CHIARA C. RIZZARDA: Hi. Good afternoon. Thanks for being here. I'm Chiara. This is Competition of Design class. We're going to talk about building information modeling. We're going to talk about how you can use Revit in order to automate certain design processes. And so take your time. Take your time to review our summary. Are you all in the right class? Because if you're looking for [INAUDIBLE] he's talking just across the corridor, no? OK. OK, good.

So in this class, we are going to tell you how to translate a design process into a Revit parametric project. We are going to tell you how to deal with your design rules with Revit and a little bit of Dynamo. And we are going to tell you how you can real time check your project against a set of standards that will be given to you by the clients. On top of that, we are not just going to tell you. We are actually going to show you.

So we are going to cover quite a broad workflow. So we've got just shy of one hour and we are on a rather tight schedule. So we are-- we will try to answer to all of your questions at the end of the class, but just in case we don't get there, and in order to save a little time, we would really appreciate if you could put them forward via Twitter. This on top is his contact, he is Claudio.

CLAUDIO VITTORI ANTISARI: Hello.

CHIARA C. RIZZARDA: The other one is mine. I'm Chiara. And we will get back to you. Even if we don't answer your questions to this class, I promise you that no question we go unanswered. OK? You all got this? Can I skip through? Yes? Now, as you might have guessed from my rather odd accent, I am not American. We actually come from Italy and this explains our peculiar names and my rather odd pronunciation of certain minor words like well, BIM. So when I say BIM in this class, just keep in mind that I actually mean Building Information Modeling. There are no beams in this class. Just be indulgent and bear with me, OK?

CLAUDIO VITTORI ANTISARI: Me too.

CHIARA C. RIZZARDA: I'm going to flake on lots of words. Just try and keep up. And if you don't understand, raise your hand and I'll try to say it with different words. Then, as I was saying, I am Chiara Rizzarda. I am Deputy BIM Manager of Antonio Citterio Patricia Viel, and this is the first

architectural firm in Italy, in terms of business value. We are leading in our country when it comes to building information modeling. We are a full BIM company, which means that everybody works in BIM, from the concept guy to the coffee girl. Everybody.

And this is pretty rare where we come from. We are like 80 people, and everybody works in BIM. And if I have to take a couple of minutes to tell you what it is that we design, I would have to borrow this idea that we have-- this saying that we have about architects from Milan-- and we are from Milan. So in Italy, we say that architects from Milan has this peculiar vice. We tend to design everything from the spoon to the town. And this is what we do. So this is us, spoon, town. These are two projects that we did with building information modeling, yes, even the spoon.

In between, we have buildings. We do high rise residential projects. We do-- we actually did one bridge that was Autodesk success story a couple of years ago. And we did it with Revit, because we are crazy that way. We do residential, I said. We do offices. We do retail. And we do hotels. We do lots of them. We are quite good at that. And between me and Claudio-- this is a map between Citterio and previous practices-- this is a map of the places we designed hotel in.

We design mostly five star hotels, and maybe you've been in one of them. Who knows? We'll chat later if you want. Now when you booked this class, you provided Autodesk with some data, and Autodesk provided us. So I was able to divine a map of all the areas you work in. And I really hope this is accurate, because this is really, really important. So if you could please raise your hand. Who deals with health care? Who designs hospitals? OK, or builds or-- good. And education, like schools? Great.

And hospitality? Oh good. Now this is important, because what we're going to show you is a workflow that is applied to a hotel. But the main idea can be applied to any rule based building. And when I say that buildings are rule based, I actually mean all buildings. Because all buildings are rule based. Let me make an example. This room is rule based too. You all come in from that door, and that is a rule. Me standing here talking to you, is a rule. You sitting there looking at me is a rule.

You at the end of the row being able to hear me is another rule. Can you hear me, by the way? Can you hear me clearly? and him sitting there really quiet while I talk is another very important rule. So this is how you design a space. You understand the rules, and then you

apply them. Even if you do it by hand, this is how you do it. Even if you don't use all computational design and generative design theories that we sow during this Autodesk University, and that we are going to see now.

Now, when it comes to architectural jargon, the language that we use, rules are parameters. And this was before BIM, building information modeling, and before Revit. This is a sketch made by a Gothic architect, who was trying to explain his builders what were the mathematic rules they had to apply in order to build a Gothic pinnacle. These were parameters. Now parameters in Revit look like this. But it's the same thing.

Now, who here does not use Revit? OK, bear with me. We're going to pull through. But I like this numbers. Then, when we talk about generative design and we talk about computational design, we are basically talking about a system in which you use those parameters, and you tell the computer what you want, and you let it design stuff for you. We were all there at the opening keynote yesterday, yes? And we saw lots of marvelous things, lots of science fiction thing, lots of things that come from the future.

Now I hope you will forgive me, but I will bring them down on earth a little bit. I will show you what we are doing right now. This workflow that we are going to showcase is something that we are doing-- something that you'll be able to go home and do it right now with no fancy tools, just providing that you understand the rules of your practice. Now generative design is often about stadiums. There was a really cool class this morning about computational design for stadiums. Anybody saw it? OK, so you know what I'm talking about.

And often we talk about stadiums-- and I don't believe I'm about to say this-- because stadiums are easy to generate. Now let me explain why. Stadiums-- this is a stadium that an Italian architect designed in the 1960s, yes, 60s. He had this idea that if you knew the rules of your space, you could put them into a computer, and you let the computer design it for you. I don't know if this sounds familiar.

He had an IBM 610, which is often described as the very first personal computer. And he had no idea on how he wanted his stadium to look like. He just knew that he wanted the people sitting at the stadium all to experience the same kind of game. So they want everybody to see the game, as simple as that. He put those parameters into the IBM, and the IBM designed this. Which is nice, so and so. It was the 60s. It was quite an achievement. Now this is-- we're very fond of it as Italians-- because this is the origin of parametric architecture in Italy. He is Luigi

Moretti, 1960s.

Now stadiums are easy to generate, because they have an underlining driving geometry that's simple. So you have a field. And the field-- a soccer field, a football field-- is easy. Is a rectangle. Even a baseball field is geometrical, and is a simple geometry. So you know what your building want to look like in a certain way, because you have a geometry underneath that is going to drive what you're going to do around it.

Now also, you have predictable behavior of people. Because people come into a stadium, and they want to do just one simple thing. They want to look at the game. Then you have people, and the people are all put into one matrix of what they need to do, almost all of them. But you can deal-- you can deal even with this guy, because he wants to do one simple thing as well. He want to see the game.

Now when it comes to a hotel, a hotel is rule based just as much as a stadium. So why aren't we talk about generative design for hotels a little more? Because you actually have many rules. Sure, you can go into a hotel, and you want to sleep there. But you can also have a drink. You want to have lunch. You want to have dinner. You can get a massage. You can workout. You can do lots of things. And it's not just you. You have guests, but you have housekeeping, and you have maintenance. And you need to take into consideration also those people when you are designing the metrics of your behavioral patterns.

And then, just to put on top of that, you have many luxury levels. A three star hotel is very different from a five star hotel. It has totally different rules. So the problem about hotels is not that they are not rule based. It's that all these kind of experiences, all these kind of people, all these kind of spaces, sum up to create actually a little too many rules. You can put them all into a computational system, but we have one hour, and we cannot show you how you can do it for a whole hotel. We have to focus.

But what we're going to show you, you can apply to a gym. You can apply to whatever space you want. Now since we have to focus, we decided to focus on the main reason a hotel gets built, which is a guest room Without a guest room you don't have a hotel. You have a very big restaurant. So the guest room is what we're going to talk about, and specifically, a five star hotel guest room, because this is what we do.

Now the other thing we need to choose, we need to focus, is which kind of room. Because rooms comes in different sizes, and for different clienteles. So in order to focus, we decided to

focus on the standard room, because, well, it's a standard. And because it's the most difficult. The standard room is the smaller module you can get in a hotel, and that one needs to work because-- this is a pie, a doughnut actually, on how the percentage of the standard rooms you have given the total surface area. It's 50%. And if you'll pardon my French, if you screw the standard room map, you're damaging the client a lot.

The standard room needs to work, and it's the smallest, so it's the more tricky to design. Everyone can design the presidential suite, given a little time and a little taste. It's not that difficult. Besides, if you own an architectural firm, you know that your people will hate designing standard rooms. Because they can't play around. The space is so tight, that they cannot play around with it. So by automating this kind of workflow, you are actually doing them a favor.

Now, we're going to start from a designer's perspective, because this is what we are. This is what we do. And when you start as a designer-- when you start designing a hotel, the first thing you do is actually not designing, is reading. You need to read the client's brief. In the building information modeling era, the brief is particularly crucial. It's called the employer information requirement. It's the very first step. And the client needs to put his thoughts together in a strategical perspective, so that everybody else coming afterwards will know what they are doing. Which is a very simple concept, but apparently it's not that obvious.

Now even in the building information modeling era-- I'm going to say BIM from now on, so you'll forgive me. Clients tend to have a very romantic view of your reading the brief. So the brief comes as four book novel, as a well of text which actually doesn't mean anything to you. Because you cannot work with it. Now if there are any clients in the room, I beg them, please, change the way you do your brief. The brief needs to look like this.

We are working with Revit, and Revit is a database. So the brief needs to be a database as well. So I can take those parameters and feeding them into my parametric software. If you are a designer and the client delivers you the well of text you saw before, what you need to do, first hand, is translate that brief into a language that a computer can understand. If you don't do that, you're going to have troubles afterwards. So it might seem like a waste, but take your time. Do a little data entry, and shape the brief like this. You're going to thank me later.

Now-- then when you're done reading the brief-- by the way these were measurements of all the things that the client wanted into the room, so how big he needs the shower to be, how big

he needs the bathtub to be, if he wants the bathtub or not, if he wants the double sink vanity. It's something that you usually find in a well of text, just put in a different way. Now when you design, you start with the layout. And these are layouts of hotels we did around the world. I think this is Milan. You have Paris. You have Dubai. And I don't remember the other two.

CLAUDIO VITTORI ANTISARI: Bali.

CHIARA C. RIZZARDA: Bali. Yeah, you're right. And they all seem very different, don't they? Pretty much. Now if you take your time and break them down into zones, you'll see that they are not so different, actually. Now the first thing you need to understand when you draw a layout of a hotel room-- the first question you need to ask yourself-- is the question that is probably the most important in life, and that is, where is the bathroom? I'll show it to you again. The main difference in these layouts is the position of the bathroom.

When you start designing, you need the bathroom to work, because you cannot play around with it. You have specific rules of the fixtures that are tied to the human body. And since you cannot change the human body, you cannot change the dimensions of the shower much, because you need one person to fit in. You cannot change the dimension of a bathtub much, because you need one person to fit in. And I'll stop here, because I'm a lady. But you get my point.

Now you have to decide if you want your bathroom to have the window or not. Sometimes you can't afford the bathroom to have a window, so you are stuck with the second configuration, which is also the third one, which is also the fifth one-- sorry, the fourth one. So this one, this one, and that one are actually the same room. And this is what we're going to do. So I'm going to stop talking in a minute, and he's going to start doing stuff. What we're going to do is, we are going to generate the key elements of our bathroom. So we are going to use parametric families.

Now we are not going to explain you how you build a parametric family Revit. We are assuming that you already know that. If you don't, there are lots of tutorials. You can see, this is really easy. But it's really crucial. You need your families to be parametric, and you need to control which kind of values you want to change. That is the first design decision you're making. Which are the dimensions that are changing in the bricks you're going to use to build the room?

Now we are going to feed parameters from the designer's brief into-- so from the designer's brief we are going to feed the parameters into Revit using Flux. Who has heard of Flux before? Who has used Flux before? OK, that's good. For everybody else, Flux is a platform originally developed within Google, and it's based on this really cool concept, that all you need to share in order to collaborate with other people is not models, it's data. So it's a system in cloud that pulls out data from whatever software you want-- actually, whatever software they developed the plug-in for-- and stores them into the cloud.

So you have a plug-in for Google spreadsheet. This is what we're going to use. You can pull the data out, and you can fit them into Revit with Dynamo. This is Dynamo. Who using-- who is using Dynamo in this room? OK. For everybody has, Dynamo is a visual scripting system. All these little boxes you see-- and when you see somebody playing with it, they're actually connecting little pieces of software that somebody else wrote for us. And I will never thank them enough for that.

So we're going to use Dynamo in order to feed parameters into parametric families. Then we are going to create a parametric room using global parameters. This is a rather new concept. Who is using global parameters? Who has ever used global parameters? For everybody else, Claudio will explain what global parameters are. And then, again, we are going to place families into our project using Dynamo, and be happy about it. So we're going to extract data with Flux. We are going to compile parameters with Dynamo into Revit.

We are going to generate the room in Revit with Dynamo. We're going to close Dynamo, look at our Revit project, and be happy about it. And when I say we, I actually mean him, Claudio. You're up.

CLAUDIO VITTORI ANTISARI: OK. Hello everybody. So we're going to see this workflow that Chiara just illustrated to you. We're going to start from Flux, then Dynamo, then Revit, and Revit. We're going to have three data set, and we will not have the time to explain every single passage, like what this block does, and what this other block does. We just want to follow the workflow. But again, we have Twitter for question, so if you're curious about something, come to speak to us. We like to speak about BIM.

Let's start from the client brief. So the client brief should come in this shape, because this is a database. So if I have this database I can start to move my data from one software to the other. The first step is to take this data from Google drive and bring it in Flux. If you don't know

it, Flux is here. It's under this tab. And at this time, I just want to upload the data. Actually, what I really like about Flux, that Flux is really easy to use. So if I have to do a comparison with Dynamo, Dynamo is way much more complex. And of course, when I click on my keys, that's just here on my screen, is already highlights what is selected, what type of data I'm going to bring into Flux.

Just to let you know how easy it is to do these things, I will remap the shower data. So I'm going to click here. Add it. Then I need just to select arrange, like I'm working on Excel. I click OK. I save. It takes some times. It depends by the connection, and how many of you are using Twitter right now. So I synchronize one and two. Now the data are uploaded in Flux, so I can close Google drive. And I go-- I can go in Flux.

If you see here, on this part of the menu, I see my data. And there are some data just get updated less than one minute ago. So is our data. As you see, it's still-- they still have the same shape. Is important to keep in mind that I'm not going to edit this data at all, because I have client data. So I don't want to do it. I just want to move it from one software to the other, one and two. And now I'm going to go back in Revit. And I'm going to use Dynamo in order to bring this data from Flux into Revit.

So this is my first dataset. Here we are dealing with a generic component. Is like a bathtub, you see. And is basic parametric family. You see? They are just like basic parameters, nothing special, nothing crazy. And I'm going to use Dynamo to get my data from Flux to the Revit. Again, I don't know how much you're familiar with Dynamo. My approach, when I have to see a node that is made by someone else, I always go to look to the last block, because the last block is telling me all of the scripts-- what's the purpose of the scripts. What the scripts want to achieve.

In this case, the last is family type that duplicate. That is a node that can be used to duplicate types, and to rename types, because it's the same concept in Dynamo. So the entire script works in order to generate types. What's my goal here? I didn't show you, sorry. This that's up right now is just a generic-- well, a generic type. This is like what I have in my office when I have to start the project. And now I have to produce six different types that need to fit in six different rooms. And we're going to have a different name, because I need to count them, and different dimension, because they are different.

Like luxury, as much as luxury, as much as bigger, as you may notice in even this hotel. Let

me go back to Dynamo. Here I press run on the first instance, and nothing happened. So don't freak out, like I did it the first time. Because it's Flux, you need to run two times, because you need to select the key. So I'm going to work with the bathtub. So now I have it. You see it's still not working. I press run. And hopefully it's going to work. Great. You never know with Dynamo, in front of great group of people.

So as you see, my final output is to generate these types directly from the client brief. But it's important to keep in mind that we didn't do anything manually, like, anything is coming from the brief. So this is-- this first part is over. I'm going to close it. I'm going to show you that I have six plus one new types in my family, just created directly from the brief. Of course, this is nothing, because I need to feed the parameters. So this is just one step. Let's go for the second.

CHIARA C. RIZZARDA: You need to do it in two times, because Dynamo is a little bit like a man. He can do just one thing per time.

CLAUDIO VITTORI ANTISARI: Not-- That's completely wrong. The reason is because when you try to do so many-- so many action in just one script, there is not the right sequence. So sometimes you need to-- maybe you're right. So this is the second script.

CHIARA C. RIZZARDA: Nobody freak out. We are going to upload the scripts so you can download them as additional material for the class. So don't worry.

CLAUDIO VITTORI ANTISARI: We're going to share the script. So again here, there are two final output of the script. One is name-- the group, I name it set parameter values. So I'm going to feed these three code block. That's our set parameter value, with value, again, coming from the client brief. And more important, this other one is-- the final point is a node called family instance by point and level. What does it mean? That's I'm going to create families. I'm going to create families from Dynamo in Revit. Why? Because I want to give it to my designer a showroom because I need to help them to not make mistakes. So I want to give them not-- OK, you need to-- you have your families and you to input this date. No, I want to give them everything ready.

So we're going to see how, again, I press Run. Nothing happened. Now I choose again my key. Now as you see, this is-- maybe you don't remember. This is the same script of the previous one. This is just a check in this scripts. It's checking that I have the right types, or if I don't have it, it's going to create it. In this example, it's not doing anything.

And here I have this massage data. This is something that is pretty common when you work with Dynamo. You need to adapt the data to the new workflow. So basically, I'm going to split the spreadsheet and divide it, in order to feed with the right value, that you can see maybe here. I have the value for these five different-- for the five different bathtub depths. And it's going to set my parameters. What's the result of this script? Is it easier to watch it? It's a showroom. I have my types.

So now I can give this to my designers. They can start to design the hotel with the right component. So you cannot-- say that you avoid mistakes. It's too ambitious. You reduce a lot mistakes from designers. Of course, we still have one hour, so I cannot show you how to generate all the component of [INAUDIBLE] but you can apply this workflow to all the element that are rule based, like the shower--

CHIARA C. RIZZARDA: The bed, for instance.

CLAUDIO VITTORI ANTISARI: The bed, the sink, and so-- and all the major component of a hotel. But at this point, I didn't do anything. I didn't do a room. I just made it a showroom. So let me do the room, otherwise Chiara won't be happy with me. And I want to avoid that. So I close the first example.

CHIARA C. RIZZARDA: Again, you need to close Dynamo, because Dynamo can point just one project at a time, just like man. So you need to close it and open it again. And that with man, unfortunately doesn't work. And the showroom that Claudio did is actually a very powerful design tool. Because imagine with a very complex project as a hotel, you have this file where you have all your elements in a row, and designers really look at what they're doing, know which components they can play with. It's really a really powerful tool.

CLAUDIO VITTORI ANTISARI: OK. So let's-- see this file is empty. That doesn't look great. But we're going to fill it with components. This is a script made up by a reference plane and global parameter. Again, if you don't know what are global parameters, I really invite you to take a look. Because it's the unique type of parameter that allows you to connect the project file environment and the family environment. And in this case, It allows you to work in a project file like in a family. So as you may know, this is a project environment. I have my sheets. I have my views, but I'm working with parameters.

And as you still notice, this is the room. These are the two measured dimension. And this is the space for the bathroom. Before proceeding, I want to say something about why we don't

do standard room with always the same size. Because it's not the interior designer that choose the size of the room. It's made by two activities, that's happened before the interior designer get in the project. The room width is defined by the building size. So how big is the building? And the room depth is made-- is dictated by the engineer, by the structural distance, by one column to the others.

So if you're an interior designer, you need to be flexible. Because you cannot have your perfect room that you're going to apply in every single project. And we're going to be flexible. I just see here I have put several reference plane to define the zone they're going to take in by my component. And I want to show you the workflow, then we come back to this file. So I go back in Dynamo. This time, in this workflow, it will be just Dynamo, no Flux, not other software involved. Because we just want to read-- to read the global parameter, and to take advantage of them, and utilize them to place my families.

So again, this is another script. As you may notice again, the final node is from a distance by point and level. So we are placing families. How do we place these families? First, we mine data from the global parameters. So we read the data that we need from the global parameter. And again, understanding which type of parameter do you need is a design process, like we're still designer, nothing different. Again, this is our just standard node in order to collect the value in the shapes that I like.

And then I'm going to create a basic geometrical rule-- that's really like just plus, minus-- in order to define the sequence of components. That, by the way, is going to be toilet, sink, shower, bathtub.

CHIARA C. RIZZARDA: It's what we call a leaner bathroom.

CLAUDIO VITTORI ANTISARI: Yes. Correct. And these are the families already selected. So here there is no Flux involved, so I should be able to press just one time. Wow. So we got our component placed. So is what we get. Why this? Because the bathroom, as Chiara was saying, is a parametric-- is a total rule based space. There is no freedom. Before moving to the next stage, I want to say a couple of things about this example. The first one, that's of course, distinct is parametric. So if I go and I change the global parameter-- I go from 4 meter to 4.5. Now it's a mess. But if I press run again, they adapt.

And if you didn't notice, this sink is adapting to the reference plane. And this is achievable only

with global parameters. So that's why they are cool. Because with global parameters I can connect the parameters inside of a family, with the parameter inside of the project environment. And you can do a lot of cool things with this. You can create a lot of cool workflow with this approach.

So now I would like to design the entire room, but we have just one hour. So I will show you the final output. Again, there are some other spaces that are rule based like the bedroom. When you have the bedroom, and you have-- how Chiara, how would you say--

[NON-ENGLISH SPEECH]

CHIARA C. RIZZARDA: Bedside table.

CLAUDIO VITTORI ANTISARI: Bedside table on the one side and the other side. So here you can create your rule with that space. So the final output should be this one. Let's try to visualize it in a real workflow. When I get this stage, I generate in my room. And this room is ready to be linked in the building model. Because if you design a hotel, you know that you cannot design every single room. Or if you have to design every single room, is a drama. You have a big problem in workflow. Someone is doing all that over there.

CLAUDIO VITTORI ANTISARI: So with this, why I want to say that-- because this is a workflow that's really appliable to a real case scenario. We have our showroom, where we collect all the components for the team and everyone goes in that industrial room and get the components. Then you have the designer that is going to work on the room, and he's going to create the modules. Then the module is going to get inside of the building.

But before doing this, I switch back to Chiara, because she want to say something else.

CHIARA C. RIZZARDA: May I have my presentation back, please?

CLAUDIO VITTORI ANTISARI: Yeah. Let me close it. Just a second.

CHIARA C. RIZZARDA: So what we saw-- what you saw is appliable, and what we are doing is using it also to explore the possibility of changes. So when the structural engineer calls you and tells you, hey listen honey, I probably need to change the module of my structure. Can you deal with that or not?

You can change it. You can look at it, and you can say, yeah. No problem man. Go for it. Real time with just one single click. So this is what we did. And now if we have-- yes, we have plenty of time-- we are going to change our hats a little bit. And we're going to step back. And we're going to try to I meant in a metaphorical way, but OK.

We're going to change our hats, and we're going to step into the client's shoes. And we're going to see how the same workflow can be applied in a reversed way by the client to mine data from the model that was delivered by the designer. Before that, I just want to say something. When you are a client, and you're working in BIM, you need to stay involved in the process. And I mean really, really involved. It's a concept that we like to explain by borrowing a paradox from quantum mechanics.

And it's the Schrodinger's cat. Now who's familiar with this poor animal? OK, good. So for everybody else, I'm going to sum it up, and I'm going to probably not do a good credit to this very complex theory. But the basic idea is that Schrodinger was this guy and he had a cat. And he put it into a box with venom, and he theorized that-- I know nobody protested. He was a scientist, I don't know. The main theory is that when the cat is in the box with the venom, you don't know if the cat is alive or dead. And this opens up for a whole set of infinite universe.

What we need about this thing is that when you are a client, the project is your cat. So if you are working in a traditional fashion, you have your design team working within a box-- with lots of coffee usually-- and your cat. And you have no way of knowing if the cat is alive or dead, meaning you don't know if the project is correct in terms of what you wanted, in terms of brief. This goes on until the end of the phase. So at the end of typically the conceptual phase, the designers gives you back your cat, and you find out that it has been dead for months.

Now this is a problem. This is a problem when it comes to BIM because it's easy to apply changes, but that's not the point. If you are working, and you stay within the stream of data, you can keep the cat alive. So you need to stay really involved. You need to ask your designer for small deliveries, like software engineers are doing with the Agile, they are pushing small updates. And you can check them as a client, and say, OK, the cat looks fine. Go for it. So as a designer you need to keep the cat alive, and you need to work with the tools that can communicate with what we are working. I'm going to kill myself, I think.

CLAUDIO VITTORI It's done.

ANTISARI:

CHIARA C. RIZZARDA: Thank you.

CLAUDIO VITTORI ANTISARI: [INAUDIBLE]

CHIARA C. RIZZARDA: OK. Thank you. Let's broaden our range, our visual range a bit. This is a simulation of floor plate of a very small hotel. So when you are working with the designer that we're are sending-- because we are the client now-- the designer will deliver you a model. And what you need in this example is just one thing, usually. Is what we call the distribution and core efficiency. Because-- let me explain to all that are not in hospitality. What's your rant about this plan are the rooms. But you need space for people to roam around.

The problem is that you are not selling that kind of space. So you need to make sure that distribution is usually with say, under 30%. This is the kind of data that you need from this drawing. Now the designer will probably deliver it to you with a thing like this, a pretty picture. Now, what you might want, probably, is something like this, another kind of pretty picture, in order to check that your core distribution is less than 30%. Now the cool thing about BIM is that this pretty picture and the other pretty picture are actually the same picture, because they come from the same source. It's the same data just shaped in a different way, and they are always consistent.

You don't have somebody that had made math, and then took a spreadsheet, and then did the graph. And lots of things can go wrong when each kind of step-- each manual step gets taken. So what we're going to do now is we're going to extract data from the model with Dynamo and Flux. We're going to analyze them in a spreadsheet. We're going to use Google drive because we like Google drive. And then we're going to visualize them with a tool that's called Venngage that we like a lot. So basically what we're going to do is we're going to use Google, Google, and Google to do these kind of things. And when I say we, I mean him, again, Claudio. You're up.

CLAUDIO VITTORI ANTISARI: Not again, I'm a different type of Claudio. Now I'm the Claudio that is working for the client. And I'm ready to check the mess that the Claudio that was working for the designer have done in the project. Why this? Because when we see a building from a client perspective, what we found out that the client doesn't really care about our project. He care only about a couple of numbers that define the quality of the investment for the client. So if you have that numbers,

your project is OK.

I mean, of course, there are some other factors. But the beginning, that's the key value when you're designing a hotel. And we have to act in a different way, because when you're working for designer, for your colleague, you can have a type of workflow where the--

CHIARA C. RIZZARDA: Sorry, that was me.

CLAUDIO VITTORI ANTISARI: When you have to work with your clients, you need to communicate the data in a different-- with a different approach. Because you need to be more clear. You have less time. There are people that are not software educated. So you need to invest more time, more efforts in the quality of your presentation. And you can do it, again, with a parametrical workflow without going out of this automation. So I will show you how we start from our revenue model. Our revenue model now is the next step. So is a 3D model. You see we have our full floor plate. And of course, again, it's just a matter of time, but you can easily apply this workflow to your entire building.

You have your layout. And you have your zoning. What is relevant to this workflow? Is nothing but the rooms. We are going to extract value using the rooms. As you see, I have two type of rooms. I will- like circulation and room areas. Of course, the rooms, they got their own names. This is the department, but this is pretty basic. So what do we want to do? We want to extract this data from the rooms, and bring it in a scalable thing, like Dynamo, Flux, Venngage, Google drive, Venngage, or Google, Google, Google. And we open Dynamo.

CHIARA C. RIZZARDA: Now what is nice is that you as a designer can extract datas, and you can deliver them to the client. And the client has his spreadsheet and everything is real time.

CLAUDIO VITTORI ANTISARI: Exactly. And plus, the client doesn't have to use Revit or use Dynamo. That is something that-- I mean, at least in Italy it's not happening. I don't know about you guys, but-- so as you again-- I like to start from the bottom with script. So the last one is a node for going into Flux. And basically, the entire node is just for collecting data from rooms. So I make it run so it will be more-- it will be easy to understand. And then I have to perform two operation.

And the first one is to get the department names. And of course, in this demo, it would look pretty silly because we have two department. But if you try to apply it to a larger building, it makes sense, because you may not be the-- the person to check the model, it may not be the

person that designed the model. So if you can automate this, is a great advantage. How do I get this? Basically, I read the value in the department parameter in the rooms with this node. So is like, I got the parameter. I got the parameter names. And using list that unique items, I group them. So I get out what I want, what type of rooms are in my project.

Then I need to perform a second operation. Is to split my list in two lists based on the department. And again, I'm going to they use Colin McCrone called it yesterday, the least favorite node for designers, list dot map. And I kind of agree on that. When you fill it with block, without feeding the element, feeding some datas, it's just a function. So I'm mapping my list, basing on the department. And the output is, again, is reading what's the name of the parameter in my rooms. You see circulation and room areas. Then I use these other two nodes that's list dot get item indices and list dot get item at index. That at least, they're not the same even if my pronunciation doesn't help.

Sorry. It's a typical workflow to identify value, and to get the data. In Dynamo you have to do, usually, two operation. First you identity-- you map the data, and then you get the data. So you cannot do it in just one-- in one shot. So that's the final output that I want, our room. A room department based on which room are in circulation, and which room are in room areas. But again, I'm dealing with the client, so I want to give them the right type of data. So I give them-- give him just a series of values, like this, I mean, in my experience it doesn't work. Or you risk that the attention of the client goes on that detail, and you cannot go-- speak about important matters.

So we use again. We map for area. And we sum the values in order to achieve this, so the totals. This is pretty, super easy. And again, we rework again the data in order to get this. That is the way that Dynamo looks at-- thinks it is a spreadsheet. So this is a spreadsheet in Dynamo, this table, this type of list. As a list passage, the last passage is to upload this data on Flux. so I press run. And the key is correct. If you see empty list again, don't freak out, like I did the first time. Because when you upload data to-- from Dynamo to Flux, when the data gets out of Dynamo, you cannot check in Dynamo if it's working or not. You need to go on Flux. If you see in Flux, my data got updated one minutes ago. So I just uploaded my datas. That's an ugly one that you're seeing, but these-- these two numbers.

And now from Flux, I just need Flux to move the data from one software to the other. I'm going to move to, again, Google drive, where I have my data. That's already placed, but in order to replicate the workflow I'm going to cancel them. This time I'm just doing the opposite

operation. I'm downloading data. Again, it's super easy. You just press. You're there. So again, this is a workflow that parametric in it's automatic.

My last activities is to use Venngage. How many of you use Venngage? No one of you? OK. Maybe this is a really nice software we use a lot. That's-- wow. At least you won't see my password. It's a software for creating infographics. Then it's pretty smart because it can read data, datas from other softwares. So in this example, I want to create a chart, a pie chart. And with this-- with this chart I can feed the-- I can feed here. I can put here Google public spreadsheet. And of course, in order to achieve this workflow in the Autodesk University, we need to make this data public. But for the one, that's our concern about the privacy, you can do-- you can achieve the same workflow within Excel, and you can keep in house.

So this is just for because we are here, and we want to show you the best workflow as possible.

CHIARA C. RIZZARDA: By the way, lots of puppets and stuff you saw in the presentation were done with Venngage.

CLAUDIO VITTORI ANTISARI: Yeah. I really invite you to take a look If you are an architect and you need to do presentation, because it save you a lot of time. So I publish my Google drive. I get it. And I'm going-- sorry. Here I'm going to feed it in Venngage. I imported it. I remove one row. This is my graph. And again, it's coming from Revit directly. There is no-- the chain is still connected. There is a link in every step that we did it. Maybe this is not what my client want to see because he want to see numbers. So I'm going to change the [INAUDIBLE] and I'm going to activate labels. But--

CHIARA C. RIZZARDA: Can I have percentages?

CLAUDIO VITTORI ANTISARI: Yes, yes, yes. Clients always like that. So we're going to fix the percentage.

CHIARA C. RIZZARDA: Can I have a line connecting them to the sector?

CLAUDIO VITTORI ANTISARI: Yes, I can do that. Nope, oh what's that? Is this one.

CHIARA C. OK. Good.

RIZZARDA:

CLAUDIO VITTORI ANTISARI: So this-- it's a clear communication of a concept. We are-- why this number important Chiara?

CHIARA C. RIZZARDA: Because the red one is the one you do not get money from. OK?

CLAUDIO VITTORI ANTISARI: OK? That's all.

CHIARA C. RIZZARDA: So can I have my presentation back?

CLAUDIO VITTORI ANTISARI: Oh yes. For sure.

CHIARA C. RIZZARDA: Thank you. Just to close it up, we have a couple of minutes. So let's sum it up, what you saw us doing, is basically going around in a circle by using lots of different instruments in order to keep the data consistent, and to keep them together. So we use a spreadsheet with Flux, and then as designers we use the Flux and Dynamo and Revit. And then we pushed all those data back to the client. And the client used the Flux and Dynamo or whatever. What's important about what we just showed you is that you can actually pick your own instruments.

So this is the set of instruments that suits our practice, because this is how we work. But as long as you understand that you can take data and communicate them around, you can choose your own instruments. Now, we like to talk about instruments, and not about tools, because there is a specific difference when it comes to this kind of [INAUDIBLE] This is not something that I'm making up. It's something that was theorized by a philosopher of science, Alexander [INAUDIBLE] and he said that basically a tool is like a hammer. You remember [INAUDIBLE] he said that a hammer makes you see everything like a nail. Now what's cool about an instrument, as opposed to the tool, is that an instrument makes you see things differently.

So a telescope like this one is more comparable to the kind of instruments that we showed you, because Flux, Dynamo, Revit make you see things differently. If you are able to see things differently, like with a telescope, you're able to see when pirates come, and you can save your family. So right now, you probably need to use these kind of things, because I don't

know how you're faring here in the States, but in Italy we need to save our practices, like a lot. So if you have a telescope, and if you're using the right instruments, you can face the changes that are coming. So please, please, please choose your own instruments.

Now, if you liked this class, and if you enjoyed what we did, we are going to talk again tomorrow, just next door. And we are actually going to explain you how you can implement this kind of workflows into your office. The class is Revit Implementation BIM Implementation. And when I say we are going to talk about it, I mean he is going to talk about it, my assistant with the silly hat is actually my BIM manager. Thank you, Claudio, for your precious support in this class.

And as I feared, we don't have much time for questions, maybe one. Does anybody-- I had just one question on Twitter. They were asking if Flux is for free. Unfortunately, it's not anymore. You have to pay for it. But we can all go to the Flux booth and protest, if you want.

CLAUDIO VITTORI ANTISARI:	We encourage you. We encourage you all together to-- against Flux.
CHIARA C. RIZZARDA:	What's the question?
AUDIENCE:	When you make changes in the model does it update all the script, or do you have to manually go in and--
CHIARA C. RIZZARDA:	You don't have to manually do anything. You just need to decide that you want to push the changes.
AUDIENCE:	You go to each individual software that you showed and you push it, or it reads from each other?
CHIARA C. RIZZARDA:	It reads from each other. So it's a chain. And each link of the chain decides if they want to push it to the next chain.
CLAUDIO VITTORI ANTISARI:	But, if I can add something. In this type of workflow, even if it's automatic, you cannot rely on the automation. You need to check your data. So if you have to press a button, it's not a big deal, because you need to-- you are sending that data to the client. So that's why I don't see-- it's a good question. But I don't see as a major problem. If it's just a button, you can press it.

**CHIARA C.
RIZZARDA:**   OK. So thank you for your time. Let me reassure you--

[APPLAUSE]