

Module 16 - A321b - Specifying Requirements for Traffic Management Systems Based on TMDD v3.0 Standard Webinar Transcript

Shelley Row: ITS Standards can make your life easier. Your procurements will go more smoothly and you'll encourage competition, but only if you know how to write them into your specifications and test for them. This module is one in a series that covers practical applications for acquiring and testing standards-based ITS systems. I am Shelley Row the director of the ITS Joint Program Office for USDOT and I want to welcome you to our newly redesigned ITS standards training program of which this module is a part. We are 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 yourself. 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 the webinars. ITS Standards training is one of the first offerings of our updated Professional Capacity Training Program. Through the PCB program we prepare professionals to adopt proven and emerging ITS technologies that will make surface transportation safer, smarter and greener which improves livability for us all. You can find information on additional modules and training programs on our web site www.pcb.its.dot.gov. Please help us make even more 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 to be helpful.

Nicola Tavares: Module A321b, is about Specifying Requirements for Traffic Management Systems Based on TMDD Version 3, Standard. Target audience for this module includes engineering and planning staff, emergency management and public safety, traffic management center operation staff, system developers, public and private sectors, and coders. 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. He is a member of the NTCIP, and PMVB [ph?] committees. Dr. Patel has 40 years of experience in the transportation field, close to 30 years in New York's City Department of Transportation, as Chief of System Engineering. He is currently teaching ITS and transportation engineering at NYU Poly, in Brooklyn, New York.

Dr. Raman Patel: Hi, everyone, I'm Raman Patel, I thank you for attending class today. We have two curriculum paths that we are following in this professional capacity building program of these first four modules that you are looking at here in the front, are preparatory and tells us what ITS standards can do for us, benefits and outcomes, and in A201, there is a detail of how to use ITS standards, so you get the system that you're looking for. It's followed by a series of standards. In this set up modules here, for example, A311a, is NTCIP based standard for DMS, followed by ESS, and that's also on this curriculum path for TMDD.

As you know, TMDD is a system based standard, and we have one module, A321a, for user needs, is also on this curriculum path. Today's module are based on SEP, is on requirements.

There are also modules developed for non-SEP based standards, meaning that they were developed earlier, and have no benefits of documentation provided by SEP process. So the first four are again, the same, and followed by the series of standards that were not developed using SEP, are also dealt with in this series of PCB program. There are certain standards that have 300 level exposure, meaning that you actually write details for your device oriented procurement and you end up with the system that you're looking for. And there's a lot of work in those standards, but we have several modules developed under these concepts. For A321b module, we have prerequisites that must be followed in sequential manner.

The last one here on this list is the module previous to this one, to this, and it is required because that dealt with user needs for traffic management data dictionary. In addition to those requirements, we also have several areas of knowledge, areas where it might be helpful to us, and exposure, particularly the last one here, Traffic Management Centers. As you know, the TMDD, Traffic Management Data Dictionary, is designed for helping traffic management operation centers, and traffic management centers to conduct the business for traffic management application areas.

There are six learning objectives for today's course. We have three here shown, are basically dealing with the information process, where the information is, the structure of the standards, in learning objective one, and also the role that NRTM plays. NRTM stands for Needs to Requirements Traceability Matrix. So that's where the information is available for our needs. Also, learning objective two, deals with requirements in detail. Now what the description of each of these requirements are, and how we can use them for building a system, is under this learning objective. Also the RTM, the RTM is also a table that deals with requirements to data concepts linkage. In other words, it's a traceability matrix that links both important elements of the requirements and the design concepts, so we will understand what role they play, and how we can use it for their purpose. Four, five and six, are following up with a little more in detail, continuing with the use of RTM. In four, we will learn how to use RTM for our specification, and where to locate the data concepts in that particular module-- matrix, I mean. Also, number five deals with, what if certain things are not available in this standard, what should we do? So there is some discussion about that, and we will deal with that. Finally, there's a guide. The TMDD also has a guide that's prepared specifically for helping users to prepare certification and acquire standards based system interface.

So together, these six learning objectives take us to where the information is, and gives us the methodology to use that information, and then prepare RTM and how we go about putting that information in our specification. There are four key areas that we had discussed under A321a, which was the previous module, based on user needs discussion. First thing was that there is a need for system interface, a system interface is required for Center-To-Center Communication, and we discussed that in that module, how and what required, what is driving that need. So Center-To-Center Communication allows us to exchange information among different centers and particularly for traffic management related activities, and that's what system interface was prominently discussed in that module. Also the structure of TMDD standard itself. What information is available where? So there was a lengthy discussion about user needs, and requirements, and the related data concepts for specification purpose.

The third area is operational needs, so where are these categories, what are the categories, I mean, like where do we use TMDD, and what is the coverage area for TMDD? So there are eight categories of operational areas that we have to deal with in this process. Also the NRTM, Needs to Requirements Traceability Matrix, how do we prepare one for our project? So those four areas were highlighted in A321a, and we will briefly discuss them here, so it will prepare us for today's discussion as well. So what is system interface? A system interface is an entity, is a software entity that has a definition provided by IEEE. This is a definition that we use in terms of creating a boundary with the local system.

So a system interface is a shared boundary across which information is passed. And how does that work, is that we have a local or native system, it could be a traffic management system, 911 system or 511, whatever system it is, the system interface makes a boundary with that local system. Once that is done, we are able to exchange messages with this system here and outside world. So there are messages coming in, and messages are going out, so this is the third attribute of system interface. First one is the definition, then it's location with the boundary, third one is the ability or capability to exchange messages. And the fourth one overall, that this setup of system interface allows outside world in an open system environment, meaning that this is open, this is based on standards, so anyone can access this methodology here in front of the local system. So that's why it's open, it's not proprietary, it's available to anyone who has that need for.

So there are four attributes that we talk about in terms of what a system interface is. So we look at this representation here, where we implement system interface, so there are two TMCs for example, in this layout here, and system interface appears in front of each of these local systems. Whatever the purpose of the system is, regardless, system interface will make a boundary in front of the local system. And in the middle here, we are exchanging different types of messages to different-- to perform different tasks. For example, how we use the system interface, these are four examples here. We could use system interface to manage field assets, you have a lot of field devices, dynamic message signs, traffic controllers, and so on, and all of those signs and devices can be managed through the system interface. Also we can retrieve information from each other, the question formation and decision formation. We can also monitor the devices on a 24-7 basis as well. And then, occasionally, we also have a need to control devices. We can make requests to control someone else's traffic management related activities, using certain devices. That can be done.

So these traffic management centers together, can do all of these tasks. And that's where the implementation is about. We can create this capability which we don't have otherwise, so system interface makes this possible. Also after you receive information from each other, we can use that information in the real time, which is what I like to say, in a way that our operations, our traffic management operations are enhanced, because we are using that information in real time. And that's a benefit that we draw, or get from this use of system interface. What it takes, what are the key components that will enable or prepare us to develop a system interface. So these three components here are shown, user needs. User needs addresses what? What are the operational needs? Why we need a system interface? So it addresses the problem or situation on hand.

All right, requirements deal with the shell written in a shell language that deals with details. How do we make this thing happen, how do we address this need? So they take us to a next level in detailed way, and they actually deal with functionality, what functionality we'll be using across the interface. And the final element here is the data concepts. The data concepts make us ready for design. They provide all that you need to create the design so that your local project is completed or prepared properly. So these three elements make system interface specification. Now the data elements and user needs and requirements, are addressed for these eight operational areas, and eight operational areas have general requirements for particular tasks on hand or whatever. But here, for example, the first three, are preparing the network. There are two TMCs involved, or more than two TMCs involved, and we need to make them connection ready, we need to authorize restriction, whatever, plus the organization behind those centers, so that kind of information is supplied through these three areas, and we deal with that in terms of user needs and requirements. So all of these eight areas provide-- are provided for user needs requirements, and then supporting data elements and data concepts, actually.

So let's look at just one, as an information here. Let's look at number six here, for example. The TMDD has covered approximately ten different devices, dynamic message signs, traffic controller, detectors, environmental sensor stations, highway advisory radios, all of those devices, they have different functionalities in them, and TMDD has actually provided user needs for those devices, as well as the requirements. All right, so these eight areas are supported by TMDD Version 3 standard in a very detailed manner. The last one here is like, if you don't have anything to send to anybody, or you don't have any more information, so that addresses this, so null means zero, there is nothing to send, and so that kind of issues are dealt with in that category. So all told, TMDD Version 3 standard has accumulated or provides inventory of 126 user needs, 134 requirements related to those user needs, and then close to 600 data concepts and a report after those data concepts shown here. We'll discuss that more in detail later. So all of this, elements of TMDD, are presented here, to show you the overall scope TMDD supplies complete needs that you may have, based on those eight operational areas that we just seen before.

So those are the three main parts of building materials that we might need to prepare a system interface specification, all right? NRTM, and RTM, both are connected to tables, they connect, and RTM, for example, connects user needs to requirements, right, and RTM connects requirements to data concepts, so both of them have different purposes, one prepares the problem definition, and then comes out with the right type of requirements, this one actually takes those requirements and converts them into design process, so you can see that the problem side, maybe perhaps one way to look at it, and this is the solution side with providing a single design for each one of the requirements.

So how do we prepare NRTM? That was a main topic of a previous module. Here we are briefly reviewing it. The NRTM is provided in Section 2 of Volume I, and we can look at those detailed user needs in Section 2, and then, by the ID number, they are already connected to the requirements in the Volume I, so you have user needs coverage in the NRTM in the front, and then you have requirements coverage toward the end of the matrix, all right? So this you will see on page 174, of Volume I. And this is what NRTM is. Now, TMDD has full detail of complete NRTM for the standard. What we have to do as users, is to take that table and convert for our

project, all right? So we have to reduce, based on our needs. So in other words, we have to prepare the project level NRTM with these two connections in mind.

One is that operational needs are connected to these user needs, and secondly, user needs are connected to requirements in section three of the Volume I. So that's what we have learned in preparing project NRTM. So how do we achieve off-the-shelf interoperability? If we use the standard in a proper way, we will end up with getting the interoperability what-- that we desire. What is interoperability? Interoperability is that, in a practical terms, it can be status saying my system is able to talk to your system. If you want a more formal definition, a complete definition, then it says that two or more systems or components, are able to exchange information with each other and use that information to exchange. So we can see that the emphasis is on ability or capability to talk to each other. And that can only be achieved if we use design solution specified in the standard. That means that we have to take each user need, and whatever the requirements are allocated to that, all right? So we have to follow the standards. And proprietary solutions, like someone trying to come up with his or her own solution, is not acceptable. If we do these things, basically, we will be working toward achieving off-the-shelf interoperability. So this is the emphasis that the TMDD has placed and we should be mindful of that.

So let's summarize quickly, what learning objective one meant to us. It tells us, generally speaking, that where the information is in TMDD Version 3 standard, and we found out that it supports the system interface development. We take whatever is available and put it in our specs, according to our project need, and we come back with a system interface a design that will work for us. Secondly, we also learned that to get a system interface that works for more than two centers, an ability to talk to each other, we need to follow the direction given to us in NRTM, and also, we'll find out later, RTM. T

The first three things, user needs, requirements and data concepts. We know that they are provided by the standards. So standard has done this job very well, and it provided lots of setup definitions for all three of these things. Our job, as a user, is to prepare project level NRTM, and to prepare project level RTM. So the module-- previous module has taught us how to do that, you know, the A module has taught us how to prepare for project level layer in RTM, and that's what we learn in planning objective one, today. Today we will discuss RTM, how to prepare RTM for project, and that's something that's a complex task, but nevertheless, has to be done at project level. So turning back to now, learning objective two, where do requirements and data concepts fit on system life cycle process?

So we know this V diagram, we've been following it to develop an ITS project, so we have to answer that question very quickly and say, well, they follow, actually, the concept of operation, where the user needs are, right? That's the first step. That's the very first step we have to deal with, and then, that we understand that in the previous module. Here, today, what we are saying, the requirements and data concepts requirements are located at the system requirements level on the V diagram, and data concepts are emerging at the high level data design concept. So both of these things are following each other in terms of sequential steps.

First the user needs, adhere to the concept of operations. Next, the requirements at the system requirements level, and then finally, high level design we deal with the data concept answers,

and then we follow up with the detailed design as well. So one of the key ingredients that we have to take with us today, in a more detailed way, while requirements attest to user needs, and requirements fulfill some aspects of a user need. If there is a user need, then requirements follow, okay? So in that sense, they are actually detailed description of what information is, and how it is exchanged, so requirements are more detailed description of how things will happen, or what will be required to make that thing happen, so that we can talk to the external centers.

The second thing is that they stand for, or they are tied to certain functionality. What kinds of functionalities are supported across the system interface? I tell you that-- quickly, that there are ten NTCIP devices that TMDD provides support for, so each one of them has different functionality, for example, dynamic message signs, they display messages on the surface, and people can read the message. On the other hand, some other devices do some different task, okay? So all of these things end up with different functionality and all of the things are addressed by the requirements in detail. A structure of a requirement, it has an ID number. This is a unique ID, and it's contained in Volume I. All of these three things are in Volume I. A requirement ID is very important, because that is the beginning of our second level process beyond user needs. So here you have ID number, then it's followed by the title, requirement title, it has a specific, unique title as well, and this is returned in italics in the standards, so you can separate title from the rest of the description.

All right, and the description is very, very plainly stated, and it tells you what purpose this requirement serves. So these are the three key components of a requirement. What is a classification, in terms of how many different types of requirements exist out there? Well, at least we have to say that there are certain requirements made mandatory by the standard. To conform to a particular standard, or particular, in this case, it's TMDD, but in general, standards also have their own main minimum requirements and in this case, for TMDD, for example, certain requirements are mandatory, the one that's shown here, this has been made mandatory. So to conform to TMDD standard, all mandatory requirements must be met, that's one. Optional requirements, on the other hand, are left to the user by the standards, so we'll have to figure out how to deal with optional user needs, how do we go about choosing this, all right? So there are two types of requirements, mandatory, and optional. In this example, one user need generated ten requirements.

This is an example, okay? So there is a relationship between user needs and requirements. It could be one user need to many requirements, as in this case, it's ten. It could be one to one, it could be one to three, you know, so this is apparent, so for this user needs, with this ID, the standard has accumulated or provided these ten requirements, so we are more interested in these ID numbers for requirements, and the ID number for this, because that's how the NRTM is linked. And in this case, we have selected the support, yes, we want this thing in our specification. And this is just for administration purpose here, okay, and then these things could change, because out of these ten, certain requirements are mandatory, and certain are optional, and we'll see in a few minute. So this is the NRTM first three columns deal with the user needs, and the last five columns deal with the requirements, and that's how the setup is there. Those requirements that you saw, here they are with their titles. And these titles tell you something, it indicates what's behind those requirements. For example, this requirement identifies an operator, and it's optional. This one says, who dealt with the information, okay, it's also made optional.

Also, the name of the organization is optional, and the last operator who worked on this particular request is also optional, so these four are made optional, right? And the rest are mandatory as you saw earlier.

So we have to figure out what to do about these things. So requirement is implemented by providing one single design. If there is one requirement, there is one data concept that will be required, or more than one, data concept will be required, and those are identified. So this is the job that RTM does, and it does pretty well, because you will see that RTM deals with each requirement on a separate level, in other words, there is a line, or there is a row for each one of these data elements that addresses particular requirement, all right? So there is a separate data concept for each requirement. Also, these data concepts are linked to a selected requirement. So these data concepts-- there are four types of data concepts. There are dialogs, messages, data frames and data elements. And this is inventory provided by TMDD standards. And they're all related to each other, okay?

Let's understand what a dialog is, very briefly here. Dialog is a sequence of messages. It prepares the system for now starting the conversation, all right? The messages are actual conversation. Messages have content in them, they mean something, all right? They may be particularly about information event, or it could be about a device. So they have very specific information that becomes part of the conversation, which is now started with the dialog. So dialog begins the process, messages are the content and this content is made using these data frames, or data elements. Data frames are a bunch of data elements coming together, they're grouped together. So these are all the things that are used in various ways, in terms of listing in the RTM, so there are four data concepts that we should be familiar with. Let's do a little activity here, and answer this question in the chat pod. What determines requirements for a system interface project. Go ahead and please type it in your response in the chat room.

Okay, RTM, user needs, user requirements, user needs which flow from business requirements. You have objectives, you have certain things that you need to do in your original aspects or project level task, and all of those things reflect. So that's where you are coming and say, well these are my needs, right, so these answers are leading us in the right way, to the requirements. So then the question is, do we select all 134 requirements? And very quickly, we will say no. We ask the next question, and say, do we select only mandatory ones? Obviously, we're not going to say yes, we're going to say no, right? And then finally, we deal with this question, and say, do we select based on the project's needs? And the answer is yes. So we say, there are three different ways we can sort this thing out, and this example helps us to understand what we actually meant. So for example, here, these user needs generates certain requirements, right, and in the example previously we discussed, we said that there are ten requirements are required, or ten requirements are necessity to fulfill this user needs. That's what we said, right? So these requirements are allocated and here are their titles. And with these titles, we have decided to react to those titles, and say, well, do we need them?

So first, we're going to say everything in the red, they are mandatory, because the standard says that you must pick them, so we pick them, right? Here, these four, we don't really know going in, so we may have to ask somebody, and that somebody could be an operator. Okay, operators usually have very good knowledge of what they want in their menus or the interface that they are

using in the TMC. So you have to decide whether on this one, for example, that says, operator identifier, do we really want to know who the operator on the other end is, if there are two TMCs, or more than two TMCs, the question come up, should we get that information, who received our request for information. Chances are very good that not all centers will ask for it, but there are some, like state level operation centers, or, if you are dealing with a 911 center, or outside world, you might want to know who the operator is on the other end that is receiving your request. So in that case, you will select, not optional, but at that point, it will become necessary, and you will select yes. That means it becomes part of your spec. Also, in the same way, the last person who locked the information under their name, saying this device is not available, right? Some of this details may not be necessary for everyone, or every center. In that case, we step out of optional and either keep optional or select them. So this example tells us that that is how you make your decisions, based on your needs, all right? And then we separate the optional requirements from the mandatory ones, so right away, you have good combination, mandatory ones are selected already, and we make progress, if we don't know ourselves, whether this should be in the specs or not, we should ask the PMC people and perhaps get a good answer from them. But it has to be disposed of. So that's what this example prepares us in terms of what should be optional and read how to include mandatory ones.

So in summary of learning objectives two, we said that the requirements are listed in Section 3 of Volume I, both user names and requirements are in one place in Volume I. Also mandatory requirements must be selected. We cannot do any other thing, we have to select them, yes. And then, select requirements that fulfill-- well select the allocation of requirements to user needs is done in Volume I, and how that requirement is fulfilled is done by RTM in Volume II. So that's what we are summarizing.

So for NRTM, the connection to user needs and requirements, is done in Section 3, and in an action from requirements to the data concept is done in RTM in Volume II. That's what we are learning from learning objective two. Now let's take a look at the nature of RTM. In this learning objective, we need to make sure that we understand the structure and the role played by RTM. So for every requirement, we have the mechanics that allows us to test that requirement, so at the level of requirement, and data concepts, this is the requirement and these are the data concepts. So they are linked by their ID number. There's ID number here, and also in the last column, there is ID number. Now by providing to us, through RTM, the standard has reduced a lot of design work that we would have otherwise to do. So this is one good thing about TMDD standard, is that all the design work to connect requirements to design is done through RTM. So RTM, in a sense, relieves us from design work, and simply, we just go out and select, based on the requirements that we have for our project. Also, it helps achieve interoperability, and why are we saying that thing? Because the RTM forces us to select the data concepts that are connected to the requirements. So if you are TMC A, you will do the same thing. If you are TMC B, you will do the same thing. So if everyone selects the data concepts which are tagged to that particular requirement, then both centers are in the same boat, meaning that their systems now, be able to talk to each other, and exchange information with each other, because they use the same design data concepts, and that's the good part of RTM to understand.

Let's look at this RTM as it appears in the standard. This one is taken from Page I-81, and when you look at the RTM, that's how it's going to look, that's how it's going to show up in front of

you. So the front portion is the requirement, these two columns, right? And the last column, the last in the green, is the data concepts. So now this is a very compact presentation of RTM. So solutions are provided for each requirement, for example, let's take a look at this for example, this one here, the fourth row is-- ends at four, also the same. For the same requirements, right, each row delivers one particular data concept. This one is for messages, this one is for data frame. So we are repeating data concept for each requirement in a separate way. In other words, data concept has its own place. And this case is a dialog, dialog, dialog, message, so everyone is separated in a data concept type, And then it's connected to a clause in Volume II, this is Volume II, all right, this is Volume II, this is Volume I, okay?

This is-- they look the same, three, but this is Volume I, and this is Volume II, so we have to be aware of that. So for example, another example here in this case, you also have one message, and you have a data frame. So when you navigate this information in RTM, it will be very clear to you that a particular requirement, there is a particular data type that we have to pick for our project, if that is what is needed in our project. That's what RTM does for us, it provides us design solution for each requirement, very, very compactly.

There are two types of data concepts that we have to be aware of, the encoding formats, meaning that, how the data concepts are coded for further use in the system design. There are two formats that we use in TMDD, one is called, ASN.1. This is based on another international standard, 14817, so this is a standardized mechanism provided to describe a particular data element and a data concept, all right? Similarly, there's an XML, Extensive Markup Language, is also based on particular standardized schema or structure. So we have two formats, ASN.1, and XML. Now we have to select one or the other, all right? We cannot mix them both. So we have to make that decision as a user, when we prepare the specs, which format we're going to use. So that's the agency decision, big time, and we have to be making it. Second thing attached to this is also interoperability.

Interoperability is something we have to keep in mind all the time. So if you are looking for interoperability, we have to select the same format. Both TMCs, or more than two TMCs that are involved in the project, then they must select either one of them in a common way, all right? You cannot have, one center cannot have ASN.1, and the other one, on the other hand, can have XML, that's not going to work. So we have to make sure that we understand this representation. Let's look at, very quickly, how each one looks like for understanding purpose. Well, this is ASN.1 representation, has several definition structure. In other words, there are components to it. There's a definition, there's a name here, but there is an ID, or sometimes you report it as OID, Object ID number, so that's there. This is a definition what it entails, what this element represents, right, so we have to understand, this example here, says that this is the current volume for a link. Alright, expressed in vehicles per hour. So up to 100,000 vehicles per hour we can express in terms of definition provided by this, and this gets coded at the end, so this is one way to express our data needs.

This is the other one, XML, representation, which is returning very English syntax, and it's between the tags. So here are certain things are not there, that you saw in the previous one. But nevertheless, this conveys the same meaning, saying that we want to get the current volume for the particular link, so that appears in terms of number of vehicles per hour, alright? So this is the

second one. So question comes up, how does RTM help us to keep track of all these different things? Well, if you look at it in a way, organized way, you will see that user needs, requirements and data concepts are linked to NRTM and RTM. We want to make sure that we understand that the role RTM plays. Well the role RTM plays is in Part A, and Part B. It places everything to requirements in a backward way. In other words, it leads us to each requirement, and eventually, requirements take us to the front portion, which is the user needs, right?

So there is a reason to term this backward compatibility, meaning that where did this requirement come from? Why are we having this requirement, and we keep track of it throughout the whole process at the system engineering level, all right? With that V diagram we saw earlier, we keep track of it, thinking that this particular requirement has a purpose. In the forward direction, we also answer the same questions, saying, well why these data concepts are included in here? RTM helps us and makes it very clear that each one of these data concepts that's written in the specification, has a purpose, and that they are linked to the requirement. So this thing works very well. It works in the backward direction, and it works in the forward direction to the design process. So if you are-- who benefits from RTM? If you are one of those four entities shown here, and let's say you represent your agency in terms of you're writing the specs, right, you are the specification writer. So you have to make sure that it appears as a checklist to you. You don't want to miss anything, and then if you miss something later on, it's going to be difficult. Plus, your project will be incomplete, and you don't want to have incomplete system interface design. So you use this as a checklist.

If you are a user, you want to make sure that you are connected with the ITS projects that you're working on, also looking at the interoperability needs for the region, you know, you work with the other people around you. We no longer work on the island in ideas, you know, we always have other people to take care of, or connected with them, so that gives us a good idea in terms of interoperability needs. If you are a system integrator, it works even better, because now you want to make sure that you are not missing anything in design, so this checklist will help you to ensure that all elements or all requirements are covered, and then, you don't have to deal with that later on, and risk a failure of a secure system interface. If you miss something, you might end up with an incomplete or complex interface that may not be necessary, and then you have to deal with the client if you are integrated, and you are working with a client. The client may not like that, so a lot of these things could be avoided if you pay attention in terms of system development process using RTM, so that's a very good advantage.

If you are a supplier, use your handle on what users want, what their necessities or their capabilities, requirements are, and so on. So benefits of RTM are multitude, and they come in a different direction, and they help us in general. Let's have a little activity here. What do you think, what are the key functions of a project RTM? Go ahead please, answer that in your chat room. Okay, verify that the system objectives are being met. Also link requirements to the data concepts. So you have a very good terminology here. We need to verify whether all objectives are met. How else are we going to do? We cannot go around and ask questions, right, so we need some kind of methodology that connects us to different objectives that we have in the system process, so that's a very good answer. Anything else? Also, make sure that every need has a matching requirement, and no requirements are inserted without a need. It's an excellent statement. That's a very good statement saying that there is a reason why a requirement exists,

okay, and we have to ensure that it's there for a purpose, and that it's not there without any support for user needs. In other words, if there is a requirement, there has to be a user need linked to it, at least one user need, minimum linkage should be to one user need. So this makes sure, RTM makes sure that there is a case, it connects us in terms of every design concept that we have inserted into a requirement that is real and connected to user needs eventually. These are good answers.

So let's just summarize quickly, in the sense that the RTM functions, projects, RTM ties the requirements, which is our foundation, to data concepts, and it provides a reference to verify that all requirements are contained in the system interface specification. Perfect. So you have requirements, you have data concepts, you have linkage for verification purpose, and it goes right in the specification. So that's what RTM Functionality can be summarized. Also, similarly, let's do this question. Answer in the chat pod, also, what are the minimum conditions for interoperability? And some answers are coming in. Based on the same standard, okay. Another one, it says, user should use same language. Yes. So, and each line set up there? Okay. All systems use same standards and language. Perfect. This summarizes very, very correctly in answering, saying that we have certain needs that must be met for interoperability, at minimum. So we can summarize saying that, use project NRTM to choose the same set of user needs. We mention that user needs should be the same and language should be the same. And NRTM will help us to do that, alright? Use project RTM to use standardized design concepts. This one will link-- one links-- the NRTM links to the user needs, and RTM will link to the design process, so the solution side of it, as you said. And then you need a common communication, which is the language someone mentioned here.

So we've got to use the same common communication protocol so we can end up talking to each other on two different ends, alright? So these are the minimum conditions that we have. Now turning to learning objective four, we are saying, that data concepts and requirements are to be tracked at each different level of system engineering process. So how do we step out from requirements to data concepts? Well, here we are saying through using-- through RTM. If we use RTM properly, then we will be connecting requirements to data concepts. So the first thing to do is to turn to an RTM and look for various requirements. They are in Volume I, page 174 to 294. So if you take this step, you will be in sync with your project level need. Next step, you will have to do is to approach at the high level design process, you will have to start selecting, or taking the specified data concepts. Now you're not doing anything, you are not designing yourself. This is a point I like to make very strongly here, that none of us is charged to design data concepts, none of us is charged to even come up with a data concepts that links to the requirement. What we are simply saying here, that using RTM on these pages here, Volume II, we are picking whatever requirements we have come up with, and whatever the RTM tells us to go to that design concept. If we can do this thing at this stage, we are doing very methodically, all right? To help you understand this subject a little more detailed, we have developed a seven box process for you to follow step by step, you will see there are seven sequential boxes laid out in supplement. So I recommend you to follow the process that we have in supplement. That will really make it very clear for you, which is first step, what is the second step, what is the third step, all the way down to the last box, will lead you right into the specification design concepts. So go ahead and please, and when you have a chance, look at it, and that will help you.

All right, so now the organization that RTM has offered, has brought different level of data concepts. There are four types mentioned before, the dialogs, messages, and data frames and data elements. Those are the data concepts. How are they organized? This example will help us to understand that. In the Volume I, Section 3, we have the requirements, right? Each requirement is separated, so we have that list there. In Volume II, Section 3, we have each one of these data concepts listed separately. So there are three generic dialogs, the first step, and there are three generic dialogs also guide us in between, and they take us to the real dialogs, which are stated here. So this organization allows us to connect requirements to the data concepts in a very significant way.

For each different device, this is a CCTV device. Previously you saw one for dynamic message signs reference, so that kind of approach is taken by RTM in providing this organization for each different devices, all right? So what are the generic dialogs? There are three ways we can get information from each other. So these are three different types of dialogs. They are connected to each other, so there are dialogs that we will conduct to get information about any of the ten NTCIP devices, for example, or event information, so the process begins with this generic dialog trying to understand. So 2.4.1 is the request response dialog. If this center, this is an external center, wants to obtain some information from owning center, or traffic management center, then this dialog 2.4.1, will come to begin-- will come to play, and it will begin the process for making a request and receive the answer, all right? The next one is 2.4.2, will be getting continuous information on what you have subscribed. This TMC or center, will continue to send information to you as you have asked for. And then there's the publication dialogs, also, that follows this subscription dialogs. So we go into detail about each one of them. Generic dialog means that if you want certain information from your external center, then you go to the owning center which, the TMC, for example, and say, please give me this information. So your message, request message goes out. A response message comes back to you, all right, with information that you asked for. Another message will go out, so on, and so forth. Sometimes you have error in the process, and some error develops, then you will get an error message. In that case, you will have to resend your request, so that's how generic dialog works. In terms of subscription, you have many, many things are out there, and you may not want all of those things. So, for example, you are a traffic controller type of center, you are only dealing with traffic controllers, and nothing else, then you don't want all kinds of information coming to you. It's your choice, so subscription means that you tell the other centers whatever your needs are, and that center will send you information based on whatever you subscribe to, all right? Just like we get the magazines in the mail or whatever. So things like that will happen during this process of making a request, so this is the subscription method and it goes out to the owning center and the response is sent to you, so there is a confirmation coming, and say, okay,

I have information that you asked for. Here it is, that kind of thing, all right? The last one here, is providing you updates, and say, okay, don't bother me every minute, or every two minutes. Give me periodically, every ten minutes or 15 minutes or half an hour. A lot of centers prefer to get their information on a periodic basis, this way they are not overloaded at TMC and end up getting information that they don't use, or most likely they don't need. So, and in any case, so periodic information coming to you is handled. Also, we live in the world where ITS is used heavily for managing incidents, for example, all right, or if you want to look at it in a different

way, saying that something happened, all right, a natural disaster or natural emergency occurred, in addition to traffic incident.

So now you say, when things change, send me information. Okay, that's the nature of this dialog here. When things change, send me information. Give me an update, or if the event has occurred, send me information. So those are the kind of pieces of information will come to you, so you're not overwhelmed with unnecessary information. How does this process work? Well, let's look at this example, illustration here. It says, okay, I want-- this dialog says, dl stands for dialog, it says, DMS control request goes out to the traffic management center, and requests information about the current message on a dynamic message sign. This is owned by this center. So this says, okay, this is the current message, and the response goes back to the center here, all right? So these kind of messages go back and forth using request and response dialog. This is an RTM that displays all those requirements that will be necessary to make that happen the way we discussed the previous example, so this is that generic dialog we just saw and now it has certain requirements that we need to fulfill. To do that, the data concepts that make that design happen, are here. So right away, what we wanted to do, what is required, and now, what the design concepts are. So all three things are coming together, all right? The functions, the requirements and the data concepts. They are also traced, dialogs are traced. Every dialog is traced to a requirement. They appear for good reason, right, and therefore, they need to be tracked, or traced, so in this case, 3161, 3.1.6.1, is actually traced to this requirement. So in the middle of the design process, every dialog is tracked. Similarly, messages are also traced to dialog, which, these messages support is for a particular type of dialog. So same dialog is supported by this requirement to comply with this requirement, this dialog is now traced to this message, all right? So the message, the dialog, everything is coming together. And here it is, that message. This message is input-output kind of setup here, and this is the target name space, TNS stands for target name space. So this binding is occurring here, in other words, the new connection that this message represents this request, and this message represents this response. So this kind of tracing is provided in terms of RTM. This is an ASN.1 representation, also here you see those messages are connected to these particular dialogs. So this particular dialog here, requests response dialog, and these are the messages that will support that.

So now, turning to our learning objective five, in terms of verification. Where are we in terms of whether the requirements are complete or not? How do we know? As you know, system engineering projects, system interface, particularly, we have to keep track of all different details in front of us, and we need to verify them at every level, every stage, so that we can say that the requirement that we begin with is now addressed, all right, and is complete in terms of the user needs that it serves. So this connection is verified for requirement. So that's one part. Also, we want to make sure the requirements are complete, that nothing is missing. If there is a requirement missing, then we will have incomplete design, and incomplete design is not acceptable for system interface, and I will also caution us that system interface is designed to alleviate our problem, which is the inability to talk to other systems, so if we leave things incomplete, we will not be able to eventually talk to each other, so that's also eye opening kind of notion that we have to make sure that we understand. So requirements must be complete, that's one thing. The second thing is that every requirement is traced to a data concept, so what if we miss a data concept, what if we miss a data element? Alright?

So we have to look at a fine observation has to be made that we are not missing anything, and whatever we have is actually tracked or traced to a particular data concept using RTM. So this is the foundation of verification in the way we understand. What do we say now to ourselves, is that we are progressing in the project. And remember, we have started this process long time back, maybe months, maybe even a year, whatever timeframe it is. Now we are coming towards the development process, probably in the final phases, and the end almost, and we need to make sure that we are following all along these different stages in terms of our needs here. So finally, when we come toward the end, we have to accept the system interface, and say, let's check it. How do we do that?

Well, we started out the user needs as a condition and it's the first step toward interoperability, for example, right? So user needs will help us to validate the system that whatever we have built is the right thing, because it meets user needs, right? So that's a good question, answer in terms of validation. When we come to the requirements, we have to do the same thing, or verifying whether each requirement that we had outlined earlier, is now actually being built, and proper design concepts are used. And how do we do that? We do that through this process at the verification level. So we say, yes, we know that you have-- sorry, you have built the system correctly. So this declaration of victory will come in two parts, and here at the verification level, we will ensure that every requirement is checked out. So this is also the time and place when you get at the TMC, and you actually try it out, you are hands on, now you're verifying in terms of the response, whether, are you actually able to talk to other system, you know, and see if that's happening or not. So that is a kind of process we do, and we do, all through this process, process checking with the requirements in this case. So condition again, achieving off-the-shelf interoperability. We have realized, so far we have said that there are two encoding formats that we have to make a decision on which one we're going to use, right? And assuming that you have decided and you have one for your project, right? Now the other center on the other end will have to do the same thing, will have to have the same format for the data representation. So condition number one, is that everybody who wants to talk to each other, or exchange information must have the same encoding format. That's condition one.

Condition two is that all centers must have the same set of user needs, requirements and data concepts, and we just finished talking about it, how and why they are connected to each other. So that makes sense, that if you end up having same user needs and requirements, and data concepts, now you are able to reach out to the interoperability needs that you have. You actually can make sure that the system interface will deliver that, all right? So the third condition now takes us to this common language that you mentioned earlier, that it has to be a protocol which is the same, whether it's ASN.1, based, or XML, and there are choices, and we'll talk about that in a minute, all right? So there are three conditions in front of us, and they appear together, so that we can focus about all three conditions at the same time. So these two standards already mentioned several times in TMDD process, is the first is a TMDD standard itself has to be used, and this is Version 3 we're talking about today, right? This standard is available so if you want to get more information, you can download that information in that file. I will recommend that you only get your standard information from this particular source. This is authentic source, okay? So that's one standard you can download. The other standards, the NTCIP 2306 is XML based, that's the common protocols. If your agency decides to use XML based implementation, then you have to include this in your TMDD specification, that's one, right? On the other hand, if you choose to

use ASN.1 based implementation, then you have to take this standard, 2304. So there are two standards, and both of them are available at this site. You can go to this site, and one time, a download is available to you, free, all right? So I suggest that you reach out to these sites and get your standards for your purpose from there.

To summarize, learning objective four-- I erroneously said number five. It's four, sorry. So summarizing learning objective four, we have realized that RTM provides design-data concepts for each requirement. All right, it is RTM that takes us to the design process, so that's what we have realized in this learning objective. Dialogs allow conversation-messaging with each other. Dialogs are actually the real sequences where information will flow from point A, to point B, between the two centers.

So there are important methodologies are used in terms of carrying out the messages. RTM itself is used for tracing data concept for each requirement. And I mentioned that, for each requirement, user needs is repeated in terms of data concept, right? Data concept appears only once in each row, so it's easier to track, so sometimes you will see several requirements rows appear for each data concept. And then, centers must use the same data concepts for interoperability. This thing doesn't go away. If you use different data concepts, let's say someone decides to use-- the requirement is the same, but the data concepts are different, then, interoperability is broken, so we don't want that situation. So we want to stay with the same data concepts in each specification if you want interoperability. So that's what we're summarizing in terms of learning objective four. Now we are now turning to learning objective five, which answers the question, sometimes raised, and say, will the standard contain everything that I need? And it's a reasonable question to ask. And I provided that inventory to you, 126 user names, 134 requirements, 600 data concepts, and it's possible something may have been missed, but not likely. On the other hand, if you have a new need, and a new project come up and say, well gee, you know, I looked at the standard, this one is not addressed in the standard user needs, what do I do? So that question is answered by TMDD in a very methodical way. What the standard is saying, that there are certain rules that you have to follow and they are laid out in Section 1.6.1 of Volume I. So standard is very particular in saying that if you follow these rules, then you can take it to the next step, alright? And this is done, so that your interoperability needs are also met. Now does the standard encourage these things? The answer is no. No one seems to say that you can make an extension, you know, in fact, all standards recommend that you do not go outside of the standard. But if you do, then you have certain obligations to follow.

Anyway, in considering, in TMDD, what we are advising, what the joint committee TMDD committee is advising, is to contact ITE, because ITE is maintaining this particular standard, and these sources will allow you to take the next step in a very logical way. Also you can consult the TMDD Version 3 Guide. The guide has given very detailed exposure in how to prepare a specification, but it also has discussed, what if your needs are different, or somewhat new, and what should you do? So supplement, a Student Supplement, page 26, deals with this subject, so I encourage you to go into supplement reading and find out for yourself, what it discusses about. In terms of conformance, very often, this word, conformance is also coming up in the context of compliance, so let's just look at conformance, what it means in terms of standard TMDD standard.

So to conform to the TMDD Version 3 standard, all the mandatory requirements must be selected, and if user has selected certain user needs, they must be also considered mandatory for conformance purposes, all right? So that's one level of conformance requirement. The second thing is that mandatory requirements, as we have discussed, must be in it, must be in the specification, alright? And if these are selected requirements for that particular project, then they should be all made mandatory as well. So the requirements are, Part A is mandatory by standard, and Part B, made mandatory by user for the particular project. So that's second. Third thing is that all the data concepts are allocated by RTM. So user, you, as a user, you are not deciding anything. Therefore, you have to follow RTM. If you follow RTM and collect all those data concepts that are attached to that particular requirement, then you are okay, so that's the third requirement for conformance to the standards. Alright, so these three together will allow you to say that your specification meets the standards of version 3 standard.

When just the interface implementation is completed, and is tested, then you will be going out and look at, for all of these different things, and to see they meet the standards requirements. So these one, two, three points, are important to realize. The more detail will be available to you if you refer to T101, that discusses the subject of conformance in general. When there's a standard, it's conformed to, and that goes to all standards in the NTCIP and TMDD and you can read, or take that particular module and see what that discusses. We are now turning to learning objective six, which is about the guide. The guide is a companion document. The guide is prepared because the TMDD based implementations are very sparse, very few and they're up and coming, and the guide will help you to understand the subject in overall context of your needs, operational needs, so this document is very recently published, and it's available for you to download. But this is a guide that works with the standard, so if you have a guide, you cannot say that we don't need standard, you know, you need both to work with each other. So summarizing key parts of the standard is what guide does well, because it takes you in Volume I, and say, where is the information, it takes you in Volume II, and says, where is the design information, that kind of thing, okay? And also, step by step it lines out the steps to prepare specification, four major steps that we have to go through, and all of them are explained at each level of the V diagram that we discussed today. It also provides implementation guidance. Now, implementation means that you need to have TMDD based system design specification, but you also need additional things connected with that common protocol that you need, so during implementation, you need both, TMDD and the common protocol that we discussed, all right?

So guide helps you in understanding what those components are in terms of implementation. And as I said earlier, this is recently published, and you can download at this site, the guide at this site. Let's take a comparative example here, that's given in terms of what's in the guide, by section, by chapter and sections, and what's in the standard by volume and section. So let's say if you are reading the material and you come up with this question, or you want to straightaway answer certain question on your mind, so well, what do I do? Well, what does the guide do for me and what the standard does for me. And before you go to standard, perhaps you look at the guide, and say, well what does the guide say, you know, or if that's what you want to do. So for example, take a look at it, it says, number 6, what is conformance? How is it different from compliance? Very legitimate question.

If you are new to ITS, and if you are new to system engineering process, this question should be answered very quickly, early on, and you should grasp everything that you can. So the guide will say, the minimum in this section it says that this is what conformance means, and this is what compliance means. Now when you go to the standard, there is a very bare minimum discussion. You know, it may not go all over, but the guide actually helps you in the context that you're operating from, and that's what guide is supposed to do. So 11, for example, question 11, where can I find TMDD design content?

Well, you go to chapter 4, and you find, in this section, everything is about design content, each one of them is explained. And then you go back to TMDD standard and you go to that, and you look at everything the guide has spoken about. So that is how these questions are addressed, and I will recommend you to look at these questions, and perhaps comprehend discussion, you know, in both the guide and the standard. So let's now-- we're coming to our concluding discussion, and we look at it in terms of what have we learned? What have we learned in terms of learning objectives and let's make an attempt to tie or link them. So in learning objective one, we learned that a standard has a structure. TMDD Version 3-- TMDD Version 3 standard has a particular structure. It's in two volumes.

Volume I houses concept of operations and user needs descriptions, and they're organized in a very sequential manner. Also requirements follow user needs, concept of operations and user needs are discussed in section 2, requirements follow in section 3. And toward the end, is the NRTM matrix that ties these two together. So that's what we learned. So if you look at Volume I, you should, by now, with the help of this webinar and the guide, you should be able to go in there and look at it, all these different things that you need to work with, to match your operational needs, for example, and then pick the right requirements that the NRTM tells you to, all right? So that part is dealt with in Volume I. So that's Volume I relationship here.

In Volume II, the solutions appear. The design solutions appear, and necessary data concepts are laid out. Again, there are four types of data concepts, and they all are discussed individually, separately, and described in a very detailed way. So that part is now available to you in Volume II. RTM connects these data concepts to the requirements, so this-- toward the very end of this Volume II, you also have now a lengthy RTM to look at it, alright? And that's this part here. So Volume I and Volume II, are separated through an NRTM for the problem definition side, and the-- through RTM, through this design solution side. That's what we learned. Learning objective two, well we're still, I guess, in one, learning objective one, has also identified the capabilities. What is available out there in the standard, so that you can build your interoperability system interface needs. Alright, so a lot of people want to exchange information, for example. What kind of information? Well, here we have two events that are occurring. It could be a planned event, it could be an stadium event, or a parade or something, or it could be an unplanned event, a natural disaster or tragic incident, and then you want to be able to exchange information. So all that requirements or needs are presented in terms of capability. If you follow that particular user need, and then translate to requirements, you will end up with a data concept that actually delivers what you are looking for, all right? So that's the capability aspect that the TMDD has delivered big time. This is the only standard that has done something similar for traffic management, and in other domains are also similar approaches to defining these different elements.

All right, next example could be also about the data gathering, and ITS is planning heavy. Also we need a lot of data, real time data, so that capability is now provided by TMDD standard. Everything through NRTM. We have learned that NRTM actually links the user needs to requirements or requirements to user needs. So in other words, if there is a user need, then there is a requirement. If there is no user need, there is no requirement. One of your answer was also speaking very clearly about that, which is very good, all right? So now we have to acknowledge that we now know how to prepare a project level NRTM. Through these webinars, we have made sure that the past module, A321a, actually taught us how to prepare project level NRTM, and we discussed that a little bit completely as well. Using RTM again, we are now able to connect single design to a particular requirement. We couldn't do otherwise on our own, because then everyone would have different designs, and then you would be back to square one again, in terms of not achieving interoperability. So RTM has eliminated that issue of interoperability requirements. We are now able to achieve that portion through RTM. And we also know now, today, how to prepare project level RTM. In terms of what we have learned for objective three, both NRTM and RTM are required. Now you don't have this question anymore that, do I need one of the other, and the answer is that we need both, all right?

We have answered that question. Interoperability is dependent on a specification that uses same data concepts, requirements and user needs. We have repeated this message several times in our discussion today, that you cannot achieve interoperability unless you have user needs and data concepts and requirements all in the same level of details and same set, actually in every specification that needs to talk to each other, the system needs to talk to each other, all right? And interoperability is dependent on the common protocol. We have to end up-- we must have one protocol that's common to those who want to exchange information with each other. In terms of extending standards, there are certain rules. Again, there's a caution that we are not encouraging anyone to go out and do that one thing, or create different user needs, or requirements, and work away from the standard. The drawback is that you will end up with a system that's not interoperable, and you will probably also make the system complex than necessary, and the results may not be conducive, the results may be out of synch and so on. So all of us in the standards community, advise users whenever we have a chance, to not to go beyond the standard if you can. So that's what we learned.

In the learning objective six, we learned that there's a guide document available, and that document talks in terms of both what should be done for particular step, and also it references to the standard. So the guide actually is a linkage document between a user and the standard, so if you want to think it that way while preparing specification and implementing system interface. Where do we find more information? So these five resources are very carefully selected. The supplement has seven boxes in them, which will lead you to navigation process which will allow you to end up with the specification that meets your project level need, all right? So supplement has a lot of good information that you can review. The guide itself has detailed information, about 70 pages, and is pretty much in synch with the TMDD Version 3 standard. The NTCIP Guide, version 4, is also a complete document in terms of field level devices standards, DMS, ESS and so on and also 2306, and 2304, which are the center-to-center communication standard. So the NTCIP Guide does a very good job in providing you overall comprehensive coverage of what the standards are about, and how to use them in a way, very concrete way. Also NTCIP Guide, if you are technical in nature, and you're looking for a lot more detail about calculation,

bandwidth calculation, things like that, that's also available in the NTCIP Guide. Number 4 here, System Engineering Guide, we talked a lot today about the V diagram and system engineering process, as the basis for our work in this PCB program, and therefore we go to this guide.

There are two sources, the number four here, Caltrans developed System Engineering Guide has a lot of deployment level information in it. Also there are case studies in the back, so I encourage you to look at those tables and understand what they reflect. For example, if you have a new project, then it will lead you to those elements and say what you should do, all right? If you are actually adding devices or certain things to an existing system, what should be happening? So that kind of thing is detailed enough in that handbook, guidebook. Last one, System Engineering, is a basic knowledge book, on system engineering. It's a very authoritative work done and we all use this, and it's a complete coverage in terms of what we need to know to build ITS projects, by using system engineering, so that's a very good, reference.

The TMDD sequence is not complete until we have this testing, T321a, which will be developed for the next year's PCB program, so when you take A321a, which is the user needs discussion. Today we discussed requirements, A321b, and then the future, we will add T321 for testing plan and what TMDD can we work with. So we have more coverage in terms of compliance, conformance, and how to ensure that everything that you ask for is actually being delivered, you know, at the time of acceptance. So that is the testing modules that are coming next year. So at this stage, I'd like to open this for questions.

If you have any questions, please go ahead and type in the chat box. Any questions? Anyone? All right, okay, here's one. Agency A-- if agency A, has ASN, and agency B uses XML, how is this resolved? Very good question. If one agency uses ASN.1, and other one has XML on the other hand, the quick answer is that it's not available, they will not be able to exchange information. On the other hand, someone will say, well, what if we design a converter, you know, it's possible that system interfaces are actually data and messages type of entities, so chances are good that you can probably develop a converter. But again, it becomes a complex process, and you may not gain the interoperability that you're looking for. So that is work that you have to do, investigate, and it's probably not recommended to have two different setups and still try to get interoperability. Second question, okay, sounds great, okay. No question. I think you did a good job. Okay, very good.

Alright, so again, if there's no further questions, I have two I would like to just take you very quickly to that, is that-- okay, there's a question here, is it possible to just have either ASN or XML as a standard? Yes, absolutely. It's possible, yes, absolutely. This is a very good question, and this is something that we encourage everyone to think at length, that you make that decision when you are developing your project, in fact, when you think in terms of concept of operation, this should come up as well. Since we are all willing to talk to each other, and we want to talk to each other, and we want to achieve interoperability, which format should we use? And that should come up at that stage, and say, well, for good reasons it's XML, for good reasons, ASN.1, whatever those good reasons are for each one of them, all right? And another thing I will say in reference to this particular question, is that you may already have started or taken a step one. A lot of ITS projects are already in place, and they may have some exposure to XML, for example, or ASN.1, for example, all right? So in that case, your investigation will start from there, and

take you what you should do, where you want to end up with, so this is a very good question. And we encourage that you use that methodology to choose one from the beginning. It was also a question that I raised, which says, which matrix should I use for establishing relationship between requirements and design concepts? And we want to make sure that everyone understands that it's RTM. We have to go through the design process through RTM, so that is a very imperative statement, everyone will make to strengthen our discussion today. The second question is, what are the minimum conditions for interoperability? Again, one you just raised yourself that it has to be one format. That's a common protocol that we're talking about, and also the data concepts we'll be using, whether ASN.1 or XML. We cannot mix them, so we have to pull same condition that we shall use, the same user needs, data concepts and requirements as a subset, and we should also use the same protocol format for encoding these elements, and then, and the end, the common protocol that we need. And overall these things, user needs are the first step toward achieving interoperability. We keep saying that thing, because that's where we begin our process, and that's where the requirements are coming out of, all right?

So make sure that your user needs are similar for your step one, requirements are properly obtained from the NRTM, and then, finally the design concepts are taken from the RTM, and then this protocol that is common, whether it's XML based or ASN.1 based. So if there are no more questions, I would like to conclude this seminar today, and thank you all for attending.