Module 19 - C101: Introduction to Communications Protocols and Their Uses in ITS Applications
Course Transcript
(Note: This document has been converted from the transcript to 508-compliant HTML. The formatting has been adjusted for 508 compliance, but all the original text content is included.)

Nicola Tavares: Welcome to the ITS Standards Training. This course is sponsored by the US Department of Transportation ITS Professional Capacity Building Program. The IST PCB Program is part of the Research and Innovative Technology Administration's ITS Joint Program Office. You can find information on additional modules and training programs on this website, www.PCB.ITS.DOT.gov. Thank you for participating, and we hope you find this module helpful. This module is C101, Introduction to Communications Protocols and Their Uses in ITS application. Throughout the presentation, this activity slide will appear, indicating there is a multiple-choice pop quiz following this slide. The presentation lecturer will pause at each quiz section to allow you to use your computer mouse to select your answer. There is only one correct answer. Selecting the "submit" button will record your answer, and the "clear" button will remove your answer if you wish to select another answer. You will receive instant feedback on your answer choice. Please help us make even more improvements to our training modules by completing the Post-Course Feedback form. Your instructor, Dr. Raman Patel, has been actively involved in the ITS Standards Development and Standards Training Program and has served as Chair of ITE's Standards Committee for the past 15 years, and he is a founding member of the NTCIP and TMDD Committee. Raman has 40 years of experience in the transportation field, close to 30 years at the New York City Department of Transportation as Chief of System Engineering, and seven years at Parsons Brinckerhoff. He's currently teaching ITS and Transportation Engineering at NYU-Poly in Brooklyn, New York and works as a road safety consultant to international organization. The next voice you hear will be of your instructor, Raman Patel.

Raman Patel: Hi, everyone. I'm Raman Patel. Let's begin with the target audience. The target audience for this C101 module is telecommunications specialist, system administration people, and anyone who is involved in network management process of putting an ITS project together, including the engineering staff preparing the specification and acquiring the systems for ITS that involves multiple devices perhaps. Also, the developer, system engineers, and implementers who have responsibilities to put these systems in place and then, finally, the topic management and operation people, who will actually use the system to carry out management strategies. The recommended prerequisites for these modules are two, I101, which is a basic overview of what the standards are and what they entail, what is included, how to use them sort of, and then A101 also had actually leads us into a detailed discussion on how to acquire standards-based ITS systems. There's a curriculum path based on non-systems engineering process. It's called SEP, non-SEP. Non-SEP has certain modules that were developed, earlier modules, that did not use the SEP process. And, generally, what that means is that we do not have the benefits of user needs and requirements in those standards. There are several of them here in series listed. The top three prepares us for introduction and background. Then 200-series standards, here listed in the middle, or two-plus-two, provides us how to acquire standards-based systems in A201. A202 is more detailed in how to actually identify and then write user needs. A103 leads us into the requirements part of the process and prepares us on the next module, A203, how to write it. This

particular module that we are discussing today, C101, is a basic introduction to the communication process protocol that is a main thrust of this module and how to use this communication protocols in these different devices and other standards that we are talking about. So, basically, the C101 module that we will discuss today lays a foundation for two additional modules that follows C201, which actually introduces the SNMP, the Simple Network Management Protocol, which is a kind of workhorse or a central protocol that we've been using in NTCIP. So we'll go a lot more in detail in that modules, and then that module, in turn, will help you prepare for how 300-series device standards are utilizing SNMP in that process. C202, on the right, is about how to communicate information from one center to another, as we call it, Center-To-Center, and it gives in details about the XML-based messaging process that NTCIP 2306 is utilizing. This standard also uses the two additional modules sort of like for the data concepts called Traffic Management Data Dictionary Standards, A321A, which deals with user needs, and A321B deals with the topic management-related requirements for that standard. There is also the new module that's available or will be available in this series is T321, which deals with Center-to-Center testing process. So there are four learning objectives for this module, C101. The first one is to be familiar with the basic terminology used in communication process in the NTCIP. The second learning objective will explain how the NTCIP framework fulfills different requirements we have to make up our operational needs in ITS. Learning objective three specifically goes over what the Center-to-Field applications are about and how we use the NTCIP standards and also what those standards are. The last one, learning objective for Center-To-Center, deals with the information exchange process between two centers. So at the end of this module, you should have a very good understanding of how the communication process is carried out and what it entails in both different categories that we have, Center-to-Field and Center-to-Center. So let's go into learning objective one, be familiar with basic terminology. So what is the protocol? What is the standard? How does the communication process work? In fact, how was NTCIP put together using the Open System Interconnect-Reference Model from ISO? An ISO is an International Standards Organization. So we will learn how that process came into being, and then we will also go into a lot more in detail about what the NTCIP framework looks like. So those are the objectives that we will do. Let's ask ourselves a question. What is the first thing we do when we meet someone? Well, if we are sitting, we get up and we greet each other with a handshake. We begin conversation, perhaps ask, "How are you?" Well, computers also observe certain protocol, and while they don't shake hands physically, they do observe the rules and conventions to begin conversation or communication between those two entities, and that's what protocol is about. There's a formal definition from IEEE. A protocol is a set of conventions that govern the interaction of processes, devices, and other components within a system. So using that definition and expanding it in an example here, we can see that a protocol is a specific set of handshaking rules, procedures, convention, the planning the formats for data, sequence, timing of data transmission between the entities or devices. And this definition that we have here actually leads us into a little more clarity in the sense that you have a management station at the central location and then you have a device in the field, so how do we communicate to that device from the management station. Well, the protocol actually guides us in terms of deployment of data, then how we transmit that data in particular sequence, and the timing. So these are the general concepts behind protocol. There are three parts. To be specific, there are three parts to a communication protocol. Syntax is the data and the data format rules. This form, the Protocol Data Unit, PDU, as we call it throughout this module. A PDU is a core concept of the communication protocol. It is actually the message's main content, and as we call it

sometimes, payload. The semantics are the rules, the transfer rules. How are we going to move that PDU from point A to point B, right? So the semantics deals with the rules, transferring rules. They also contain information if there are errors. How do we handle errors? How do we move the information, in particular, following particular rules so both ends understands how this process works. The Timing, the third part deals with performance requirements, how fast the data is going to get to the other end. Things like that are covered in timing and the response mechanism for each message. So what is a standard? A standard can be defined as rules for exchange of a message for data and establishes a format, so both a mandatory definition of the data elements and the sequence of exchanges. This definition here also leads us to realize that not everything about standard is mandatory, but most of the part of the standards are mandatory, and they are called normative, meaning that they must be complied with. Those requirements are built into this definition. Also, certain information in the standard is optional. It's left to the user and it becomes informative for us. So there are two parts to the standards, Normative, which is mandatory, expect. And optional, which is informative, that the user has to deal with. So if you read this definition, you will get away with the understanding that it is a definite structure. It defines a particular process that we have to follow in a standardized way. Here's an example that does this. This is a standard that all of us, most of us in the industry are familiar with. This is the standard that we cannot be without in traffic management, and it is both informative. It has a lot of information in, but it's mostly normative part that we are referring here in the sense that a stop sign, for example, stop sign is not left to anybody's definition, but the standard itself provides us exact definition and says it shall be like this in octagon. Or color-wise, it can see red and white letter. So these are the kind of two bits that make up a particular requirement in a standard. So MUTCD is a core standard in the US, and it has created consistency all over traffic management devices in the country. No matter what localities you are in, you are complying with the MUTCD requirements. So here again the standard helps us to bring this consistency in our practice. Now, ITS standards deal with data information structures and communications, and some standards also deal with hardware, but generally speaking, data structures and communication is a discussion point today here in this module. So what are the benefits of ITS standards? Well, benefits can be looked from different perspective. The owners might look at a benefit in terms of how all these different devices are working together, how they're interoperable. So that's a key benefit for the owners, people who manage their infrastructure and operate. It's very important that different devices work together. That's what standards help us do, attain. The second thing is that procurement process, which is everyone's business. We want to procure devices which can be made available from more than one vendor, so we want to create and work within the multi-vendor environment. That brings the cost down. If we have one vendor, the cost will be according to the conditions that the vendor will impose. If you have multiple vendor, the market will decide. So something like that tells us that the procurement process is a key when we have standards in our favor. Also, standards defines the interface design details. All the NTCIP standards we will discuss today have the design details in that. And then we benefit from their design details. It also reduces the work for a system manager. You develop the system once, and perhaps you can utilize more than once. You develop the software device drivers. So it also tends to reduce the work that we are constantly engaged in. So this is a very good benefit for system development perspective. And, finally, standards also help us make a whole out of parts, as we say sometimes. The connectivity that we are looking for in the transportation sector is coming through this process of ITS standards. We are now finally able to bring all of our different devices together so that we can make a whole out of parts. So what does communication protocol

and compatibility mean to us? Well, compatibility is the ability of two or more systems or components to perform their required functions while sharing the same hardware or software environment. This definition from IEEE is very important for our work here in this module because without compatibility, devices cannot coexist. For that reason, the common communication protocol plays a very important role. When both ends use the same communication protocol, we achieve compatibility. System work with each other in a harmonious way without interfering with each other. So that's what we take away from this definition of compatibility. Let's illustrate this. Well, here we go with one vendor supplying us to original equipment, for example. Now, we have different types of devices coming in at a later date, and perhaps we procure that from Vendor B, Vendor C, Vendor D, and that environment we had now making all this different devices compatible with each other using the same communication common protocol. So that's a very good benefit. So we start with the small standalone system, and we gradually upgrade to more than one device and build our system in a modular way because the systems are compatible. So how do we achieve interoperability? The interoperability-- again, definition from IEEE helps us-- interoperability is the ability of two or more systems or components to exchange information and use the information that has been exchanged. So interoperability expands for expectations at the very high level so that the devices can communicate with each other and exchange information and then the user, in fact, uses that information for a very productive use. So second bullet here, we can understand that both ends have compatibility and use the same message definition for the information level. Then they achieve their interoperability. In other words, interoperability is dependent on the system being compatible, the two different devices being compatible. All right? So we need combination of compatibility, meaning the communication protocol has to be the same, and we also need the same message structure, the data definitions, so that the information can be consistently interpreted. So this is the central thrust of our discussion. Interchangeability brings the product in front of us in the sense that, okay, if I have this kind of hardware and later on I buy something from somebody else, can I swap it? Can I interchange it? Will it continue working the same way previous hardware worked-- previous device was working? And that is very important because we want to maintain the performance and the durability of our functionality in the field. We want to continue providing the same service. Sometimes there are issues with interchangeable devices, you know? However, if we have devices that follow the same communication protocols, at least we don't have to do messy changes in the software and all those other different things that we look in summary. So that's the interchangeability issue. Let's look at what a dialog is. Well, both humans and computers exchange information in orderly conversations called dialogs. That's a very good definition because orderly conversation is what dialogs are about. All right? So dialogs are a sequence of exchanges, messages, that is. The dialogs carry out a specific purpose, and to conduct a dialog, we need a connection first between point A and B here in this sketch we are showing. So we need to establish connection, and then devices will conduct the dialog from each other's standpoint. Now, let's look at the benefits of common protocol. The keyword here is the procurement through multiple vendors and see how it brings us cost savings with the common protocol user page. All right? You have a management station and you start out with remote capabilities. The common protocol allows us to manage our devices remotely. We start out with the one device and we can do that remotely without going at the local junction or the previous segment. Wherever the device is located, we can do the control and monitoring from the management station, all right? That's one device at a time or fewer devices. Now, we expand system, and most idea systems, as we all know, are not installed once. They gradually take

shape. So it's much easier for us to expand from one or two devices to more than two devices. So the expansion of our systems can visually take place in a very systematic way. That's a very good benefit that we have with the common protocol first. Then we have more than one device, and the idea is to mix some devices that we use today. So, for example, we have a traffic control system, and then we are now adding a dynamic message site. So another standard coming from the industry is now also working in unison with the traffic controller standard, for example. So we have combination of industry standards working together through common protocol. Then we have this final outcome, which we all very much appreciate. When we acquire a device, we want to be able to be able to acquire it from more than one vendor, right? So that's a marketplace issue. Other vendors can supply the same device with the same functionality, and we can get the lowest-cost vendor. So that's one particular aspect when you look at this common protocol. The other one is the market itself. The market becomes compatible, and if you are a vendor or supplying a product, you will be also concerned that I can sell my product now to more than one customer. So it also expands the market, so there is some nature of compatible market being healthy in terms of how common protocol play a part in that area, as well. So what is a communication process? A communication process handles transmission of information of data between point A and point B, and it involves systems and different devices, and it requires protocols. So communication process has two entities in it. First is the common communication medium. You've got to have same medium so you can have- different devices can work on it and operate on it. And then you'll need a common protocol and common language. So there are two requirements. One is common medium and the other one is a common protocol and common language. So this combination is what communication process is made of in ITS. So we turn to this Open System Interconnection, OSI, reference model. And there is a compelling case why we turn to the Open System Interconnect. This is a very complex mechanism of communication. There are compelling reasons why OSI has been chosen by- or was chosen by NTCIP for our ITS applications. But take a look at it. In the old days, we had a communication process (that) was closed process, was considered as monolithic, very complex, unstructured, and only the person who put it together would know about it. And if you want to make a change or you had to write a driver, a device driver or something, you would have to go through a very complex process and perhaps costly, as well. So from there, we have now migrated to a structured approach, from monolithic approach to a structured approach, where we have seven layers helping us to understand the task of communication. So these seven layers are what OSI reference model has provided us with and NTCIP actually took those seven layers and created its own framework by mapping certain layers and its functionality. Let's look at them all more closely. The understanding of the logical structure is very important because the idea of the upper three layers, of the concept behind those three layers tells us that if your application services and your requirements can be comfortably, or I should say, can be handled by the upper three layers. So those three layers are generally talked about or discussed in terms of the applications and how they will work in and fulfill the requirements. The middle layer is a transport layer. It's intermediate layer, and it actually separates the upper three layers from the bottom three layers, so it's masking the activity in that sense. Then you have the lower three layers, which provide end-to-end networking. Everything about network, the mediums, the physical plans, and all of those things that we normally deal with, the low-level protocols in the real world are part of these lower two layers. And NTCIP used this concept and created its own framework so that there is a lot of efficiency that also has been built into this process, even beyond the OSI reference model abstraction, All right? Let's look at it closely how the

framework works. The framework was put together knowing fully well that all over different devices that we have in ITS do not have the consistency in the information or the data presentation structure. So the first level here is information level that ITS standards have now available. So there are data dictionaries. Each different devices has a particular data structure, and all of them use consistent format, something that we call ASN.1 for the Center-to-Field devices. All right? So at the information level, we have a bunch of dictionaries available to us, both Center-to-Center and Center-to-Field, and then at the application level, where we have actual protocols. Now, if you look at this diagram, in the parenthesis, at each level, you will see a number. That number represent a profile. A profile is a collection of standards. So at each level of NTCIP framework here, you have a sort of mapping of standards, and from there, a user will make its own selection according to what the project requires. We'll go through that process in a short while. So application level houses all these different protocols. After that, we have transport layer level and the transport level basically takes the message and moves it from point A to point B, and additionally what we have available to us, for example, TCP/IP or UDP/IP and the T2/NULL also produce another combination of transportation transport null protocol, which is designed for we if don't have route the data, and many times, our transportation devices are connected physically from point A to point B. There's nothing going in between. So that kind of helps us through this third protocol at this level. So we get the idea that transport-level protocols are also created for our need and our use in ITS. At the sub-network level, we have protocols which are allowing us to actually move the device data. Two devices are sitting next to each other, we are able to use particular protocol and meet our requirements. So let's look at from the plant level, from bottom up. We could say there's six different ways of medium use in communication medium. If we want to have a dial-up environment where we don't need a fixed circuit, we can use telephone company service and use the Point-to-Point protocol. We can also use Point-to-Multiple Protocol, or twisted pairs, where we still use FSK modems out there in the country. And dial-up works with the V series modems. And, again, the other mediums-- the fiber, coax, and wireless-- also have combination, so this framework is independent of what medium you are having at your local level. NTCIP framework is independent of any medium, and it works with any combination that you might have. So this framework gives us a big picture. It allows us to think that different people deal with different issues in ITS development, project development, project integration, project implementation. Take a look at, for example, if you are a procurement person and you have a responsibility to put a specification together. In that case, you will be looking at making selection at each different level of a protocol, right? Which one is for your project. So you'll be creating a stack. So that's how to. So this framework guides you in that area. If you are a communications specialist trying to work with the bandwidth capacity analysis, error analysis, loading channels, how many traffic controls can I put on one telephone twisted wire pair, right? You have to figure that one out. So all of those issues will be coming to you, and you will see this big picture and understand those issues. If you are a programmer/computer scientist, then the upper two layers are for you. You'll have to be aware of what the data dictionaries are, what they present to you, how the data is structured. Then you have a protocol- the mechanism how the protocol works, what is the messaging structures. So those issues will help you understand the complexity and the sequence of activities that you've got to deal with. So that is everything or almost every aspects of a project in this framework, and this is one of the reason I took a little more time in explaining this framework because this is the heart of our work in all the communication activities we carry out in ITS. All right, a little more detail. Actually, at the information level, as I said earlier, there are dictionaries that are Center-

to-Field dictionaries are labeled as 1200 Series. So, for example, 1202 would be the traffic controller data dictionary. 1203 is part of dynamic message sign. So these vocabulary standards-definitions in the standards help us to pick each different device separately. That is a data dictionary separately for each of the device that we have out there in the ideas. Also, there are functional area dictionaries. For example, traffic management, it's a domain. People practice it in traffic management centers. So that is a dictionary for that domain, TMDD. So that is also out there at the application level of a level. Then you have another dictionary for emergency management centers, 1512. So these are the examples. This kind of domain dictionaries exist at the information level, so there are two types of dictionaries, Center-to-Field and Center-to-Center. At the application level, we have standards that deal with how to move that data that we talked about earlier. All right? So these standards at the application level provides how to exchange data, and there are several ways to look at these things in terms of which device, but basically, there is a Center-to-Field communication interface and then Center-to-Center information exchange interface, all right? So if you are dealing with the devices, you will go to Center-to-Field, and if you are dealing with TMC to TMC information exchange, you will be working with Center-to-Center application protocol. The transport level,, there are three choices. One is the TCP/IP, which is connection based. It allows us to have a connection and then send the data point A to point B. This data transmission is at a cost while it's connection based, but it has higher overhead. A little in terms of the overhead. The UDP/IP combination gives us a connection-less environment. However, it has a low overhead. So there are times when we make these choices what is best to use in our application. The T2/NULL, as I think said earlier, is that when there is no routing involved, the top two, the TCP/IP and UDP/IP, require the routing- they are used during the routing process. Now, sometimes devices connected directly from point A to point B, there's no routing involved. There's no moment of data transport in between. Then T2/Null is a better approach. So there are three choices that we can make here at the transport level. At the sub-network level, there are several issues. These issues talk about in terms of what we use, what protocols are available. For example, do you have Point-to-Point Protocol available here, which would allow us to use the dial-up circuit using a V-series modems? Still out there traffic control systems use that. In the case of we have multiple devices sitting next to each other, in that sense, the Point-to-Multipoint Protocol can allow us to use a shared connection, so one communication channel taking us to different devices, and we can use that protocol to communicate. For example, we have a twisted pair environment where we have FSK modems out there, all right? So at least we have this, choices to make at the lower-level protocols. At the plant level, which NTCIP has created for us to refer it as a plant level because we are familiar with this and plant level also in terms of physical layer of ISO, so there is some connectivity to that layer. At the plant level of NTCIP, we are dealing with bits and bytes. This is the environment we are familiar with mostly. We put our hands physically on something. It could be a connector. It could be a wire. It could be a wiring closet, all right? So those issues are relevant to bits actually. Finally, something is on that wire. The connector is now connecting to systems. So there are examples of that nature. That could be a fiber optics cable plant. It could be a wireless excess network that you rent from telephone company or you have your own. So these examples tell us that we need to perform a communication bandwidth analysis, whether what I have, for example, will it do the job that ITS systems is expected to be used for? Is there enough bandwidth capacity? Bandwidth is a reference to the capacity of a channel. And we need to analyze how much data can we place so it can be comfortably or without error can be moved from point A to point B. A lot of these issues are really practical, and why I say that, because

most of the time the communication cost that we incur- or I should say the project costs we incur in ITS are related to communication aspects. All right? And physical plant is probably the most expensive component in that environment. So we have to- we have understand this logical issues that actually create complexity and allow us to use the best solution that we can possibly inject. So those are the kind of issues that plant level will bring to us. And let's us illustrate in two examples. One, in this station here is Center-to-Field. You have a central management station on the left, and you have a dynamic message sign out there in the field on the right side, and you want to communicate using Center-to-Field. So a very simple question could be how does the user select standards. Well, we can say that the user will have to select standards at the information level. In this case, it's a dynamic message sign standard call NTCIP 1203 will be the driving the standard. Then you need a protocol to move the data in 1203 from point A to point B, so you will need an application-level protocol which happens to be SNMP for DMS. Now, by convention, by practice, DMS standard has made it very clear to use SNMP. There are other choices, such as STMP, which we'll talk later. It's also out there. Can be used, but it's not for DMS, all right? So there is a distinction. So user has a role to select particular application level standard, and in this case, it is SNMP. All right. So also at the transport level, we have to make choice, whether it's a TCP/IP or it's a UVP/IP or it's T2/Null. So whatever choice we make will also be part of this protocol standards. Now, notice here between the management station and the dynamic message sign, we are not saying that you need a particular type of communication medium. Whichever medium you have in TCP/IP protocol will work. All right? So that's the point. In the second illustration, we will now talk about the Center-to-Field. Previously, we talked about management station to the device out in the field. Now, we are talking about center talking to another center. It could be a similar topic management center, or it could be a 911 or emergency management center. Whatever the center is, then there is also a need for user to select certain protocol stack and make a stack out of it. So at the information level, for example, this is a traffic management center you're talking about here in this illustration, so we'll select traffic management data dictionary, TMDD. But if you are having emergency management center on the right, people choose also the IEEE 1512 standard, all right? So it depends on what centers you're talking about. But that choice the user will have to make at that information level. Then at the application level, basically, issues remain the same. You're talking about here NTCIP 2306 XML Profile, which is currently predominantly used in the country, in the US, so we'll stick with that notion and select our standard accordingly. But it could change to another choice which the NTCIP framework also offers you, which is called 2304 DATEX, all right? There are certain systems out there in the country, legacy system, which are data ____. So these choices are out there, so point we are making here, learning point we are making here actually, is that the user has a responsibility to create a stack based on the NTCIP framework that we discussed earlier. Let's have a little activity here. Center-to-Field device data standards are located at which level? A, information level? B, application level? C, transport level? Or D, subnetwork level? Go ahead. Make your choice for the correct answer. Okay, the correct answer is A, the information level. The answer is correct because information level is where the data dictionaries are. All right? The answer B is not correct. It's incorrect because Center-to-Field, there are standards define data only while the application defines the protocols. Two different things. Transport only takes your message from point A to point B, so it does not define data. And answer D is incorrect because sub network-level protocols, low-level protocols are used for sharing communication channel, as we discuss earlier. All right? They are not device data standards. So let's summarize learning objective one. We were expecting to be familiar with the terminology, and I believe we

have discussed that part just that, that we are now familiar with what the protocol is. We discussed what the standard is. We also dealt with the compatibility issue, how can we make the devices compatible with each other so that they don't interfere with this function, so we discussed that, and that is through the common protocol. We also discussed interoperability, how do we make new system so that they can talk to each other, exchange information, how do we make system compatible, and so we discussed that they've got to use the same dictionaries, data dictionaries, and dialogs and the messaging structure and the protocols. So to achieve interoperability, this condition, system must be compatible and must use the same messaging structures and data concepts that we discussed. We also discussed the detail of how the NTCIP framework came into being, what is the OSI reference model, seven layer structure that got translated into NTCIP framework with the file level. There's a difference between the layers. OSI uses the term levels, right? NTCIP uses the term labels. So please make sure that we summarize it properly. Also, the file levels of NTCIP framework. At each different level, we have different activities taking place and the sense of the file levels is that the users will have to make a selection at each different level and create a stack for local project. So at the end of the day, we understand that this terminology has made it very clear for us now that what the roles are for the users in terms of different aspects on the project making. All right, so let's turn to learning objective two. Learning objective two is covering what problem is NTCIP addressing, what it is that we're trying to do using NTCIP. Then what is organization of different level of dictionaries. What's in those dictionaries and how the data structures are conceptualized. Then what other operations? There are several areas where we use field devices and how do we actually deal with those aspects of contributing, controlling, and monitoring of field devices. Then we also need to understand how information exchange takes place between one center and another center or several centers, all right? So that's what the expectations really have in the learning objective two. So what is a problem NTCIP is addressing? Well, as we discussed earlier, in the old days, we were unable to have a proprietary protocol understand because it's closed protocol, for example. As a result different devices and different vendors using different proprietary protocols, we were not able to communicate between different vendors' devices. And it was sort of a dream to have one channel being shared by different devices. That didn't happen in the past. So each time we have a device, a type of device, for example, a traffic controller, we need a telephone line for that. If we had a dynamic message sign, we'd need a telephone line for that. If you want to control a camera pan, tilt and zoom, you need a separate telephone line for that. All of these things create a problem for us because they require different protocols for proprietary solutions, right? So when the specific device functionality actually was an issue because we didn't know how to combine all different devices in one place or one channel, so that made-- that difficulty actually also created a cost structure with it. We can certainly solve an issue, but each time we tried to help proprietary solution, it adds the cost and the complexity. With the closed protocols in communication, procurement became an issue for agencies. You have to go to the sole source. You've got to the sale vendor again and again and end up with a proprietary system. So you never got out of that old circle of dependency on one particular vendor. All right? So next time you want to expand your system, you've got to go back to the same vendor or your system don't get done or the integration suffers. You have isolated system sitting in the corner in the traffic management center, and that difficulty is widely known. We have come out of that situation now-- happy to say that. I'm sure all of us agree on that. But that is the environment that NTCIP had help us alleviate or remove from our professional careers. All right. So the solution NTCIP has brought to us-- interoperability and vendor independence. These are the two keywords that

ring everywhere in implementation stages of ITS. The interoperability again means that our systems should be able to talk to each other, all right? And the vendor independence means that you should be able to buy our products regardless of which vendor is selling those two. There shouldn't be more than one vendor out there. The marketplace should be such that we should not be concerned with where the product is coming from. All right? So NTCIP framework has dealt with these issues in a very methodical way. There are two stages, there are two steps, actually, in here. What is the common data format for definitions? We want to define our data, and we want to define our data based on a particular type of standards, which all different devices are using. So NTCIP use ASN.1 standard from ISO, which now help us to define data for traffic controllers, data for dynamic message. Everybody, all defined devices uses the same definition. So that part NTCIP has solved. The second part that NTCIP has helped us solve is the communication protocol standards. It is open. It's a common protocol. And, therefore, all these different standards that we use-- and we went through that in the NTCIP framework-- they're all common. They're all available. There is no dependency on one particular vendor or something like that, all right? They are in a public domain. They are available, and they are commonly available. So that's good part. So solution NTCIP has brought to us has brought to us interoperability and vendor independence. Let's look at this inventory. This slide can be used in many ways because it allows us to look at different things from a different perspective. You can look at this slide and say, okay, this is a good inventory. It tells us that, okay, there are all kinds of standards around that. They are listed in one place. And they're also shown here in the center-- they are shown here as they are like sharing a communication channel. So if you look at this diagram, all ideas, devices that we are currently using or will be using in our projects are present because there are standards here for each device that we use, okay? So now we say this device is compatible. Remember? If they're going to sit on one communication channel, if they're going to be collocated on one communication channel, will they be compatible? Will they allow each other's functionality without interfering? The answer is yes. All of these standards can be shared from a central location. They can be controlled remotely from one station. That is also good because they all use the same data structure and same communication problems. All right? So different devices working on same communication channel in interoperable environment is very good for owners/operators. Remote management for each different device is also a desire, which is available. And then in the deployment scenarios, not all different devices gets deployed one time in ITS, as we all know. We start out with traffic controller. We gradually add other devices: CCTV, DMS, ASC. These are very commonly deployed devices. And also, there are other standards shown here will join the integration process and get combined. So what do we take away from this particular slide is think about the parallel we can draw from ITS and the Internet. If you have 10 different applications on the Internet from 10 different independent service providers, for example, or 10 different vendors, for that matter, and they all have different platforms-- you know, one has a Unix platform, the other one uses C++ language-- whatever the environment is, in the Internet, you are using common infrastructure no matter what your network or local environment is, you use it in a common way. So if you look at this particular diagram for ITS, it will tell us the same thing. It will say that we have different devices no matter where it came from, which vendor it came from, but we will be able to use the same communication infrastructure because it's a common protocol. All right? And I think it gives us a good feeling to think that way, that we are finally helping ourselves to bring all these different technologies that we see out there in the other industries now actually coming to us and helping us. And I think NTCIP has contributed in that sense in our achievement so far, right? So the data

experience information level, they are marked as the 1200 series, and they contain only objects. They contain objects because each functional feature-- and a feature of a device is actually the behavior of a device, you know? So if you want our device features, then there is an object for that particular feature. So that's how these dictionaries are providing us of help. In the 300-series training modules, we are actually discussing this whole part of 1200-series standards. For each standard, we have a module, more or less, dedicated. So information-level standards are included in the 1200-series standards. So what is an object? The object is a data structure used to monitor or control one feature, attribute all controllable aspects of a managed device or manageable device. This is according to NTCIP standards. The general feeling is that it is a consistent and very clearly laid-out structure that allows us eventually to monitor and control features that are resident in a device. All right? And each object is by itself, so it is defined in a very concrete way. So when you bring certain number of related objects together and you put them in one place, you'll be calling it Management Information Base. A Management Information Base is human readable file format. It contains all the design object so that we can pick and choose as we need them for different purposes. So, for example if you want to configure a device, you want to set parameters in a device, you will be using certain objects which are labeled for that. There are a lot of controlling objects we'll be using for controlling a device functionality. And if you want to monitor or get certain data from a device, you'll be using certain objects to achieve that goal. In this case, for example, your station here of CCTV standard has some 70 objects in it, all right? So you can guess that- you can relate that in the sense that every device has its own MIB, M-I-B, all right? And sometimes people refer to as a collection of objects of that particular device and call it database. So what is the relationship between a protocol and a MIB? There are two concepts here, so there's got to be reason why they're related to each other, right? So the MIB gives us the objects and their value, right? So those are the functions that device performs. SNMP, for example, in the middle here, is manipulating those objects. All right? So the relationship is one to one. They have very solid relationship. You have objects and you have values in them, and the SNMP will manipulate those objects. All right? The result or the outcome is that device behavior will change or will be altered, all right? So if you want to do something different than what the device is doing now, you will use SNMP to alter that behavior, and that's the relationship between protocol and a MIB. At the information level, there are also Center-to-Center data dictionaries. Some of them were developed using SEP. Actually, I should say only one of them so far. The traffic management data dictionary was developed using SEP process. All right? And the TMDD, for example, currently is also in a revision process. Version 3 has been device. So the point is that each time we learn something, we use SEP process to help us and then we modify and update. So TMDD is pretty up to date. They're the dictionary for traffic management and was developed using SEP process. The other domain, such as IEEE Emergency Management 1512 standard was not developed using SEP, so it has certain things missing in them. For example, the dialogs are not there. All right? So, hopefully, next time when it goes to the revision, that will be updated and brought to the level as TMDDs, all right? So these dictionaries are located at the information level. Our traffic management in our dictionary supports Center-to-Center system interface device/design. We cannot have system interface produce, which is based on a standard, without TMDD, for example, for traffic management, all right? So our Traffic Management Data Dictionary comes with the user needs and requirements. The dictionary was produced based on what are the current needs or anticipated needs a center might have to communicate to another center. And those user needs were translated into requirements, and then requirements actually created the need for design concepts. So the user

needs, requirements, and design concepts are all related to each other, and that's what TMDD offers for traffic management. Particular mention about the data concepts, data elements for TMDD, we have data concepts available in two forms. One is for XML, which is currently the predominantly used structure in the country, and we also, in case if you want to use a DATEX that are also data-defined using ASN.1. So, again, you have a choice, but the dictionary is complete and has both types of data available in it. Now, the local projects will have to select the capabilities that matches or maps to that project and pick whatever they need from these dictionaries. The dictionaries are pretty open. Not everything in the dictionary must be mandatory, and the user has a responsibility to, again, select whatever is needed. But to conform to a particular standard, there are minimum requirements the dictionaries also outline. There are other domains, as I mentioned several times. The Emergency Management 1512, for the information, Traveler Information System, you have ATIS as it normally is for you. You have J2353 standards. For transit, we have the TCIP. Those APTA standards also have their own data dictionaries. And in the supplement, you have references available, so if you need to consult those standards, there's a reference in supplement. So, how to combine Center-to-Field? Let's create an example here by selecting protocols at each different level, all right? So hosting data dictionary for a device. Well, here, you are talking about the NTCIP 1200 Series. So you will require, for example, for Dynamic Message Signs, you will require 1203. And at the protocol level, to exchange data in the previous bullet, you will require 2300 Series. So, again, you will select either SNMP or STMP. So if you are talking about a Dynamic Message Sign, you will be selecting SNMP or if you are having your application with ASC, Traffic Controller, for example, you will be selecting protocols SNMP as well as perhaps STMP. So those choices you are making. So what we are learning from this slide here is that we are gradually building our stat as per our project requirement. The transport-level protocol will be deciding whether we need the TCP/IP or the UDP/IP or T2/NULL. So one of those three combination will be working here at the transport level. At the subnet-protocol level, we'll be making sure that if you are dealing with the, for example, PMPP, Point-to-Multipoint protocol, then you will be making that choice or you will be dealing with the PMPP, Point-to-Multipoint Protocol, you will be selecting that for your implementation. At the Communication Plant level, the physical plant level is whatever you have that your project already has decided. Actually, a project is based on what you already have in place, those are the choices that you make. So this is how you create a stack for a center to fill. Something similar, here's an example, for DMS, as I mentioned to you before, that if you have a situation where the traffic-management center wants to talk to a device in the field, then you will be making those choices, and actually, the number will come out like this, 1203, 2301, which is SNMP. If it was an STMP, the number will change to 2302. So this number designation is very important when we manually prepare the specification. That's why this slide is showing you as an example here. And the UDP/IP in this case, it could be the TCIP, but if it's TCIP, the number will change to 2201. And this happens to be a Point-to-Point protocol because physically, we know where the Dynamic Message Sign is. So this is how you create your deployment stack and you write it in your specification. Something similar for a traffic-management center, for a center-to- center purpose, here we are selecting Functional Area Data Dictionary, TMDD, we are selecting 2306 XML, we are selecting TCP/IP -- that's what XML works with -- and you have a built-in network in process in your agency or in your TMC environment. Then chances are very good that you will be dealing with Ethernet, which is 2104 profile. And that's how you do that. And then you, of course, need a connection to the Internet so that you can send messages to the other centers. So this is how you create a Center-to-Center stack. Let's have a little activity here.

NTCIP 1200 Device Standards provide what? A) Management Information Base for each field device, or B) application protocols such as Simple Network Management Protocol and Simple Transportation Management Protocol? Go ahead. Make your choice. Okay, the correct answer is A, Management Information Base for each field device. The answer is correct because 1200 Series standards are about the specific MIB. They deal with objects, particularly for that device. Answer B is incorrect because at the application level, you only have protocol and protocols are not the data dictionaries. They are not MIB. They work with the MIB, but they are not themselves data content. All right, so let's summarize Learning Objective 2. In Learning Objective 2, we have discussed the problems being addressed by the NTCIP and the proprietary protocols. With NTCIP, we have now a common protocol, so that issue of proprietorship of building systems has pretty much diminished. Second thing we discussed is the Information Level dictionaries are housed, 1200 Series are at the Information Level, and they provide the objects for our operational needs. They reflect, actually, the features of the device. We also discussed the domain specifics data dictionaries level, TMDD, for example, for Traffic Management. So now we move on to Learning Objective 3. Learning Objective 3 is about describing Center-to-Field applications and related NTCIP standards. Here, we want to go a little more in detail in terms of what are the operational needs, define SNMP Network Management Model, how the SNMP actually works. We want to also understand the structure of our SNMP Messages and PDU format. I also tell you that PDU is a core content of a Message. And then we have different protocols that we want to just prepare to view what they are about. So that's what we will do in Learning Objective 3. The operational needs come from a real environment. The very reason why we are building a system, because we have certain operational needs and to satisfy those needs, we have to conduct certain operations, certain tasks that we have in front of us. So the Center-to-Field standards support Management Station for these four key areas that are listed here. First is to configure a device. Every device has certain characteristics, certain parameters, certain content, behavior issues, for example, timing, how to reset the timing references, and all those parameters are set in in this configuration, how the device actually will work as when it starts working. Second thing, we monitor a device and gather data. The device itself is helping us to see what's going on by way of providing us the data set, the gathering or collecting the data. Then we are actually influencing the device behavior by some control functions. We tell the device what to do, and if the device is doing something, we modify the device telling it to do something different, something else. So those actions are the controlling actions that are attached to what the current device's function. Then we move on to retrieving data. Well, we don't always have a connection to field devices. So in the meantime, the devices continue working. So if something goes wrong or something happens at the device level, they keep this log data. They keep the reports locally so when these reports are available, we want to know what about it, we can retrieve this data. It's always available, both live and also when they are not live, the connection is not there. So sometime when the connection is not there, we can next time when we have the connection, we can actually go into these logged/event reports and get whatever data we have and understand what really happened. This is generally referred to as exception reporting. This is also stated in terms of how the data will be actually brought to the central management station. How do we know what's happening in the field? So a lot of good things can be done with this kind of operational set-up. So Center-to-Field protocols, there are four of them actually that we have to know about and understand in which context they are used. Simple Network Management Protocol, SNMP, is the primary protocol used by all devices. It is the core protocol that is present in almost every device that we have been talking about in ITS

deployment. Simple Transportation Management Protocol, STMP, is a variation of SNMP. And there are differences between the two. SNMP is a little more overhead. STMP is much less overhead. So it decides when to use or where to use. So right now, ASC, or traffic-control standard, allows the use of STMP along with the SNMP. Also, the STMP is also used in other standards related to traffic control. It is called SSM, Signal System Master. So that's also STMP is used. Other than that, most other devices are using STMP, so that's the difference there. The File Transfer Protocol, FTP, is a very practically used protocol. We all use it every day, or if not every day, very frequently. And a variation of FTP is Trivial File Transportation has a much lower overhead and simplicity to that. So just let's look at quickly each one of them a little more with the clarity. The Center-to-Field communication protocol, SNMP, it is simple. It's in its name as well that it is simple, because it has very few commands and it's flexible how you use it. Also, it is used in general network management terminology. It's called network-management tool that monitors and controls devices remotely. So SNMP is a remote management tool. It allows us to monitor devices remotely, which is the capability we did not have before in a standardized way. Some might say that we need that information available to control and monitor devices. Yes, but that may have been the case, but using proprietary protocol. With the SNMP, there is nothing proprietary about it, okay? The SNMP has put NTCIP framework in the driver's seat, and all devices are now remotely managed without reference to proprietorships, as was the case before. So that's a good thing. So now we can manage, in the Internet we data-manage routers, bridges, firewalls, connect closers, wiring closets, switches, whatever, right? In the ITS, we are monitoring the same thing for same situations, actually, monitoring and control for our traffic controllers and Dynamic Message Signs. So the nature of use of SNMP has changed from the Internet devices to actual field devices which actually control and display messages and things like that. So the environment is slightly different but the principles are the same. This is how SNMP Network Management Model looks like. It has two components. That the management station, you have a management application that tells the SNMP Manager, and there is a MIB, as we discussed earlier, so there are three things present, in general, at the management station, the application, the SNMP manager, and the MIB. At the device level at the far end, you will see management information, inside the device, SNMP Agent and the MIB. So we are talking about the software environment here on both ends. SNMP Manager is a software. SNMP Agent is also a software. So this discussion between the two ends is going through so that the remote capability can be achieved. Three parts, the key part, SNMP Manager, it's an application program telling SNMP Manager what to do. SNMP Manager waits until application tells it to do whatever the message is about. The SNMP Manager will create a message and send it to the agent and it will either be a query, getting something from the Agent, or it will tell the Agent what to do or change something in the database. The Agent itself is a software piece which runs on the device and it maintains the information about the integration and the current conditions in the database, whatever the activity is is reflected in the Agent's process. The third part here shown is a MIB. Now, MIB is the same MIB at both ends. It cannot be different. And the reason is that both ends need to know what you're talking about is actually inside the MIB. You know, both of the software pieces can be only referenced inside the MIB, and therefore, the MIB is present at both ends. The SNMP Operations, the summary is in three parts. First is that we actually retrieve something. We retrieve in the way of a value of the object. And that is done by reading for GET operation. So the SNMP Manager will say, "Get me this," all right? So that's a GET operation, and that is a read method that is used on the object. Second is modifying the value of the object, and that is done through writing SET operation, or setting something, you

know, from previous to now. So the change in the step is achieved by setting process. There is a third element in SNMP Operations that also is called Reporting. It's condition reporting, or exception reporting. And that is about an event. Something unexpected has happened at the device level, and the Agent is notified by the device. Now the Agent's role is to tell the Central Manager, that, "Hey, you know, something happened, it may be worth your while to know about it." So in the Internet, we call this a trap operation, and in NTCIP, we don't have this capability yet as the Internet has in reporting the event. So we will deal with, in a little while, what the NTCIP does in the meantime. Hold on for that. SNMP Messages, there are three messages in the form of request message from a center to the agent. The GetRequest message actually initiates requests for information. We will say, "Give me information on the following object," and that message will go there at the Agent level and get the information and the Agent will respond that. So second one is the GetNextRequest, which is getting more information of the same type. Instead of issuing GetRequest message for each individual object, if you have more objects related in one table or something, you can get them with one shot. So that's what the GetNextRequest message does. And the SetRequest message, actually you are telling the device to do something. So that is an active way the device will have to perform that. So that's the SetRequest message. The Response Message has only one form, from the Agent to the Manager, and it's called GetResponse message. This structure, and said, "Here's the information, you asked for it, and here it is," that kind of thing. So exception reporting is the fourth issue that we always, I mean, not always, but we generally run into or that if there are conditions out there at the device level, there are two ways to handle. The device keeps the logs, right? The device has the logs, as I mentioned earlier, and when you have the connection to the device next time around, you can actually go into the log and read and see what happened. So that's how things could work. So something called NTCIP Event Logs allows us to do that, and the Traps is the future. The NTCIP working groups have developed this standard called 1103, but it's not ready yet for the Traps mechanism. So Event Logs allows user-defined events to be logged with the timestamp, when did it happen, and then next time when you have a chance, you can retrieve it. And you can retrieve it by using GetRequest. So, like anything else, if you want to have data coming from a device, you can also get the same event report data from the device the same way. The Trap allows, actually, reporting immediately. Something happened now, and a few seconds later, the message arrives at the Traffic Management Center, you know, not yet, but it will be in the future. But in the Internet, this mechanism is already available. So let's just summarize saying that, okay, we can wait for the traps mechanism in the future, but we don't have to. We can still get the information we need if there is something happened. So there is some relief here that NTCIP has created. Let's have an activity here. To gather data from a detector station, the central SNMP Manager initiates which message, GetRequest message, B) SetRequest message, C) Trap message, or D) GetResponse message? Go ahead and answer one of the four. Okay, the correct answer is A, the GetRequest message. It's correct because GetRequest message retrieves the values from Agent. In this case it's the detector station, so you will get your data from the detector station through this message. Answer B is not correct, incorrect because SetRequest message modifies the value. It does not retrieve, but it modifies the value. Answer C is also incorrect because Trap message is a one-way message from Agent to the Manager, all right? And last answer, D, is also incorrect because GetResponse message only returns the value requested by the request message. So these answers tell us that each message has a particular task to perform. Now let's move to the Simple Transportation Management Protocol, STMP. STMP is a variation of SNMP, and it works only with the 13 specifically designed dynamic objects. These

dynamic objects were created based on the needs of the needs of the ASC traffic controllers, for example, and it would say, if I'm going to conduct this transaction with the traffic controller, I can eliminate certain overheads, because it's all related to the overheads in a sense. So it is a reduced version of SNMP, keeping in mind that STMP has much lower overheads so that it can be made a little bit more efficient when there is an issue of bandwidth. Not all communication channels can provide the bandwidth that you need. So in a situation where you have very limited bandwidth, how do you make the best use of it? And STMP is your friend there for traffic controllers. So you can comfortably put X number of traffic controllers using STMP, which you could not do with SNMP otherwise. So the difference is in the amount of overheads each protocol has. So in the wireless network recently in the country, there are several applications that are out there, and also in the Internet environment, you can combine multiple objects and we packed them just like those dynamic objects we mentioned before. And you pack them in one block. In a block means related objects can be treated as one block. And then you can do any of those four things that you want it to do. Perhaps the GET operation is a very good example because you get certain things from a wireless network, using wireless network, so there is an efficiency of data channel, communication efficiency. So that's a good point. Modern systems are also being flexible and wireless networks are a good example to begin in that direction. The NTCIP 9001 document, as is labeled here, is a guide. This guide document has a very good discussion on STMP. It actually provides the calculations and bandwidth analysis, how many traffic controllers can you put and what are the differences in terms of overheads between the two protocols, SNMP and STMP? A very practical discussion. And what kind of lower-level protocols will be required to work with this process, whether there is routing in-wall, not in-wall, all those issues are very, very much indicative and are presented in the NTCIP guide. So if you are a traffic-controller application person, I will definitely urge you to consult the guide for all those examples. FTP is a very widely used Internet protocol. All of us use that to get a file from somewhere in the Internet now. And in the NTCIP structure, the framework, it's labeled as 2303 and it is TCP/IP based. You've got to have a connection to get a file transferred to you from a server. So there are examples in terms of the TCP/IP applications, ESS, for example, Environmental Sensor Stations standards does use to get a snapshot of a condition and the pavement, weather conditions or the rain conditions, that kind of thing. So that is a good example to cite here. Not too many examples are our there in terms of FTP uses. However, I will tell you one thing here is that in the Center-to-Center, we will be discussing FTP as an XML direct. There, you will see FTP used a little more differently, and perhaps more widely, as well. The TFTP is a variation of FTP. It is also labeled in the NTCIP framework, but it works with the UDP/IP. Unlike the TCP/IP connection service, this one is without connections. It's a connectionless service, so you have to make a connection each time you want to get a file. But it's a very convenient way of doing things for small files. Let's take in an illustration here to bring our point home. For example, here, the Central Management Station is talking to the field device, in this case it's a Dynamic Message Sign, and the Agent is communicated to using GetRequest. So if you use a GetRequest, this central communication, Central Management Station, will say, "I want to know what the device is doing, what message it's displaying," for example, all right? So the status of the device will be known to you through the GetRequest message, and the response will be issued by the Agent, who is saying, "Here is what the device is doing now." So this is the way this process between SNMP Manager and SNMP Agent is working out. The information you get at the Central Management Station gets translated, and the response is translated for the benefit of the user. The computer software program, the applications, the native

application will turn this information that you get from the Agent and present it to you as operator or the user in a form that you are used to or something you can use practically. We don't have time to analyze it and everything, but the program, or the computer program, the software will analyze for you and present it to you so that you can look at the data and say, "Yeah." So if you're not satisfied, you can make another request, NextRequest, and you get more data, and if you get enough information from the central device, then you make your management decision. That's the whole idea, is that you want to have information, current information, updated information so you can make the proper decision based on that. This could be an example in incident management. Things constantly change out there in the field, and you need a lot of information, leading information so you can make your decisions whether to bypass a route or to send a patrol car there or whatever decision you need to make will help you get the insight. So such things do occur at the DMC and SNSP can help you to meet your needs. Let's have an activity here. Which of the following is a preferred protocol for monitoring a DMS sign, Dynamic Message Sign, A) SNMP, B) FTP, C) STMP, or D) NTCIP 2306? Go ahead and select one. Okay, the correct answer is A) SNMP. SNMP is typically used by convention for the DMS standards. DMS standards actually does say clearly to use SNMP. So by convention, or by practice, we choose particular protocol. In this example, it's DMS, so we chose SNMP, all right? So answer B is incorrect because FTP is only used for file transfer, right? We're talking to a device, not talking to another entity to get a file. So FTP is clearly not the right protocol. Answer C is incorrect because STMP can be used only in ASC application and, as I said earlier, for SSM, the Signal System Master for use there. So STMP, by convention, is not used the DMS applications. So that's why it's incorrect. Some people could argue and say, "Yeah, we can use STMP," but the standard is discouraging you to do that. So we have to pay attention to what the standard itself says, and if the standard doesn't say, then we have to make our own choice accordingly. So in this case, in this example, STMP is an incorrect answer. NTCIP 2306 XML, D, is also incorrect answer because this is specifically, 2306, specifically used for Center-to-Center. Here, in this example, we are talking about Center-to-Field application, all right? So there are two different aspects of how protocol is coming to working. So let's summarize Learning Objective 3. Learning Objective 3 is about Center-to-Field applications, and we reviewed operational support, those for important aspects of operation, configuration, monitoring, control, and exception reporting, as well. Then we also discussed protocol us in the Center-to-Field application, Dynamic Message Signs and all those different messages, how they provide us the different aspects of information that we're looking for. Turning to Learning Objective 4, now we are going to focus in this learning objective on Center-to-Center. So previously we've done all the Center-to-Field discussion. Now we will enter into Center-to-Center application areas and say what are our operational needs for Center-to-Center and how or what communication interface it is about and what are the TMDD concepts and what are the components involved in 2306 and then what are the messaging models used in this process? So this file, key points that we will have to raise and answer in this Learning Objective. Now, Center-to-Center has brought new terminology. When we say Center-to-Center in today, now, we are actually talking about World Wide Web. So World Wide Web Consortium, W3C, is a terminology that you're going to run into right away. XML, I'm sure everyone taking this course and will take the course in the future will be aware of XML, because XML is predominantly the eXtensible Markup Language being used now in ITS, in transportation sector very commonly. So XML is another terminology. Then we have WSDL, W-S-D-L, Web Services Description Language. This is the format, this is the language that is available now for us in transportation

sector to let everyone know that, hey, just like the Internet is using WSDL, we in transportation, and ITS particularly, we're going to use this format so you know what information I have. So I'm telling you, one center telling another center, say, hey, you know, I got this information. And we do that through WSDL definitions. The SOAP, S-O-A-P, stands for Simple Object Access Protocol. This is the acronym that was there. Nowadays the W3C has stopped using this as an acronym. They just simply say SOAP. So let's just continue saying SOAP, alright? It's a communication method for transmitting XML message. It is a transport mechanism where XML message can be encapsulated and sent to someplace. HTTP, Hypertext Transfer Protocol, all of us use that. Even while taking this module course, you have used it by logging on to Internet. So HTTP is a web browser transport protocol, very commonly used. The term PRL, Profile Requirements List, is a table that has been developed in NTCIP 2306 which lists what the components are required to conform to NTCIP 2306 standard. Two specific mention, 2304 DATEX is an Application Level Profile Standard. It uses fixed messages and a fixed connection between two centers. It used to be, years ago, the only thing available. Now with the Web services through XML, 2306, we have two choices, DATEX and XML, and XML is predominantly used in the U.S. for various applications such as 511 and many other large-size applications, as well. So I urge you to familiarize yourself with this terminology because without this terminology, we cannot discuss the concept of website assisting in the C2C discussion. So operational needs, what are the Center-to-Center operational needs? So for one time, we also have to understand, whenever we say Center-to-Center, we are talking about what goes on between the two centers. So that's the communication process we are specifically talking about. So keeping that in mind, the system interface that works at both ends, both centers, will be subjected to certain user needs. It is, therefore, certain user needs. So those user needs are generally categorized in four headings. There is a need to share event information. There is a need to provide control of devices, share status, and data. There is a need to provide roadway network detail data. And there is also a need to share data for archiving. So these four needs are generally behind the development of a system interface in Center-to-Center. And these user needs are handled or addressed in Traffic Management Data Dictionary. So Data Dictionary provides dialogues in a logical sequences. Dialogs are a logical sequences themselves, and they use messages so that the conversation takes place. So consider a dialog as a conversation and sequences follow with particular types of messages. So there are three types of messages that we will see being discussed in these modules, communication modules. Request/Response paradigm is like it's a combination. You make a request and you get a response. So you share your information using Request/Response paradigm. The second is a subscription. People want to subscribe one time and say, I want the following type of information from you. So you are almost subscribing just the way you are subscribing a magazine. Every month you get a certain type of magazine that you're looking for from a publisher. It's the same thing here. So you are subscribing. And the publication is something available, some information that you ask for that's available and that information is sent to you. So subscription and publications is another pair of activity that we see in the Center-to-Center arena. 2306 Profile Standard we also want to learn correctly, so I will urge everyone to see NTCIP 2306 XML Application Profile, all right? So this is a good title to keep in thought that way because it has a combination in it. The TMDD is not physically described in a detail in 2306, but to use 2306, you need TMDD. So what does TMDD provide us? It supplies us data concepts in XML format. That's where your messages are coming from. And that's where the dialogs are presented. That's where we have data-frames and data-elements from which a message is constructed. So there are modules in this C-eries that stick to

data TMDD data concepts. So TMDD supplies data concepts, XML messages. 1104, NTCIP 1104 is another NTCIP standard which we need to use out of 2306 because it provides us some naming conventions, as the title says. It tells us which center you're talking about, the organization, who's making the requests, the name of the operator, how do you put all that thing in a message? So that information and that structure format is coming from 1104. So it's a companion standards. 2306 supplies XML schema, the methodology, how the messages are put together, the WSDL announcing the rest of the world how to get that information, and SOAP, the transport protocol to move the XML message from Point A to Point B, from one center to another. So those are the things present in 2306. There is a transport mechanism also offered to us in that 2306. There are two transport mechanisms. We could use HTTP just like we are using in web terminology now, technology, you know. So HTTP, or you have XML. You can also send an XML message directly. So these two choices two we have to make. And then, of course, the TCP/IP, we need message to take it to the other node. So those are the kind of things you'll see in 2306. So let's introduce the web service now. What is the correct way of saying a way of service? We heard these names. It could be misunderstood in the sense that to get to this seminar today that you logged on, probably you had to go through the browser, right? So the browser allows you to get connected to the Internet. The web services allows you to offer whatever you have to other people, other centers, and you don't want them to call you. You say, this is what I have, this is available, where to find it, and you put your URL resource location and people will go there directly without human intervention. So web services is an automatic process. So that's clearly defined here, saying that a web service is any service and is in the queue during an operation that is available over the Internet or intranet. So first time in the transportation we are talking like this b/c we are now saying, whatever I have at my traffic management center is available to you, another center, using Internet or intranet product matter. So that's a definition of web service. And it uses XML messaging system. XML is predominantly available, common methodology now in every application, Internet or otherwise, and we can make use of that. So that's the second point. The third point is that it doesn't matter which system you have or what operating system you are coming from, what your native application look like. It doesn't matter. Any and all kinds of computing systems will be able to deal with these web services and get whatever information is available out there. So that is generally how the web service should look like to us in terms of combination of the system interface. And the system interface is the method where we are touching base with the native system, center system. And the message is input and output. There are always two messages in conducting a web service, all right, a message coming in, input message, message going out, output message. So that's one operation. If there is another operation, you will have a similar structure for that operation. So let's look at the SOAP. Now, what does the SOAP do? SOAP is a transport mechanism for XML messages. XML messages cannot be placed on their own unless there is efficiency involved and whatever, right? So you decide, but SOAP encodes an XML messages. So if you look at this whole box here as an envelope, the XML message is contained in there. So there is an organization how to do it. So this whole thing is called Envelope. So the envelope for SOAP messages are required. Then, there are headers. There could be one header or two headers, and these headers are optional. They don't need to be required, but they can be. So that's the second part to a SOAP message. Third is the real content of the XML message. This is the real things. This is the payload. And this message is what we are talking about coming from Traffic Management Data Dictionaries, the TMDD in our case. So this message is carried out by this SOAP envelope. So 2306 also tells you that you can do these three things. You can do the XML transport message with SOAP,

without SOAP, directly, or you can do it through FTP. So there are three methods, and these three methods of doing the Center-to-Center communication are listed in something called Protocol Requirement List, PRL, and this is Table 3 listed in the Standard 2306. So this PRL is a guiding document. It's a guiding table for us because it allows us to choose whatever we need. Not everybody needs to transfer their messages using SOAP. Not every center would want to do that, all right? Not every center wants to do just XML. Or somebody will say, "FTP is good enough." So those are the kind of choices we will be making using the PRL. So PRL is a guiding table which is required, and every Center-to-Center project that builds a system interface requires a PRL. We cannot do without PRL. So PRL is a very important document, and it tells us what section we should contain. For example,, in this table here, we are showing 1.0 SOAP over HTTP. So we can say SOAP can be transmitted using HTTP, which is the predominantly done methodology right now. So what will be the requirement? So you are telling everyone else that you need to do the following through the WSDL. So that's one. Suppose if someone wants to get SOAP message directly, SOAP over XML, for example, so that choice is available as 2.0. Then, there is also using FTP. So those choices will be made by a user preparing a specification using this PRL. Let's have a little activity here. Which of the following is not applicable to Center-to-Center? Now, this activity occurs because you already know now about Center-to-Field and Center-to-Center. So at this point, we should be able to answer a choice correctly. Which of the following is not applicable to Center-to-Center? Go ahead.

Raman Patel: Okay, the correct answer is B, SNMP. It is correct because the statement is not truly, properly done. SNMP does not apply to Center-to-Center. It only applies to Center-to-Field. So that part, SNMP, is not the correct protocol. 2306 XML is the heart of Center-to-Center. That's what we really are talking about. So it's not-- it's incorrect the way this answer is framed. Answer C is also incorrect because SOAP is the transport mechanism of the NTCIP 2306. And WSDL is the language standard. So this variation in answer tells us that the terminology that we use to understand, that they have to be properly realized in our minds so that we can use correct protocols. So I mentioned earlier in the PRL that we have seven choices. So now we have to also summarize the discussion in Learning Objective 4 and say, well, we reviewed these Center-to-Field Operational Needs. There are four key needs that users have in terms of what the system interface is supposed to do. Then, the system interface itself is built using XML messages in a pairing, request input, output kind of things, and we produce in a system interface that works with the native system. We also discussed Generic Dialogs process in TMDD. These processes are contained in a separate standard under the TMDD. And we also discussed the profile, XML profile, has several choices available and we need to make one choice from that profile standards which one will that be. And introduction of web services, what it is, we discussed that. SOAP is a key protocol in the web services foray, and WSDL is the language with which everyone knows what is available at a particular center and how to get it and where to get it. So I realize that we have not provided as much details in this Learning Objective 4, but I also want to tell you that we have a Module C202 which deals with Center-to-Center XML 2306. So a lot of those details are being discussed there and we will urge you to keep an eye on that. So what we have learned? Well, we have learned that the NTCIP Family of standards are based on ISO's OSI Reference Model, Open System Interconnect Reference Model. We also learned two categories of NTCIP communication includes Center-to-Field and Center-to-Center. We also discussed that NTCIP objects are based on ASN.1 language representation, which is an ISO standard. We also discussed and learned the 1200 Series device

standards are located at the information level. We also discussed that SNMP is used for remote management of ITS field devices located in the field, specifically, SNMP performs a retrieval, modification, operations to manage a field device. A Center-to-Center standards in the traffic-management application includes TMDD and NTCIP 2306 XML. There are resources out there that are available. The participant student supplement provides a list of references. Also, it provides you with a good discussion on the OSI reference model. What's in there is that the discussion is on what each layer does and how well it does and all those related issues. NTCIP Guide has very significant details in terms of the benefit analysis and how SNMP and STMP work and there are a lot helpful diagrams, actually, when you route, when you don't route, what protocols will mean what. So I will urge you to consult NTCIP Guide for a lot of good learning concepts in there. The TMDD Standard itself and the guide -- there is also a companion guide available at the ITE website listed here -- will give you very good detail of what those XML messages are and how to use them in terms of putting a Center-to-Center interface design specification together. There are questions that came up or were being asked at various times. Consider them as a Frequently Answered Questions. I will discuss them now here. The first question that can be asked is, can an STMP be used for a controlling Dynamic Message Sign? And I think we discussed that in this module earlier, that by convention and practice, Dynamic Message Sign is controlled by SNMP and standard guides have said that SNMP is the preferred protocol for Dynamic Message Sign. Second question is that, can there be a vendor independence in the ASC applications? The answer could be that, yes, there could be vendor independence in the applications if the agency identifies ASC MIB. However, there is an issue out there about nondisclosure agreement from different vendors keeping the private MIB. They would like to keep it private, and if that's the case, then it's contrary to what the standards are telling us. So the advice is that we should stay away from a private MIB, objects, and if there is a case where you need such things, then the nondisclosure agreement is contrary to the application. So we have to handle that properly. There's also a related question about 2070 Traffic Controller Standards. Yes, there are certain issues with it, but it does separate hardware and software, so stay tuned for that kind of discussion on those standards. Lastly, I would also raise this question and answer briefly, is that, can one TMC be allowed to take control over another TMC's devices? Could that be DMS or whatever? The answer could be yes, and if there is a prior arrangement to get a request from another center and both centers agrees, you can share information among the centers and exchange information about a particular device. However, this is a very high-level abstractions. You are not going into the parameters of the particular device and alter it unless the agency who owns it allows you to do so. So there all kinds of combination in this discussion that can be worked in this thing. But the standards do not prohibit or in any way create a hurdle for you to not to do it. They are very conducive to carrying out the information exchange process. So that's how I will leave it at that point. There are additional modules, as I mentioned earlier. The C201 deals with SNMP module, actually, specifically focuses on SNMP and everything about the SNMP messages and how they get carried out and how they are implementing is covered in this module. So we will recommend you to take that module. Also, that module will help you to implement your 300 Series Modules courses on devices. A C202 deals specifically for Center-to-Center, and it goes and completely deals with 2306. It has a significant amount of guidance, as well, which profile is for me, subprofile, whether I do SOAP or whatever. So all those answers are there in C202, and together with this module today, C101, C201, and C202, you will have a very good handle on the communication principles and how they are used in the NTCIP

framework. So thank you very much for attending these modules, and this concludes the presentation of C101.

End of 2013-01-29 10.12 C101 Final Recording.wmv