**PRESENTER:** I usually like to start off with a big idea for the day. I'm a little mind blown by big ideas between the product thing and everything going on. I'm not making super movies and doing things five years out.

I'm from Wisconsin. I do support for people doing everything from the end of surveying mapping all the way through construction in our engineering software, which happens to be Civil 3D. And we have everything from megas down in Milwaukee that are going on constantly all the way up to the other end in a little town I call superior, which is where I grew up.

And they're dealing with [? five-mile ?] mill and fills. Redoing intersection. Adding a turn lane to something. And those are the people that I typically deal with. There's one person on a job. And they're doing all the reports, and they're doing all the engineering. And they're doing everything.

So a lot of the time that I spend is helping people push this software and push this functionality down to the lowest level and see if we can get value out of that. How much of these big, awesome ideas can we do on the half a million dollar intersection rebuild or the $2 million small town reconstruct? So that's what I wanted to bring today, cross section labels.

Not super exciting. I don't have a laser show about it. But you know what? If you spent a day or two banging your head against stupid labels that aren't doing what you need to do, then this is what we need. So this is my little idea that, hopefully, we can make happen here today to help you guys do the things that you do on a daily basis.

So while he's setting up, I'll give you a little more background about what we've been doing in Wisconsin. We've been on Civil 3D in production since about 2010. And our big focus has been on construction ready models out of design because our earthwork firms, a number of them, have been reverse engineering our plans for, like, ever.

And they said, hey guys, could we figure this out? So we've got a team of about five internal people. About as many of that consultants helping us out. We've been building up the system, building up the system, building up the system.

So a couple of key points about what we've been doing. We have not tried to address the how do you stamp a digital model. How do you make it a contractual document? We've been really

focused on making sure the models are good. So we've been focused on work flows and deliveries.

**AUDIENCE:** [INAUDIBLE].

**PRESENTER:** Are you 2016?

**AUDIENCE:** That's it? And this one?

**PRESENTER:** Yep.

**AUDIENCE:** And then?

**PRESENTER:** And you just grab the whole thing, and move it over.

**AUDIENCE:** Yeah, OK.

**PRESENTER:** All right. So we've been focused on the technical end of that. So what we've been doing is we haven't gotten rid of any of our 2D deliverable. What we've been doing is making a 3D deliverable that meets our construction AMG spec. So it's got to be within something like 2/10 xy and a couple of hairs on the z so that when they set up their base station stuff, they can do basically earthwork.

We've got in mind getting ready for paving. But none of our firms have gone there yet. So what we do in our cross-sections, and I'm faking it a little bit today, but we generate surfaces out of feature lines only. So if you haven't done 3D stuff for construction yet, I'll just tell you anyways, all of our firms say the cross-sections are great, but what we really need are the feature lines.

You could throw everything out. But give us the feature lines. That's what we're going to build off of because they're going to suck that into Trimble. They're going to suck that into [? Topcon ?], and build their own surfaces off that.

Some of them are building from subgrade up. Some of them are building from top surface down. And we're trying to support that delivery model. So there's two questions that come up all the time. And one of them is, when are we getting rid of cross-sections? That usually comes from 25 and under.

And from 45 and older we get, when are you gonna [? bring ?] cross-sections? And the reality is, basically, not yet, OK. We're still in a PDF world. That's a contractual document. But what

we've done is make the workflows and the deliverable say the cross-sections need to match the model because the model's going to be more accurate.

We're aware of that. The density and the detail that you need for AMG is far greater than you'll see in cross-sections. But the damn thing is if you're cross-sections are wrong, that's your contractual document, OK. So that is why we still are doing cross-sections.

And I've only wasted 15 minutes of my time. So let's get this party started. So what we're doing is our design model is the surface. Well the multiple material layer surfaces. So we've got [? a ?] [? top. ?] We've got [? datum. ?] We've got all those.

But none of our contractors are using Civil. And no other software can suck in a surface. So what they want is the feature lines that built the same surface. So we are doing two exports, basically. We're going to DWG basics.

So you know, faces and 3D polylines. And then, [? o:XML ?] export, [? wear ?] it, merits it. And this is one of the challenges when you're not design build. We're not a design, build state. And we've got a handful of those kinds of projects.

But that handoff, especially when they're in a completely different software package, that's where we're at right now. So [INAUDIBLE] [? Casey? ?] All right.

**AUDIENCE:** [INAUDIBLE].

**PRESENTER:** All right. Press on.

**AUDIENCE:** [INAUDIBLE].

**PRESENTER:** Thanks, man. OK. Sometimes it's worth putting the effort into a slide deck. So I've got a couple of captures here. And we will do our best. So back one. So automatic or, as we would say from the presentation this morning, automagic labels today and manual labels.

These are not Civil 3D Autodesk terms. These are my terms for what these labels are. Neither of these are an m text or a dimension sitting on top your cross-sections, OK. Both of these will be live to your section view.

They've got some differences between them. And I'm going to spend a lot of time on the automatic stuff because that's really where you want to be when we're talking about corridors that change right up until submittal and keeping them alive with things.

Automatic section view labels, basically, means becoming a code set style ninja. You have got to dig into that and completely understand it for it to work. Corridor data wise likes to go to section views. And it likes to go to a surface.

If there's anything that's broken in your section view or your surface, however, the fix is back in the corridor, or in the assemblies, or in the code set style. You do not want to be tweaking your corridor surface because as soon as you do a rebuild, it's going to blow your edits away anyway.

So automatic labels are totally in the code set styles. Manual section view labels are basically this. Off the annotate tab or from the contextual for section views, there is something called a section view. There's an offset elevation label. There's also slope labels.

These translate into points and links. However, [INAUDIBLE] offset elevation label is, literally, just that. You can click anywhere on the section view and say, boing, there's a label. And it'll stay with your section view. It won't stay with your corridor. OK?

Interaction between assemblies, corridors, and code set styles. Blah, blah, blah. This is where I'll probably spend a lot of time today since I can get a nap. Can I go back here? I can go back here. I can point too with my pointer.

So this is the DWG way up at the top. Over here, we have the code set style. This is where all the styles are. Now when you build an assembly that's built up from multiple sub-assemblies, the sub-assemblies are containing all of your codes. Your point codes, your wink codes, and your shape codes.

I haven't seen anybody do a ton with shape codes. I mean, the most you could do would be a hatch pattern or some kind of fill pattern. Point codes are where your elevation offset labels are going to come. And then, the link codes are your [? sulps. ?]

So this is where the data is. That gets built into an assembly. The assembly gets applied to your corridor on a baseline and region. So we, basically, got our cross-section. And we apply our cross-section across the region. Now we've got targeting. And we've got conditional subs. And we've got all kinds of crazy that can make it happen.

But what's going to end up on your cross-sections is those points and links. The corridor then gets fed to sections under your alignment and a sample line group. And you get a section view

group and a section view.

The other place that corridors go to, obviously, is a corridor surface. Now the one thing that I want to point out in this whole deal here is everything that is orange can have a code set style applied to it. So you can style all this stuff.

Couple of key points here. You can style an entire section view group. Or you can style just one section view. So think about that in styles. I mean, if you're in Civil 3D, you got to at least wrap your brain around there's a style for everything. And you probably want 5,000 styles for every single thing that there is.

So you can be applying a code set style to an entire group. And you say I've got a 500 foot stretch that's just a mess. Tweak it. And apply it to just that 500 feet. The other thing is that we can apply it at the corridor level. It's not too interesting because we're just in model and plan view.

OK, so code set style for the corridor view, typically, just drives as your feature lines. So you can code your ditch to be a blue line. And you can code your slope intercepts. We do a dash deal so that when you look at it from the top view or in model space, color and line weight can be giving you some information about what that feature is.

But where it really comes into play is in the corridor section editor. Now the thing that I want to impress upon you is that these are two different places. I'm going to call this design. This is where we're doing design. Corridor section editor and model.

And your section view groups are what you're sending to your cross section sheets. So that's your output. That's your plan set. So what do you want to do with your code set styles?

How many of you are managing your code set style for your firm or your office? Or how many of you are lucky enough that the guy or the gal down in the office or whatever is managing that for you? OK. Even if you're not managing codes, or styles, and templates and everything for your firm, you're going to need to understand code set styles because I have not seen a code set style that you can set out of the box, run that on every single project, and it'll do exactly what you want.

There's always some tweaking that goes on because of all the crazy things that cross-sections can do for you. Another thing that I've added is adding some point codes that are strictly for labeling. So general, the way I see most people deal with point codes and link codes is they're

driving both data and labels.

It doesn't have to be that way. And I'll show you why you might want to consider doing that. Obviously, if you're doing the same edits again, and again, and again, you might want to think about expanding your style templates so that you've got something more to load into new projects.

And one of the things that we went to, probably, about two years in is elevation offset labels are awesome for your cross-sections. They don't tell you a whole lot when you're in design. When you're in design, you need to see what those codes are because those codes are driving everything.

So in design, we're viewing actual code names in section editor or something like that. And then, we're doing the elevation offset labels over in our sections. So what does that look like? It looks like a hot mess to be honest with you because of how many flipping codes are everywhere.

But they are what drives your offset and elevation labels. Right? So this would be a view for a section that's got a conditional daylight for cut fill that I'm showing just the point codes. So those are the actual codes for those points.

You can do links. But a lot of times you're not labeling much with links. And they're pretty 2% slope. OK, whatever. So we've got a style set that is just point codes and a style set that is point codes and link codes.

So when you get over here, this would be our labeling and our cross-sections. Well say we had-- for whatever reason, the logic that goes through a subassembly-- say we had another point getting built on this [? four to one ?] that was in the middle and was just making a whole mess.

In some situations, maybe, this is a 20 foot long fill section, and it's breaking there, and you need it. But sub-assemblies are, kind of, their own little gremlins. Our cut fill sub-assembly is awesome. It'll do everything. But it sucks because it will do everything.

So in a simple little scenario where you've got one slope, you can get points in there and stuff. So if I want to get rid of that point, I don't want to push the Delete button. I want to go back to my styles and say, OK, what point code is that? Somehow, I've got to get rid of that point code

so that this section looks good.

And especially because it's run 1,000 feet down the road, and I don't want to go delete, delete, delete, delete, delete. So multiple methods for managing automagic labels in section view groups. OK, so in my experience, like I said, if you ever come up with a code set that works for urban, and rural, and retaining walls, and raise median, and ditch median, let me know because I've never seen it happen.

The best that you're going to be able to do, and you should do this, is try and come up with something that looks good most of the time. So maybe grab two or three of the projects that you've done. And you start turning on and off the labels and the code set to make it look good. And if you're lucky, you're going to get 80% of the way there.

And that's going to be what you're going to use for your basic template. Give it to your designers that aren't diving into this stuff day in and day out. And say, here, start with this. Next, once you've gotten a code set style that gets you pretty far down that road, once you apply it to an actual corridor, you're going to have to mess with it.

You're going to have to change the labels. And you're going to have to turn some on and some off. And the more you can do that at the assembly level, the better off you're going to be because, remember, these code set styles don't-- every style is individual to the DWG.

So we've got a corridor file or multiple corridor files. And our cross section files are over here. So it's always an [? xref in ?] and suck it all in. I'm a big fan of not putting any more into a corridor file than you absolutely have to because they usually get huge. And you start handing them off between computers.

And Jerry that's got the flaming hot rod gives it to Kim who's got a somewhat normal machine. And blah! So that actually is a good thing in this scenario because you can twiddle with your standard code set style, make it work for those sections, and you're not you're not messing with anything else. It's just that one section file that you're getting to look right.

So custom point codes and marker label set styles. I'm just going to go to what we have right now. So what I'm showing you here is eight different marker styles with-- no, 16 styles with eight different codes. I wonder if I can do the-- there we go.

All right. So these are our standard codes here. And the only one that we use in default is this one right here. So we label them around the horn. 0, 45, 90, 135, all the way to 315. And so

this is down at a 45 degree angle. And it's applied to every single point.

That doesn't mean that I might not need some other style. You got two labels that are smashing on top of each other. Maybe I just need to flop the other one over to the other 45. Or maybe I need to send that one up. And then, the ones over on the far side there, they're the same deal, except they've got an extension line on them.

So you got two points right here. And they're smashing on top of each other. If I turn one into an extension line, all of a sudden, they're extending nice. And they're not overlapping. And I haven't done anything to drag or manual edits or something. I just change the style that I use from the 315 no lead to 315 extended lead.

Now the last thing that I'll tell you that we've developed is these are actually point codes. Elevation offset. Whatever. So we're used to dealing with point codes of ETW, ETW sub, CL, CL sub, DP, daylight, and that kind of stuff. These are obviously never, ever going to show up in an Autodesk stock code set. And they're not supposed to. These are built just for labeling.

So the thing to remember about subs is some of them have hard-coded points. And that's terrible, terrible, awful, awful, bad, bad. It should never, ever happen. Some of them you can change the point codes on. The ones that you can change the point codes on, you can add multiple point codes to.

The other thing that this opens up is everybody knows the mark point sub, right? You can click it on anywhere. You give it a point. It does whatever the heck you want it to do. You combine this with a mark point sub, and you can label anything you want anywhere. All right?

And I will talk my way through that-- I still feel a little queasy-- later on. But I'll show you what that's like-- and good. Just all the way. All right. There we go. So corridor codes. Data. Label both. Like I said, often times, they're used in both. Your ETW sub is going to become a future line for your surface.

And you're going to label it something. That's a very convenient, easy way to do it. But it doesn't have to be that way. You can just use point codes and link codes for data. And you can just use point codes and link codes for elevation offset labeling.

So this is what it looks like in the code set style. This is in the pull down of your point codes. And the labels-- is that OK to read? Or do you want me to do the zoom thing again?

**AUDIENCE:** [INAUDIBLE].

**PRESENTER:** Oh, good. OK. So these point codes are sitting in my standard code set styles. And they're basically doing nothing until I add them. And when I do add them, then all they're doing is firing off whatever I want to label.

So how do you apply this? Well you can change the point codes. So that goes into once you've got these custom codes built up. And by the way, all these codes that I've shown you, I put in the additional file. And you can grab them.

Please, if you haven't started this, even if what I showed you isn't exactly what you want, if it's something like half way there, steal mines. Steal somebody's who willing to let you steal them. Don't start on these things on their own because between drag states and line objects, the marker labels are a whole little sub genre of things in Civil that if you're just a regular designer and you're hammering out projects, this is not somewhere you want to spend your time.

Start with something that's somewhat close to what your production standards are. And build off of them, please. So you can either change the label styles. So that is like, OK, ditch point is doing the wrong thing. I want to change it to one of the other styles.

The thing that I want to mention here between whether you change point codes or whether you change the styles is back to my crazy picture here. So this is all in one DWG. But like I said, we typically have our corridor file somewhere. And we've got our cross section file somewhere else.

That means you got two DWGs. That means you've got, potentially, two different code set styles in each one of these. The point codes are coming from the assembly, which are being applied to the corridor. So point code modification has to happen back in the assembly. If you're changing the styles, that's going to be happening in your cross section file.

So if you are not putting everything into one mega, super, uber DWG, which is a really good idea, you just gotta keep in mind what change you need to do, and where you need to do it. So styles are going to be happening over here. And changing the point codes is going to be happening where the assembly is. Make sense? OK.

All right. Am I in the right slide? MarkPoint sub with label only point code. If I could go white board on you right now, I totally would. But I'm going to do my best to talk through this, OK?

So changing your label styles works awesome if your point codes are different.

But if your point codes are identical, you can't make things different. So for instance, if you want the left side codes getting shot to the left and the right side codes shot to the right, we don't have left side, right side codes by default in Civil 3D, right. It's just daylight. It's ditch point. It's whatever.

One of the reasons why we're changing all of our sub-assemblies over to have total control of your point codes. But if you've got identical point codes and you want to change the labeling, what you need to do is turn off the labeling for that point and go to those custom point codes. That's where this mark point sub-assembly comes in.

So let's put on our imagination hats. And you can imagine a rural two lane, OK. If I want my ditch point on one side to label differently than the other side, I can turn off ditch point, and do a mark sub that does down and to the left on my left side. I click it down to the right on the right side. And I've just changed the same point code with different point codes.

Mark sub doesn't do any logic other than snap to a point. Give it a code. And then, the label takes it off from there. So in my example of elevation offset, whatever, my right side would be elevation off 315. And my left side would be elevation off 225. That sounds right. Yeah, 270 minus 45.

So let's expand that idea. I've got this big four lane divided urban reconstruct. I got bus turn outs/ I got left turn lanes. Everything's crazy all over the place. I've got multiple lanes. ETW sub is getting labeled all the dang time. I need to control it.

I can turn off all of those point code labels in the code set style. So these data points here, I just change the code set style so that none of that labels. And I go to just depending on my elevation off point codes. And I throw in mark subs wherever I want it to label, and make them do whatever I want.

Up and to the left over. Over and to the right. Whatever. Whatever. Whatever. So I've completely taken control of my labeling. And I don't care if the same points are being used in the same places, and I want to label it differently. I can do whatever I want.

And people really like mark point sub cause you're not digging into every single sub and adding codes and stuff. You just say I'm going to build a mark point that's got up and to the left. Select it. Copy. That one. That one. That one. That one. That one. And then, I got a mark

point sub that goes up and to the right.

In that one, I want to go there, there, there, there, there. And it's all good. I haven't touched anything in my section views yet. This is all in my assembly. This is all being driven by the data. So any corridor update I do, raise the grade three inches.

Extend that turn lane five feet. It's still keeping dynamic and live with that corridor, which is the whole reason why we're even thinking about doing this in the first place so that you don't actually have to touch those labels in the section views. Does that make sense? OK. All right.

Now there are still some changes you can make to individual labels. You're going to have to push the I believe button with me on this one because the controls for them are crazy. I find them crazy. But if you're doing this more than a handful of times, you might want to start thinking about one of the previous methods that I was talking about.

This is not what you want to be doing. What we're going to be doing here is we're going to be turning to off labels. We're going to be applying a drag state to a label. Or we're going to be changing the label style of a point.

It's doable. It works. You'll get exactly what you want. It's a labor intensive. And I can't give you specific numbers, but I will tell you I've seen drag on files that have a ton of these. This is memory Civil 3D does not want to be allocating.

Drag states are like-- what are the manual overrides in the parameters? Those are bad, evil, awful, awful too. But you don't want to do this if you don't have to. But it's absolutely in the toolbox for you to use.

So dragging an individual automatic label. This is probably the most common thing you want to do. This is where the I believe button comes in, OK. With nothing selected in your cross section file, you control click on that label. And it'll give you a grid. Don't tell me why. That's the only way you can grab one of these labels, OK.

But nothing selected. Control click. And it'll say, bing. I've got a selection on you. And you can drag it. Watch out for your oh snaps, OK. We do full grid on our cross section so it looks like an old grid sheet. Oh snaps in this environment is aah. It's all crazy.

So do a temporary override so that your label doesn't go zinging to the far edge of one of these thick lines, and you'll be a happy person. Now of course, if you want to change the style

of visibility or flip condition, that's a completely different selection method. It's not control click.

It's click on one of these labels, and everything will go dashed. You select the label set. And by magic, you right-click over the label you want to change, and you'll get a right-click menu for label properties, flip label, or reset label.

And it'll go to a property that you can change the default style, or do a flip, or a visibility. So if you have one label that you want, it's going down and to the right, and you want one label to go down and to the left. You click the set. You right-click. If you have those styles available, you just say down and to the right, or down and to the left, and it's done.

OK, you fixed it. Control-click selects one for a drag. Click on the whole set. And right-click over the one that you want to change to do visibility reset or flip. Now I'll give you a warning. A lot of times, you just want to turn one invisible, right?

Once you've turned it invisible, guess what? You can't grab it anymore. So you turn invisible and you, oh dang, I need that back. You can get it back. I've had a couple of calls with people in hot sweats because [INAUDIBLE]. These are the controls. And all this is in the handout.

My slide deck is pretty much verbatim from my handout. But it's a right-click on the corridor section properties and going into the codes tab. And you can do a reset on that section. So that will wipe out anything that you've changed.

So any drag states that you have, any visibility changes that you made, and any style changes you made, it'll set it back to default. And then, you can change it back. So it's not awesome. But if you need that label back, that's the way to get it.

Finally, in the manual section view labels. So this is a going off of the annotate tab like I showed earlier, or from the contextual ribbon if you've got a section view selected. You can add them like the controls that I just showed. You can do drag states, and style changes and that kind of stuff.

And they are dynamic to the section view. However, they aren't dynamic to the corridor. So what that means. Why this is still better than m text or a dimensional label is say you get rid of two sections. Well more likely, you add a section for a pipe or for whatever, a transition point, that you didn't have initially.

That's going to stack all your section views further down in the sheet and make more sheets.

And you got to add a layout and all that fun stuff. But these labels will still hold to the section view. So if you clicked on offset 12 elevation 990, it will stay there with that section view and ride with it.

It's not dynamic to the corridor. So if you click on a point and you change that elevation or you do a little tweak, it's not going to refer to that. And that can be really dangerous because if it's a small change and it, kind of, looks like the value is still good but it's really no longer any good, be very, very careful with these.

I would spend time getting comfortable with a filled out code set style and knowing how to operate that rather than doing these. If you absolutely need just a quick and dirty, they're there. But that's why I don't talk a whole lot about them. I don't know if filling out a survey for this class is awesome.

[LAUGHING]

They always tell us get them to fill out surveys. Get them to fill out surveys. Somehow I ended up right on time. Is there any questions that anybody has? (HIGH-PITCHED VOICE) Could you show us [INAUDIBLE]? No, I can't show you.

I do way more stuff than cross-sections. If you guys have any questions about anything that we've done or experiences that we've had, you want to come up and give me a card, or set something up this week, I'd be happy to do that. I have a huge thank you for all of you for sticking through this with me. I hope this was valuable to you. I appreciate your time and attention. Thank you.

[APPLAUSE]