

Module 11 - A313a - Understanding User Needs for ESS Systems Based on NTCIP 1204 v03 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'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 A313a: Understanding User Needs for ESS Systems Based on NTCIP 1204 Version 3 Standard. Target audience for this module includes engineering staff, TMC operation staff, systems developers, private and public sectors, users of Clarus initiative, and public safety and other stakeholders.

Your instructor is Ken Vaughn. He's president of Trevilon Corporation. He also develops software using the NTCIP standards. The course developer for this course was also Joerg "Nu" Rosenbohm.

Ken Vaughn

This is Ken Vaughn and thank you for attending today. As Nicola mentioned, my background is with the NTCIP. For quite a while, back in 1995, I got involved and have developed various topics. So feel free to ask any questions during this webinar through the chat module and I'll be happy to answer them.

Just a few recommended prerequisites for this course. We assume that you've already taken or have familiarity with some of the various topics shown here. The I101 training module, Using ITS Standards: An Overview; A101 Introduction to Acquiring Standards-Based ITS Systems; A102, which is An Introduction to User Needs Identification; and A201: Details on Acquiring Systems-Based ITS Systems. Those modules fit into a kind of a standard curriculum path as shown on this slide, just one after another. And this module fits in immediately after the A201 module.

The ESS standard does contain systems engineering information. And a lot of this process of understanding the user needs will be relying on the actual content of that standard. So these user needs are already defined and we will be going over them in this particular training module. The value of this particular curriculum path is that the ESS is just a part of various modules. This particular module focuses on understanding user needs. There's one additional module that focuses on specifying the requirements for the systems for ESS systems. So you first define which or identify which user needs you might need and then you look into the requirements that you will have.

There's also in a separate training module curriculum path. There is a special ESS module for testing ESS standards. So if you're involved with testing, that would be another module that you would be interested in taking. The specific

learning objectives for this module, include understanding the structure of the standard, identifying specific ESS operational needs using the PRL or protocol requirements lists to select the user needs and link to requirements. And finally, explaining how the PRL fits into an ESS specification. And we'll go through several slides for each of those learning objectives.

The overall outline as we'll talk about the ESS standard itself and then we'll talk about the improvements made to this particular version of the standard from previous versions. Then we'll discuss some of the user needs addressed by the current standard. And discuss the protocol requirements list particularly in relation to user needs for the purpose of why the PRL exists and how you start filling out the PRL. As an overview, the NTCIP family of protocols, as you may be familiar with, includes a variety of different standards that fit together in a modular fashion. At the lowest levels, shown at the bottom of this figure, you have your plant level standards. Those are IT-based standards that define your actual physical infrastructure. Within the sub network level in the yellow, we have specific standards that have been—that IT-based standards that the NTCIP community has adopted and adapted for our particular environment. So those are NTCIP standards but they're all based on general IT standards. And the same is largely true at the transport level where you see UDP/IP, TCP/IP.

T2NULL is something that has been somewhat customized for the transportation environment for very low bandwidth type of environments. So there's a mixture of general IT standards there and customized standards. And that continuation continues since the application level where there are some specific ITS standards such as STMP and a lot of just general IT standards. You finally move up the information level, which is where NTCIP 1204 exists. It defines the actual data that is used to communicate information across that general IT infrastructure. And most of the ITS specific work occurs up in this information level where we're defining the information for ITS. And the ESS standard is just one of the standards with a specific focus on environmental sensor stations, including weather stations, environmental sensors, and pavement treatment systems.

So what is NTCIP 1204? Well, it defines a communications interface standard. So it defines how different components communicate with one another. And it specifically specifies the interface between the environmental sensor stations and its relevant monitoring systems. So you may have a weather station out in the field or some other environmental sensor station out in the field that communicates with a single monitoring system back at a central office. Or it may be simultaneously communicating to multiple different monitoring stations. For example, one might be associated with the weather service and another one might be associated with your traffic management system. So a single field site might communicate to multiple monitoring systems. So this standard applies to all of that. It's between the field and that central monitoring system. It specifically contains the object definitions or the vocabulary used to monitor an ESS. So, for example, the real intent is to find exactly when we say temperature, exactly what does that temperature mean? Over what period has it been averaged? How is it communicated? Is it in Celsius or Fahrenheit, etc? All of the details about exactly how we communicate data so that both sides have a full understanding of what is being transmitted.

A little bit about the history of NTCIP 1204. The development effort started, I believe, in 1996 and was finally published in 1998 as 1204 version 1. That led to several deployments out in the field. Those field deployments, as is common with any brand new standard, revealed some issues with the initial draft concept. And amendment number 1 was approved in 2001 and those revisions were due to those actual real world implementations so that we had different manufacturers that once actually started deploying the standard that identified some minor ambiguities and those were corrected. Now, despite the fact it was a first version standard, their revisions here were fairly minor. There were some issues along the way but the bulk of the standard was proven to be fairly stable. In 2007, we updated that standard to 1204 version 2 and this was the main systems engineering process update. So we added the concept of operations, the functional requirements, dialogues, some slight updates to object definitions, once again. We added some minor new features and other things, some additional real world experience, and we also included the protocol requirements list or PRL and the requirements traceability matrix.

As we'll discuss in the future, in a few slides from now, those two tables are the things that connect the concept of operations and the functional requirements in the case of the PRL. And the RTM connects the requirements, the dialogues, and object definitions. So you now have one single structure within your standard with clear traceability from initial concept to the full design detail. NTCIP version three took this one step further and we added test procedures. And we also updated some of the systems engineering process, once again, based on real world feedback,

some slight refinements, the context we had there. We also added the test procedures and once, again, a traceability table that connected the requirements to the test procedures that related to those requirements.

The general structure of that standard follows that same systems engineering process. So it starts out by defining user needs supported by the standard. An example user need is monitor the temperature sensor of an ESS station, a very kind of general idea. That then relates to functional requirements. And those supported by the standard include things like retrieving the temperature. So now, rather than just monitoring the temperature, which may be through a push service or a pull service, now we're implying that this is, in fact, going to be a pull service. We're going to retrieve the temperature manually. And that's trace the need. Each requirements ends up being traced to the particular user need it applies to. Finally, we design a single design for each requirement. This is very important so that once you specify the requirement, if you're procuring it, your work is now done. There is only one defined standardized solution to that requirement as defined in the design details. And that way we support interoperability. You define what your requirements are and everyone implements it the same way. So a little bit more of the structure of the standard, the general lay out, the table of contents.

Section one is just general overview comment, pretty much all of the NTCIP standards. Section two goes on to the concept of operations. And each user need there is defined in text, in very readable text that everyone should be able to understand. Section three then includes and defines the functional requirements. The PRL is also included in section three, which provides the traceability from each individual user need defined in section two to the specific requirement defined in section three. Section four defines the dialogues. This is how a monitoring system sends a message to your field device. And that field device responds back to the monitoring system. Then perhaps some additional exchange, some sequence of events of messages going back and forth, those are clearly defined in section four. And then section five gets into the detailed data definitions for each piece of data that's included in each one of those exchanges.

Annex A includes a traceability table that connects the section three functional requirements to the individual design elements of section four and section five. Annex B includes the positioning of each piece of data that we've defined within a tree structure so you can have kind of an easy reference of where things exist. Annex C was added in version three and those are the test procedures as we mentioned before. Annex D provides revisions of the document throughout its history from version one to version two and version two to version three. Annex E defines user requests. These are things that we've had people in the community say well, it would be nice if the standard could do this. Well, in some cases the standard actually does support that capability. It just may not be really obvious the way the standard is written. So it's kind of a FAQ, kind of a frequently asked question in that case.

In other cases, maybe some features that have been requested have not actually been fully implemented yet in a full design of the standard. And in that case, that's described there in Annex E as well. Annex F includes some generic clauses. These will eventually be moved into other standards such as the global objects standard, NTCIP 1201, but did not exist there right now. So rather than putting those clauses in the body of this document, we created a separate Annex for them. Annex G and Annex H relate to primarily for implementers to give them some practical examples of putting all of the rules together exactly what do you see go over the wire, for example, message. So that's the basic content of 1204 version 3. But what's the advantages of going through this whole process? While the systems engineering process defines a very easy to use document, okay, one of the key things we wanted to do by going through this process was not just hand out the design details and force every agency to figure out the full design in order to specify what they want to buy.

Now, they can look at the user needs. And they say, well, does my project actually have a real world need for the sample user need? If so, I can select that user need and now I can very quickly identify which requirements might apply to my project. And once again, very quickly read through those requirements and identify which ones might apply to my project. Once I do that, I can then quickly put together my procurement specification without ever getting into the low level details of how a standard actually works. So it's a much easier use standard. And as a result it's much easier to specify for a lot of agency personnel that quite frankly have no reason to get into the real low level design details of ones and zeros on the wire. But it also makes it easier to test because now we have clearly defined requirements to test against. We have a very clearly defined requirement that we have a very clearly defined test procedure. And now you will create automated tools to automate that test procedure.

And if it fails, you now have traceability from that failed test to a requirement. You can say it did not meet this requirement. And now, because that requirement is traced back to a user need, I can also define which user needs are not going to be fulfilled for my users because I have all of this traceable through a defined systems engineering process. It also means you can more easily have drop in user needs requirements and design content to support your broader systems ensuring activities. What we mean here is if you have a particular project and you identify a need that's not included in the standard, it gives you a clear pathway to say oh, well this is how I define a user need. Now, I need to define a set of requirements and I have a clearly defined path of how I define those requirements. And then go all of the way down to the design details. So it becomes much easier for you to extend the standard because you have this example right there in front of you.

Now, as I mentioned, the first thing we do is we define user needs. That's defined within the section of document that is called a concept of operations. The concept of operations gives you the overall structure of the architecture, if you will, of the environment we're dealing with and what you want to achieve out of that. It focuses on the system and its users. The timeframe is a lifecycle of the system, which means the development time you'll need to operate and maintain the system. So you can have some user needs that really relate to say, diagnosing problems in the system, as opposed to real life data management. It also defines the user needs supported by the standard—so, your real live data. So I need to monitor temperature. In addition, for the life cycle, I need to monitor how accurate my temperature readings are or if I'm getting anomalies in my data, I might be interested in checking out. The price of an operational context for the system, so it says why do I need to monitor the temperature, for example.

Finally, it reviews and builds on previous PCB modules that we'll talk about here. So as we go through this, we kind of assume that you've already taken the A102 and A201 courses. And already have a base concept of user needs. If you don't, you can certainly go back and review those modules online and bring yourself up to speed on anything that you feel uncomfortable with. So what are the primary uses of ESS data? Well, first, you have timely, accurate, and relevant weather forecasting and knowledge of existing weather conditions. So this is particularly true of traffic management systems and that sort of scenario. That results in more accurate traveler information. So now we're not only traffic management, but also traveler information, more effective, safer transportation system use.

Then finally, improved management of facilities, maintenance resources. So now I can manage my snowplows, decide where they go, and things like this. All of these things fit into the overall context, including with weather forecasting you may be providing now that you have new data in the field, providing this data to National Weather Service and other entities that might get into really heavy detailed weather forecasting. And through this whole process, we get a better view of the environment that's out there. It also allows us to make more effective use of advisory and regulatory mechanisms.

So, for example, in some parts of the country, they're monitoring weather conditions. And they may be posting warning messages on signs or perhaps even altering speed limits to reflect prevailing conditions. Enhanced monitoring of potential hazardous conditions. And that not only includes weather conditions but also environmental conditions whether it's air quality or other things. As well as sharing data with the broader weather community as I mentioned with National Weather Service and such. And also effective use of advisory mechanisms and regulatory mechanisms would also include things like being able to spray chemicals on your roadway through either your snowplows and other de-icing equipment or even through bridge pavement sensors and pavement treatment systems.

Now, that brings us to the first activity. And with this you can just use the chat pod, type your answers in there. The question is, what is the purpose of the system's engineering process? So go ahead, type your question into that chat pod and we'll talk about that once you're finished with that. So what is the purpose of the systems engineering process? Why do we go through this whole process? And, as some background, we have the user needs and requirements getting some responses back. So easier for use, test and specify and that's certainly true. To make sure that user needs are met and traceable through design and testing. That's certainly true. And to align objectives with designs. And that's all certainly true.

The answer we came up with was it provides a structured and reproducible approach to designing, creating, testing, and ending a system with checkpoints at each various stage to ensure the system will deliver what is needed. So a lot of it is that your checkpoints along the way, it's a proven process that is reproducible. And, you know, a lot of people

have tried to deploy systems without going through a rigorous process. And very, very often they get through whatever they do kind of by hand and realize that they should have been more rigorous about documenting things. The systems engineering process just makes sure that you hit these major checkpoints and that things are being done correctly. So it's a review process.

Well, a second question is what is the purpose, in particular, of the concept of operations? Once again, you can use the chat pod to answer this question. But so we have concept of operations, requirements, and other portions. This module focuses on the concept of operations. But what is really the purpose of that concept of operations? Why is that important?

Getting a couple of responses back now. Focus on system and its uses to define a shared vision of the system between stakeholders. And identify roles and responsibilities. High level of methods to employee, to define system, and how it will function. So all of those are good answers. An answer we came up with is define the why and what of a system via definition of user needs and selection of user needs by the system designer. And then probably one of the things that you could probably add to that based on one of the inputs here is yeah, an emphasis on the stakeholders that this is very much a shared division. It's not just one individual creating this document. It is all of the potential users and stakeholders of the system working together to clearly define what they want out of the system. So what's actually developed meets what they intended.

Well, some basics on user needs. Fundamentally, only users have actual needs. So that may a traveler, someone out traveling on the roadways or on your transit system, wherever, they have a real world user need. Team C operators also may have user needs. So they need to monitor the conditions remotely so that they manage their devices and otherwise warn the public. Likewise, maintenance personnel. If you're operating a snowplow or something, you also have some user needs that are specific to you and what you do for your job. Likewise, implementers such as system developers, vendors, system designers, and testers all potentially have user needs related to this sort of system and what they want to see in this overall standard. And all of these communities were discussed, were contacted as we developed the standard. Now, a transportation system does not really have needs.

So a TMC, for example, does not really have needs. It exists in order to meet the needs of travelers, TMC operators, maintenance personnel, and others. But it's those other people that actually have the needs. So that's how we've written all of the user needs within this document is a focus on the actual end user as opposed to this intermediate group. Discuss the problems. So a quick overview of the user needs. So the problem statement addressed by the standard is covered within that concept of operations. And the issue being addressed is to manage generic information. So this may include, for example, the ID of a device, so I know which devices I'm talking to and things like this, just very basic generic information. There's also a need to detect, sense weather information from sensors in the field. So, for example, I need to be able to monitor precipitation, visibility, all sorts of other things, air quality, for example.

And then finally, I need to be deployed on stationary deployments and potentially also on moveable vehicles. So I may have a weather station that is a fixed point in the field. Or I might have weather sensors on a snowplow, for example. And each one of those can provide me kind of a unique sense of what field conditions are like. And we've designed the standard to cover both those use cases. But clearly, if you're looking at standard or a deployment that is just for weather stations, a physical point in the field, then you don't need to worry about some of the details you would if you're on a mobile platform. And the inverse is true as well. If you're looking at a mobile platform, there are going to be some features that don't really apply to you that would apply to a stationary field deployment.

But regardless of whether you're movable or not, there are some basic characteristics that we developed that applies to both those deployments. And we modeled the standard in this fashion so that we understand there's a management station. And that management station talks to this field device, whether it's a mobile deployment or whether it's a fixed stationary deployment. And that link is the subject of the NTCIP standards. And that field device or—that in total—is called an ESS, or environmental sensor station. Within that environmental sensor station, our terminology is if there's a controller, that controller has three major components. There's an ESS manager that just kind of manages the device over all.

And then there's a manager for the sensors. Sensors may include weather sensors, air quality monitors, all sorts of

cameras and whatever else. And then there's a PTS manager and that's pavement treatment system manager. So, for example, if you're on a snowplow, this might be dictating whether or not you turn on and off the sprayers to spray chemicals on your roadway. Or if you're on a bridge deck, maybe a stationary deployment where you're just spraying from fixed nozzles on to your bridge deck. So in either case, it could be mobile or stationary. In both cases, you can have sensors and you can have pavement treatment components. And then that controller controls individual devices that are connected to that controller, which include all of your detailed sensors, your sprayer for chemicals, and your chemical tank. So that's your basic architecture that we've assumed for our ESS. And that terminology is used throughout the standard.

It also means it relates to some of the physical architectural needs of our deployment. So physical architectural environmental needs supported by the standard include the ability to, if you're a remote weather station, you might have one set of needs. A sprayer combined with a payment sensor would be a second set of needs. And a payment treatment station would be a third set of needs. So, you know, in the first case, it's a weather station, all you're doing is really collecting information from sensors. The last scenario is you're just in the treatment station, more of a control device.

But in many cases, or in some cases, these two functions are combined together. The standard works in all three of those scenarios whether it's just monitoring, just control, or a combination of the two. It also relates to in architectural needs in all of these cases what we discover is there's a need, basically to exchange data. And as we mentioned, there's a need to gather information from the field for your sensors that would mainly be using your get requests. There's a need to control sensors in the field or control devices in the field. They heavily rely on the set request. And there's also kind of an overall management capability of exploring what data is out there and that's where you get next requests. These are all very generic capabilities within all of the NTCIP standards. But it's very important that the reference within the NTCIP in this particular NTCIP standard to give you that full path of a full definition of what we're talking about. In each case we're basically dealing with fairly simple exchanges of ask a command from your management station to get something from the field or set something in the field and then you get a response back saying what has happened. So you have verification that your commands have been received.

Now, another architectural need that we discovered, actually, this was after we deployed the first version of the standard, we had a lot of people indicate to us that some ESS devices, particularly weather stations, are deployed in remote locations. And those remote locations - they have very limited bandwidth. In fact, some of them were connected via satellite communications or something else where it's very expensive to communicate data. Or perhaps they'll only communicate data once a day that they're dial-up and they connect their data, collect all of the data from the previous day. So as a result of that real world user need, we enhance the standard and define compressed data. And what this has done is define how you collect a lot of information and a single request in a much more bandwidth-efficient method than the standard SNMP operations. So if you have a need to for, you have a low bandwidth capacity on your link, or it's costing you a lot, then compressed data is probably—you know it's worth doing the software to make your communications more efficient over the line. And as I mentioned, there's also the possibility of offline log data. If you only dial up your device once a day rather than being constantly plugged into it, then the offline log data allows the sensor to collect period over time store it in the log and then allows your central system to come collect that log at the end of the day, or at the end of the week, or at the end of hour, however often you want to do it. You can record it as it happens, timestamp everything, and then upload the log whenever you dial in. That moves us now to operational needs.

So up to now, we've largely just been talking about the whole concept of how do you communicate with the device, under what scenarios? And how much of an issue bandwidth is. Now, though we're focused more on what's the actual content, what information do I need to exchange? Those are covered in your operational needs or features within the 1204 standard. And it relates to the information that users need, what information you need, and this is where you have the three major components of your controller, the ESS manager for your weather data and your air quality data, your monitoring type information; your sensor manager; and your PTS manager. So for ESS manager, which is just covering the whole device as a whole, this allows you to collect device identification. It allows you to external device control, monitor door status, monitor power, monitor mobile station data. So your latitude, longitude of your device, the speed of the vehicle, things like that. So all of these are kind of generic information about your ESS as a whole. And really most of it applies to almost any device. Some of it may not apply to any device, but a lot of this door status

applies to signs and other things, device identification. All of these sorts of things apply to most NTCIP devices. So that's a very high level ESS manager needs.

You then have your sensor manager needs. And the sensor manager needs are divided into several sub items. You monitor weather conditions, so various weather conditions, temperature, precipitation, whatever, monitor pavement. So you're looking at is my pavement icing or whatever. Monitor sub surface conditions which can be very important if I want to predict when my pavement might start to ice. Monitor human readings so not only am I monitoring automated weather conditions but now I can collect data from humans, what do they perceive as current conditions in the field. Monitor water levels, which a lot of these devices are deployed in low lying areas so you can see if your creek is rising or something. Monitor air quality and biohazards, monitor mobile weather profiles. So each of these are sub clauses within the sensor management needs section. And then you talk about the pavement treatment system manager needs. And there's two main scenarios here. One's a stationary spray system. The other is a mobile spray system. So as I mentioned before, stationary spray system may be just on a bridge deck. Fixed nozzles that spray chemicals on to a bridge or a mobile system that may be hooked up to a snowplow or some other device.

So that brings us to our second poll. And in this case, we'll use the polling capability which I will launch here in a second. The question is - what if a user need is not supported by the standard? And I'll launch this poll. We have the various options here. One is the standard like the entire suite of NTCIP protocols allows for extensions. So you can extend the standard however you want to. The second option is proprietary extensions are not desired because of interoperability problems, but sometimes they're necessary. The third option is proprietary extensions might become part of a future version of the standard. And the fourth option is the standard supports interoperability for all contained features. So you can select multiples of these if you wish to but whichever standard—whichever ones of those options you think apply go ahead and select those. And we will take a look at how you're doing here in a second. And we're getting a fair number of votes in so far but we'll wait a few more seconds here for people to go ahead and answer. Once again, it's multiple choice. So feel free to select more than one. And with that, we'll go ahead and close out the poll here. We'll close and I'll show you the results.

A fair number of you suggested, actually, all three of the items, the standard like the entire street of protocols allows for extensions. This is true. Every NTCIP standard allows for extensions. The standards define how you achieve certain objectives but it does not prevent you from extending and going beyond the standard. And that's—we recognize most successful standards in the world only are successful because they still allow manufacturers to go over and beyond the standard and do additional things beyond the standard. In the NTCIP, we knew if it was going to be successful over time and especially in such a rapidly changing industry, we had to support these extensions so that as new technologies came along they could be added without being stymied by the whole standardization process. However, even when using such an extension should be aware that any sort of proprietary extension is or does potentially create an interoperability problem.

Okay. The whole purpose of the standard is to ensure that everyone does things the same way. If you go beyond this and develop a standard or develop a feature that is not part of the standard, then you can no longer get that feature, most likely from two different manufacturers and expect them to interoperate. So it is well worthwhile getting things standardized first before deployment. But that doesn't mean you have to delay your deployments. Third is proprietary extensions may become part of a future version of the standard and that's certainly true as well. A lot of times what happens is people in the real world have experience with a problem. They identify a need for an extension. But as soon as they do that, they also submit it to the standards working group. And a few years later that feature becomes part of the standard.

Finally, the standard supports interoperability for all contained features. So yes, for all of the features defined in the standard, the standard does support interoperability. So, in fact, all four of the items there are, in fact, true. And there we go. So that brings us to the protocol requirements list. It's a table that maps user needs to the requirements. So as we discussed before, there's two major tables in the standard. One is protocol requirements list, the other is requirements traceabilities table. Protocol requirements list is the main table that is used by agencies procuring equipment. They go through identified user needs they are interested in. And then it maps those user needs to the requirements that are related to each of those user needs. It really should be a part of the agency's specifications. The intent as you take this PRL and you take the PRL and you include it in your specification. And, in fact, the copyright

of the NTCIP standards have been written specifically to allow agencies to do this. The PRL itself contains references to the standard that then define communications interface. So the PRL contains specific clause references so that you can identify very quickly the exact clause number in a standard to read the precise text and trace it through to identify exact design detail of that communications interface.

And all of this has been done to help you specify the particular interface you want to include for your project. So as a quick example of what the PRL looks like, the PRL includes the first line or the headings of the PRL table and then that includes a number of rows in the table. And we show one example of user need in this case. So the second line is the user need itself, in this case. The PRL starts out with a user need section number and that is the specific clause reference that I talked about. You can look into the standard, pick up that standard and go to clause 2.5.2.1 and you'll see that that clause is monitor weather conditions. And it gives the exact text of the user need. So if you're going through the list of your PRL and you're not quite sure if a particular row applies to your project or not, you can go to that particular clause in the standard and read about it and decide whether or not this is needed by your particular deployment. And then the second column there is the user need itself, the title. So it's just a clause number and the title. So using user need number, the corresponding text allows you to find. So this is, as I mentioned, going to that particular clause item, you can actually see the actual text of that user need.

For monitor door status, it would be a transportation system operator may wish to determine if any doors on the ESS equipment are open. This may assist the operator in determining whether maintenance crews have properly secured the controller. So you can read that whole clause and determine whether or not this user need applies to your particular project. And then go back, maybe discuss it with your stakeholders, determine whether or not you need this user need and then select it or not as the case may be. Once you select it, then you determine whether you have requirements underneath it. That's the subject of another module.

Now, another clue as to whether or not you might need this is related to the conformance column in this table. The conformance column may indicate that it's a mandatory item. If it's mandatory, clearly, you need to select it. Many of the items, though, are shown as optional. And if it's optional, that means you have a choice. You can either select it or not. And then several of these items you'll see the "O", meaning optional dot-two, well, that means as part of a conformance group. So there's a list of several different options, all a part of option group number two. And then the parenthetical statement shows you the rules related to that particular conformance group. So, for example, in this case, option group number two I must select at least one of the user needs, but I can select many of the user needs in option group number two. Okay. So I have to select at least one in order to claim conformance to the standard. I have to do something related to weather conditions or something related to one of these items in order to call myself an ESS conformant to the 1204 standard.

However, as you mentioned, some devices are mobile, some devices are stationary. So at times, some items only apply to certain environments. I have to select one or the other. So that sort of scenario is I have to select at least one. But in this particular case, I can select many as long as they must be selected from the same option group which is the dot-two portion of that expression. So that brings us to our activity. Considering what we just learned, please look for other user needs within the conformance column of value 0.2. And you can do this by, if you don't have it up already, if you go to your material section in your window there, go to materials and open up your student supplement. And within that student supplement, you'll find the table that lists out all of the different options. And so, go ahead, go to that, and then identify for us the optional requirements, those labeled 0.2 within that table.

Alright, well, that brings us on to the—there's a total of six different answers. All six monitor weather conditions, monitor pavement, monitor sub surface conditions, monitor human readings, monitor water level, they monitor air quality and biohazards. Now, when you look through these, you might notice that, in fact, they all are in the same overarching clause. And this isn't an absolute requirement but it's frequently done this way just because it's the logical way of setting things up. But they're all 2.5.2 and then .1, .2, .3, .4, .5 and .6. And if you remember back, we said option group number two you had to select at least one but you can select multiple. And this is very common. You may be an NTCIP-conformant weather station if you only support weather conditions, monitoring weather conditions. But you can also support monitor pavement, monitor subsurface conditions, etc. You could also be conformant if you only monitored human readings, for example.

So by standardizing NTCIP 1204, we've not standardized the exact equipment you've put out in the field. That is up for you to decide for your particular project and that is the importance of going through these user needs for your particular project and identifying these are the features I need out of my system. So when you go through this PRL, everything is laid out kind of in menu fashion where you've made it fairly easy but it's still very important you go through this process. Otherwise, you just say I want an ESS and you might get something that only supports human readings and you wanted a completely automated system. Technically it would still be conformant but it would not be meeting your needs.

Now, some user needs are, in fact, basic needs related to an ESS. That includes retrieving ESS characteristics or configuring an ESS manager, for example. Those needs are considered mandatory but most needs are optional. So there is a basic set of user needs that are essential for device to be in ESS.

That brings us to our next activity, which if you look back in your student supplement, identify all of the user needs with conformance set to M or mandatory. So go ahead and take a look at the supplement, pages four and five in the PRL table. Go ahead and use the chat pod to indicate which user needs by just the clause number, which user needs are mandatory. I'm going to give you a few seconds to do that here.

Alright, so you should be pretty well through that. Which ones are mandatory? And if you look through the document, you realize that there really are not all that many that are truly mandatory. Architectural needs are mandatory. And then underneath that, generic architectural needs is the only one that's mandatory. Under features, you have ESS manager features is a high level concept, but only generic features at the detail level. Monitor pavement is a high level item that is optional. And underneath that, you have a mandatory monitor pavement surface condition. But what's interesting, as we'll learn here in a second, is because monitor pavement is optional the more detailed item which is mandatory is not optional. And we'll talk about that here in a second. Likewise, pavement treatment system manager features, there's a couple of needs under there that are mandatory. And then there are some generic architectural needs provided by data as mandatory and retrieve the device identity is mandatory. But really overall within the whole complex standard, there are not really all that many items that are truly mandatory which means, which emphasizes the importance of going through the document, through the user needs and identifying which features you want on your project because if you don't select what you need, very likely you will not get it on your project.

But there is a relationship as I mentioned, a relationship between the higher level parent user need and the lower level user need. So in the case of monitor pavement, we mentioned there's a monitor pavement surface condition. Well, that detailed item is shown as mandatory. But if you look over to the right, the column for a project requirement you actually have a choice. You can either say yes, support this feature or NA, not applicable. Okay. You can't say no because it's a mandatory item. But if the parent requirement or the parent user need I should say is not selected, okay, because that's optional and I select that no, then the more detailed items become not applicable. So they're mandatory if and only if the parent user need has been selected.

That brings us to the next poll, which I will go ahead and bring up here. So support project requirement. What is the project requirement value for a mandatory user need? And we'll go ahead and put that up there. So if the user need is mandatory in the conformance column, what is the project requirement value? The example here shows an optional conformance item with values of yes, no, or NA. But if it is a mandatory user need, what would the options be? And we've gotten several people to vote. And I'll go ahead and close that out.

And you'll see that most people, or actually everyone, said yes, which is the correct answer. And I'll show that here, a value of yes, versus a value of yes, no, or NA. So if it's mandatory, you must select yes, and that should really be the only option given. So exercise two, let's say that a DOT wants to deploy three weather stations to supplement weather data from a TV and radio available for TMC. So they're going to have these three weather stations deployed. And their goal is to have each of these ESS include wind temperature, surface fog, and sub-surface sensors and to report all of that data back to the TMC.

Now, my question is - has the DOT defined any user needs in this paragraph. Or in particular, what are the user needs? Can you think of any user needs that you could derive from this paragraph? And you can go ahead and chat those and put those into the chat pod and send them in. So we're essentially just looking for, you know, if that's what your

manager came to you and said we want to deploy these three stations and we want to collect this data, what user need do you think you would be looking for when you start looking in the 1204 standard? What user needs do you think you need to be looking at? So go ahead and type those in. You just take a look at that statement and we're starting to get some responses back. Okay.

And as the responses are more or less what we've seen. Yes, user needs are to monitor and report back temperature, wind, surface visibility, and sub-surface readings. And each one of those are user needs that you should be able to look for in the standard and map your requests to user need. Now, a second example, DOT wants to deploy three ESS stations. So the basic condition, what are two user needs that are applicable? So specifically, if you look at your student supplement, go in there and find out which user needs from the standard are applicable. And you just type in the clause reference from that table there in your student supplement. And we'll give you a few seconds to look those up in your student supplement. And that PRL table you'll find that listing the user needs and which ones indicate the wind, temperature, etc. So I've gotten some back: 2.5.2.1.2, 2.5.2.1.3, 2.5.2.1.7. And you see that those are, in fact, the right answers. Also 2.5.2.2.1 - a monitor pavement surface condition and 2.5.2.3 - monitor sub surface conditions.

So user needs—I mean this has all been a great exercise but essentially, why are we doing this? We're trying to describe the features of the device and not only what the features want, but why we need them. And then that allows us to better clarify the functional requirements that we need. And the requirements refine those user needs into detailed measurable specifications. So, whereas user needs say well, we need to monitor temperature, for example, the requirements will start saying, okay, what sort of resolution do we need on that temperature range? What sort of measurement? Is it Kelvin or whatever? And within the PRL, the relationships between user needs and functional requirements are standardized.

So the PRL gives you a very clear rule. As soon as you select a user need, you immediately become aware of the functional requirements related to that user need. And that facilitates the job of the procurement officer by simply saying, okay, the user need is a very high level concept, select that. Now, I know which portions of the requirements I need to look at more detailed to identify whether or not these requirements were not. And then once I select the requirements, as we discussed before, that traces to a specific design solution then that promotes interoperability. And as we show on the table, the actual full table includes not only user needs as we've talked about to date, but underneath each one of those user needs is a set of functional requirements that trace to that particular user need. And as you see here in this example, you can have multiple requirements related to a single user need. And in each case, you have a functional requirement identifier and a functional requirement title.

So this is very similar to what the format was for the user needs themselves. You have user need ID, user need name, and then the next column's functional requirement ID and then the name of the functional requirement itself. The requirements that are associated with user need are found underneath that user need. Each user need will have at least one requirement associated with it. Otherwise, I mean, that shows you that we're fulfilling the user need. So every user need has at least one requirement. Each requirement in the standard must also be associated with one user need. So we've done a check already as part of our design review of the standardization process. We make sure that every single requirement of the standard fits into at least one user need. Otherwise, it's an unmapped requirement and it's just wasting space in the standard. There's no need for the requirement.

So we've done double check making sure that every requirement has at least one, every user need has at least one requirement. Every requirement has at least one user need. The result is the standard has no unnecessary requirements, that all needs are satisfied. So a full PRL will look something like this, where you have several different lines, your darker grade rows indicate kind of a high level concept user need. Your lighter grade cells indicate a detailed user need. And then underneath those detailed user needs, you have your specific requirements that relate to them. Now, in cases, you'll also see a predicate. In this case, you have monitor mobile station data. You see the conformance clause there is mobile column, colon M (:M). In that case, you look for the concept of mobile. And if you did a search on a document, you would find that mobile applies to a mobile or a moveable station, if that's not too much of an oxymoron for you. But basically, a snowplow type device or some sort of weather system on a moveable vehicle. So if it's movable, then you need to support the monitor mobile station data including the ability to retrieve mobile ESS movement, things like this. So it's just as simple if you support this feature then the condition applies either mandatory, in this case, or maybe optional.

So a little bit of definition about mandatory and optional. A mandatory requirement is only mandatory if an associated user need is selected. So before, we talked about a parent user need and its child user need. That same sort of relationship applies to a user need and its children functional requirements. If an optional user need is not selected, its associated requirements are not necessary. But recognize that each requirement may appear multiple times within the PRL, designed to have a single requirement related to multiple user needs. Now, under that scenario, if a requirement is ever selected as being required for a project, then it is required. Even if it's elsewhere in the table, it's not selected, if it's ever selected once, it is required. You'll also notice in the table that we showed before there's also a projects requirements column. This allows you to enter additional notes in requirements for your particular project and provide further details about the implementation. In cases in the standard, we have filled out standardized text with a blank for you to fill out. But in many cases, it's empty. We don't really envision that field being used. But if you do see a need, you can certainly put information in there.

Well, that brings us to our next activity. Assume you're procuring a stationary ESS system that monitors the conditions of roadway pavement. Which user need is mandatory for this RFP? So go ahead and just type your answers into the chat pod. Do you believe that the monitor door status is mandatory or optional? Do you believe monitor pavement surface conditions is mandatory or optional? Do you believe monitor subsurface conditions is mandatory or optional? And do you believe retrieve the device identity is mandatory or optional? Which one of those would be mandatory?

I'm getting some responses back and they seem to be let's see 2.5.2.2, okay, 2.5.2.3, alright, those are both—I'll show you the answers here. Monitor door status, no. Monitor door status is shown here as an optional in the conformance column. So it is not mandatory for this RFP. Monitor pavement surface conditions is, in fact, required for this. It is mandatory because it has that M next to it. Monitor subsurface conditions is no; it's optional. And retrieve the device identity is mandatory. Functional requirements will be covered in the next module. So that largely reviews and covers our discussion of the user needs. But module A313b will continue discussions and discuss how you go about specifying requirements for the ESS systems based on the 1204 version 3 standard.

From a vendor's perspective, even if a user need and resulting requirements is not mandatory, a vendor may still optionally fulfill that user need. This goes back into the well proven concept that a product provided by a vendor needs to meet or exceed the specification. So it can do other things beyond what's being required, but only has to do what has been required. So just because a user need or requirement was selected as no in a RFP, does not preclude a vendor from providing that feature. It just means that it's not required. So the other thing is a vendor can also provide a PRL, can fill out a PRL for their own product to give agencies an idea of the product that they can sell. So this lets agencies know that they can select this particular set of features in a device and that they actually have a product that meets that particular specification. And then an agency can collect these PRLs from a variety of different vendors and find out what's available in the marketplace so that they can make sure that they're not specifying something that is unique to one vendor or perhaps even not available on the market at all.

From an agency's perspective, the completed PRL must become part of the overall specification. The PRL, as I mentioned, there's a specific copyright rules within the standards that allow them to do this. And it should be part of the specification, the completed PRL in the case of the requirements for the communications interface and by extension the user needs that ESS must support. So it indicates the requirements, the communications interface and all of the user needs all in that PRL document. So it really covers what the standard—what the agency is looking for. Specifically, that PRL document is one portion of an overall project specification. Most projects are large documents, large contractual documents. Part of the contractual documents are product specifications. Part of the product specifications are hardware specifications. That's an environmental requirement and all sorts of other things. You have software specifications. And then you have communications interface specifications.

Generally speaking, the NTCIP PRL is a part of your communications interface specifications. And specifically related to the functional requirements of that communications interface. Now, there are certainly interrelationships with other parts of your contract documents, in particular, your hardware specifications. Your functional requirements defined in the communications interface need to line up with the functional requirements in your hardware specifications. But the PRL itself would typically go into the communications interface specifications. The completed PRL needs to be consistent, as we mentioned, with hardware specification. And that the PRL shows the intent. Interested vendors can view the PRL and understand the intent of the requirements. And one of the huge benefits of using the standardized

PRL is that the vendors then become very familiar with what's meant by the PRL. So you minimize misunderstandings. They look at it. They know exactly what to expect from your project. One also note on the conformance versus compliance, conformance is a term that means simply meeting a specified standard.

Now, strictly speaking, it means that it, as a minimum, it meets the mandatory requirements of that standard. To claim conformance, the vendor shall minimally satisfy the mandatory requirements without violating any rules. But as we discussed here today, many of the user needs and requirements in the standard are shown as optional. So you don't have to support everything in the standard in order to be termed conformant to NTCIP 1204, which means I can have a conformant device in my system but it doesn't do what I need it to do it. That's part of your job as an agency procurement officer - to make sure you specify which features of the standard you want in your system. And that's where compliance comes into play. That it meets a specification.

Now, strictly speaking, you can be compliant without being conformant to the standard but the intent is compliance as a part of your compliance, your requirement, and your project specification is you'll require conformance to a particular set of features. So vendors that provide additional features beyond that are still conformant. But they meet what you've required in your project and that's compliance. And hopefully you've required conformance to the standard. So another poll. Can an implementation that monitors wind speed but not air temperature be conformant with the standard?

And I'll go ahead and launch this poll real quick. And there we go. So can an implementation that monitors wind speed but not air temperature be conformant with the standard? So go ahead and put in your answer there. I'll give you a couple more seconds to do that. It looks like we got most of the people there. So yes, the answer is—80 percent of you answered yes. It is conformant with the standard and that is true. Now, technically you also have to support some others things but you do not have to monitor air temperature. So it monitors wind speed, but not air temperature, as long as it supports all of the other mandatory requirements. Yes, it could be, it would be considered conformant with the standard. It does not have to support air temperature. So, I'll go ahead and hide that.

An ESS, another poll, an ESS has a gate to close the roadway based on detected roadway surface conditions. Is that conformant to the standard? So I'll go ahead and launch that poll. So an ESS has a gate and that's going to close or open based on roadway surface conditions. Is that conformant with the standard? Go ahead and answer that. And we'll go ahead and then close that out.

A little bit more divided vote on this one. Sixty percent of you answering yes, though. And we'll show that the answer really is kind of potentially. The standard neither defines nor prohibits a gate closure or function. The device conforms as long as the standard—as long as it properly implements the standardized features and does not violate any rules adding the functionality. So as long as it supports the standardized, mandatory standardized minimally conformant features, it could be termed conformant, but the feature we're talking about here is really beyond the standard. So it's really kind of a non-issue.

That brings us to the next and final poll. Would an implementation that does not support the ESS device ID be conformant with the standard? And let me launch that poll real quick. So go ahead and answer that. Would the implementation that does not support the ESS device ID be conformant with the standard? And I got most of you voting, 100 percent of you voting no and you are correct. An implementation does not support this requirement it's a mandatory requirement, so if it does not support that feature, it would not be conformant with the standard.

A final note on backwards compatibility. The 1204 standard version 3, actually all of these standards, are backwards compatible. We've taken great pains to make them backwards compatible. That doesn't mean that you won't find any issues, okay. The version one and version two standards did define approaches to certain functions differently than those in the version three. But here's the catch. The problem is real field experience we discovered that the original definition had an ambiguity. So rather than actually redefining those particular pieces of data we said well we're going to basically say these are deprecated. These are no longer the preferred way to communicate that. What we prefer you to do is the new version three-way. But we still recognize the old data, okay. And if you buy a new version three device you can say I want to make sure that I'm backwards compatible with my version one or version two information. And I need support and be able to recognize that older way of exchanging data. And that's what we mean

by backwards compatible. It is—the capability is there. There's still a way to unambiguously identify that you're exchanging this older data. But recognize the reason we went away from that older design is because we found ambiguities with the design. What that means is if you require conformance to a version one or version two deprecated feature, you need to specify the interpretation, your particular interpretation of that older definition that we discovered was ambiguous. So it may make perfect sense. If you already have equipment to put in the field with one interpretation, then you want to make sure, well, if I have new equipment, I want to make sure that it interprets that old data the same way and you can put that into your requirement if you need to.

But really what you should be looking at, you know, just like with any upgrade, you want to be moving or migrating, have a plan to migrate to the newer definitions because they are more unambiguous and more interoperable throughout the industry. So in summary, the systems engineering process goes through user needs, functional requirements. Those are mapped to each other through the protocol requirements list. You then have dialogues and object definitions. Both of those are mapped to the requirements through the requirements traceability matrix. And then finally, we have test procedures with its own table. The version three standard does include test procedures and it also addresses backwards compatibility. The contained features include kind of very high level generic features of all devices. Those are called ESS manager features. They also include sensor manager features. These include all of your weather conditions, your air quality conditions, etc., things you tend to monitor with sensors. And then it also includes pavement treatment capabilities, whether it's a mobile system or a stationary system. The PRL links the user needs with the requirements. It allows specifiers to quickly select your user needs and requirements. And it is strongly advised to use this PRL as a part of your agency's specification.

So what did we learn today? Well, NTCIP 1204 version 3 defines the concept of operations and user needs for environmental sensor stations. In fact, it follows the SEP approach. And there are three major categories of operational needs: ESS manager, sensor manager, and PTS manager. A protocol requirements list is used to link the user needs to the functional requirements. And finally, a completed PRL should be integrated into the specifications as a part of the PS&E package.

There are additional resources, the standard itself at NTCIP 1204 version 3 and it's specifically - v3.08r2 is the final version that was published. That you can download from www.ntcip.org. The NTCIP guide, or NTCIP 9001 version 4, can also be downloaded from that website. And you may also be interested in the IEEE standard that defines how you go about defining a concept of operations. That's IEEE 1362 and that can be purchased from IEEE. Next steps, as we mentioned, this is module A313a. There's a follow-on module talking about requirements that's called A313b. It reviews the requirements supported by the standard, shows the relationship between requirements and the design that satisfies those requirements. And shows how to select and refine requirements using the PRL. And with that, we have time for questions. You can go ahead and type them into the chat pod. And I'll be happy to answer any questions you might have. Thank you for attending.

And we'll just wait a few seconds for anyone to submit their questions. Some of the questions I've seen come up before include things like how widely deployed is the standard? And this particular standard is very well proven. In fact, I have not seen a recent deployment of weather stations that have not used the standard. There may be some out there but I'm just not aware of them. Virtually every deployment coming out today is using the standard for NTCIP and I've seen that both for mobile systems and for stationary systems. So lots of deployments of the standard. One of the better proven standards, one of the more widely referenced standards. Well, with that, I don't see any questions coming in. So I thank you for your time and this concludes module A313a, Understanding User Needs for ESS.