

Module 15 - A313b - Specifying Requirements for ESS Systems Based on NTCIP 1204 v03 Standard Webinar Transcript

Shelly 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'm 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

Today's module is A313b, Specifying Requirements for ESS Systems based on NTCIP 1204 version 3 standard. The target audience for this course module includes engineering staff, traffic management center staff, operations staff, system developers and integrators, coders, private and public sectors, users of Clarus Initiative and public safety and other stakeholders.

Your instructor is Ken Vaughn who led the development of all three versions of the NTCIP 1204 standards and in 2010 founded Trevilon Corporation. The course developer was Joerg "Nu" Rosenbohm.

Ken Vaughn

So with that we'll go ahead and get started. We've assumed by, or if you're taking this course that you have some basic background in ITS standards. There are various modules that this module is a part of. It's a curriculum course. We assume you've all ready taking I101 Using ITS Standards: An Overview; A101 Introduction to Acquiring Standards-based ITS Systems; A102 Introduction to User Needs Identification; and A201 Details on Acquiring Standards-based ITS Systems; as well as A313a Understanding the User Needs for ESS Systems based on NCTIP 1204 version 3 Standard. All of those lead into this course and provide an understanding of how the standards are structured, how the user needs will lead into the requirements which is the focus of this particular course.

That's laid out on the standard curriculum path of SEP. Those five courses that are pre-requisites are shown there before hand and then this one is highlighted at the end. It is the last module that you'll need to go ahead and specify your equipment. It is the part of a group of ITS standard modules. And this group consists of both the four environmental sensor stations. This group consists of the A313a which focuses on the environmental sensor station user needs. This module focuses on the requirements. And there's another module that focuses on the test plans for an ESS standard or an ESS device. And those three together are the three very specific modules related to environmental sensor stations that you may be interested in if you're in a market for an ESS device.

The learning objectives for this device include discussing the structure of the standard, using the protocol requirements list, and requirements traceability matrix to specify the standardized structure of requirements. Using the RTM to specify the standardized design including the requirements from the PRL and RTM and the specification. Understand how to specify requirements not covered by the standard and to infer the relationship between selecting requirements and testing. So we'll discuss those. In the upper right corner of most slides you'll see which learning objective we'll be

So a little bit of an overview of the NTCIP family. We've seen this module before if you've taken those prerequisite modules. The NTCIP is a family of standards. And some of the standards are information profile standards, this is where we define the data in the specific exchange process for information. And then there's the underlying communication standards and those are called the protocol standards. And those protocol standards line up with a communication stack you see here. The NTCIP 1204 ESS Standard is an information level standard. That rides up that top section there with the red figures. And it's a specifically NTCIP data dictionary standard. And that rides on top of your application level standards such as SNMP, your transport level, your sub network level and your plant level all dealing with the type of communications network you have.

But this standard deals with the actual information content you want to exchange between a weather station and a central system. The NTCIP 1204 defines communications interface standard so it's a standard defining communications interface but strictly focused on the information itself. So it specifies the interface between the environmental sensor station and the monitoring systems. And notice that it can be a mini to mini relationship.

You can have multiple environmental sensor stations with multiple monitoring systems. How you manage that as part of your communications stack of profile selection but in theory, at least, you can have multiple monitoring systems, monitoring a single station or a single monitoring system managing multiple sensor stations. The host system includes a traffic management center, central software or, for example, a national weather service or some other type of weather system or perhaps a maintenance system if you're doing snow plow operations. The standard contains the object definitions or the vocabulary used to monitor an ESS. So it defines all of the low level details. And when we talk about temperature, for example, we know exactly what we mean. We're talking about air temperature and degrees Celsius or tenths degree Celsius and all of the details there.

Version three is an improvement over older versions of the standard, specifically the big change is we added test procedures to this version of the standard. The functional requirements section of the standard includes a tutorial, scope of the interface, protocol requirements list which we mentioned a little bit in the user needs module, architectural requirements, data exchange requirements and supplemental requirements.

Now, once again, just as a reminder, section one of the standard was just an introduction. Section two is where we define the user needs that led into this section, section three where we defined all of the user requirements and provide that mapping from user needs to new requirements in that protocol requirements list.

So what is a requirement? Well, according to INCOSE Systems Engineering Handbook it is a statement that identifies a system, product or process, characteristic or constraint which is unambiguous, clear, unique, consistent, standalone, not grouped, and verifiable and is deemed necessary for stakeholder acceptability. Generally speaking, they are presented in the form of shall statements and in module A103 has a detailed discussion of well-formed requirements. The center starts out by discussing architectural requirements. Architectural requirements are those requirements that are related to the communications capabilities of an ESS system.

So once again, we're still talking about information level exchanges but we're talking about how that information is now exchanged. So there's three key components here within the 1204 standard. One is being able to provide live data so when I select air quality, for example, or maybe air temperature I know I'm getting the current air temperature. That's what provide live data is. Provide compressed data is perhaps needed in environments where you have high communication costs. For example, if your device is connected via satellite because some of these weather stations are in remote areas, then you're paying for every byte you're sending across that link.

So there's a desire to compress the data more so than what is normally done in NTCIP. And we've provided a mechanism for that within the standard. So for that environment you want your data compressed so you can gather your air quality and pavement temperature and all of your other conditions simultaneously. And then the third option is to provide offline log data. And in this case, maybe you only dial into your device once every hour or once every day depending on conditions. And rather than constantly monitoring in an always on communications link you only have a periodic communications. And in this case, you still want to collect very detailed data so you're going to have a history of temperature profiles and other things. This allows you to time stamp locally every measurement, keep it in a log and then at the end of the hour or at the end of the day you can go query the device and upload that entire log.

So in all three cases there's configuration variables and other things that have yet to manage information-wise in the device, that's why it's a topic for NTCIP 1204 but it's all related to the architecture of your system. So it's architectural requirements. Then the second major piece is the data exchange requirements. And, of course, this is where we really get into the meat of the standard.

The data exchange requirements are divided into three main modules. It's defined in section 3.5 of the standard. And the ESS manager is the first of the three major portions of the data exchange requirements. Now, the ESS manager it is the logic overall managing the device, no actual sensor type information. It's just managing the device itself. So this includes things like configuration requirements for retrieving, configuring ESS characteristics in either normal or compressed mode. Monitoring requirements including retrieving ESS door status, retrieving battery status, retrieving the line voltage. And data retrieval requirements for moveable ESS equipment. So, for example, if you're monitoring a snowplow or something that has ESS sensors on it, you can also monitor the speed and location of that vehicle. So, once again, all of these are kind of high level information but in many cases you need to know that information, in particular, if you're tracking a vehicle through your network you need to know where the temperature is being recorded from that vehicle, in addition to the time and everything else. The next section of data exchange requirements deals with the sensor manager.

Now, this is really kind of what most people think of when they think of the ESS Standard. It is the sensor configuration requirements including retrieving and configuring atmospheric pressure, wind sensor, temperatures, pavement sensor data, sub surface sensors and even camera metadata. So you can take pictures of the scene and see what the roadway looks like. It also includes sensor data retrieval requirements, retrieval of weather profiles from mobile pavement treatment systems, weather information, pavement conditions, sub surface conditions air quality, bio hazard conditions, water levels, snapshot images and sensor control requirements including controlling the snapshot cameras. So configuration, data retrieval and control of all of your sensors, all of your-- even your cameras and other equipment. And then a third major component of your data exchange requirements deals with the pavement treatment system. Now, a pavement treatment system is any sort of system that disperses anti-icing or deicing chemicals on to a roadway. This may be part of a mobile vehicle that's distributing sand on the roadway, for example. Or it could be imbedded bridge sprayers that will spray deicing chemical once the temperature drops below a certain degree. So they're going to be stationary or mobile. But in either case, they're pavement treatment systems. And the reason that's included in the standard is because these sensors or this equipment is frequently deployed along side and managed by the same controllers of your weather sensors. So it's all contained in one standard. But this includes, once again, configuration status monitoring, and control requirements for this sort of equipment. And then one final set of requirements for data exchange are the backwards compatibility requirements.

And you may remember when we talked about the user needs that the standard is backwards compatible. However, when we deployed the first version of the standard, and the second version of the standard, deployment experience made us realize that there's some ambiguities in those initial versions of the standard. Rather than changing those implementations to a common interpretation we added a new way to do it which means any old equipment you have is still fully conformant with the previous version of the standard. And nothing has changed with that version of the standard.

But we recommend new deployments migrate towards the newer definitions which is a different piece of data so that there's no confusion that when you migrate, you know which piece of data you're dealing with. So your central system during this migration period may have to support both the older versions and the new versions. And likewise even your field devices you may want to support the newer versions going forward but right now today you're dealing with the management system that only understands the older information. So in both cases, we give you the option of saying go ahead and support the older version in addition to the new way of doing things. And if you're dealing with this migration issue that's something you'll have to consider. And also understand for each one of these features the reason we made changes was because there were ambiguities in the original definition. That means when you specify, support the older version, you also need to specify exactly what you mean by the older version, which interpretation of the standard do you want your devices to support so that it interoperates properly with your equipment.

With that, that brings us to our first poll. And the first poll is assume the following user need, the transportation system operator may need to monitor the temperature at the ESS location, which of the listed requirements fulfills this user

need? Now, we'll go ahead and launch this thing here. And launch that. So is it retrieve temperature, retrieve humidity and retrieve daily minimum and maximum temperatures? And you can select multiple if you want to. So go ahead and make your selections, which requirements would fulfill that particular user need. We'll give you a few seconds here to fill it out. I see a good portion of you have voted, and we'll go ahead and close that out.

Most all of you said retrieve daily minimum and maximum temperatures, which is correct, retrieve temperature is also correct. Most of you selected that. Retrieve humidity is not necessarily one of the-- I'll share those results with you real quick. So those are the results. And the actual answer is retrieve temperature and retrieve daily minimum and maximum temperatures. Those are the two requirements that relate to temperatures. So as you see it's pretty easy to select. Once you know what user need is it's easy to figure out which requirements are going to be needed.

So that brings us to the protocol requirements list. The protocols requirements list or PRL is a table that maps the user needs to the requirements. It must be a part of the agency specification. This is how you tie your particular specification into the requirements so there's no ambiguity as to what your project is requiring out of the standard. It specifies the standard, i.e., if gives specific clause references to the standard and you say either yes, I want to support this feature, or no I do not want to support that feature or requirement. And by selecting those, you're fully defining a communications interface for each one of those requirements. And it's designed to help you specify what you want the interface to do. So it takes you through that user need concept that most users find fairly easy to understand. Then it takes you down to the requirements which, as you saw, it's relatively easy to figure out which ones you need. And then the standard itself will define one precise solution for each of those requirements you've selected.

Now, what does this look like? Well, it's in your supplement and if you look at your student supplement on page seven you see the protocol requirements list from the NTCIP 1204 version 3 standard. The first few columns start out with the user need ID and user need text or title. These two columns we discussed during A313a module. The next two columns include the functional requirements ID and the title of the functional requirement. So those four columns are shown there highlighted in green. And first off as we talked about an A313a you only select the user needs that you need for your particular project. And then once you've selected those user needs you can then look underneath those user needs to define or to identify your requirements that relate to that particular user need. All of the other user needs, all of the other requirements, you don't need to worry about. You just select your user needs that you're interested in for your project. And then look underneath that user need to see which requirements may be of interest to you for your project. Based on the user need selected the PRL identifies requirements associated with those user needs.

So that brings us to our first activity, based on the knowledge of the PRL please describe the purpose and use of the PRL. What needs to be selected first from the NTCIP 1204 to support the desired functionality of the system to be created? And you can go ahead and type your answers into the chat pod. So go ahead and type those answers in. I'll give you a few seconds here to do that. So what do you need to select first in order to support the desired functionality?

And we have some answers coming in and they are correct. User needs. User needs that are needed for the proposed implementation. Now, the second question please describe the purpose and use of the PRL. When you have identified the first item which we determined was user needs what does the PRL provide a specification developer.

So once you specify the user needs what does the PRL, how does it assist you? So I'll give you a few more seconds here. All right. And once again, we're getting answers back and the answer as suggested is, in fact, the requirements. Based on the user need selected the PRL identifies the requirements associated with those user needs. Now, finally, when you have completed the PRL, what do you do with it? So it's been a great exercise to go through and select user needs in your requirements. But what do you do with the completed PRL? So go ahead, once again, and type that answer into the chat pod.

And moving on, right so we have-- it defines the requirements and specifically that those requirements will become a part of the project specification by including the full PRL in your project specifications, the PRL references, the selected requirements by specific clause numbers and then that provides the full traceability from your procurement into the standard and as we'll see here in a second, those clause references will then trace to specific design characteristics to make sure you get an interoperable system.

Now, that brings us to the next column. We've discussed the first four columns. The next column is the conformance column which we did discuss a little bit in A313a. The conformance identifies if a user need or requirement is mandatory or optional. Now, sometimes you'll have a predicate which is a conformance that may depend on a condition or whether a feature is supported. So, for example, in this column right here we see several items that are mandatory and a few items that are optional. In some cases, near the bottom there you see O.6. That means those two items there are part of the sixth-option group. So you need to consider those two requirements jointly together. And the parenthetical statement next to it indicates how you can make that consideration. The one-dot-many indication says from this option group, from those two requirements of the sixth-option group I must select at least one requirement, but I can select multiple requirements. So in other words, if you support the monitor power user need which is that parent user need then I either have to support, retrieve battery status or retrieve line volts. But I could actually support both of them and I'm still conformant.

The next item is the support project requirement. Support project requirement is the procurer selects the user needs and associated requirements desired for his or her implementation. So in each case if it's mandatory it must be a project requirement. So you only have one option there, yes. If a feature is optional you have a choice, it's either yes or no. Now, if it's mandatory underneath an optional user need to then you have a conditional case. If the user need is optional and I select no then every requirement underneath that is not applicable. So I select the NA option. And in some cases such as under monitor power you have requirements underneath an optional user need so NA is an option. And then the features themselves or the requirements themselves are optional underneath that optional user need. So they also have the ability to select either a yes or a no in addition to the NA.

But this gives you the layout that when you're going down selecting these things it's really just a checklist, which options do I want to support my standard? If it's mandatory, I simply select yes. If it's optional yes or no. If it's underneath an optional item then it's not applicable if I've selected no for that optional parent item. The final column is the additional project requirements column. And in some cases, some requirements have additional details. Now, the field is always there. You can type in your own details if you feel a need to.

Generally speaking, the only cases where you'll need to fill this out is where we have provided sample text for you with a blank. And this is commonly done for things like determining the number of sensors or something like this. So the ESS shall support at least x-number of atmospheric pressure sensors. Some devices will support multiples. Your device may only need to support one or two. There's no reason to require a system to support 50 sensors, for example, unless you have an actual need for that for your project. And yet, our standard had to standardize on something. So by default whenever you see a sensor being required by default we say well you have to support at least one as you're saying you support that sensor. But we fully recognize that a lot of projects for some sensors may need to support multiple sensors simultaneously. And in that case it's very important that you fill out these additional project requirements to identify clearly how many sensors you want to support for your particular deployment, otherwise, you're likely to get only one.

So as a review, requiring support for monitoring air quality and/or biohazards would only increase costs and limit vendors who do not offer these capabilities. Air quality and bio hazards are a part of the ESS Standard. And there are very real user needs for that equipment. However, most deployments around the country don't include these because their weather stations are more focused on monitoring temperature and pavement conditions and things like this. So as a result, we designed the standard to be very flexible. If you want to use the standard to support air quality monitoring very likely you don't need any of the weather data. So you can just select air quality. But if you wanted to support all of the weather data you don't have to support air quality. Both types of devices can claim conformance to the standard. But it's very important when you go out and specify the device you want on your project you have to make sure you tell the vendor which options you want so that you make sure you get the equipment that you need for your project. Because clearly if we required everything to be part of every device it would have simply just increased the cost of every device and given people things that they really didn't need. So the user need, the example of PRL usage, the user need monitor atmospheric pressure is an optional user need for the standard. But if it is selected all of the requirements associated with this user need become mandatory. 's the one there at the bottom. And we've selected that user need with a yes. And the two requirements underneath that user need are shown as mandatory. Therefore, we've selected both of those items.

So that concludes the topic of the PRL, protocol requirements list and now we're going to move on to the requirements of traceability matrix. But I will take a second here to stop and answer one question that came in which was does the NTCIP provide a fillable PDF for the PRL or an Excel spreadsheet?

The quick answer is yes it does. Anyone can go online through the agreement with the federal highway and download these standards for free. When you download the standards you get a PDF file. So you can then take that PDF file and use like an Acrobat Pro version to edit that file to highlight and circle the yes and no like we did here. So it is available. You just go to the NTCIP website. So now we'll move on to the requirements traceability matrix. As we discussed before, the PRL focused on tracing user needs to the requirements. The requirements traceability matrix takes that to the next step it traces from each requirement on to the design details. So it describes the standardized design for fulfilling a requirement supported by the standard. Requirements can be traced in a standardized way and this reduces the amount of design work. So the design is all ready done for you by the standards effort. And what we've done is by adding the requirements since the standard, now you just select the standard and that immediately directs the implementers precisely to the design elements that they need to implement. The design consisted of a dialog which is a sequence of data exchanges and objects to be exchanged. The intended audience or beneficiaries of this are primarily device vendors who need to look at this to understand what messages they'll be receiving and how they're supposed to respond. Essential system developers so that they know which messages they should be sending, in what sequence, testers so if they can develop their test procedures and they can make sure that the device responds properly and that the central system queries properly, and finally the agencies because now they don't have to get into the details of the design aspect so much. They can focus on what they really need and the requirements of those needs.

The definition of RTM is to conform to requirement in the standard, an ESS System shall implement all objects and the dialogs traced from that requirement. Now that is a firm requirement of the standard, so if you want to claim conformance for your device or for your central system you have to implement the objects and the dialogs. Your requirements are traced to dialogs and then to objects. This order is key to interoperability needs. To achieve interoperability all systems shall satisfy a specified requirement the same way. So there's only one design that's traced to every requirement. And if you want to claim conformance to the standard, you have to implement that one design or if it conforms to a requirement you have to implement that one design and we can show we know that will provide you with interoperability.

So what does an RTM look like? As you can see an RTM is somewhat similar to the PRL table, but in this case the far left column is Requirement I.D. rather than the User Need I.D. So everything is driven by the requirement now. So Requirement I.D. The second column is the Dialog, so which dialog does this requirement use? The third column is the Requirement title which is the same exact title we had before, and then you have the Object I.D. and the name of that object in the final column. There's also periodically some additional requirements that may appear in that final column. So now you'll notice the shading as well, the darker-shade items are kind of high-level groupings of requirements and it's all in the lower level individual requirement that has any detail underneath it as far as objects and such or traced to a dialog.

So the dialogs and the objects are specific to a specific requirement. The Object I.D. show the section number of the objects that will fulfill this requirement. So just like with the requirement ID you can go to a specific clause number in the standard and you immediately see the full definition of the object being referenced now. So for an example requirement 3.5.1.1.1, Retrieve ESS Characteristics is fulfilled by using standardized dialog which is F.3.1, once again specific clause reference in the standard. In this case it's an Annex-F which is a generic SNMP Get Interface and it relates to the objects `essNtcipCategory`, `essNtcipSiteDescription`, `essTypeofStation`, `essLatitude`, `essLongitude`, and `essReferenceHeight`. So all six of those objects have to be supported by the standard and be retrievable via the SNMP Get Interface and a single request. That's what the generic SNMP Get Interface requires based on those set of objects. The SNMP Get Interface is shown here. It defines a generic process by which a management station can retrieve data from a device. This gets process consists of a Get request and a Get response. Both the Get request and Get response messages contain a list of objects as defined in the `varBindingList` structure. RTM customizes this generic process by calling out the appropriate objects to meet specific requirements.

So that's the full definition. On the previous slide you have a list of objects we've defined and that list of objects must be included in the Get request and then it comes back with one single response containing that list of objects. So we

fully defined how that is supposed to operate and that means we can set that up to test the device coming in whether it's the central system of the field device. The order of the data exchange is important, unless the dialogs state otherwise. So in other words I shouldn't be getting my response before my semi-request. And a lot of dialogs are more complex than that and therefore that's where this issue really comes into play that you have to do things in a certain order. Interoperability may be compromised if the sequence of data exchanges is modified. Now there are other rules and procedures within the standards that are not nullified by these dialog requirements. So it may be that other sequences are possible. However, even in those cases those sequences very likely were never tested out in a test procedure. So for your highest probability of your system working, your central system should use these standardized dialogs. Now in cases they may want to do different things in order to lower the amount of overhead and other things, but recognize there should always be a fallback operation to the standardized dialogs because that's what we know is going to work because that's what all the test procedures are going to be based on.

So that brings us to the next poll. Which of the following elements are not part of the RTM? And we have a few options here. I'll go ahead and launch this poll. So which of the following elements are not part of the RTM? User needs supported by the standard. Requirements supported by the standard. Standardized dialogs to fulfill requirements. Or Objects to fulfill requirements. So which one of those are not part of the RTM? I'll give you a little bit of time to fill that out. Which one of those are not required? I'll go ahead and close that poll.

Most of you were correct: user needs supported by the standard are not an RTM. Those are the contained in the PRL or Protocol Requirements List. The requirements traceability matrix on the other hand does indicate the requirements and it traces them to the dialogs and the individual requirements - the objects. It does not include the user needs. So we talked a little bit about the completed PRL being included within your project specifications. Specifically the PRL should form a part of your functional requirements within the communication interface specifications. That's just one small piece of your overall system. Communication Interface

Specifications are likely to also include performance requirements and protocol requirements, your product specifications will also include software specifications and hardware specifications which make up the product specifications which only once again is one small piece of your overall contract documents. So this is just one small piece of a large project and you need to make sure that the information contained in this piece is consistent with what you've defined elsewhere. So for example if you've gone through and you've said that you want to support temperature sensors within this PRL you need to make sure that your hardware specifications also describe what those temperature sensors are, what their operating ranges are and things like that. The product specifications include hardware specifications, software specifications, communications interface specification and the contract requirements define activities during system development, testing, integration, operations and maintenance.

So the vast majority of your contract materials is separate from this PRL but your PRL still serves as your key component into the specification so that you know what you're going to get. So the PRL defines the requirements for the communications interface. But for this standard we're only dealing with the information level. It's the application data. The underlying communications standards still need to be specified. So you still need to specify that you're going to be operating over twist pair network or that it's going to be a wireless network or it's an Ethernet network, fiber, whatever. And select the protocols that are appropriate for that communications environment.

Some of the considerations to reference interface standards, be specific to the version and date of issue of the standard. Include the PRL from that standard. And include value ranges for all objects to clarify the parameters or specifically in that additional project requirements column make sure you identify the number of sensors you need and things like this. You also need to be concerned about performance requirements. Make sure that gets into your project requirements. Performance requirements for the system are not covered by the standard except for response times because that was a critical component to interoperability of equipment. But for the most part the number of devices on a channel, the time lag when employing device and pulling rate, etcetera, are all part of your response times that have to be considered to make sure that you - that your design allows for your system to work in responsible ways. So response times are a trust. But all of your other details are not. But even in the case of response times, it's only measured from the device receiving a request to the time it takes for the device to respond to request.

So the standard does not define your full communications analysis. It does not consider how many devices you have on

your channel, what time lag is when pulling the device or the pulling rate. In all of these you're going to affect how many devices you place in your channel in your bandwidth analysis. That's a key important factor when designing your communications systems to make sure that your communications facility has enough bandwidth to support your exchange needs. So the requirements for the communications interface must be consistent with hardware specification. You should include a statement that requires implementation for all standardized design solutions, as specified in RTM where there are requirements selected via the PRL. Include a completed copy of the PRL plus a reference to RTM as a source for the design of the system and the test plan. Now there are some requirements that the standard does not cover. The standard was designed specifically to allow for growth in the industry new inventions. We're dealing with a high-tech world, very fast moving technology world and we don't know what the future holds for a lot of these devices. The NTCIP standards as a whole allow for growth and innovation so any agency can specify new requirements, new features that are not part of the standard and likewise, manufacturers can develop new features that are not per the standard and include them in the products they provide to you.

But recognize that any use of those features you're now customizing your system beyond the standard and it may be difficult to acquire that same interoperability with another manufacturer, so it's there for you to innovate with but recognize as soon as you start innovating you kind of start going back into that non-standardized world where you can't just buy products off the shelf. So user needs and requirements not supported by the standard may result in user specific requirements. And procuring agencies should specify the dialogs and the specific objects fulfill those specific requirements. The reason you want to specify in complete design details rather than leaving it to the manufacturer is because by you defining the details, you now have ownership of that intellectual property and you can provide that then to the standards community and say this is what I'm doing on my system. Perhaps something like this should be standardized for a future version of the standard. Otherwise what may happen is the manufacturer may come up with their own design and then not release the design details to you, at which point you're kind of locked into that one manufacturer if you want to keep operability with that feature. I actually got one question in just now. "If an extension is used, is the device still compliant?" Well, yes unless you've explicitly prohibited that feature, it would be compliant.

Let's quickly review the terms though, compliant means you meet all of the requirements related to a project specification. Conformance it is I meet all of the mandatory requirements of a standard. So if you go beyond the standard and you provide some feature that is over and above what the standard requires, then you can still claim conformance because the standard doesn't prohibit it, and as long as your project specifications don't prohibit it, you'd also claim compliance. As long as this new feature doesn't interfere with any of the standard definition of how I operate my device. I can switch modes but ultimately there has to be at least one mode of my equipment where I operate fully conformant with the standard. So it would be compliant unless you prohibit it. Activity about benefits and drawbacks. What do you think the benefits and drawbacks associated with extensions are for the NTCIP standards? So go ahead and type your answers in the chat pod.

What are the benefits and drawbacks of using extensions or even allowing extensions? Right so one of the benefits is it allows procurers to use the NTCIP family of standards and still support operational user needs not supported by the family. So I can do things like I don't know, I can implement my own infrared sensor for example which may not be supported by the device. Actually I think that one is. I implement a bell or whistle to warn people about ice on the roadway and that's certainly not in the standard yet, but you could allow it. You could do that if you wanted to. You have a little light that goes on when the road ices. Whatever you want to do, you can do beyond the standards that allows that growth as long as you implement that feature in a way that does not interfere with the base standard. So every device, even central system will still be able to communicate with that device for all of the standardized features. But if you are able to control that additional feature, you have that added functionality. So let us pretend that agency A wants to add an over height sensor to an ESS installed in a tunnel. Such a sensor is not currently covered by any NTCIP standard. This requirement is therefore not supported by NTCIP 1204v03. Agency A could define a new object and could start to monitor the over height sensor using an existing dialog. So that's a real world benefit. There may be a real world need out there. Simple little sensor rather than creating a brand new device separately I can just tack this on to an existing device as long as I convince my manufacturer to build it, I have an easy way to monitor a simple little device out in the field with little cost so that's a great benefit.

But there are some drawbacks. Interoperability may be compromised. Other management stations that do not support the new objects will be unable to exercise the new capabilities. So I have this feature on my device now. I upgrade my

central system and I forgot to tell my central system provider that there's this feature, then all of a sudden I lose access to that feature. So it's very important that you document what you get so that you know when your central system gets upgraded you know what to tell them to include. If the agency is not consistent on how the requirement is fulfilled for all devices interoperability cannot be achieved without custom integration for each deployment. So if I go to manufacturer A and I tell them to implement one thing and they come up with their design, I go to manufacturer B and they implement it using their design well now my central system has to support both designs.

Now this perhaps isn't quite as bad as it sounds. Most likely they will both use the base level protocols of NTCIP to support this feature which means a lot of the work will be common but the details will be different and of course the devil is always in the details of making sure these IT systems work properly and ultimately you would have to have different software or different device drivers for each one of those pieces of equipment. So it's much better for the agency to figure out what that design is in full and then say "This is the design I want to have implemented, because that's what my central system supports." Other agencies with the same requirement may not have the same design though. So that's why that agency when they design that feature should probably also submit it to the NTCIP committee so that perhaps that design can be considered for a future version of the standard and then that way the whole industry standardizes around one particular design and you have the whole benefit for the marketplace. But that's essentially how the design matures over time and stays relevant with the changing technology.

The other drawback is that if you're going beyond the standard then there aren't any standardized test procedures to test that new feature so you'll need to develop test plans and test procedures to support that new requirement and if the feature is relatively untested that means that you're much more likely to run into bugs and everything else. So if you're buying something that's off the shelf, most of these equipments, you just go through, make sure it passes and then after most vendors have passed that test several times, it's pretty straight forward process. But when you're adding new features, you're much more likely to start finding bugs not only on the new features but potentially even in some of the other code that there could have been overlapping changes to. So a little bit about user needs and requirements in testing. User needs really relate to validation making sure that you're building the right system, and requirements relate to verification which means making sure you built the system you were told to build.

Testing must be considered early on during a project development process. The V-diagram shows the relationship between the user needs and the validation requirement to verification. So at the low levels you have- you work down the left side of the V-diagram, concept of operations requirements, high level design, details- and across the way of concept of operations you see validation. You have system validation. And during the process once you have completed your concept of operations you should immediately start considering how am I going to validate what I've defined in my concept of operations? Likewise, when you've completed your requirements you need to start considering how am I going to verify each one of these requirements? Because part of the task of defining a requirement is making sure that your requirement is testable and if it's not testable or you're not going to test it why are you bothered to finding it as your requirement? Because if it's never tested you're not really checking to make sure you're getting what you've required anyway. You're just adding a lot of text that confuses everyone. So you need to have a verification plan that can be developed as soon as you finish requirements well before you actually start your verification.

Same thing with validation. You start developing your validation plan after your concept of operations but well before you actually perform your system validation. So there is a relationship between selecting requirements and testing. Within the version three standard there is standardized test procedures now. These test procedures verify the requirements. And that's defined in Annex C and it's a separate within that annex there's a separate table requirements of the test case traceability table. That traces every requirement to the test cases that relate they can check or verify that particular requirement. In some cases there's only one test procedure. In other cases there may be multiple test procedures for a given requirement. An example of the test procedure and we'll discuss this more in a future module but an example test procedure would look like this where you have a title of the procedure. You have a specific number for the test case, and you have a description.

This test case verifies the ESS accurately reports its type, category and location. And allows a management station to edit the site description. In some cases you have to test the variables. In this case you would say well in this case I'm going to test sensor three because I know I actually have five sensors but I only need to test one for a particular test

procedure so there are variables in some cases. This particular example does not include any variables.

And then finally you have pass-fail criteria which generally speaking for every test case is the same thing. The device under test shall pass every verification step included within the test case to pass the test case, otherwise it fails. Then you have your step- individual steps of the procedure and in this case the way it's listed out is I'm going to get the following objects. Get those six objects as part of one request and I either pass or fail that step.

Then the next step is I start verifying your responses, is the response value appropriate? Now exactly what is appropriate for your given requirement and how is it deployed in the field? That's up for the tester to determine. If it's reporting one degree Celsius and it's actually freezing you have to decide whether or not that's within a reasonable range for your device that is outside of the NTCIP that's part of your hardware standards as far as accuracy of your equipment. However if notice that the temperature as it goes up and down the scale is following Fahrenheit rather than Celsius that is an issue of NTCIP because NTCIP says it has to be in Celsius, actually in tenths of degrees Celsius so sometimes it takes a little while for you to figure out is this a problem of the standard, the NTCIP standard or is it an accuracy problem of my device? So and then at the bottom you see that this is a form that can be used by a real world practitioner. You can indicate who the test was conducted by, the date it was tested on, whether or not the whole test passed or failed and you also record some notes. Let's see so a testing of project specifications, procurement specifications should include language to use the standardized test procedures to verify conformance through requirements selected in the PRL. So your project specifications should have a clause in there that makes sure it references and requires the vendor to have some sort of proof that they have tested it probably by a third party or will or assurance that it will be tested upon receipt that will follow these procedures. The test plans and procedures for an ESS is the topic of the next module which is T313.

So in summary NTCIP 1204 v03 defines requirements, dialogs and objects. Defines test procedures. It relates the user needs that we discussed in the last module through the requirements and dialogs in this module. And also the requirements to the test procedures. It also contains user definable predefined protocol requirements list which is the PRL. It also contains requirements to traceability matrix that automatically relates user selected requirements so the dialogs and object definitions so you have full traceability, user needs to requirements and requirements to the dialogs and objects. And finally it includes a requirements test case traceability table that maps requirements to your test cases and procedures.

So what did we learn today? Well a protocol requirements list is used to link user needs to the functional requirements. Tracing of selected functional requirements to dialogs and objects is standardized within the requirements traceability matrix table. Interoperability is achieved via the use of PRL an RTM as well as standardized test procedures. Extensions may be used to specify non-standardized functions that are sources for interoperability problems. Finally a completed PRL and a reference to the RTM should be integrated into the project specification's as part of the plan specification and estimate package. Now there are a variety of other resources you may be interested in. The standard itself, NTCIP 4 V3.08R2 is available online at www.NTCIP.org. The NTCIP guide is also available on that website and that's NTCIP 9001. You may also be interested in IEEE830 which is a recommended practice for software requirement specification. And finally the federal highway has a systems engineering website at the address shown there fhwadotgov/sadiv/segb. And the ITS standards website at www.standards.its.dot.gov. That takes us to the next steps. That would be module T313 applying your test plan to the NTCIP 1204 v03 ESS standard. That shows how to apply test plans to verify an ESS system meets the agency specifications and requirements supported by the standard. It also shows the relationship between testing for compliance to the standard and how it fits to the overall testing program. As a part of that module you may also want to take some of the other initial testing modules as a prerequisite.

With that I'll open the floor to questions and I have received a couple of questions. I'll look back here. So we talked about the one about the fillable PDF that is available off of the NCTIP.org website. Just go to the documents links, download the standards, and you can take the standards. They'll be downloaded in PDF form and then you can take that into Acrobat Pro or any other PDF editor and highlight the features you want and X-out the features you don't need.

Second question was "If an extension is used is the device still compliant?"

And the answer is yes as long as it doesn't actually break any actual requirements from the standard. Another item there was someone suggesting a benefit of extensions a system based on agency-specific needs, drawbacks, excessive costs, may not be interoperable.

Let's see. "In step two what is the intent of verifying? What are the steps to do this? Is something observed? Is there a requirement for a diagnostic mode of protocol analyzer?"

The intent of verify is that you actually test the requirement. How you verify it can be done by inspection. It can be done by demonstration. It can be done by a variety of means. You can go in and inspect the code. You can demonstrate the system. You can actively test the system through online device such as an active tester that sends the device sample messages and you check to make sure the responses are appropriate. You can also use as you selected a protocol analyzer to just monitor live communications to make sure it's accurate so there are a variety of ways to verify user requirements. In general most of the NTCIP requirements are verified, well verified in a device you can use an active tester. You can use a bit of test software that kind of mimics a management station and it will send test messages to the device and make sure that the device responds appropriately and it looks at all the communications, makes sure there aren't any errors in the data packets, make sure the responses are appropriate and in some cases prompts the tester or the user of that software package to make sure that the responses are appropriate in what the user is expecting. But you can also use a protocol analyzer and that's typically more often done for a central system. You can also create mimic devices in the field or simulated devices that you can hook up to a management system and then perhaps even create errors in those field devices and those simulated field devices to check whether or not your central system will work properly. So there's a whole host of ways you can go about that and that's really a part of your test plan of how much time do you want to spend testing these various options.

And let's see, what other questions do we have here? "Does FHWA maintain a list of manufacturers that meet the mandatory requirements?"

The answer to that is no. What most agencies do is they include and what we recommend is that your procurement specifications should include a requirement that requires testing. Now most agencies I've seen do this by requiring the manufacturer to demonstrate that they've been tested by an independent third party and they get a signed letter stating that they've gone through a test procedure. In the case of ESS that's fairly simple because you actually require them to have that third party in a test lab use the standardized test procedures and then you know what they're being tested against and you have confidence in that test procedure. In other cases there are widely distributed test procedures such as for CCTV or for message signs. In some cases you may have to require the vendor provide the test procedures for your review prior to acceptance, but in all cases there is no certification lab or single certification authority. It's all done by an independent third market.

So that's all the questions I have received so with that that concludes A313b Specifying Requirements for ESS Systems.