NATHAN KING: OK, I guess we'll go ahead and get started. I'm Nathan King. I work with Autodesk, developing design robotics labs around the world, most recently the Boston BUILD space, which is a new flexible, AEC research lab dealing with kind of automated tools, CNC, robotics, and so on. Also do a lot of other consulting work.

The goal of the course-- it's really to get-- we get a lot of questions about how do you even enter the kind of space of robotics if you're not an industrial engineering person or working in factories. How can we allow design robotics to actually expand a little bit, and with firms, practices, universities, to purchase machines? So I wanted to put together the class that would go through the kind of wide range of things that have to be considered, and try to get into some of the pragmatics of actually executing procurement and commissioning of a machine.

Two work cells that we set up for AU are out on the exhibit hall. These are both kind of a result of some work at Pier 9. These guys are more like an industrial robotic work cell, in that their single purpose, built for specific application, specific robot, limited flexibility. But they have all the kind of requisite safety pieces in there. So the things we'll go over today, they're exhibited there as well. So there's kind of a tangible piece of it.

Designer robotics is a little bit different than industrial robotics, and we'll get through that in a moment. But more flexibility and creates more risk. Also creates more kind of inability to communicate with the standard workflow, in which you would normally have an integrator that would set up your system for a purpose--built machine. And we need actually flexible systems that allow you to do whatever kind of work you need to do for a given research project or low-volume production.

So we'll get through some of these things during the talk, but essentially-- Starting out with getting the robot, there's the commissioning phase, which is installation. And this is involve setting up the cell, tooling an environment, which is the actual thing that goes on the end of the arm, the work-- the end effector of the work object and other things you'll be operating on. And then there's the operation of the machine, which we'll talk about.

Design robotics, as compared to industrial robotics, is again, this kind of flexible platform. Where industrial robots we have an automated control, reprogrammable machine. Typically this kind of generic-- this is a six-axis manipulator. It's kind of the common thing. When we say robot in this context, we're talking about that thing.

Other CNC machines, 3D printers, FDN machines, these would actually all be robots as well. But when we say robot, we're talking about this industrial robot. Designer robotics is kind of set up and emerged out of the academic setting, where we have researchers that are allowed to adapt new tools based on a lower cost.

So around 2006, 2007, 2008, cost of the robot started to drop significantly. That's also coupled with a few key developments in software to allow us to program the robots directly out of the design software. So there's a number of things that can be used for that, which we'll go through. These two things combined kind of allowed a lot of academic institutions to purchase robots rather than a single-purpose CNC machine to do-- you know we might spend several hundred thousand dollars on a heavy duty milling machine or much less money on a robot that can actually do milling and a bunch of other projects as well.

Six-axis robot has six joints. Starting at the base, moving up, we have joints one, two, three, four, all the way to the end is six axis. All of these things play a role in positioning the tool, also how the kind of configuration works for the machine. There are external axes, there are seven-axis robots and so on, which kind of add to complexity.

We have new robots that are coming online, like this universal robot here. And a number of other manufacturers are making these, they're collaborative robots. What this means is that, kind of, human robot collaboration, interaction between the user and the robot, where typically, if we're operating under kind of a safety protocol, we're not in the same environment that the robot is in. We're outside. We're away. We're protected by guarding and so on, with the Universals and other groups that are making [INAUDIBLE] got a few.

There's a number of these collaborative robots coming on online that the user of the robot can actually work together and still meet safety requirements. So this has a lot of new potential, we hope, for kind of, the AEC research side of things. And on one hand we talk about this is a research, kind of a pseudo-scientific experiment, which is a conversation we've had a lot during AU, about what are these things are research based and what are kind of production. Or what type of users are we?

And in all cases, it seems like we're moving out of what is typical of an industrial robot, where we have a machine, preprogrammed to do the same thing over and over again, thousands of times, throughout its lifetime. And then, it de-commissions, another robot goes in its place, and continues to install an engine block in a truck, for example.

We're using the robot, in the field of design robotics, using the robot-- it might do thousands of different movements on a given task. That task might change every day, every week, every month. So we have to look at kind of flexible ways to do it. Collaborative robots have this potential to allow this interaction between the user, which is great for the facilitation of research.

It's still kind of new. And everything's kind of scrambling to catch up with the collaborative robotics human robot collaboration. Another thing we can talk about are robot-to-robot collaboration. And this is simply two or more robots working together for a single task.

Systems like MultiMove, which is an ABV product. Other manufacturers also have similar systems, whereby two robots are synchronized through the controller. They have dominant computer systems to allow actual synchronization. Other multi-robot operations might have two different robots, two different controllers, two different programs that are sequentially operating and not necessarily synchronized.

Another piece of the kind of flexible robotic platform are tool changers. And tool changer adds a little bit of complexity to the system. And in a typical application, you might have the same tool, like a gripper. In the example of the engine block, you'll have a gripper that can hold the engine block. That gripper is installed in the robot. It never has to come off. It stays on there for the lifetime of the project or that product.

But to allow some flexible applications, there's a lot of product available that is tool changer. So any tool that you build as a fitting. That fitting would interact with one of these types of tool changers to allow you to switch out tools rapidly. Most academic settings we use these things to allow us to switch from a hot wire to a milling spindle, you know, to any other kind of end-of-arm tooling that might be designed or developed.

It also could be as in industrial setting. For example, a milling versus a sawing end effector so you can switch during process between the two. These are available from a lot of manufacturers. They add costs and complexity. So the simplest is a manual tool changer that has just a physical interlock. You actually have to touch the tool to change it. That's fine from a small robot, small tools. Things are happening here in process.

And then the automated systems, like these pneumatic ones. They range in complexity from something very simple to something that has all of the kind of electronics, pneumatics, control system, sensor feedback, and so on. So it can go from very cheap, very easy to use to very complicated, very complex, and expensive.

Other things we consider with the robot. Set up or cable routing. So it's typical to get pushed into a lot of features. And what ends up happening is we have cables routing to the end of the arm. They carry the power supply. In this case, there's a milling spindle on the arm. They carry the new pneumatics and so on. And so, as you increase complexity in the system, you increase the amount of cables and kind of junk you have to deal with on the robot.

So on one hand, this adds functionality. On the other hand, it makes things a little more complicated for use as an application. Generally the robots come with some cable routing up to the third joint. And then you have to kind of user-supplied routing to the end of the arm. So all the way out to a joint six is usually something that we have to add a dress pack or some additional routing.

Other features of the robot that are kind of critical for operation, or some safety features, emergency stops, and in this case, motors-on indicators. So in that robot cell, when we're working with a robot, and we have people actually interacting, or at least in the same vicinity of the machines or know when the machine's actually going to run.

So they're relatively quiet. You don't really know when the motors are on or motors are off. This is true if you have a small robot on a desktop or a massive robot on a factory floor. Typically there is some kind of indication that shows that the robot is on or off. So you notice the robots up in the exhibit hall, they'll have some kind of flashing indicator here when the power is on. When the robot's actually moving, these things will be solid.

This is true of almost all robots that are kind of in the category of industrial robotics. Typically robots have brake release. So there are buttons on the machine. In this case, it's a very small robot, on the controller. Press this button, it disables all the brakes on the machine, so we can reconfigure, calibrate.

It also means that there is no more support for that joint. So on the big robot, like the one that's up in the exhibition hall, if you were to press the brake release, the robot would just collapse on the ground. So from a safety standpoint, knowing where the brake release is is important. From a user interaction standpoint, it's not something that we typically want to deal with very often.

There's a lot of application-specific considerations for the robot. So if you want to buy a robot that is, say for a university, where you might have faculty or students who want to one day pick and place tiles. On the other day mill, and on Wednesday, cut something with a hot wire butter. You've got to consider this. These are application specific. Each one of these things have a specific kind of duty and requisite protocol.

But on other hand, you might have a factory where you want-- or low-volume production facility where you want milling, and that's all you're going to do. So in that case, you buy purpose-built tooling for milling. And you don't have to worry so much about flexible operation.

Payload and reach are two of the parameters for kind of the robot. They're kind of the driving parameters. If we get into the weeds a little bit, the accuracy of the machine, and these other-- all robot manufacturers have different features. But the big thing we consider is reach and payload.

Maximum reach is essentially exactly as it implies. The robot can reach a certain distance. This is typically some kind of geometry like this. It's not really hemispherical. It's more of kind of a torus that has some blind spots. Some robots can rotate kind of indefinitely around and around, but a lot of robots have a blind spot down at the back, where the kind of cabling comes in and things like that.

Robot reach does not mean that you're definitely going to be able to operate on an artifact at that given reach. So you can imagine trying to reach on the table, reaching over something to actually grab something on the other side of it. You actually have to move everything closer. Just because it has a four-meter reach doesn't mean you can actually operate on a full four-meter working area.

Payload-- skipped over a little bit. But this is again, another, the other parameter that's kind of critical within the thing. You can get a robot that has a 0.3 kilogram payload, all the way up to three 500 kilograms and ranges in between. Typically there are multiple variants within payload and reach, so same robot and model-- or same family of robots will have a long reach, low payload. Short reach, high payload.

So there's a little bit of a trade-off. The longer the robot reach, typically the lower the payload. And vice versa. The shorter each, big robot has really high payload, but has a very short reach. Other considerations in buying the robot are the mounting protocol. And this is a diagram for a big robot. Storing massive bolts, 12-inch anchors, you know, anchoring into the slab.

The robot companies will provide a specification for that anchor. But they don't really support the engineering of that piece. So there's usually an extra bid involved in actually making sure your slab is compliant, your anchoring strategy is correct, and so on. And there's-- it's obviously kind of a professional engineering situation, but in many cases, this is not actually done. So if you know you're going to get a robot that's going to be table mounted, there's a little less concern as long as the thing is rigid.

But as you get to the larger robots, serious anchoring considerations are in play. You can tell it's-- down there in the exhibit hall, the huge steel bases that are there. Those are kind of bare minimum, even though they look giantly overbuilt. They're kind of the bare minimum to meet the tipping requirements for those robots.

So as you go from small robot, which is on tabletop or some other cart or something, to the large robots, the cost of kind of anchoring and fixing things goes up. So the higher payload, the more expensive the tool changers are. The More complicated it is to mount. The more things you have to consider beyond just bolting to a table.

This is also-- usually these recommendations are calculated at full reach, full tangential velocity, and a full emergency stop carrying full payload. So if you have a robot that has a 500 kilogram payload using 2000 millimeters a second, and you slam on the brakes, it's going to have a totally different kind of behavior than if you're moving, like the robots in the exhibit hall, where 250 millimeters a second, moving very slowly to do things. So it's not a catastrophic issue if you slam on the brakes. But in other situations, it is. So there's a trade-off between meeting the maximum requirements of the robot versus actually making it reasonable for you to have that robot in your facility or your shop.

So mounting can go in a number of ways. Do a lot of work with these small robots, these can be basically unbolted by one person, moved to a different location. And this is where the $50K or less comes in. We're talking about robots in this range, with all the requisite hardware and tooling.

But the small robots can actually be used to do large things. In the research sector, and the kind of academic area, we're working a lot of these machines because you can have basically more for less money. So we can have six or seven different student groups working on different robots. And the consequences, while they're still very high from a safety standpoint, failure on the smaller robot is different than failure on the 300-kilogram, four-meter reach robot.

So we can work a lot in the academic setting with the smaller robots. Totally non-- in this case, it's a six-axis machine being used for what could be a two-axis positioning system. If we have time, I'll go through this example later on in the talk.

Another bit of thing, and it's really important to consider when purchasing a machine, is really the end-of-arm tooling or the end effector. So the payload of the robot we mentioned, say it's 100 kilograms. That machine-- the end effector [INAUDIBLE] load, plus whatever. In this case, we're printing ceramic, or have clay inside this tube.

So this a 40 kilogram payload and a really grainy image. My apologies there. This thing, you know, this might weigh 10 kilograms, and then it has 30 kilograms of clay inside of it. So the aggregate of the tool plus the material, plus all this other junk is all what comes into the payload. So a three kilogram payload, which is a smaller robot, and this image, this has a three kilogram payload, much lower on capacity than the 40, but still when you start to aggregate all these things on the end, we've got to reduce payload.

Other tooling, tooling conversation with the end of arm tooling or end effector, here this actually has automated tool changer so you can see this disk that mounts this bit of hardware on. This is a project developed [INAUDIBLE] pull the machine apart so you're going to cut some building facade. So there's not a robot tool to pilot custom building sheet metal facade.

So there's not somewhere you can go to buy this tool. So we have to develop the tool in place. And this is something where, on one end, kind of a smaller production, or with student activities, they're going to work with 3D printers, you're going to work with small scale kind of crude fabrication techniques to make the end of arm tooling. And that's why I think it makes it very valuable for academic settings because you have to make the tool, In a production setting, you could be a little more stringent with building out the tool. But that's the opportunity [INAUDIBLE] of the robot is that [INAUDIBLE] dumb arm. And you have to develop the tool [INAUDIBLE].

Some tools are available standard milling. [INAUDIBLE] common, you can buy this off the shelf, add it to your robot. Most other tools are custom. Grippers and things like that are available. Things like this are custom in this case the sheet metal has to fold. It's bending about this axis here it's because a laser cut, a laser cut, and then the little tabs that are remaining. Each one is folding at a prescribed angle, and when you bend this thing, typically the sheet metal deforms or oil cans. So the tool has to address all those material properties or issues within the process.

So there's a section grouper on one side, so the part that's raising up is actually being pulled through the suction grouper. The part that is getting pulled down, so the bottom here, is getting pushed with this. So you get uniform bending across this axis. This thing is connected to a pneumatic, there's pneumatic lines running through the tool changer. [INAUDIBLE] interior device that creates suction so air blows in, suction comes out, and then we have this tool to make the facade.

Other ways to figure out the tooling, you can actually put the tool on the table. So you want to build a small robot. We want to [INAUDIBLE] something. [INAUDIBLE] I actually work with a small robot, a very tiny one [INAUDIBLE]. It's useful for some things, not for others. So there are strategies in which you can design the tool to mount on the table still using a small robot. But, in that case, the small robot's holding a work object instead of a tool. So in most applications, the robot holds the tool, and the work object or the thing that you're operating on is on the ground table some other apparatus or thing But you can also flip that around, and have a tool mounted somewhere else.

So some desire [INAUDIBLE] tooling. This is-- as you're buying tools that are available from [INAUDIBLE] anyone else that makes pneumatic grippers or any kind of grippers, these things are already kind of pre-engineered. Plus we're going to actually have to develop [INAUDIBLE] tooling yourself. So there's a critical juncture where the tool mounts to the robot. This is where the tool changer comes into play for whatever prescribed goal. [INAUDIBLE] a lot of accidents happen where the loose parts fall of the custom tool. So making sure there are not a lot of loose parts on the tool. [INAUDIBLE]

And the guarding solution, which is the next part of [INAUDIBLE] talk about. Creating a digital model for the [INAUDIBLE] simulation. And then make sure at the end of August consider the perimeter guarding solution. So we have a small robot with a one-meter reach. But if that has a 30-meter tool at the end of it, we've got to obviously make [INAUDIBLE]. It's counter-intuitive, totally intuitive, counter, but often forgotten when setting these things up.

The other piece of the robot itself is a robot controller. These are included with the robots. These things can train. The computers control all the motors on the robot, whether it's a [INAUDIBLE], along with a series of other control mechanisms. [INAUDIBLE] there's a number of IOs-- inputs, outputs-- that send control signals to the end of arm tooling. So this is something to know when you're going into buying a machine. Features like-- controller features like inputs and outputs are important.

All robots, for the most part, I can't remember not ever seeing one that doesn't come with some level base IO meter 16 IOs, typically 24-volt signal that can send command lines. For command lines, you can send information to the end of arm tooling. On/off digital switches and an analog signal. But having a little bit of foresight saves a lot of money up front. So if you put these things into the controller ahead of time saves you exponentially. Because if you buy it after the fact, it costs a lot more. Controllers are very-- they're robot specific. Even if you have two identical models of robot, you have two different controllers, you might have slight different configurations.

So you have multiple robots. You can't necessarily interchange the control system. These are all features that are available on the controller. This is an ABB controller, but all robot controllers for an industrial setting have a very similar setup in that we have an on/off switch [INAUDIBLE] operational protocol [INAUDIBLE] talk about. Emergency stop, the motor's on indicator, some kind of keyed mode selector. So talk about operational mode [INAUDIBLE], and some kind of access to the inside of the cabinet. Features that are worth adding onto the robot are external data, or PC interface.

So this is a service port. It allows to plug an ethernet cable in and have access to the controller while using the machine. But there's another option [INAUDIBLE] that actually gives a real iterative feedback between the controller and the [INAUDIBLE]. So that [INAUDIBLE] the interface. For almost all manufacturers, they offer something like that. If you check inside the controller, it's a jumble of wires. There's a lot of features on the door and some IOs that users might actually need you to access. But this is typically-- once it's set up, it's locked down, you don't have to go into it ever again.

This is something that presents another hazard. So [INAUDIBLE] machine. [INAUDIBLE] some protocol [INAUDIBLE] static eliminators and other aspects. But going into the controllers, usually a technician or an operator that really knows what they're doing, not necessarily the flexible users, but you kind of see here on the bottom right the little green inputs are all places that you can customize things by adding additional IOs connecting to tools. So the example for the suction gripper that I showed you before, it would have a wire going from here to the base of the robot, up the robot arm, probably internally, that then tells the server to turn that vacuum on or off [INAUDIBLE] solenoid.

Ready to roll? All right, so last piece of the tool is the teach pendant or the flex pendant. This is the thing that will come with the robot. Most robots, this comes with it. Flex pendant allows you to control the robot via these joysticks. This is for calibration positioning. It's also your interface under the controller, so loading software, loading programs, setting up different IOs. These are different for every machine, so this one at the top's an ABB. The one on the bottom is a KUKA. Even within a controller, five robots with five of these teach pendant might have a slightly different configuration.

So once you are in one, it doesn't necessarily mean you can directly translate to the others. But very straightforward in that this is the device that gives you access to the robot. So this is where we're loading files, it's where we're programming IOs, setups, inputs, and outputs. You can also set up logins on these things so your students or your workers can access certain things, but then the experts in the office in the company can access more advanced features. Not really a lot more say about these.

Lastly, area guarding, which is something that on the university setting we've ignored for the past decade, and now we're getting more. As more users are using more robots in the university setting, I'm going to get more stringent about understanding as an academic community what gardening is all about, what's safe operating protocol. If you use the robots in an industrial setting, you're underneath the ANSI standard, and the guarding is really a critical factor. So it has to be budgeted. It's also within that 50k entry price point that's mentioned in the title of the course.

Different types of guarding, physical guarding is what is done in the exhibit hall. Physical guarding is usually required any time you're cutting anything. So if you're milling, cutting with a saw blade, a chain saw, a weed eater, whatever you have on the end of the robot, you got to be inside of a plastic blast shield. So this is a plastic shield, polycarbonate, or a mesh gate, or something that-- if it's milling, we generally want a polycar barrier. Other things, you can have a mesh. There's plenty of companies that provide these things as prepackaged systems.

Things to note about this, all the emergency stops have to be outside, which makes sense. If you have the thing inside, you have an emergency, you don't have to go into the cell, turning the emergency to turn off the robot. So you have safety e-stops outside of the machine. Also, these doors have to have an interlock. So that means if somebody opens it, the robot will shut down. And our factory setting, where we have the robot doing the same thing more or less all the time, this is also wired into the end of arm tooling.

So if you have a chainsaw on the end of the robot, open the door, the robot stops and the end of factor or the chainsaw stops. That's not necessarily guaranteed in the more flexible situations where you might have people designing tools in a daily or weekly basis. So physical barrier, pretty straightforward. That's all we needed to do usually if you're cutting or have a high incidence of things breaking, smashing, cutting, and these kind of processes.

Simpler and less invasive solutions are light curtains. This is the next level in simplicity. The physical putting a physical barrier around is very straightforward. Light curtains are the next level, also very simple. In this example, we have three mirrors, one transmitter. So this will send a light all the way around to this receiver. And as long as that chain is complete, robot's operational. And someone steps in between this or breaks the laser, the light, the robot shuts down because the chain is incomplete.

There are things to consider with these mirrors and there's a certain distance you can reflect the thing. So you have a very big work cell. You might end up with more than one system, more than one light current. So in this case, you could have a transmitter and a receiver here and one here to extend the work area there. It's around 48 feet is the biggest that any of these will really translate with the reflectors. Next level in complexity of the white curtain is also governed by some resolution.

So if you're in a process where close to a machine, a small machining center or something, you need a high resolution because the user is getting pretty close, and if they stick their finger in there, it's going to be a safety hazard, so you can choose a resolution between the very low res, which you can still reach your arm through this guard, but you couldn't pass your body through. And in this case, you can't cat reach your arm in.

Like current resolution, something to consider when behind the light curtain, for typically for the flexible applications, we say go high res so you're protected no matter what process is being done. I keep getting caught here. Next level in complexity, area laser scanners. And then finally, there's a vision system-based safety piece, which we won't go into. It's a little bit more complex than I think is necessary for most applications that we have.

So the laser scanners can be programmed so your operator could actually be part of the cell, and again in time during the process. So the robot might be picking something up over here and putting it down over here. And when it puts it down, you need to go in, and remove that thing, and take it out of the cell. You can actually set it up so the user can be in that area when the robot's not. So these add a lot more opportunities for interaction. But their added complexity in terms of programming and setup.

Essentially, the light curtain projects a little bit beyond a 180 degree arc. And you can usually-- not all, but some-- most have zone definition. So in this case, when someone passes into it, you might get an alert, or an alarm, or a light. And the second zone, the robot might slow down. So it's not actually full stop emergency shutting down your system. And then the final safety zone is where the robot will actually turn off or shut down through an e-stop.

So decision making for the guarding, if you have one process that you know exactly what that process is, you can go through a standard risk assessment, and it will tell you exactly what guarding. It's a quantified metric accepted by ANSI, and so on, and others to actually be a safety rated solution. If you're buying your robot to do 500 different things a year, you've got to really look towards the future, and find their most risky proposition, and guard for that, and then you can back off or other things.

So typically when there's a branch, when you're cutting things, or milling, cutting, sawing, chipping, grinding, these things have some kind of physical barrier. Most other things don't necessarily need a physical barrier. So that, to me, is a critical check. Am I cutting or am I not cutting? And then you can start to bifurcating go down the line to the laser scanners versus the curtains, or versus the physical barriers. So, again, there's some trade-offs here.

Cost is a big one. Physical guarding is cheap and easy. The more complex the system, usually the more expensive the piece is. Some mundane details that are critical for that for the ANSI safety standard, every guarding solution has to be safety rated. If you go online, you can find hundreds of options for light curtains. A lot of them are not safety rated. They have to be a type four or class four to be actually meeting in that standard for any kind of liability purposes. Anything below that is typically not safety rated and still available.

These are things like garage doors, things that make sure that your garage door doesn't close on things. These kind of things, they don't have the speed in which the robot needs to actually have a safety stop. So if you're moving at 2,000 millimeters a second, the latency between the emergency or the perimeter guard and the robot has to be very short. And that's where the classification comes in and the safety rated comes in.

It should be tested. It should be set up by someone who knows what they're doing. A lot of times these are wired wrong and don't actually function. An enabling device is part of the teach pendant, which it's actually a dead man's switch. So it's typically on the flex pendant, the teach pendant that we showed earlier. There's three positions on the switch. There's a release. When you let go of it, it shuts the robot off. When you squeeze too hard, it shuts the robot off.

So the assumption is that if something happens, you're going to have one of two reactions. You're going to let go or you're going to tense up. And that will shut the machine down. During operation, you have this normal grip. We're in the center position, and that's what allows the robot to have motors on. Let's talk about the work area a little bit. We've gone through a lot of this already.

Limitations on typical industrial robotic work cell for purposes of research development and low volume production, specifically in AEC industry is that it's activity dependent. If your work cell is set up to package frozen waffles into a box, and that's all it's ever going to do, very hard to make that do something else. In our case, we want the robot to be able to do whatever process or need might dictate at a given time. So typical industrial robotic work cell does not really apply in the setting. So we have to adapt a little bit to the design, robotics work environment, which is trying to figure out this term.

If you have any thoughts, we could use them. So work cell automatically puts us in this industrial production category that we don't want to be in. So this is the bus and build space for Audodesk. So you can see two robots on platforms connected to a controller, some mobile light curtains that are anchored. So every time the system is reconfigured, these things are reconfigurable. So on one hand, we have meet the strict safety standards, which are designed for a rigid fixed systems, and have to adapt that then to these flexible systems.

We already talked about most of this. I'm going to skip it. So operational protocol for the robot, we'll get into a little bit of the programming elements. And this is critical to understanding the work cell and also for programming the machine, whatever interface you use. There are different definitions or different terminology for each robot manufacturer. But ultimately, we're dealing with the relationship of the robot to its work environment, the relationship of the robot to the thing that it's working on, the relationship of the robot to another robot in the space, and then the relationship with the robot to whatever the user is.

So the physical barriers, the light curtains, the laser scanners, these are the things that dictate the relationship between the robot and that person. But they're separate from the relationship of the robot to its tool or its work object. So tool center point is what the robot is typically programmed based on. And the tool center point is simply the actual tip of the tool that's doing the work. So in the examples of the two 3D printing robots downstairs, the one that's squirting the ABS, right where that is coming out of the nozzle is the tool center point.

And so all the programming that we input to the robot, that tool center point is going to the target. So it's the part that dictates which piece of the robot goes to the target we tell it to. So as the robot is printing, it's that tool tip is following the line because we define that tool tip as the tool center point. The work object is simply where we tell the robot where in space the object is that it's working on.

So if I was to have something operating on the table, we might tell the robot that this is the origin of the table, and for the rest of the calculations for movement, the robot, the controller will adapt our positions to this work object. So it becomes a difference between modeling Cartesian coordinate system where the robots is at 000. So that's the typical configuration where the robot, its base is the origin, and everything isn't programmed relative to that. But when you're in the design software, it's more difficult to actually design based on where the robot is. You want to design based in your own origin.

So when you actually move from the design software to the robot, we tell the robot the location of your work object origin. And then the robot will automatically translate that data. World origin is years when we're trying to define multiple things in space. So typically, the robots base is its origin. Everything is defined off of that. The tool center point is defined off of the base. The work objects defined off of the base. But that changes when we have multiple robots. So you can tell we drove out where things are in relation to itself. And when we have to tell the system where things are in relation to each other, we use World.

So imagine that on the shop floor we have three robots. That would be position A, position B, and position C. those are programmed in that configuration. And then it will interpret all the data from there. So the three robots can actually work together. Multiple robots set up is probably outside the realm of what most work would be needed. But it's opening up some new opportunities. So some of the research labs of the university setting we set up are multi-robot, multi-move cells. This is a diagram that describes what I just discussed a little bit.

So from the World origin, we can tell where the work object is. We can tell where robot one is, robot two, and so on. So ultimately, all those figures are derived from here. So once this system knows where the robot is, and robot two is, it can then interpolate all the target points that you have there on the work object. So Autodesk just opened recently the Autodesk Build Space, which is a first from the academic setting, where we have hundreds of labs around the world that are working from a design standpoint through automation to arrive at some new ways of making things. Autodesk st up the build space and also a robot lab at Pier Nine in San Francisco to allow this research to happen and actually move into industry.

So one of the limitations of design robotics as an academic practice is that it, in large part, has little relationship to industry. And so now through the build space, what is hoped is that we can allow collaboration between industry and practice with the academic institutions working in this field to allow things to actually become applied in industry. So rather than having a design firm purchase a large robot to do experimentation, which some do, some architecture firms have robot arms, lots of the robot arms purchased by firms or by schools even stayed dormant a lot of the times. And there's a certain threshold of entry that for the large robotics are actually applying the technology to the building industry where the build space should help fill that in.

So if there's a proposal that we have a project in our office to develop a custom facade system for a project, we want to collaborate with University of Innsbruck on the development of this based on something they've made in their lab, we together can write a proposal, go into the build space for weeks or months, and actually develop this thing as a research paradigm or as an actual low volume production setting. And this is one of the first industry examples that are really jumping wholeheartedly into the flexible robotics or design robotics space, but I think presents some interesting potential now that industry is involved. And that's where hopefully this talk can play in little bit.

So aside from all the physical aspects, which aren't really barriers of entry, they're just things you have to know about, how big a robot, what are you going to do with the robot, cost opera-- basic operations, and so on, is robot programming. And as I mentioned, two things drove the influx of robots into AEC research over the past few years. And that's one, the cost of the robots has gone down a lot, and two, new ways of programming the robot based on visual scripting or directly from design software to the robot. And this has enabled both at a very low level through plug-ins and into a very high industrial level through purpose-built automation tools.

So there's a number of ways to program the robots from a design software. Some are Autodesk, some are not. And any number of a dozen proven plugins for Grasshopper, for example, things coming from Dynamo, Dynamo-TORO, other Dynamo workflow that are available on the Dynamo GitHub to program the robots. And these are plugins that are written by researchers, typically, and are used to allow the flexible use of the machine. They're are not a replacement for an industrial robot programming environment where we might want to precision machine something.

So the plugins that are these, HAL, Taco, robots I/O, Dynamo-TORO, a couple others that they have no problem programming the robot, they typically take geometry, target geometry, convert to robot code, and ADD's case rapid code, or KUKA PRL, a variety of other things. They're not a replacement for things like power mill robot, which is a Delcan product that allows you to program in a precise way from an industrial milling or machining program for flexible uses. So there's two different ways to go with this.

On the area that-- most of our work end is down here. And so we deal with free tools that are super flexible. The more the flexibility of the tool, typically the more troubleshooting we have to do or the less reliable it is. So any of these plugins that we're using on a day-to-day basis, we're constantly evolving and adapting them. It's not a plug and play system, nor are the purpose-built tools. But on the high flexibility, the high research side, we have things like Dynamo-TORO, which are super flexible. They don't cost anything. They still allow you to program the robot. But because they're less reliable, there's a lot more troubleshooting that goes into it.

But then we have purpose-built tools, like power mill, and some others that are more expensive, much more reliable, and much more suited for a legitimate working environment beyond the research. So this is something to consider also, is what type of user are you? What operations do you want to have within your work environment?

All robots have multiple operational modes. Typically, it's a manual and an automatic mode, and then some varying degrees in there. Operational mode in manual is where you have to have that dead man switch, the enabling device to operate the machine. It also governs the machine down, reduces acceleration, reduces top speed. And this is the lowest risk, the least requirements to use the manual mode. So in manual mode, as long as whoever's in the robot reach has a flex pendant, you can be within the work cell, or you can be within the area the robot is working.

Anytime you move to automatic mode, full speed, or non-restricted access, that's when you have to be outside in this more complex guarding environment. So for a lot of the research side of things, manual mode is actually totally fine. The two cells in the exhibit hall are both printing in manual mode. They're not automated. They're not running an automatic mode at the moment. For milling or something, you have a long milling project, that's probably going to be in an automatic mode. You're outside the guarding. The robot is moving.

But if you see videos online of academic research institutions using the robot, you'll often see four or five students working in the work cell. They're probably in manual mode. But typically, everyone has to have an enabling device. So something else to consider when purchasing the robot is if you want a lot of people to work inside the cell, you have to have an enabling device for each person. So differences between automatic and manual or restricted and unrestricted, manual requires, for the most part, offline programming or is used more or less with offline programming. And that's where we take the geometry, run it through whatever programming interface we're going to have, we get a text file, we load that text file to the robot, and the robot does its thing.

Other operations allow you to screen for the robot. So instead of having that command, or that step where we actually have that text file to go to the robot, we're going to put the robot in automatic mode and then move on from there. So we have the design environment, say Dynamo-TORO, pulling geometry in, spitting out rapid code directly to the robot controller in real time. So it's a little bit different workflow, but typically, that's when you're going to need to move into automatic mode, although there's ways to bypass that. For manual mode, it's typically offline programming. Take the file, plug the jump drive in, order it to the machine, and run it.

So in the process I just described, there's a file that you actually load to the robot. That file needs to be simulated in an environment. The plugins, like the ones that we mentioned that are more on the low cost less reliable side, say Dynamo-TORO, for example, there's a simulation embedded in these things that gives you a visual simulation and a kinematic solver. It's not necessarily exactly a virtual controller. So the typical requirement is to simulate in the virtual controller, which is usually a proprietary software provided by the manufacturer. For

ABB, it's called Robot Studios. So you want to simulate end Robot Studio to get an actual representation of what the robot will really do.

That's not possible when you're streaming commands directly to the robot controller. But this is something that is dealt with through on-board safety pieces within the robot controller, so designing in World zones, other safety stops to ensure the robot doesn't leave certain areas of its work environment. There's not enough time in this class to go into those things into too much detail. But beyond the physical guarding, there are ways to program in stops within the robot so it only operates in certain areas. And if it goes outside of that, the controller will not allow that to happen.

So, again, it's important because almost all of the research-based, academic design robotics labs are using one, or all, or of some kind of mix of these plugins for different-- Mia to robot, Rhino to robot, Revit to robot, Dynamo-TORO. These things are all really common on the research end, but much less common on the production side. But they're not really a safety rated or virtual controlled simulation, just a little limitation to know. And that's where you get into software that's on that more expensive, more reliable side, which actually has a safety rated virtual controller.

Within the simulation, we have the same data that we talked about earlier. So we have the tool center point. In this case, the robot doesn't have a tool. So the center point is just the end of its fist. And that's actually what's going to the target. So if you see, this thing has a blue z-axis coming straight out of the robot. When programming targets, that z-axis is always consistent with the target.

So we have it set with a CNC machine, a three-axis machine. We have a position, Cartesian position that we're sending the tool to. Well, in the case of the robots, because you have the ability to touch things from different orientations through the six-axis manipulation, we need to tell the robot, both the target and its orientation to the target. So we asked everyone to touch this thing on the table here. We can all approach it from different directions, the same true with the robot when you tell it actually what orientation that is.

Again, all robots are different. It's important once you get the machine to familiarize yourself with the machine, and don't assume that if you use one machine you can jump to the next. Getting into the purchasing-- trying to speed up a little bit so I can go through an example, but may not make it. Lots of different robots available. Doesn't necessarily matter which robot you get as long as you're getting a robot that meets your payload and reach requirements.

Typically, robot manufacturers will advertise different types of accuracy or precision. And it's very hard to compare robot A with robot B because you're getting different terms for your accuracy. Typically, for at least for AEC research that we do, they're all much more accurate than we need. But there are trade-offs, again, with the reach, and payload, and deflection as the robot's moving and so on. The one for most flexible applications is this term. Absolute accuracy is what we need.

Repeatability is what the manufacturers want to sell because that's typically when you're doing something in an industrial setting over and over again, like putting an engine block in. You're going to make sure the engine blocks in the same place every single time. For us, we're very rarely going to the same place over and over again. So absolute accuracy is the metric that tells me how close we get to a target. And so that's where-- so if you have 1,000 different targets, I don't care if it's repeatable because we're only going to do it once. But I wanted to go as precise as I can to that target.

So most manufacturers use absolute accuracy for this. It's probably something you'll have to request. It's not part of the-- [INAUDIBLE] robot is not part of the standard parameters. But they all have some metric for absolute accuracy. So to buy a robot, you have a couple options. One is to go directly to the robot manufacturer. And this works in some regions and doesn't in others. So the manufacturers have partnerships with integrators or licensed dealers.

So in some areas, you can't go directly to KUKA and buy a KUKA. You have to go through an integrator or through a sales channel. There's some trade-offs for those two things. The integrator, they're typically a service provider that allows-- they'll provide the robot, the tooling, all of these things that we just talked about. Tell them what you want to do. They'll help design the cell. They'll procure all the parts, have it delivered.

It's not really a turnkey operation. By no stretch of the imagination do any of these things operate as a turnkey system. And there are some benefits there. If you know for sure you're milling in your shop, or you're using the spindle for machining, going through an integrator, having that set up professionally and done, it tends to make a lot more sense than trying to do everything yourself. And if you're much more on the flexible side, there's very few integrators or sales channels that have caught up with the desire for flexible operations.

So it's very hard to say we're going to do everything, and how do you design around those parameters. So often, we do this ourselves. There's a few people that are consultants that are popping up that will set up these flexible shops. But typically, you have to know exactly what you want to do when you're going to the integration channel. Similar hurdles going directly to the robot manufacturer in that they're used to selling robots to the Volvo truck plant, where they might have hundreds or dozens of machines that are bought in a given cycle. Slowly but surely, the robot manufacturers are much more open to selling one robot to an individual, a volume shop, or a school.

So there's been a paradigm shift in the past few years where now it's actually relatively easy to buy the machine, where it used to be a bigger hurdle to actually get to the thing that you wanted to buy. Things to make sure that are included in integration service are the safety portions, whatever the guarding solutions are, the IOs, and these things. And specifying all those things in advance, it's important, because if you try to add it on later, it's more expensive and also more complicated. Depending on what type of shop you operate, or if you're an administrator, and you're not actually going to be the boots on the ground, you might want a turnkey, or-- again, use this term falsely-- but a turnkey solution that's delivered for operators that can use it.

If you're the person that's actually on the ground operating the machine on a daily basis, you might want to consider integrating everything yourself so you know exactly what parts are what, where do you buy replacement parts, and how this system is set up. Typically, if you're doing a single process, or just a couple of processes, the integrator route, it makes a lot of sense. When you want super flexible system, everyone has a hard time wrapping their head around that because it's flexible. So you can't design for all those potential outcomes.

So the tool changers, second part of it, companies like ATI, Grip Schunk, others will provide the tool, these automated tool changers. This is, again, if you're doing the integration yourself. If you're going through an integrator, usually they'll procure all the stuff and select it. The mounting solution, it could be a table, base, flow. We talked about the complexities of that, whether it's a huge robot on a slab or a smaller robot on the table, to get consider that. And then an example, again, with an entry level-- this is a really high accuracy machine, three-kilogram payload, roughly one meter reach, a little less.

For a nonacademic price, you're looking in that $42,000 to $50,000 range, which is much cheaper than a lot of the other CNC tools that you might buy. So, again, it doesn't have a tool. But instead of buying five or six different CNC machines, one that cuts foam, one that mills, one that does something else, the robot tends to be a flexible solution and makes sense if you're wanting to do flexible things. If all you want to do is machine, maybe the robot's not the best solution. But this gets you around the ballpark price.

All the manufacturers offer some kind of discount for academic institutions. So the point of entry is relatively low compared to other CNC commercial CNC devices beyond that do-it-yourself examples that are on the very low end. We've got about five minutes left. I'm not sure if it's worth going through this example or stopping for some questions. Is there any questions? Run through some example here.

So quickly, this project is a wooden grid shell made with a three-kilogram-- the robots are mounted. They're under the table. I'll go through the robot code a little bit. So this is Dynamo-TORO, which, again, is the low-cost, less reliable version, developer version of this tool that we use to program the robots through Dynamo. It also happens to be what's driving the machines downstairs. So it is a functioning tool that's available for free on the GitHub.

So here we have, basically, the code that's telling us what's happening with the machine, different joint configurations over here. And ultimately what happens, once we have our parameter set up here, we can dump the geometry in for this pavilion. And then the robot won't have those codes there. So let's skip through this quickly. So on the robot coding side, this syntax is called rapid code. It's for AVV robots. KUKA's different. Universal's different. Everyone has their own proprietary version of the robot code.

But here, this is telling us that there is a robot target, which is this plane or oriented point that we're telling the robot to go to. Its Cartesian position is-- let's say a 40, 0, 0. It doesn't really matter. So there's a position, and then there's a quaternion orientation that tells us the positioning of the tool. So we need, again, to tell the robot a position, and then an orientation for that tool to go into. Other parameters that are defined here are the total data. So how big is the tool-- again, the joint targets, different routine parameters that we can set up to have the robot avoid certain problems, or issues with programming, some other commands within the robot, and then the actual motion control.

So this is dumped in through the Dynamo to robot interface. And this is where you download it from the GitHub. This is what it will look like. Again, there's about a dozen of these pretty functional plugins for various software that can be used. And so then that links with a material process. In this case, it's welding. So we have the cylinder. There's a tool, a chuck that holds the cylinder. In this case, the robot is a glorified positioner. So we could have done this with a two-axis positioner. But for the project, there's around a 3,500 custom parts, which would have been way outside the range of anything else feasible, welding in a traditional way, clamping things together, having jigs and fixtures where all these steel joints are in place.

It takes about an hour per joint to jig and weld each part. In this case, we're doing each joint about three minutes. It's a very crude setup. These are students at Virginia Tech actually setting this up and modeling this project together. But if we start to think about the potential to work with real materials in a real environment, for AEC industry to make custom parts in a real viable way, and afford some of the opportunities where I think there's a potential for industry practice in the academic institutions to get together and actually do something real-- so these are the joints.

Every joint, every flange is different. Cylinder is the only thing that's the same, and that's the part that enables the automation. So that's the robot interface. Every other part is distinct. And then it ultimately comes together in this. So the wood parts, they're all custom as well, but they're made by hand, no automation whatsoever. And then the steel joints are the thing that actually controls the geometry of the whole grid shell, and then meet all the structural requirements, and so on. But they're made, again, with a small robot. So I don't think you have to have a huge robot to do big, big projects. And it's typical that the first instance is to-- our first inclination is to buy a big robot.

And then you have a big robot that you want to do lots of smart things with. So it's a big trade-off. Especially on the academic side, the designer's pushing which university has the biggest robot lab. And it's not important what robot. It's important the kind of process, the reason for the robot. So that's all I have for this. It was a really long-winded overview of the robot process.

AUDIENCE: [INAUDIBLE] also the idea that setting up the robot and all that ended up paying off for [INAUDIBLE].

NATHAN KING: Yeah, so I think we would've been able to do it without the robot. And if we look at about an hour or so per part, it's 376 hours of a welder, of a fabrication time. And here are about three to five minutes per joint. So it actually enabled the project to work. And the reason we talk about, for the academic setting, we could have worked for months, welding those things, not a big deal. But in practice, actually talking about how do you do customization, even if it's low volume, and still keep the budget in check-- and that's where I think thinking about new ways of actually doing things with the tools this is where we are at this point, and trying to move out of the academic setting into practice or industry. Anything else?

AUDIENCE: [INAUDIBLE] or some [INAUDIBLE] studies, where they have cases like this, that they done them [INAUDIBLE]?

NATHAN KING: There are a lot of examples. Case studies maybe [INAUDIBLE]. But there's a lot of project examples in the proceedings for the International Conference on Robots and Architecture Art and Design, RobArch. And it's published by Springer. Every couple of years, I think there's three of them now, the volume of examples that talk about projects and proposals, new material studies with the robots. And there's not a lot of example in that area that actually starts to quantify the cost benefit of using the tools.

But that's what we're kind of hoping to get to. But the *Journal of Construction Automation* has a lot of-- there's typically a lot of papers in there that talk more quantifiably about the value of the automated tool versus traditional or standard ways of doing things. And others are in additive manufacturing journals. The robot fits in there. It's essentially comparing automated tool to a traditional way of doing something. And usually, you can find some rigorous data there.

A lot of these things, like I showed, there's a lot of academics doing really interesting things. I'm an academic. Whether I'm doing interesting things, it's to be determined. But it's very few people that are actually going to step, and actually quantifying, and trying to find out what is the real trade-off relating to skilled labor versus unskilled labor, material costs, variation. I think that's where we are now. There's been about five, seven years of play. And now it's at the point where we have to start quantifying. That's where you hope to get real data with partnerships, with industry, and practice to actually get those numbers so you can start justifying the cost of putting a robot in your shop. Yeah?

AUDIENCE: [INAUDIBLE] for robots [INAUDIBLE] operator positioning on the robot by hand, or is it [INAUDIBLE]?

NATHAN KING: Is an accounted? That type of programming, is it accounted for? Yeah. That's the way, a lot of times a robot's programmed in the industry. We don't do a lot of that kind of thing. But that's all set up with, actually, the controller. We don't need a separate software to do that, to actually position the robot, record that, and then feed back. But a lot of the work is moving. And you can see some downstairs with the metal printing, where there's a closed loop feedback.

So actually printing, scanning, adjusting the path in real time-- so there's this kind of cyclic trajectory where you have the robot adapting to real time material parameters. Maybe that's a totally different question that I just answered for you. But--

AUDIENCE: I was thinking more-- it seems like the most approachable [INAUDIBLE].

NATHAN KING: Yeah. The universal robots, which is the collaborative one, that's set up pretty well to have-- and that's actually the way it's designed. You press a button, you move the robot into the place, and then you record that, and then you can repeat that motion. I think that there's others that do that as well. And I think you're right in terms of early access. But most of the work, at least that we've been doing is trying to get the design model-- going from the design model to the fabrication without any kind of hurdle that I think as novice users-- and even I've done some workshops with children where we have that kind of robot where you can just drag it around, and you draw something, and then you can repeat it over and over again.

But universals do that. Some of the new collaborative robots might do that as well. It's not typical of the big robots that--

AUDIENCE: [INAUDIBLE] simulate it in the model space.

NATHAN KING: Yeah. We generally don't trust any of the plugins, any of the IK solvers that are in the plugins. So in the labs that I set up, we require that you simulate in a virtual controller. So that's Robot Studio or other variations from other manufacturers, or in the case of Power Mill, or Robot Master, which is a plugin for competing software, those tend to be much more robust simulation environments. But they require simulation, which is just a question. If you're simulating it, is that actually what the robot is going to do, or is it more of just a visualization?

So that's still a little bit of a hurdle there. All right, I think that's a little after 8:00. If you have questions, feel free to email me if you have thoughts or anything else that you'd like to ask.

[APPLAUSE]