ALI SHRIH: Good afternoon, everyone. My name is Ali Shrih. I work for a small company in Wisconsin, J3 Engineering. And what we do is precast engineering and drafting. So we deal with precast buildings and design.

And today, I'll talk about some of our process problems and things that we would run into using Revit, Robot, and how Dynamo may offer some solutions to the things we face during design or drafting of precast engineering buildings.

Today, we'll start talking about BIM in general, Robot Structural. Can we use it to do precast engineering, to what extent. And I'll do a very, very basic Dynamo introduction, how to use it in Revit, and how to connect to Robot.

Now, I've been to one of the Dynamo sessions today. I know it may get a little bit advanced and complicated. I'll start from the basics, and this was the intent of my presentation today.

So by the end of the session, we'll learn about how to extract an analytical model from Revit and to send the bars to Robot through Dynamo. It's not everything. We'll talk why, and we'll talk about how we do design using Robot, the problems we face, and what we can do, what we can't do. And again, we'll go over very basic Dynamo exercises.

In our work, I've looked up the attendance for today. I know some of you are internationals. The work we do mostly involves precast and mostly prestress. Members. I know on the international side, you may, for wall panels especially, you might use mild steel. But 90% of our wall panels are prestressed.

So usually, we use either Revit. We don't use Tekla in our company. Some of our clients use Tekla for BIM. And in a lot of cases, we just use AutoCAD to do the drafting for precast.

Why we haven't yet moved into Revit? There are some issues. I didn't want to fill up the page with the challenges we face. The main issue is our company and our clients need to invest an initial good amount of money and time into training and creating the foundation, the families, and all those stuff that will allow us to work with them and create their shop tickets and 3D models.

That's our biggest challenge. Because we're sub contractors and we work for multiple clients, each has their own specific hardware, ways of laying out panels, and all of that. If we want to adapt our BIM, every client, that will take us a lot of work. But currently, some of our clients are themselves adapting BIM, and they're creating their own standards.

Or also you may have been to other presentations and you've seen the add-ons and the tools. There are multiple tools that are available that will help us get Revit to be fully, completely useful tool in precast. But we're not yet there. Even with the addition of those tools, we bought some programs and we used them, so it's not completely there.

Whatever you are using, we probably either have seen, tried, or we have already in-house. And unfortunately, it's not yet there. But the tools that are available make some of parts of our work go faster.

Piece ticketing. Again, if there is a BIM requirement for a project and if they require a BIM model for coordination of our trades or [INAUDIBLE] with that thing, sometimes we do that. Then when we get to the piece ticketing in Revit, it's a little bit harder. It takes time. It's an excruciating, painful task if you're only using Revit only.

And sometimes our drafters will just take the 3D models, send them to AutoCAD, and do the ticketing there. But again, we use some of the tools available to create tickets in Revit, but we've still have not implemented Revit completely in our piece ticketing.

The integration structure analysis software, that's also not completely there. We will talk when I get to talking about Robot. It's a very good 3D fine-element software, but initially, Robot Millennium came from Europe, and when it was introduced to the US, there are few things that need to be adapted to our specifications. There is a prestressing in panels and all that. We can talk about that when we get to Robot.

Somehow our clients have already adapted 3D modeling in Revit. They have started that, they build their own hardware, and wall panels and process, and they're moving into 4D, and 5D and 6D. That's the product management, calculating materials and cost, and managing buildings after even the building is complete.

We know at least one of our clients who have by the end of the year, they'll done with 3D modeling. They offer us, their 3D hardware so that we can use them when we do our BIM stuff, they would require us to do BIM. And then they're planning in the next few years to use BIM to manage cost estimates and all of that, but they're not there yet.

I'm sure if any of you attend PCI, the Precast Concrete Institute committee meetings. We have a special committee to discuss BIM, promote it, encourage producers to use it, not specifically Revit or Tekla, not a certain program, but BIM in general, not only 3D modeling, BIM in general.

Because it will make our jobs go faster, less errors. If you've dealt with hardware and concrete and fixing an error, it's much harder, much more expensive when it's in the field. Fixing it in 3D, it's just moving something back to where it belongs.

The PCI committee for IBM, so far they've done only 3D modeling, and they used both Revit and Tekla. So it's 3D models of our common connections and DABS, [? NDABS, ?] and columns, and so on.

Like I said, this is a project I worked on. In 2014, we used Revit 2014. We put in the reinforcing and wall panel like that, the hardware. Now, it's a little bit easier dimensioning, if you don't use any plugins, all of that, creating sheets.

And it's hard, but when we do the job, all the plates, everything goes in where it should be. We can easily get counts, the [? bond ?] tables and everything out of our project. It can be done. We've done it, and others have done it too, but, again, Revit is not yet the perfect solution for precast.

A perfect solution will give us design, drafting, all just by clicking one button, and that's way ahead. We still are maybe 5, 10 years from that. And depends on development. I don't think Revit is specifically being developed for precast or prestressed engineering yet, so we have to buy the add-ons and the other software.

Our experience with Robot, before I start talking about Robot, last year, because we have the license and we have the [? prod, ?] we use AutoCAD, all the Autodesk programs. And Robot is a structure analysis adopted by them. We wanted to go over that route, and we did that. Most of our projects are either wall panel projects with beams or double T's or residential buildings where you have hollow core, planks, beams and columns, or precast, probably with prestressed, maybe mild steel, depends on the client.

And on top of the precast, you'll find the upper floor is mostly what? We get load map on the precast, so we have point loads, line loads, and area loads. And we get the plans, either in CAD or PDF. If you get them in PDF now, we just import in 2017 AutoCAD. You get the CAD, kind of.

When I tried to do that in 2017, you will get 15.99 dimensions, but it's close enough. It can put your grid lines and start from there.

Anyway, last year when we started our trials with Robot, came react. And I was told wait, there's a new, better software coming from Autodesk. Why learn Robot? Why you do anything in Robot because there is React, looks much nicer. There's a Ribbon in there, and that's very fantastic.

But if you look deeper into React, React is a better interface, something modern, and a much welcomed change from Robot. If you've used Robot, it's hard to find stuff. It's hard to get answers or results. There are multiple ways to get to certain things, but you get confused which way you should choose.

React is not complete yet. It uses the same background analysis, the same engine that Robot uses. And if you want to wait until React is done, you will miss out. If you start doing your work in Robot, the transition to React will be much, much easier.

You'll learn that there are bars. Red lines may be called axes. There are a few changes, differences between programs, but you'll learn the intricate stuff that-- you'll have some familiarity with the program. Or you can download the Technical Preview and try and use it, but that's not complete yet. I mean, we can not use it on a real project. My bosses will refuse, of course.

So for why while we stopped working on Robot, then we are back. I think, eventually, we'll move into React when the final version is released.

Now, in my opinion, we start with Robot. We continue with the Robot until React is released, if you want to wait. And then in React, if the Dynamo is fully implemented, there's a Dynamo embedded in the program, and you can easily program for React.

For Robot, they've released some libraries. They're releasing libraries for Dynamo. You can download those and add them to Dynamo. But those libraries, first are not complete. Second, I think, personal opinion, they'll go nowhere. By the time they're complete, React will be done, and we will move into to React.

But you can use them. They are available, and they use the same structure, same DLLs that are used by React. You'll see it's the same procedure to program for both programs.

Now, our work process in designing using Robot Structural. And why not wall panel projects? Because in Robot Structural, we don't have prestressing in wall panels. And most of our clients will use prestressed wall panels, so that's something we have not used it to design wall panels

There are a wall panel. We can use that in the few cases where we get mild steel wall panels. There's handling. It's good, it's usable, but so far we're still using BlackWall, if you've used that program, it's atrocious and old, but we are still using that program to design.

So I have a sample project that we used Robot in, and we start from the plans, beams, and the planks and loads. I don't think those are clear on them, but we have a load map over here, and we have that in CAD. What I do is I import the CAD drawings, and to Robot, that would be my overlay.

Then on that, I'll draw the columns, beams, walls, and you can trace all of that, and it's an easy process. Well now, the question is, are you using the Robot to do the analysis, then you're sending the results to Revit for creating the piece tickets, or are you just doing the analysis and creating the Revit tickets independently?

So would you start by creating the Revit model then sending that to Robot or doing Robot then-- that's a question, also, we have not yet had. I think the process should be from create the Revit model first and send the analytical model to Robot then send back the reinforcing to Revit. But if you're not using Revit, you can use CAD overlays, and trace your beams, wall panels, the walls, and the planks.

So what does Robot offer for precast design? For this project, we needed to use hollow core, and there's a predefined hollow-core section. You can do the one-way plank, basically. The way it works, you enter the properties of the hollow core and equivalent section. Rectangular section would be calculated. The properties of an equivalent section would be calculated for you.

So the hollow core is not prestressed. So if you're going to do a prestressed hollow-core design, what we're doing, we're using software given to us by one of our clients.

Beam sections and beam design, for our precast hollow core beam plant columns, we can have an IT section. You can have an L section by defining a slab on one or both sides of the beam. You can define a T section and give it an angle, 180 rotate the section.

Because that's a slab and a beam, that would produce not the right answers, I think. Someone from Robot here can correct me. So if you want to use an IT beam, define the slabs on both sides of the beam, and that will work just fine.

Unfortunately, if you're doing double T's, bolt T's, there are no predefined sections from the double T's, and we have to use other software that have-- it's easier to have predefine sections and just change a few numbers. And we have databases for clients. We can just pick from Client sections because they keep the sections for double T's the same, just go and pick them. Unfortunately, for beam design, the predefined sections are [? dangle ?] TI and go on the slabs. So that's a limitation on the program.

The program, a Robot will-- for beam sections, only bar sections-- you can have prestressings forces calculated, but not for plate, not for plank or wall panels. So you can have prestressing forces calculated for beam sections.

In our model and modeling only precast, the concrete, the way we get the deliverables that we get from our clients, they give us a few architectural drawings, structural drawings. And from those, we have the sheet that gives us the load map, usually, calculated by the EOR. And from that sheet, you can have that as an overlay, background. And you just click where there is a point load, draw a line load, and define your area loads.

That's one way of doing it. The problem with that, sometimes we have to do some manual work because the loads given by the EOR, most of the time are not very accurate, let's say, and that causes problems for us. When you have a load that's very big where it shouldn't or very small where we think it should, we have to go back and verify the loads by ourselves or ask our intern to do that, then enter the loads manually.

If you don't want to do that, you can do the same thing for the wood framing. You can add wood framing on top of the precast and let Revit do the calculations for you. It's just drawing the wood framing and adding the loads on the wood framing. And that, it can do, actually, pretty well.

Usually we don't design the wood framing. This is why we verify our loads and put them on the precast directly. If you are someone who can draft it for you, just drafting, you don't have to be an engineer to do that.

If you have attended the Dynamo session this morning, they talked about wind loading in Robot. The wind load generation in Robot, it's very good. It looks fantastic. You must have seen pictures of that.

But that wind loads, it's the load created through a CFD analysis. It's the load as if your building is in a wind tunnel and you get your loads on it, it's not by the ASC 710 or 705. And what they did in the morning session, they calculated those in Excel, then they'd send them back to Robot. You can check that out.

But although there is wind generation, they've done that through Dynamo and Excel. And things got complicated in Dynamo, but you can check that out. It's a very good session.

So customizations for precast. For steal input, you can do handling for beams. You can determine where those loops or bars are. Here's where we do LOR 5. It default in this file as LOR 4.

But we usually do LOR 5, and can find that in the reinforcing pattern. And you can also define, you can choose that your beam is a precast beam, and the supports for that beam is so and so, and this will help when you get your results for that beam.

The program is very powerful. It will give you all that bells and whistles of the CFD, FEA analysis. You get moments, stresses, deflexions. You can look at those, and you can print whatever you want. There's a report. This is part of the report on the left side. Reports are detailed and they're good. You get a lot of information out of those.

And for your beam design, it's not an exactly-- you can't use this as a piece ticket, but it calculates points of steal, lengths, and Rebar. And you can send that to AutoCAD-- it's a 3D model-- or you can send it to Revit. If you're doing engineering only, then this would be our engineering markup, and we give it to the drafter so that they would know how many bars, how many stirrups, and so on, and so on.

So we can use this for beams or in the case of columns. Also very basic, it's like our engineering markup, we design columns, we show the bars, the stirrups, give that to a drafter, and they will do the actual piece ticket based on whatever our client require and their specifications.

So Robot has some capability to do precast engineering. It has its own limitations. We cannot still use it for everything we want to, but it can make our job easier and a bigger job that have beams and columns and planks.

There is a link. You don't have to program for it, but there is an add-on in Revit that you can create your analytical model, adding loads, and then send that to Robot, and then Robot will send back the steer in forcing. That link works somewhat well, but it's not perfect either.

Because we're in precast, it's rare to have a rectangular section. We have maybe one because we have a garage door beam, but everything else would be ledger beam or an IT beam. The reinforcing you get back into Revit from Robot. In many cases, we had issues with getting reinforcing back in the Custom section.

Now, what we do if you have a problem, your software is not perfect for it, and that happens with every software, even with CLL and we have issues, we go to VBA or our own code and fix it.

Using APIs and Dynamo, to me, I think if you want to use Dynamo and structure analysis or in drafting, I have a background in programming. if things get very complicated, then I would prefer to go back and write code for it.

If it's a few nodes or something that I can manage in Dynamo, and Dynamo is a very good solution, easy to understand, easy to build, and you have a graphical interface, and that works also just fine.

So what is an API? Simply put, if you are a programmer, you write some code. That code would be something to draw a line. Then you write another code to draw a line in 3D, a circle, and polygons, and all that. You put all that in a file. Those files are called DLLs, Dynamic Link Libraries.

The Dynamic Link Libraries, those are files that contain code you are allowed to use. How we are allowed to use? By using that public work. At the beginning, if it's private, you can't use it. You must be allowed to use. This is why, for Robot, they are releasing those DLLs.

Some of the stuff you can use, others you can't. While you can't, probably you crash a program or the system. They have not tested those yet. So they write some code, put it in a Dynamic Link Library. In Dynamo, it is called in libraries. From that library, when I use the code to draw a 2D line, you just draw a 2D line, give it the two coordinates, and then you draw the line.

So Dynamo, which is based on visual programming, Dynamo is a program that will show you the contents of those DLLs, the libraries, in a visual way, in what they call nodes. In order to be able to see the nodes or the code and the DLL, Dynamo must know where that DLL is.

So if Dynamo does not have access to the DLLs of Robot, you will not be able to see the nodes for Robot. Same thing for Revit, same thing for React, when you install Revit, when you install React, those will add their nodes automatically to Dynamo, the button you click in the Ribbon. But for Robot, you need to install a package in Dynamo, and I will show you where that is.

And the way it works, every node is either-- think of it as either a container of data. The first three nodes, those are just boxes with either a number or some text or some data. And the node in the middle, point by coordinates, that just hides whatever code behind, and it will do something. What will it do? It will add a point if you give it the coordinates.

Every node has ports. These are the ports, either output or input ports. And in order to send the data to or from a node, you need to connect a wire to the appropriate port. So I want to draw a point. I need an xyz coordinates, So I need to connect wires from the output port to the input port of each of those nodes.

Then from that point, you can draw something. I just added the watch that will show you what's in, what the output is. And then you take the output, show it, or plot it, or do something with it. Every node hides the code behind it so that you won't have to see it. You won't have to be bothered by it.

If you see in the previous slide, we use Excel a lot in our company, and I use VBA a lot in our company. That code on the right side matches what those notes do. And if you are not used to using code, it's boring, it's intimidating, and it's a lot of typing. So if you don't want to do that, the nodes are for you. The nodes hide that ugly code.

Why do I use code? Because I'm used to doing it and because my Excel sheet that I'm using that code in, it already has a lot of VBA. We use it to fix a lot of things, and I don't have to rely on Dynamo. Dynamo 0.9 does not use Robot yet. You have to use Dynamo 1.0, which runs in Revit. It's complicated, so I'll use code.

I'll run through a couple of examples on how to use Dynamo. To start Dynamo from Revit, you need to go to the Manage tab and hit the Dynamo button. The 0.9 version is the one that came with our 2017 package. That's something I installed recently. They released that after we installed the 2017 version, and I'm using the 1.2 Dynamo.

The main graphical part in Dynamo, that's the workspace. That's where you add your nodes, and in the background, you can see a 3D model of your lines, node points, and surfaces. On the left side, you have the library, which is a list of libraries that are added or that Dynamo can read.

Because I ran it from Revit, the Revit libraries, or DLLs, are already embedded in there, so you can click on them and find whatever you want to use. It's easier said than done. If you try and look into these, you'll find a lot of nodes to work with and which one you want to use. The question is always, where to start. I know we've talked about that.

The Robot is not automatically added to Dynamo, but you need to add a package. In order to add a package, you install a package, search for it. I've already installed it. It's called Structural Analysis for Dynamo. This is a beta version, for release recently 2. This is under limited license that it will expire by the end of the year.

So Dynamo for the Robot, the construction analysis, you need to add that as a package. Go to Packages, search for a package. You type in the package you want to-- there are custom packages that access Excel files, custom packages that people have developed.

Today we use the package that was developed for the wind load that reads the wind loads from the Excel files. You can search for it here, and add it. But lots of these libraries are free.

So I told Dynamo where the DLLs for Robot are, and I told Dynamo how to access Revit. So if I want to create something, if I want to add instant, if you use Revit, you know that you have families and you add instances for certain type. So if I want to create a column, if I want to add an instant of a column and I'm not sure where to start, the best way is to search for a node.

You can search for a node in the Search box over here. So if I am looking to add an instance, I'll start typing instance in here. And you can see multiple nodes up here. There's description. With time, you'll learn what these do. But the plus sign means that it will add something. The question mark that will look up something, read, query, something for you, and the red lightning bolt, that will do something for you.

I want to add an instance of a family by defining its coordinates, and that would be another way to search, and this is a little bit faster, is to right click in the empty workspace area and type my family instance by point. So I want to create an instance of a family by defining a point.

I have ports and input ports and output ports. If I want to create that instance of a family, I need to add a family type. I need to add a point to create that family type. To add a point, I'll search for point, and you can add a point by Cartesian coordinates, let's say by coordinates.

So I'm just trying to figure out what the input port is required and just looking up the word point. And to create the wire, to wire the output port to the input port, you just drag and drop. So that's the wire.

If you don't give x, y, and z, then those will have a value of 0. What's the output that you get? You can add the watch, and sometimes I would add a watch node.

The watch node will show you what the output is. The output is a point that has a coordinate that has the coordinates x, y, and z, whatever those coordinates are. Or you can click, pin This and that's the same as using a watch node.

Now, I need to define a family type. The family type-- I'll go the family type, I'll search family types, and from My File, I have Multiple Family types. Say I'll create a column. I have a 12 by 18. That's my family type and then add that.

The columns, when you add an instance of a column, you define two points, bottom and top, and you have to define a level or you add it. So it's saying, I can do what you're asking because you're creating a family instance from a family type and a point. I need to create it from a family instance, a point, and a level. So I'll change that to family instance. by point and level.

I need to add a level, so I'll search for levels. In the background, your column now is created, and if you go back to Revit, I have added a column in Revit. It's much easier to go in and insert the column using the Revit end phrase.

And this was a question that was asked today in the first session. Why do I have to go into coding in order to draw something that I can just draw? There are windows. There is coding behind the same code that I'm using. Why would I have to go and code it?

If you're doing something repeatedly, if you have repetition, and our projects have that kind of repetition of building, define the distances between active grid lines and your columns are created, beams in between those, if you have repetition programming works, if you don't have that kind of repetition, you actually don't have to do the programming part. But I know that you can do that easily in Revit.

So this is a very basic column that is located at 0, 0. If you want to change the values of the x and y, you can add a value for these here by adding. You can insert a number node, or you can insert the code block. To add a code block, which is easier, this is why I use it, you can just double click and create a code block.

And the code block, you can just put a number or string, so 10 and that's x; 20, and that's connected to y. And I have moved the column to a different location, where x ix 10 and y is 20.

That's a very basic how to start where. If you want to start with programming, you need to have used Revit before, you need to have used Robot before because the questions that will come into your mind, I want to insert an instance. Then the key word is instance. You go and search for instance, and you add the node to add an instance.

And I made a mistake. I forgot. I'm adding a column, then I'm looking for instant by point and a level. Search for instance again and get the one that works for you.

Now, I'll do a little bit more advanced example. Because I added the column, I'll just reopen this.

In this example, I have a model that I created in Revit, an analytical model with loads, and I define find [? light ?] load, dead load, and beams and columns. You can use the Analyze tab, Robot Structural, and Robot Structure link, which will transfer the data for you to Robot.

If you run into problems using the link and if you want to intercept some data going back or coming from Robot-- and again, I would recommend doing this for using Dynamo for smaller programs. If it gets too complicated, I would prefer to use coding. But you can simplify Dynamo programs by using custom nodes.

So if you write something, make it a custom node, then it becomes smaller and easier to manage. But I would still prefer to use coding. And if you're using Dynamo, you may have to use code blocks like this one. And code blocks are actually, you're doing coding in Python.

If you've done coding in Visual Basic, Visual Basic for Application, or C#, the principles are the same. You still have the semicolon at the end just like in C#, but there are a few differences.

You can use it for your iterations, very few differences. And if you're going to use Dynamo to it's fullest, you may need to learn a little bit about Python.

So the way this works-- and this is the one that's equivalent to the code in VBA, I wanted to send the analytical model bars to Robot. So I linked Robot and I added with the analytical bars, and need to get the analytical bars based on element type. I chose the analytical model, stick model, so that leaves out the plates, all the elements in the stick model take those elements, and there is this step.

When you're dealing with an analytical model in Revit, the analytical model is actually made up, if you see the colors, there's green, orange, then red. Those are three segments-- start, middle, and end.

Those segments are bending lines when you send them between Robot, and you need to merge those lines into one curve, so you get one bar for each beam. And this code will turn those into one line.

The code uses lists or raise, if you've done programming before. The list goes into geometry. I had to add, also-- Robot uses, by default, SI units. We use [? VS ?] customer unit. You have always to do converging of units when you are sending anything to Robot or when you are taking, you're receiving anything.

So this will do a conversion the length is being converted I use the conversion factor. This is meters to feet. So I added the scaling node to change the output length-- divide it, multiply it by a number. And that goes to an analytical bar. The analytical bar goes to Robot, and then I also had to assign the cross sections to those bars.

And if we run the code, I have watch nodes just to look at the contents of the lists I had. But if you want, you can look in the background and see your structure or going to Robot and look at the result. I would do that only-- why did I need to do the code in VBA? The reason is because we can't use Robot in all of our projects.

This is a very old method that was based on a very old Excel sheet where we traced the loads and beams and put them in tables, then those tables would be transferred into point mine loads. We calculate moments. We designed the beams then we get loads to columns. Then we get them to beam ledges.

So the sheet actually connects also our other programs, can read the results from our other programs, not only Robot. So we have multiple stuff going in the background in VBA, and I needed to intercept or send data to Robot. That's why I needed to use the code. You can use Dynamo with Excel, but because I have a lot of VBA in the Excel sheet already and you can't prime point with the Robot code, so I have the VBA in Excel. So that's why I use it.

And I have not done any really complicated Dynamo solutions. Mostly the size grows bigger, and if I need to use a lot of code blocks, then I'll just use code. So do you guys have any questions, any comments?

You can rate the session and-- yes.

AUDIENCE: So you're talking about you can do pre-compression for the beams in Robot?

ALI SHRIH: Yes.

AUDIENCE: All right. Does that take into account the harping or drape for the tendons or it just a pre--

ALI SHRIH: Yes, you can do that, but I believe the losses are not included. They were not included the last time I checked. If you do prestressing losses, that's not included yet.

AUDIENCE: And also would it be possible when you kick Robot back out to Revit, it includes, I'm assuming, end reactions?

ALI SHRIH: The prestressing strand is not modeled as a strand.

AUDIENCE: Not as much the tendons, but just the gravity loads. Does it bring out the gravity loads?

ALI SHRIH: Yes.

AUDIENCE: Could use Dynamo to, essentially, use a smart connection location so it knows what your reaction is and it can grab from a library?

ALI SHRIH: Yes. Exactly. For that, if you have something in an Excel sheet to design the connections, you can just grab the load from Revit because you get it back from Robot, grab the load, use the Excel sheet to design the connection, and that would be a very good use for Dynamo.

Grab just a piece of information, and use that in designing a connection. That's also one of the things we use it for. Any questions?

Well, thank you all, and hope you enjoyed your time at AU.