like when you open it up to put in the profile information.

The next part was the next tab of TIRCE, is component selection, what's talking back and forth. In this example, we have an example of a CAD/AVL system over here that's going to send the fleet location to the traveler information. Remember, these are machines that have no idea how to communicate. They don't know the protocol, they don't know the language. What we're trying to do is teach them all of that through TCIP. And we're doing that in a very, very detailed scale here, because we have to do it exactly. We want to have the CAD/AVL system be able to successfully send to the traveler information system a data file that updates where all the buses are and it could be done every two minutes, every five minutes, whenever. So that's what we're talking about here in this particular example. Where do I find these? Well, in module 3 where we talked about part one of TCIP, we showed you a TCIP model architecture, which was a diagrammatic representation of all the different kinds of business systems. Within that, there are a

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

number of the basic business systems that you might want to use to build systems for a transit agency. You can actually pick out the components from a menu or a diagram that's supplied within TIRCE. So it actually allows you to pick out the components that you want to have. You want to say, I want Joe to talk to Harry here and exchange this particular information. Remember, these systems, they're pretty dumb. They may be complicated in what they can do inside, but they don't necessarily know how to talk to each other. So here, this is the situation. We are going to use a dialog. Remember, this is a sequence of messages and that dialog is already provided for you in a library within TCIP. It's published fleet location, so the traveler information system, when it comes online, is going to send a message, the first message in this dialog, CC fleet location subscription. I want to know where all the vehicles are and it sends that message to the CAD/AVL system. The CAD/AVL system can actually have two possible responses. It can say, "Here they all are, buddy. I know where they all are and you got them." On the other hand, it can say, "Wait a second, this system may not be working, or maybe the radio system is down" or maybe a lot of different things, so it would send back a message. The other possibility is an error notice. "Sorry, I don't have it." The traveler information system might be programmed to say, "All right, I'll just use the one I had from before." It might be programmed to say, "I'm just going to tell people I don't know."

TIRCE is going to actually produce a document that you can use. This is a cover sheet for the document. Remember, the substance is really going to be XML schema that you actually have encoded. This is the cover for the PRL. It talks about the project name, when it was created, the agency, and a lot of other metadata about the project. It also talks about some of the specifics. This is a CAD/AVL system talking to the traveler information system. This is what it looks like and all this comes out of the TIRCE tool when you start using it to specify a profile requirements list. I just want to show you an example of the XML encoded TCIP message that's produced. I'm not an XML coder, but those of you who are may recognize some of this. It does exist within TCIP. It's actually providing you XML schema.

Okay, we now are at our next activity, and learning objective number 5, the activity question is, which of the following is not included in the preparation of a profile requirements list, PRL? And the answer choices are: a) profile information, b) component selection, c) vendor conformance, and d) interface tailoring. Okay, let's look at the correct answer. Remember, the question is, what's not included. Let's look at the answers now. What is not included is c) vendor conformance, correct. That is not in a PRL. That is something that is produced when you compare the PRL and the PICS using the diff function. Yes, it does have profile information in the PRL. A lot of that we showed in that cover sheet. Component selection, yes, remember, we were talking about CAD/AVL

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

talking to an independent system, traveler information, and interface tailoring. What is interface tailoring? Well, we'll explain that a little further. We're not making a suit here.

Summarizing learning objective number 5, defining a profile requirements list and explaining how it is used to specify TCIP requirements. Remember, the profile information includes entering data to describe the project and identify the agency developing the profile. It includes component selection and we had a simple example of one component, a CAD/AVL providing information to one subsystem, a traveler information system. Interface definition involves selecting the file transfers and the dialogs. We actually showed you the published subscribe dialog. And tailoring involves selecting specific messages to be exchanged. TCIP will bring up a long list of messages that could be included in a dialog. You actually get to tailor it by checking off the ones you want and the ones you don't want. The final step is document creation, which produces the PRL specifications and the XML schema.

Let's turn now to learning objective 6. We're going to articulate and describe the uses of each tool in the TCIP suite of tools. We talked about TIRCE, but we've got some others down here that we haven't talked about, message builder, interrogator, responder. So we have talked about TIRCE and interface definition. We talked a little bit in procurement, we're going to lay this out better for you, I think, as we go along. We're going to talk about how we can develop TCIP messages and dialogs using the message builder, interrogator, and responder, and how we can verify that the systems are exchanging messages properly.

The tool suite includes, as you already know, TIRCE. It includes message builder, interrogator and let's talk about this. The first thing here is in building TCIP interfaces starts with interface definition. We then go through procurement. We can then develop messages, and we can verify them. So in the middle here is kind of a diagrammatic representation of the tools that we use. For interface definition, we use TIRCE. It defines the requirements, it specifies the interfaces, and TIRCE produces TCIP compliant documentation. In the procurement process, TIRCE provides us with an automated comparison tool that gives us the ability to compare the vendor response with actually the requirements we put out in our procurement.

The development phase, which we're actually looking at implementing our messages, is using the TCIP message builder, the interrogator, and the responder. We actually call the interrogator and responder, the test console. This gives us simulation capability. It gives us the ability to generate messages and we can log them and actually view the log messages. Finally in the verification stage here, we have the ability to automate

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

compliance verification of the messages and to make sure that they've been transmitted and received properly. Let's go through, starting from the top, the interface definition uses TIRCE. We have, remember the five tabs. We specify profile information, we select components, we define interfaces, we tailor interfaces, and we produce documents.

The next use of TIRCE is in the procurement where we have the profile requirements list and the profile implementation conformance statements and we're going to use the diff function to compare what the agency requirements were with the vendor product. You can see that's where we are in this chain here of using the TCIP tools.

The TCIP message builder actually enables you to generate XML encoded TCIP messages. These messages can be used by the test console. You can also load, save, view, and edit TCIP messages. You may actually already have components that are generating TCIP messages. You can save them in this and the message builder will allow you to store them, the messages that you already have. You know, you can actually borrow messages or steal them from somebody else who developed them. I think that cooperating agencies who are developing TCIP compliance standards can very much benefit from talking to the other agencies who've already been through it, and we can save a lot of effort through that. This is an example of the screenshot of what the message builder looks like. Again, many of the messages here are already stored in a library and the message format allows you to input your own data. For example, here, this particular message for fleet point location has space for an expiration date. You can enter that, or your system, another system could enter it. Expiration date, expiration time, so you're actually populating the message fields that have been created in the library of the TCIP messages.

The test console includes the interrogator and responder. The interrogator requests TCIP. Remember, this is a subscriber, and a TCIP responder supplies the TCIP data. Again, this is two way communications. Somebody's generating messages, somebody's receiving it. The interrogator can simulate a TCIP compliant device. It stimulates the client side. It subscribes to a publisher of data. It will establish a TCIP communications link with a device under development or test. It can transmit or receive TCIP messages. It can view and log the messages, and it provides you with the automated message verification. The responder simulates the other side of the exchange. It simulates the server and it provides TCIP data upon request of a subscription. The data can be preloaded. Again, it could be prerecorded TCIP messages or sample messages that you've built and populated with sample data.

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

This is a diagrammatic representation of how the interrogator and the responder can be used. This is testing a component. For example, here we have a unit that's being tested. It could be a logical unit, it could be a physical device. Here we have a TCIP message builder that's part of it. We'll start back here with TIRCE. TIRCE actually created the PRL. The PRL includes the message structure. That can then key the message builder to send a message to the interrogator, which then generates a TCIP request to the unit that's under test. Maybe this is a CAD/AVL system and this is a traveler information system, or what we think may be a traveler information system, or it could be a data repository. It's sending a message to the CAD/AVL system here saying, "Hey, send me something. Send me a particular file. I want to know where all the vehicles are." This unit that's being tested now is going to be asked to provide the TCIP response. So we're going to issue the request, we're going to log the response, and we're going to verify that the whole exchange took place in accordance with the TCIP standard. We can also simulate a TCIP component. We could actually simulate both ends of the exchange, and that's what's being shown here in this diagram.

Okay, let's do an activity. Which of the following statements about the underlined TCIP tool function is false? A) the interrogator simulates a client component requesting data, b) the responder simulates a server providing data, c) the test console corrects errors in XML schema, and d) the TIRCE diff function compares a PICS with a PRL. Okay, let's look at the correct answer, and the correct answer is, which is false remember, c) correct. This statement is false. The test console does not correct XML errors. Okay? Interrogator, this statement is true. So remember, we're looking for a false statement. It does simulate a client side. This statement is true as well, it does simulate a server. And the diff function is true as well.

Let's summarize learning objective number 6. We're going to articulate and describe the uses of each tool within the TCIP suite. We talked about TIRCE and how it can be used to specify interfaces. We also talked about how TIRCE can be used to prepare procurement documents. We talked about the message builder, which can be used to construct TCIP compliant messages and it gives us a place to set up a storage log of messages that maybe we've already created. We can use them multiple times, test out multiple components. And the test console, which includes the interrogator and the responder, allows us to simulate one or both ends of a TCIP interface. This is a really good suite of tools to allow you to use TCIP. That's why this standard is different from a lot of the others. The standard is one thing but it is accompanied by these tools, which are not part of the standard, but are part of what is the kit that is provided to allow you to use it effectively.

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

Let's go to learning objective number 7. This is the final learning objective. So we're going to summarize the range of TCIP applications, implementation tools, and additional training. We're going to again specify something we talked about in module 1, the differentiation between non-normative TCIP content and normative content. We're going to talk about examples of where TCIP is being applied at transit agencies and give you a little overview into some more free on-site training that you can have.

Non-normative TCIP content. What we talked about in module 3 and showed you again here is the model architecture and concept of operation. These are things that are provided in volume one of TCIP that can give you an enormous leg up on trying to develop the interfaces. They show you the different components that you might want to connect using TCIP interfaces, and the concept of operation gives you examples of where the data flows take place, how a particular component might want to be connected through interfaces to a number of other components. Very helpful. And so remember, non-normative model content includes the TCIP model architecture and the concept of operations. Something else that's non-normative are the tools. You don't have to use the tools. Non-normative content, the model architecture and concept of operations, where do I find it? The American Public Transportation Association, Transit Communications Interface Profiles standard development program, Volume 1. You can get it by downloading it free. You're not going to remember this hyperlink. All you have to do is Google APTA TCIP documents, downloads and it will bring up that hyperlink.

Normative content. What is the normative content? Normative content is that which is actually used in the messages and in the data exchanges. Where you find certain information, it's in Volume 1. That explains the basis for conformance and in Volume 2, there are four annexes that are basically libraries. Annex A has a library of data elements. Annex B has a library of data frames. Annex C has a library of messages and Annex D has a library of dialogs. Interestingly, these are all linked together with hyperlinks. So you click on a data frame and you may find out what data elements are included. You may find out what messages that data frame has used. TCIP Volume 3, the TCIP XML scheme, Annex E includes the actual XML schema that you can use. So these are libraries of stuff that's already created and these are normative elements. They are actually part of the standard.

The tools, remember the tools are not part of the standard. You don't have to use these. You may have a wiz who's found some different ways or you may just want to write things from scratch. Where do I get them? The tools are TIRCE, message builder, and test consoles. They come from the TCIP standards development program from APTA. Again,

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

they are downloadable from a long hyperlink that you don't have to remember. Just Google APTA TCIP tools and you should get directed right to the proper location.

When you do get to these places, and you're going to bring up some windows here, and this is an example of a screenshot showing the TCIP setup, installation instructions. It's real easy. You just check off what you want. You click install, everything gets downloaded pretty quickly and you wind up with these three icons on your desk. The TIRCE icon, the message builder, and the test console, which includes the interrogator and the responder. So all this can then be put on your desktop and you can use it right away.

Okay, who is using TCIP? Let's give you some real world examples. There's six examples that we're going to show you. The first is LYNX in Orlando. This is an example of an agency architecture. All of these different yellow rectangles indicate different business systems. They may be physical entities or logical entities that are communicating information in order to manage the transit system and provide information to the customers. In one part of it here, we have been adding some different new components by a particular vendor who is showing, in this particular diagram, by the red lines, message flows—these are going to be real-time flows—that are using TCIP. You don't have to use TCIP to replace legacy flows in an existing system, but you may want to do that, or you may want to actually use TCIP to add new components and make some components easily replaceable. That's what LYNX is doing here.

The next one here is King County, where they are using a TCIP message to actually augment and notify a transit signal priority request generator. The actual requests generated that actually go to the traffic signal controllers use another standard, an SAE 1211 standard, but TCIP can convey messages to the request generators saying, "Hey, I've got a full bus." You may need to say whether the bus is empty or full. You may want to say how much it's behind schedule or give other information that the request generator can use to make a decision on whether it's going to request traffic signal priority for you. So that's how it is used here in King County, to provide supplemental information.

This is an example of the largest transit agency in the United States, the MTA, which has about 6,000 buses. They have a concept architecture of their customer information system. On the left we have the bus, the vehicle, which has a head sign, a payment terminal, enhanced GPS, and it has a wireless 3G modem that is using TCIP in conjunction with JSON and HTTP, other standards to transmit the location of buses and it's transmitting the location of buses, about 5,000 buses, in a matter of seconds. So it is providing that to the customer information server. They also have implemented TCIP XML with another communications flow that's going to the bus server with crew dispatch

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

data over here. So that's an example of the MTA using TCIP for real time fleet to base information.

This is an example showing how two centers, this is a center-to-center exchange of data, between two agencies in Montreal, Canada. This pyramid in the middle here is actually a kind of a representation of something we talked about in learning objective 1, the communication stack. What this shows is that we have communications elements here using Ethernet and the TCP IP protocol for Ethernet communications as a base. TCIP is on top of that and has dialogs, and then we have XML exchanging particular messages between these center-to-centers. It allows their respective CAD/AVL systems to communicate with one another. This is a more detailed view of that, showing where TCIP is applied and it is here in the graphical user interface stage between a database and a firewall that communicates out through the physical layer of communications and the communication stack here through another firewall. It's decoded using TCIP here, and then that provides information to a web server, which is going to feed web applications developed by third parties, and it's also going to feed the database of the other transit agency.

This is another example here from WMATA in Washington, DC. Interestingly enough here, TCIP is actually providing a translator function. It is reading data that's been produced and stored in data systems and it provides a common translation using TCIP data elements, building blocks that go to a test console, subscribers, and third party applications. So it actually provides now a standardized interface to exchange information with all of these different possible users of the WMATA data.

So that concludes what we showed you in terms of the possibilities for different uses of TCIP. Now I want to talk to you a little bit about the National Transit Institute course. It's called Integrating Data Applications: Defining Data Interfaces Using TCIP. It is a two day, hands on course. It covers material presented in both of the online modules that we have presented here, module 3 and 4, and it gives you a chance for hands on instruction. All you have to do is go to [www.ntionline.com](http://www.ntionline.com) and it may be that you can host one of these at your organization.

Now let's get to our final activity. Now we are going to go back to the very beginning of what we talked about in learning objective 7, distinction between normative and non-normative. So the question here is, which one of the following lists includes only normative elements of TCIP? And the answer choices are: a) dialogs, TIRCE, model architecture, b) TIRCE, dialogs, XML schema, c) dialogs, XML schema, data elements, and d) model architecture, dialogs, XML schema? Hmm, only normative elements, which

## Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two

one? Okay, let's look at the answer. The answer is c) correct, dialogs, XML schema, and data elements are all normative, all provided in the annexes in TCIP. Incorrect. The model architecture is not normative and TIRCE is a tool, but it's not really part of the standard. B) is incorrect, TIRCE is non-normative. Dialogs and XML schema are. And d) the model architecture is non-normative. So remember here that the dialogs, XML schema, data elements are all normative parts of TCIP.

Well, let's summarize our final learning objective, the range of TCIP applications, implementation tools. First of all, as we mentioned early on, and again in the quiz, TCIP contains both normative and non-normative information in four volumes. The normative is actually part of the standard. It's standardized so that everybody can understand it. Non-normative is contextual, helps you understand the environment and to use TCIP and it also extends to the fact that we have tools. TCIP does include a set of tools to implement the standard using XML schema, and you can also use it for ASN.1 notation. It is not solely limited to XML. TCIP documents and tools are downloaded free from APTA's websites. A number of transit agencies are using TCIP in a variety of ways. And finally, the National Transit Institute offers a two-day onsite TCIP training course. That course is free to agency personnel. There is a charge for consultants, but it's free if you work for a government agency.

What we have learned. Okay, the fill in the blanks part of it. TCIP is used in the top layer of the communication stacks. This is a long one. The blank is a key tool for use in transit ITS procurements. Well, I would say it is the TCIP implementation requirements and capabilities editor, TIRCE, is the key tool for procurements. The building blocks include data elements, data frames, messages, and dialogs. Four, messages are the highest level for TCIP encoded data in two things, file transfers, uh-oh. Okay, and then must be dialogs. File transfers and dialogs. A blank is used to specify an agency's TCIP requirements in a transit ITS project. A profile requirements list, PRL. And the suite of tools includes TIRCE, message builder, interrogator, and responder.

There are a number of resources that we've used. These are all in the bibliography here. Some of them are downloadable, if you would like to read more. And there are a number of modules that you can go on to study as part of this professional capacity building program. Module 5 is transit management standards, talks about the different kinds of systems, part 2. There is a part 1 of that. Traveler information part 2 of 2. Arterial management and transit signal priority and that again is the second part. Electronic fare payment systems, and module 11 talking about transit and the connected vehicle environment, emerging technologies. So there are a lot more learning that you can do in this webinar format. Okay, I want to thank you for completing this module and we do want

## **Transit Module 4 - Transit Communications Interface Profiles (TCIP), Part Two of Two**

your feedback, so please click on the link to get us feedback or provide us our feedback using the attached link. This concludes our webinar for today. Thank you.

#### End of Module4\_Final.mp4 ####