Blender in architecture

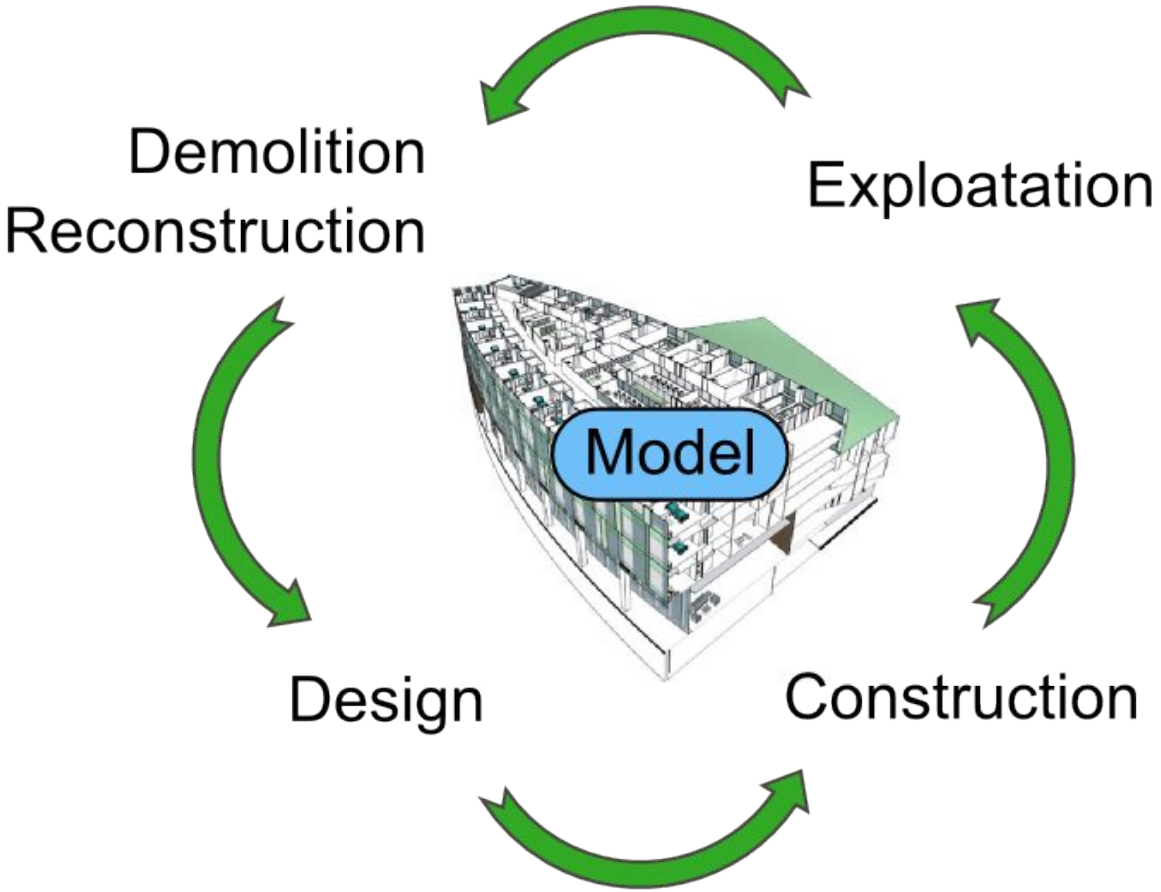
- Everyone knows that blender is cool handling 3dviz/modelling jobs.
  - Some people use it for architecture even now!
  - Still.. there is space for improvement.
- Why it is (could be :)) better than other tools in market?

CAD vs BIM

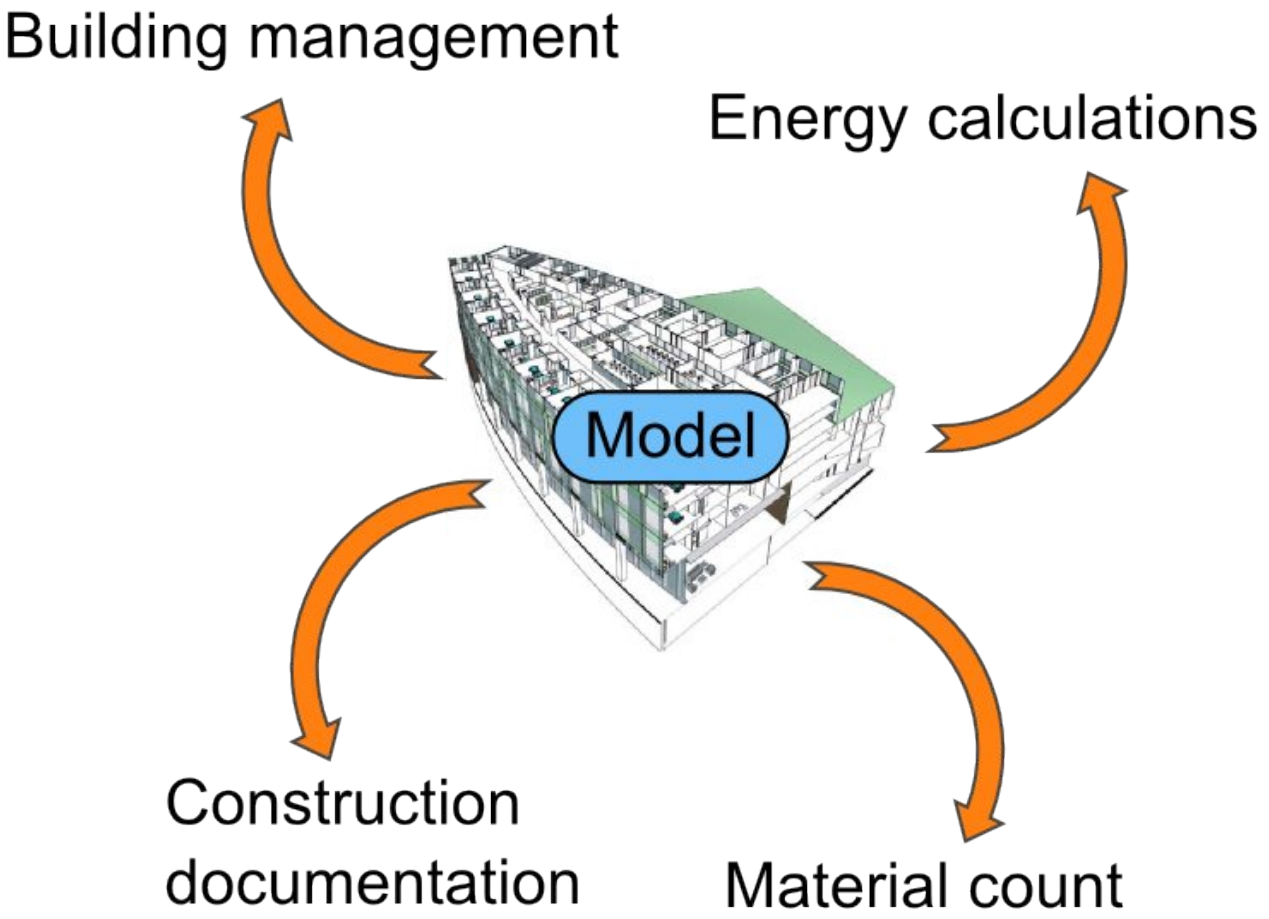
- CAD is about lines, solids and meshes. Out of date.

- BIM is about model in general. You describe the essence of the building. Blueprints should just pop out of it magically.

Building information modeling



Model

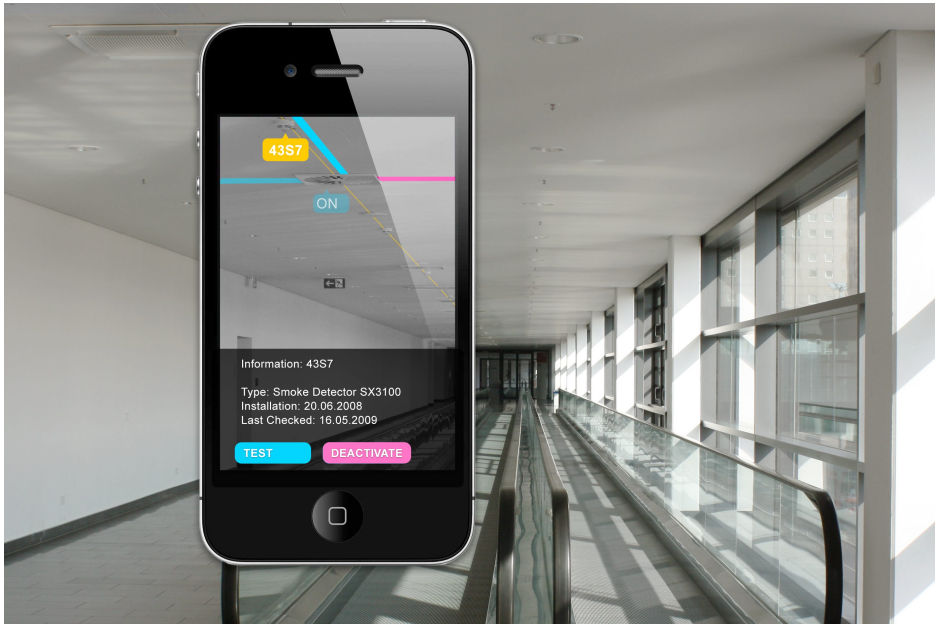

Building management
Energy calculations
Model
Construction documentation
Material count

Situation in market

- Still, *** A LOT *** of architects use old-school CAD here.
- Market demands BIM:
  - Better decisions
  - Greater predictability
  - Less conflicts and collisions
  - Faster project delivery
  - Better project maintainability through all lifecycle

PLUS

- Augmented reality:
- Ability to see through the wall – pipes, electricity cables etc with iphone
- Easier construction & maintainance



Big four

- Autodesk Revit
- Autodesk Architectural Desktop
- Graphisoft ArchiCAD
- Bentley Systems

Competetiviness

- Power of open source!
- Big projects repetetive jobs costs could be cut by customising software – API's available for closed-source projects don't always work out
- Better competition for training, support and developement service providers

Leaps and bounds

- Lack of tools & special functionality

  - not too hard to solve!

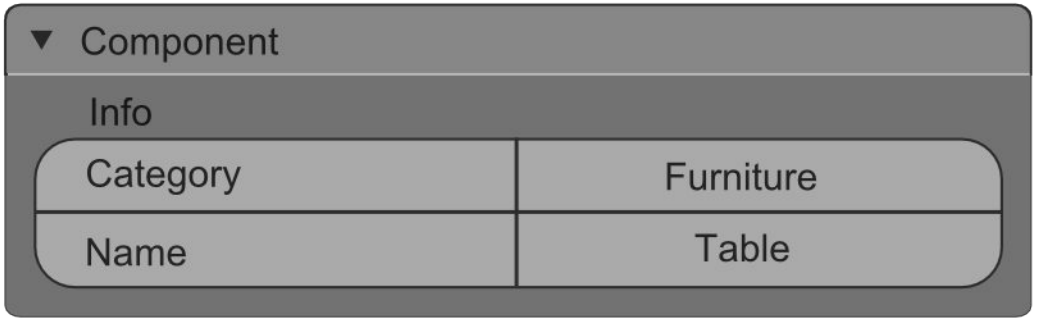
  - details in our feasibility study

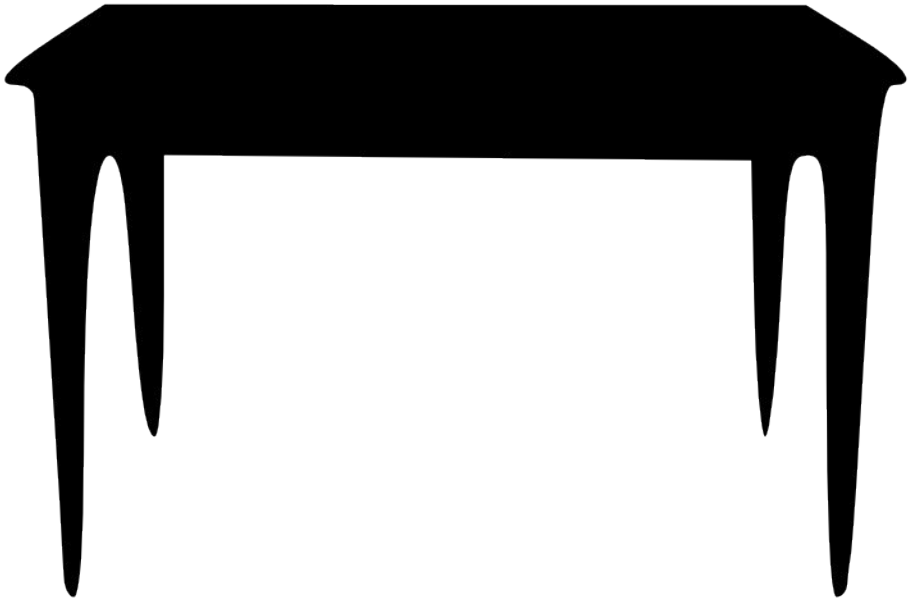
- It's worth to color the gray area between CAD/Modelling and 3DViz -to have everything in one box

If talking about BIM...

- Real things are made from objects. Not lines. Not polygons.
- It has to be possible to customize the object without re-designing it completely.
- In example, change height of the table by modifying height parameter.

Basic parameters





Component

Table

Hosts

Level	Level 1
Level offset	0

Spatial

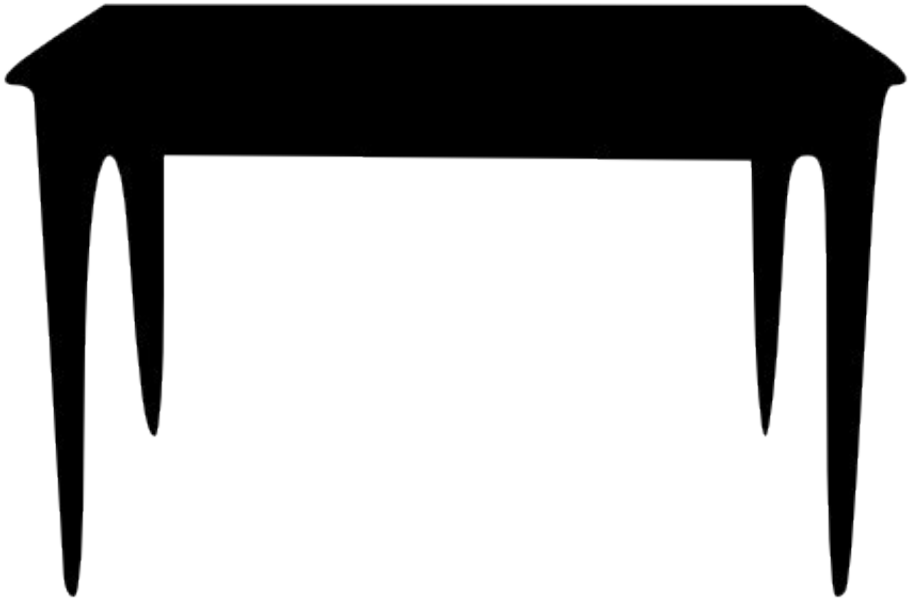
Width	200
Depth	300
Height	500

How is this possible?

- Reference planes (Refplanes). They are everywhere.

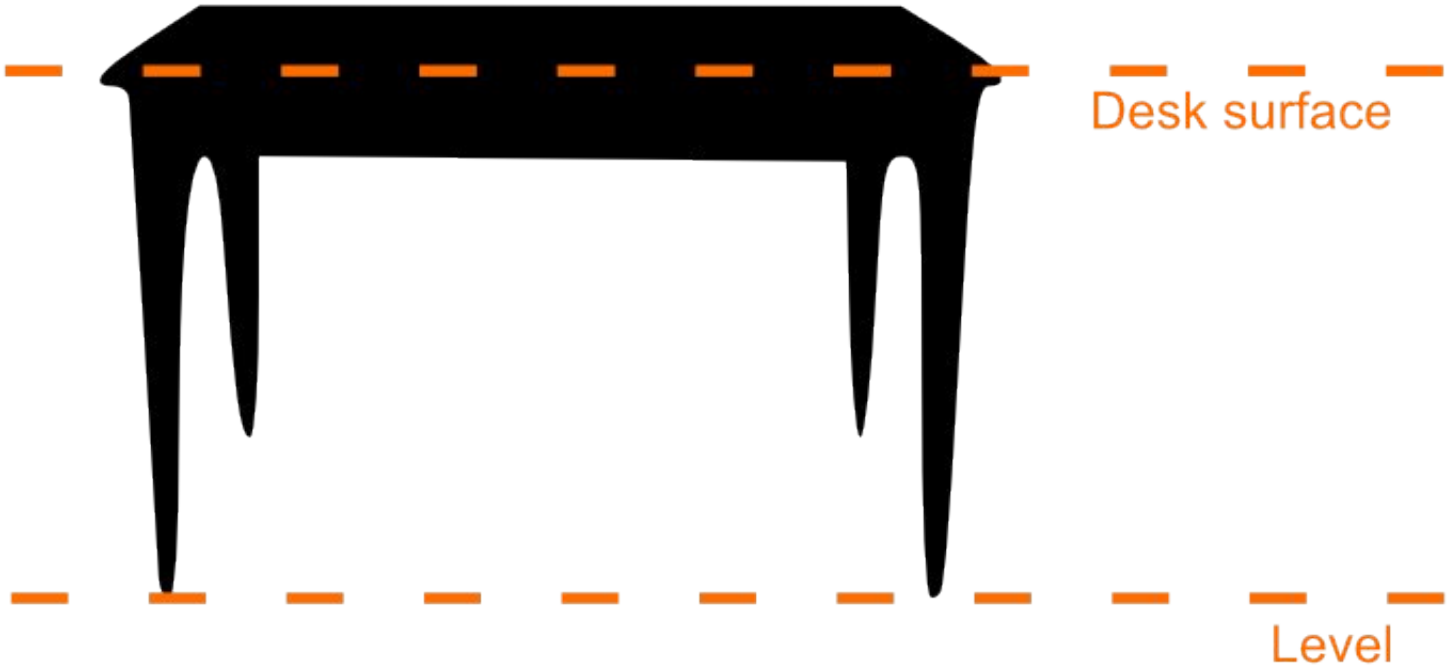
Refplanes

We have a object. Actually, we have a table.

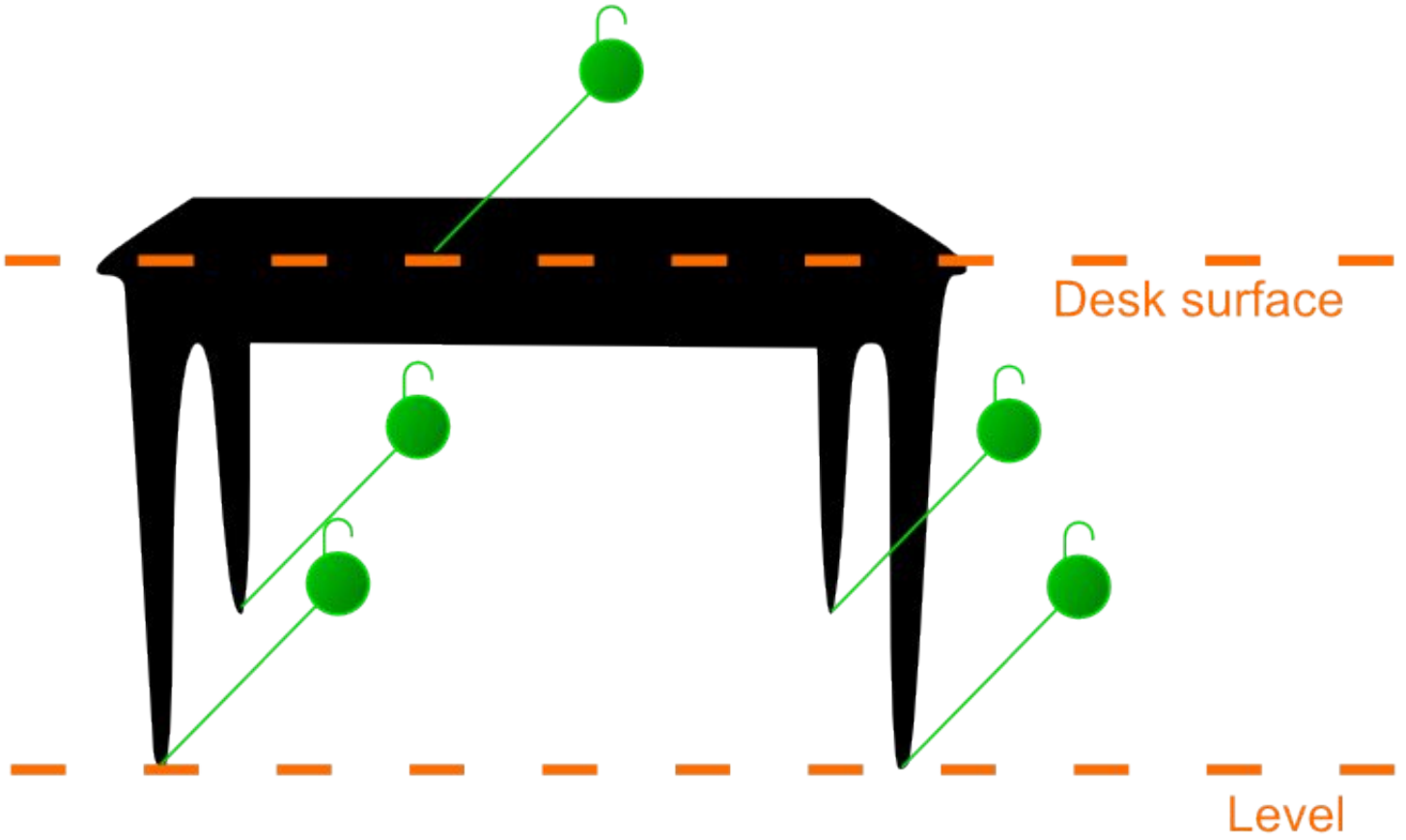


Refplanes

Let's add it a surface and floor level Refplanes.
It will be possible to lock it to other objects (in example, room floor).



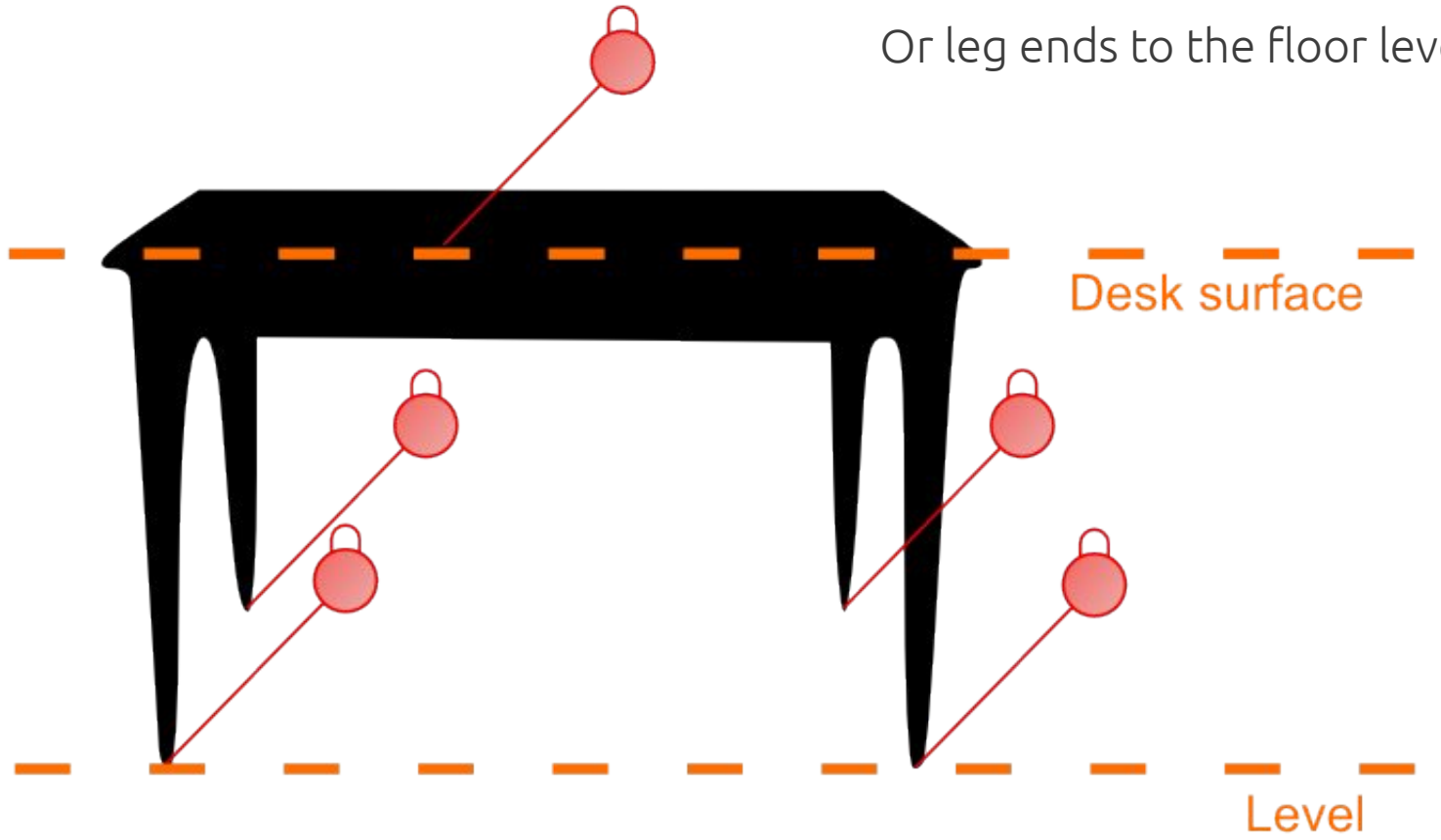
Locking geometry to Refplanes



Locking geometry to Refplanes

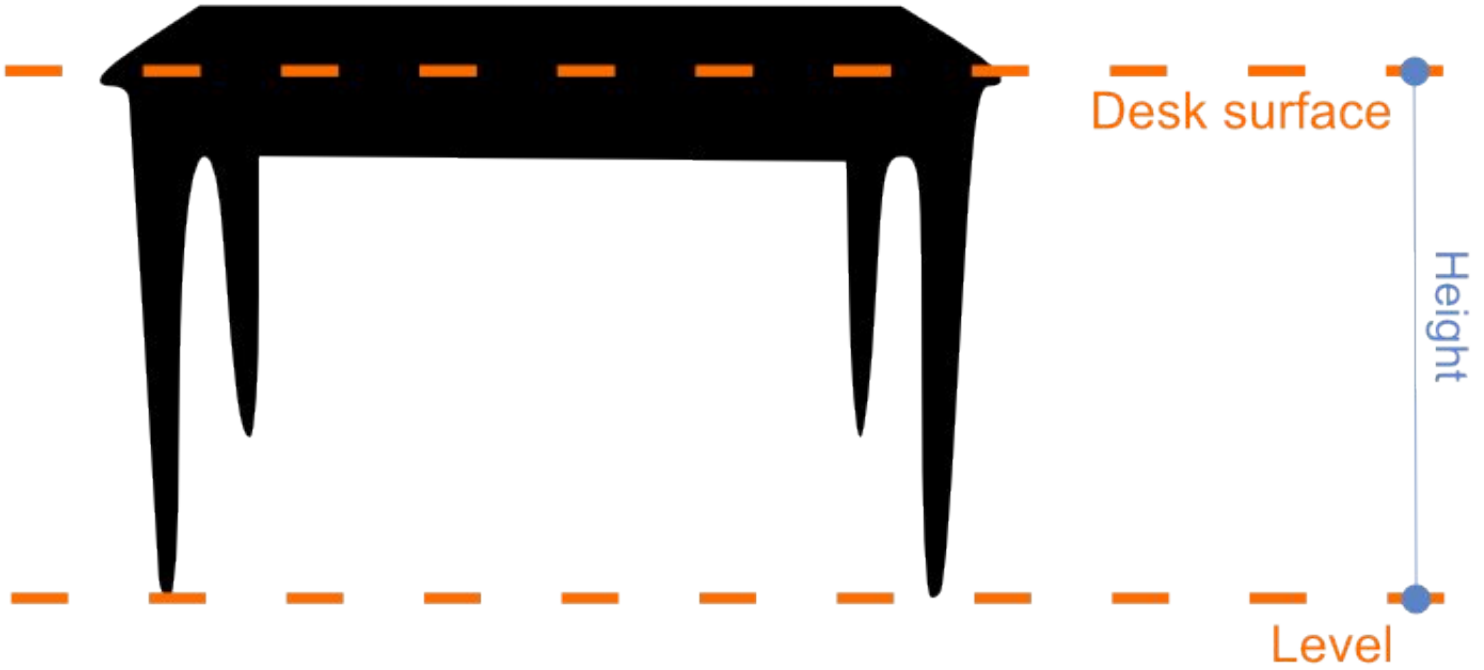
You can lock object parts to refplane.
For example, desk surface to desk surface refplane.

Or leg ends to the floor level Refplane.

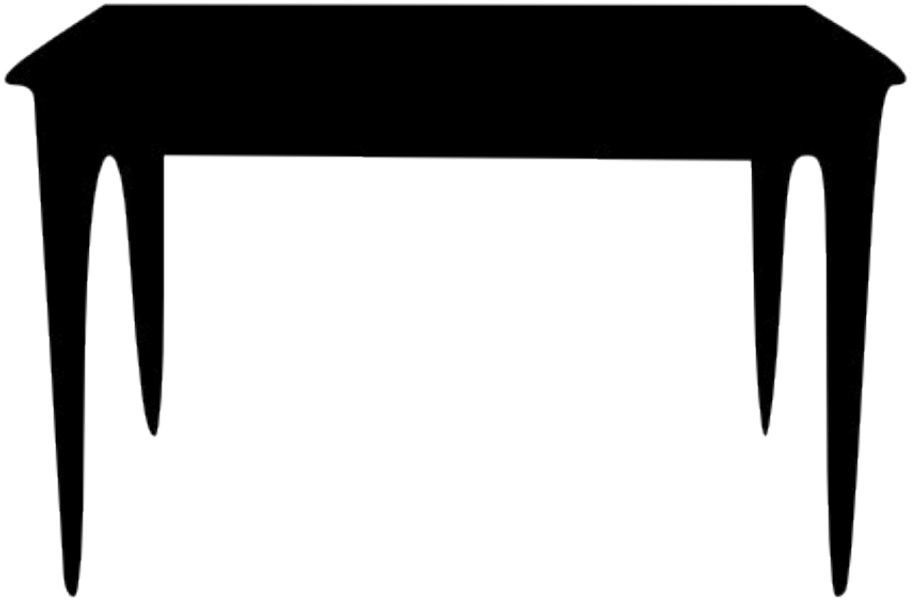


Adding parameters

Then, it is possible to add parameters between refplanes. Let's specify table legs height for instance.



Geometry changes according parameter values

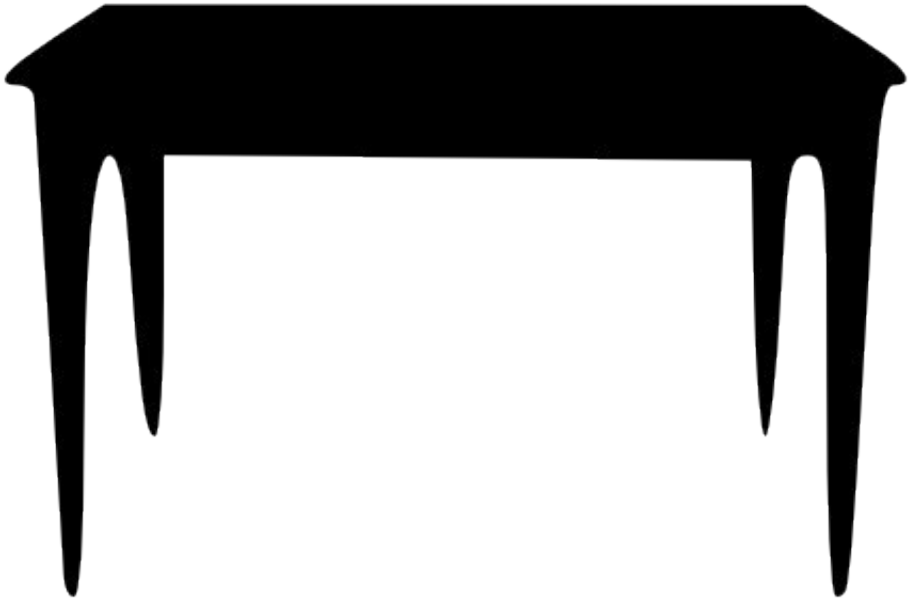


Then, when we have instance of the object, our new parameter appears in Object properties box.

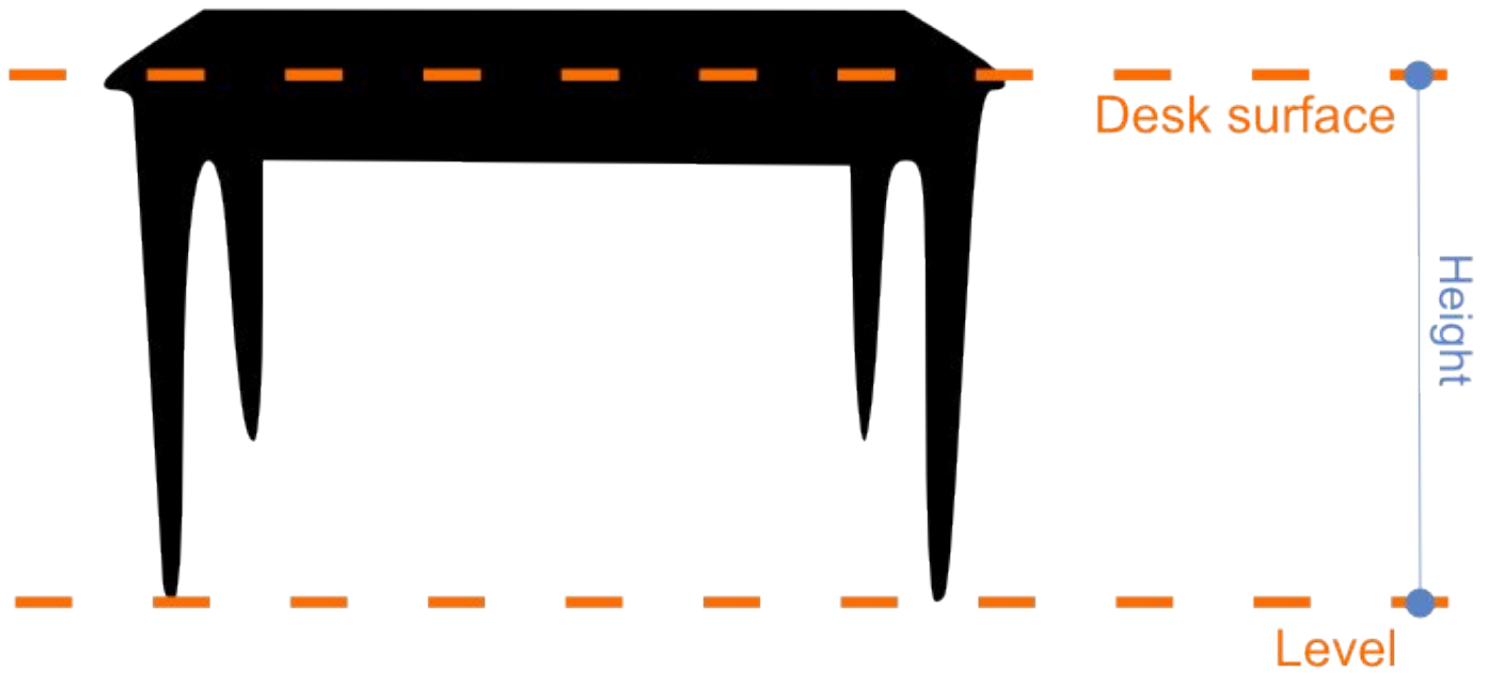
Geometry changes according parameter values

We change it, and table becomes lower.

Geometry changes according parameter values



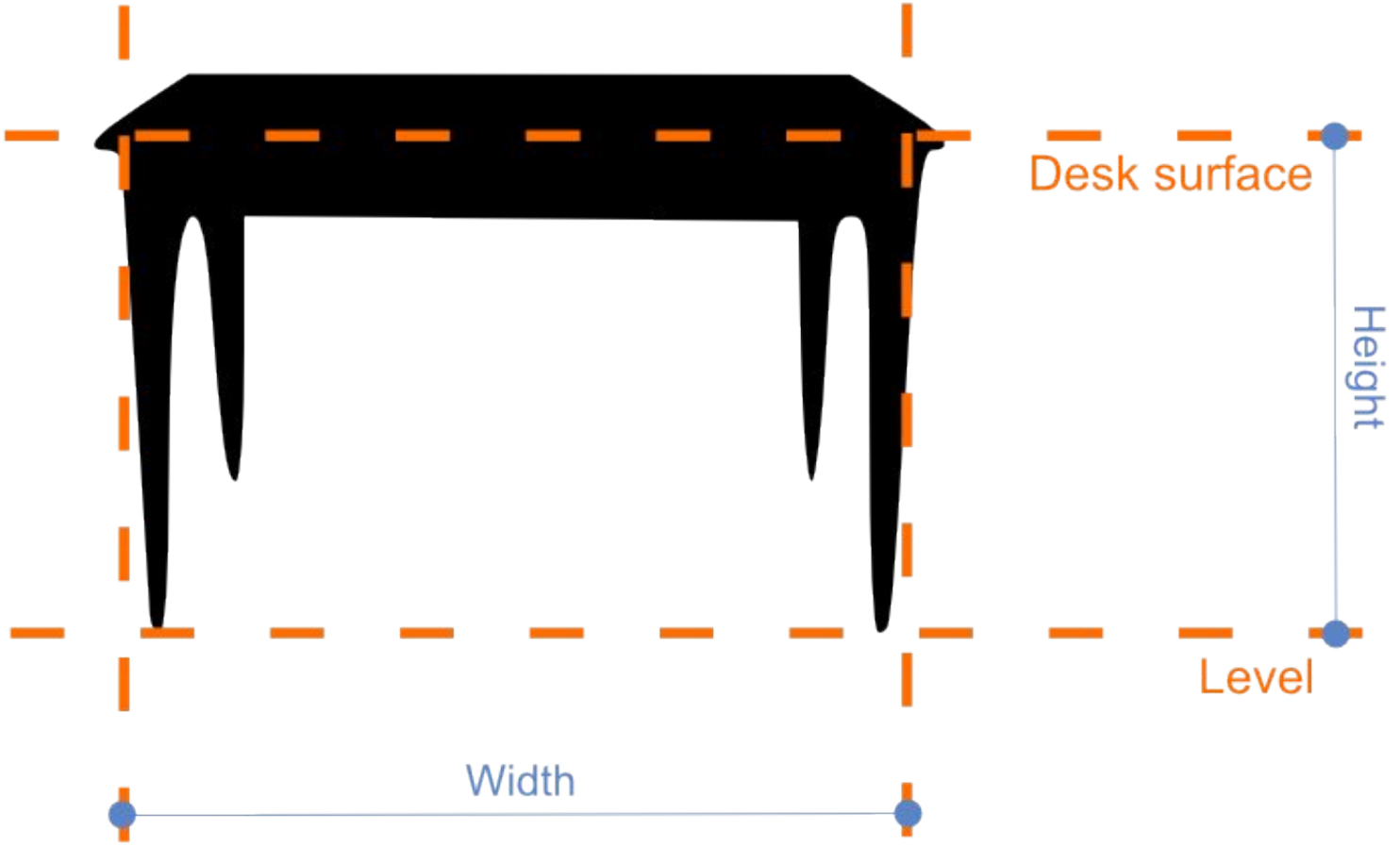
Controls



Another way to change parameter values

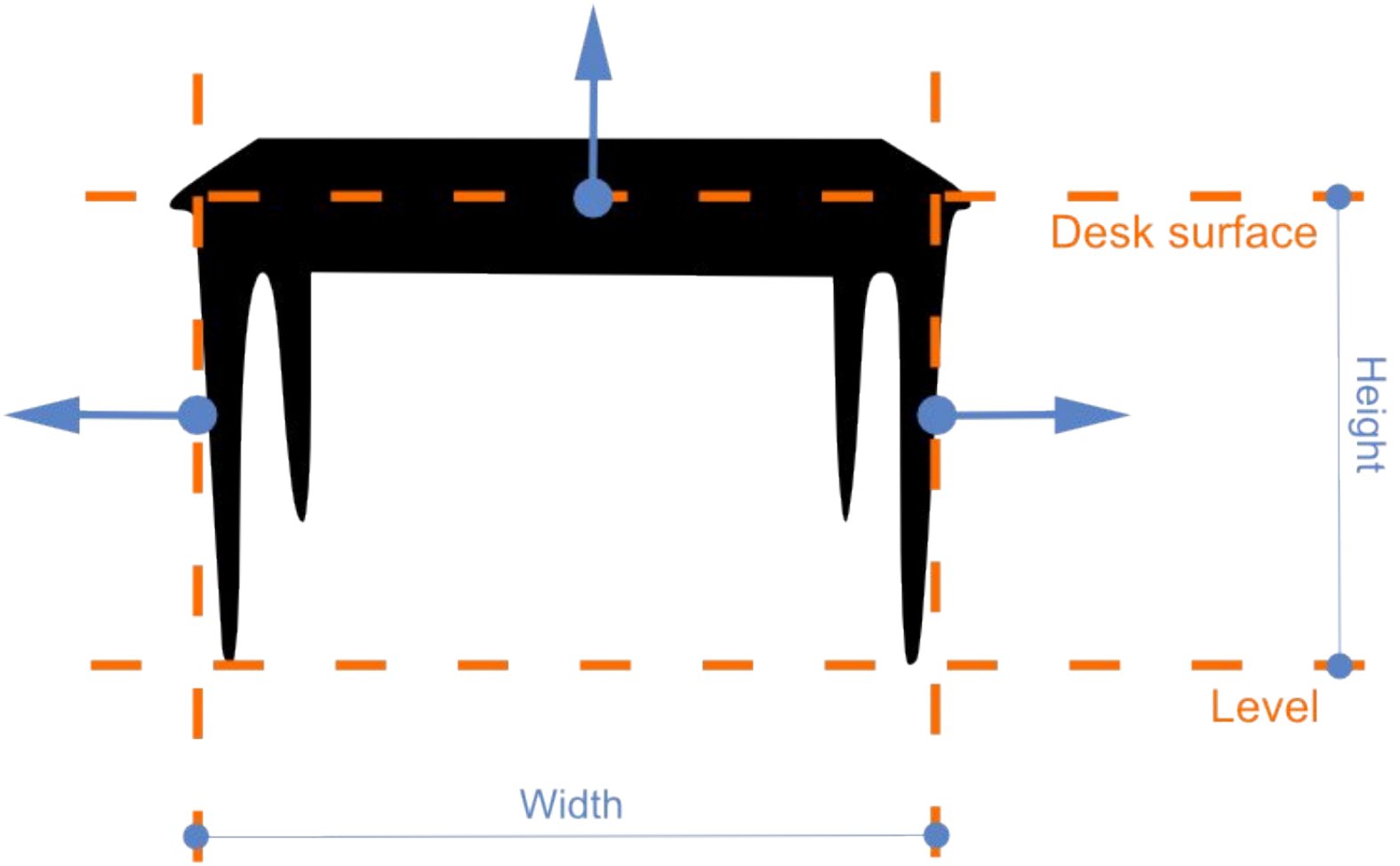
Controls

Let's add some more refplanes

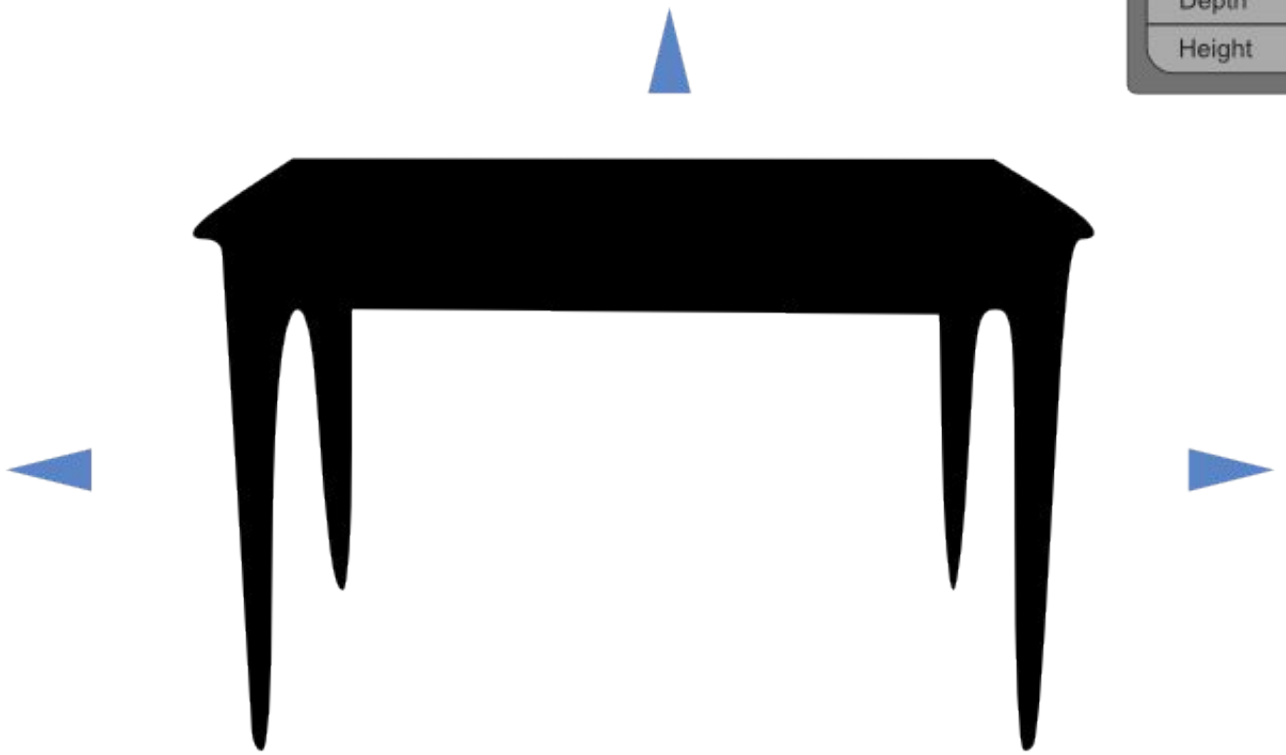


Controls

And objects defining wich refplane user could drag



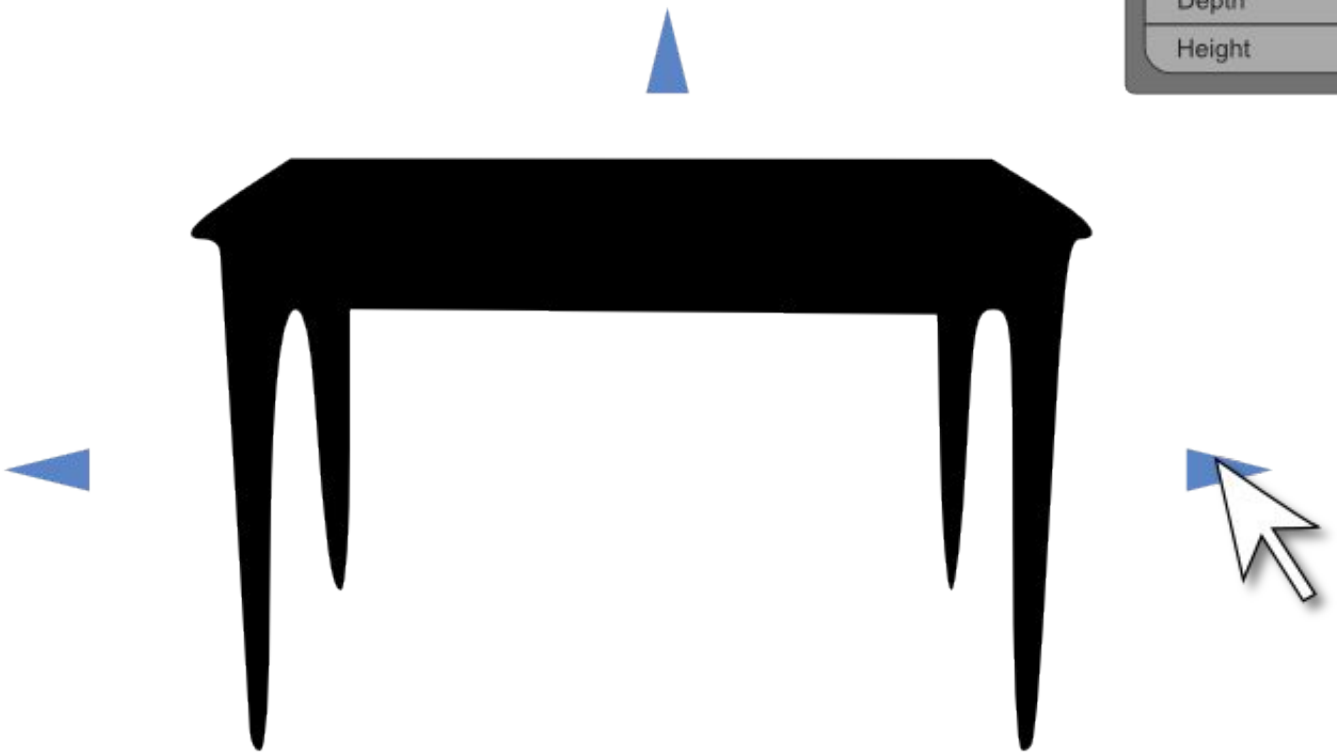
Controls



So even if we exited edid parametric component mode we can drag now hiden refplanes

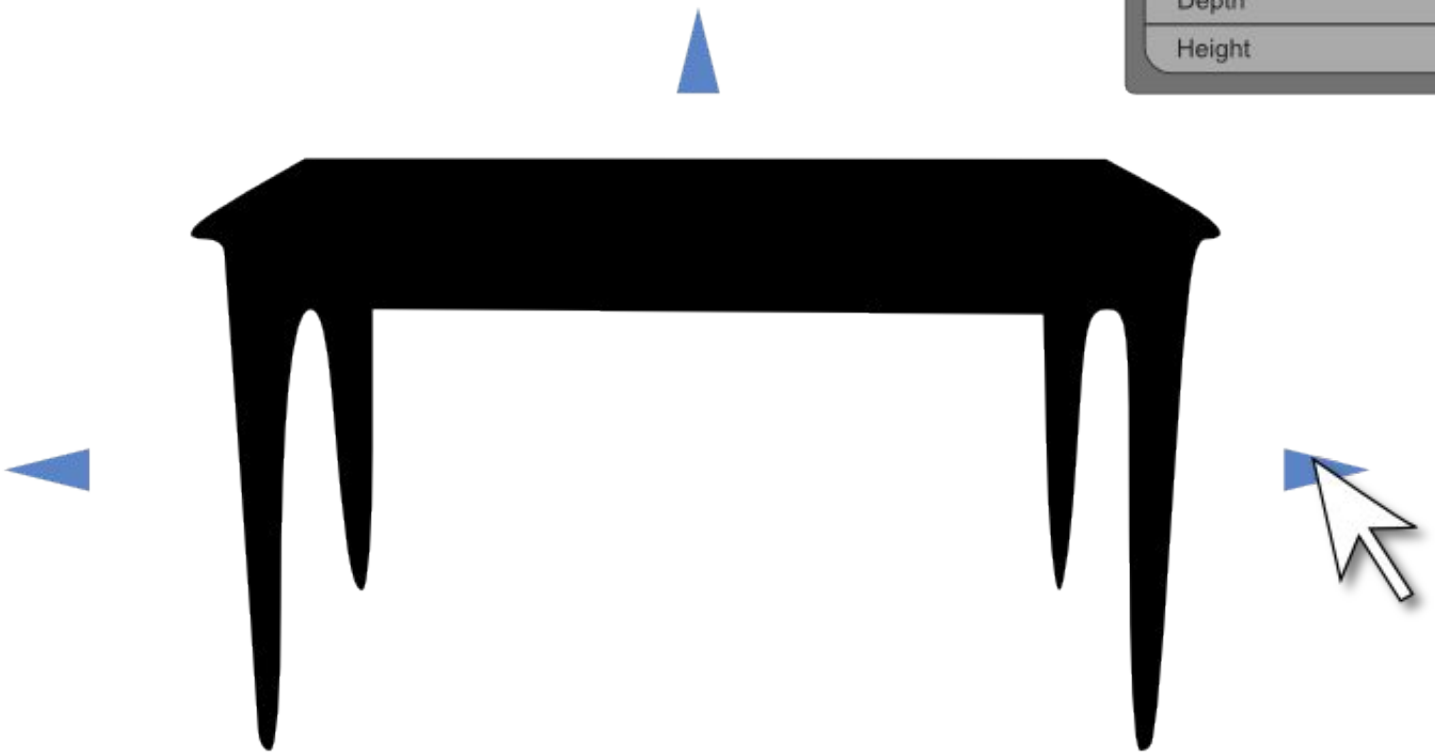
Controls


Component
Table
Hosts
Level
Level 1
Level offset
0
Spatial
Width
1200
Depth
800
Height
750

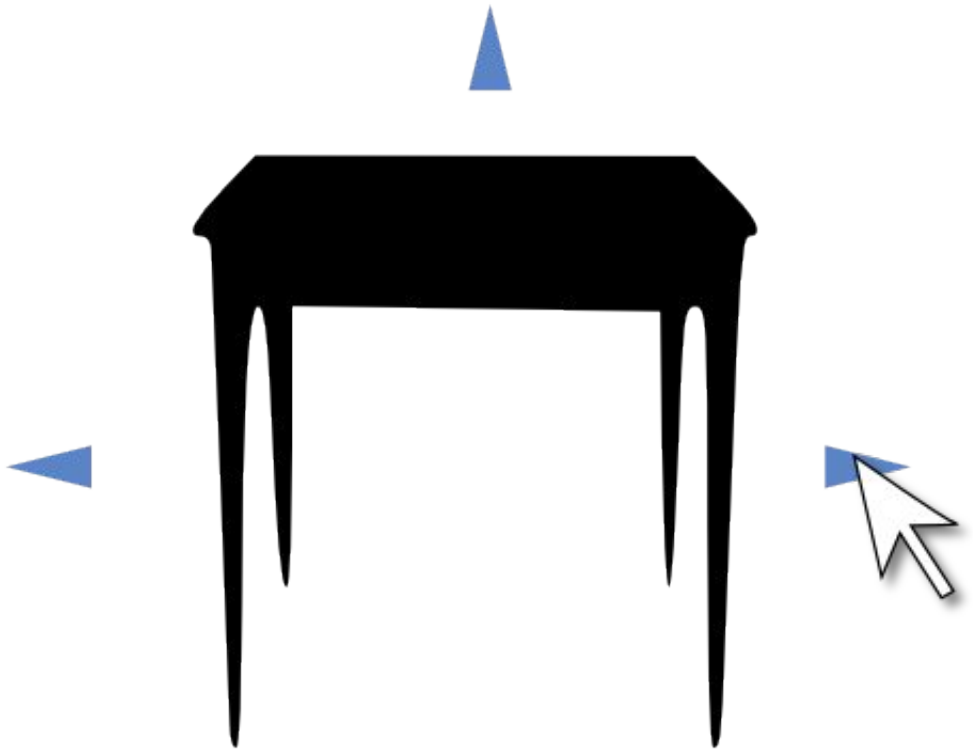


Controls


Component
Table
Hosts
Level
Level 1
Level offset
0
Spatial
Width
1330
Depth
800
Height
750

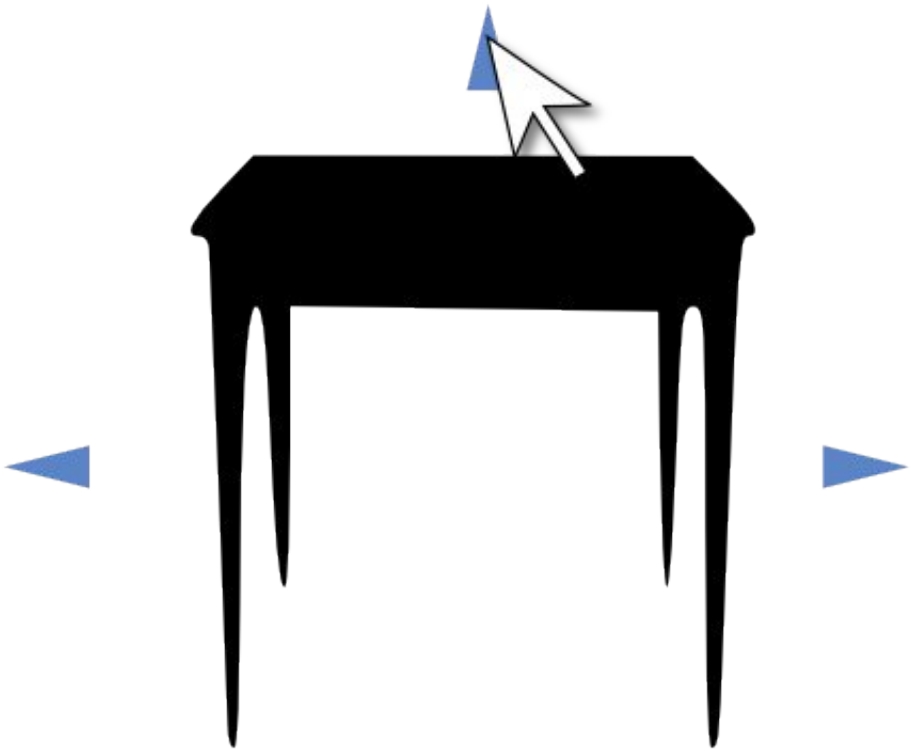


Controls




Component
Table
Hosts
Level
Level 1
Level offset
0
Spatial
Width
630
Depth
800
Height
750

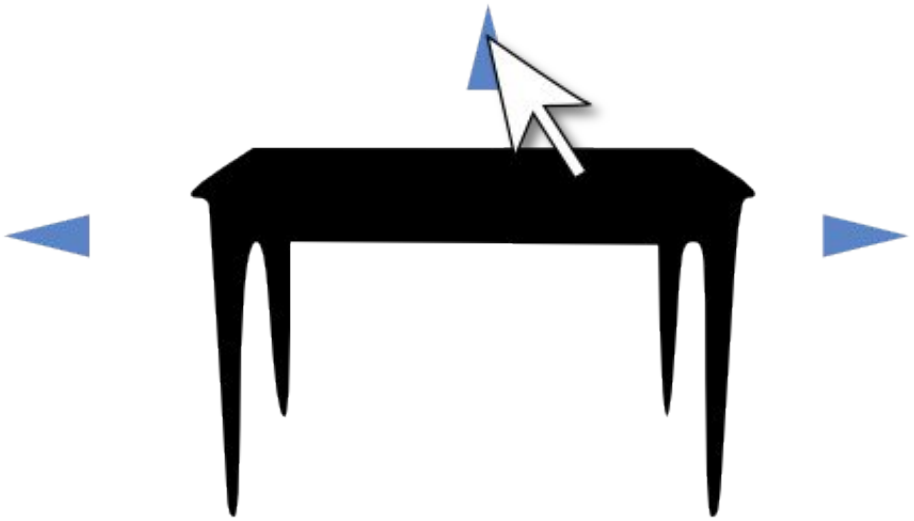
Controls




Component
Table
Hosts
Level
Level 1
Level offset
0
Spatial
Width
630
Depth
800
Height
750

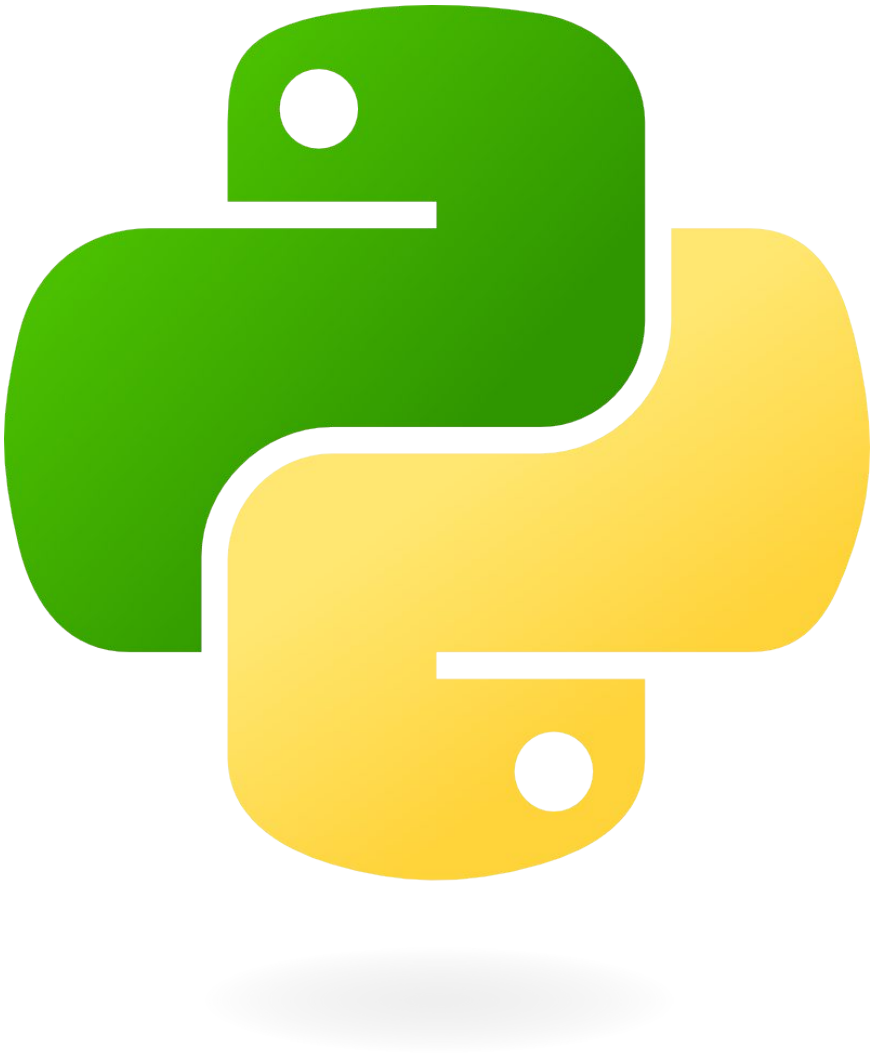
Controls


Component
Table
Hosts
Level
Level 1
Level offset
0
Spatial
Width
630
Depth
800
Height
600

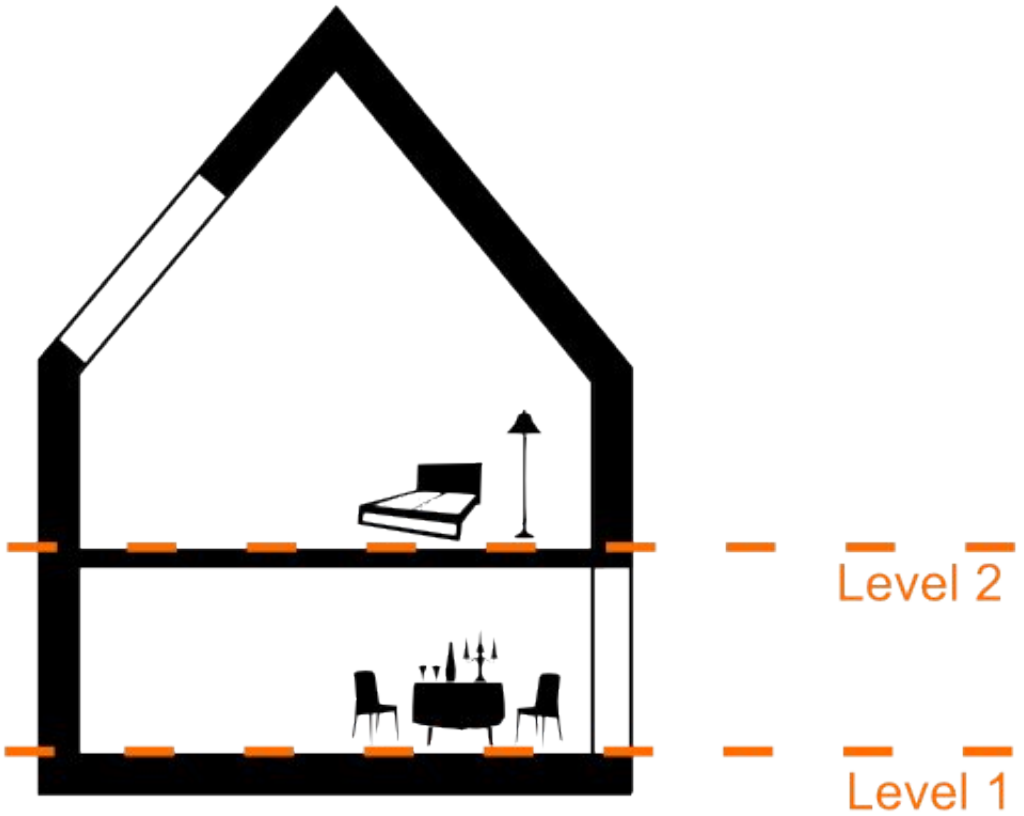


Very complex situations

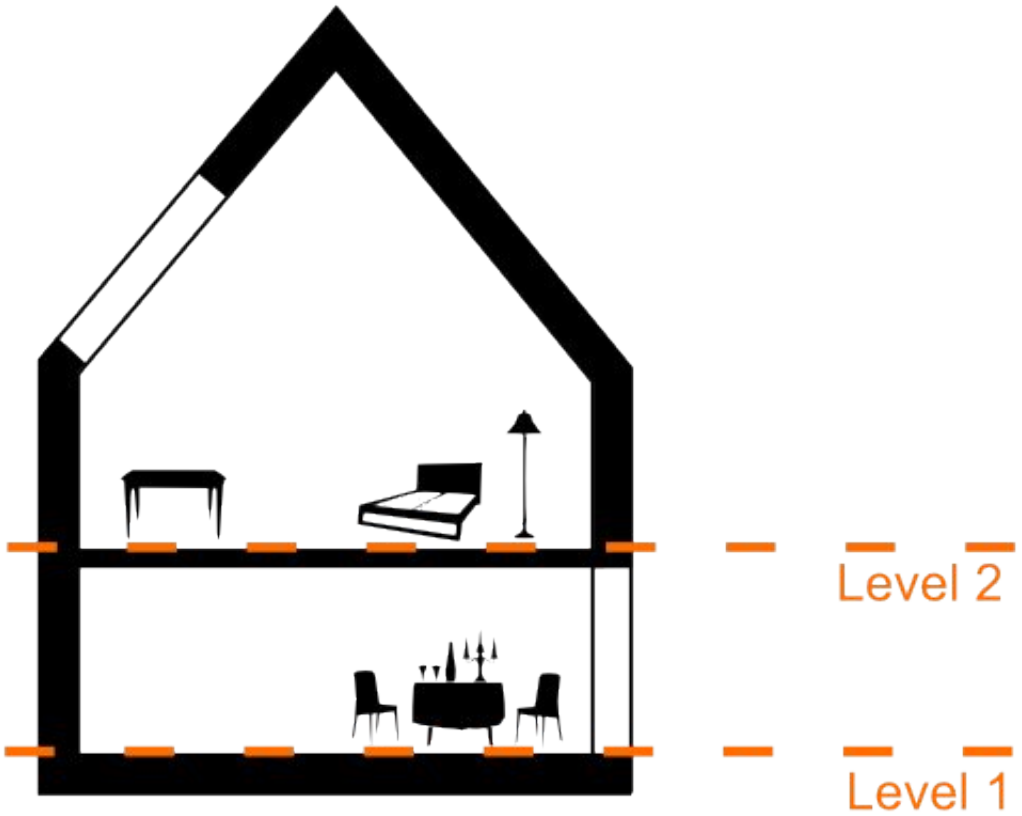
- There is nothing that couldn't be programmed by Python on the Parametric Model side
- Powerful API of POs to be developed



Refplanes and hosted components

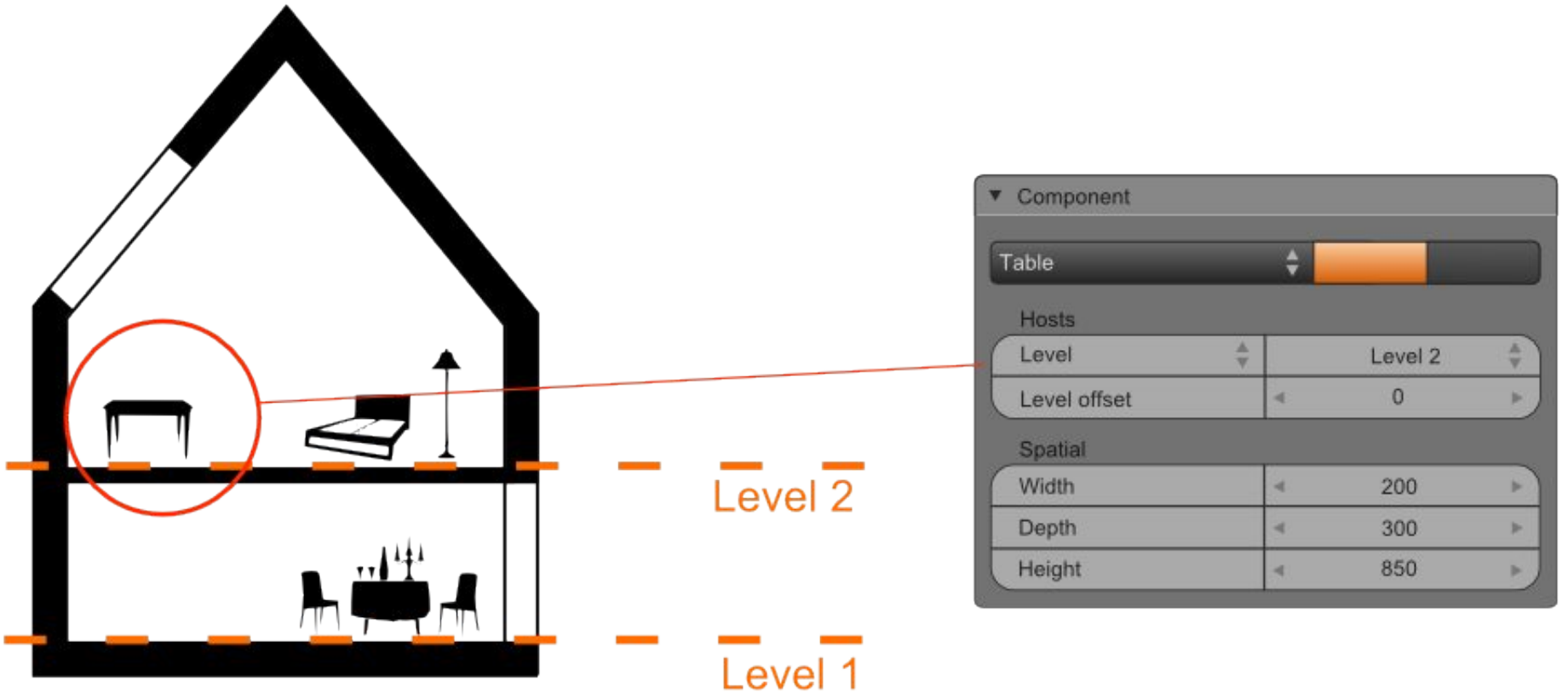


Refplanes and hosted components

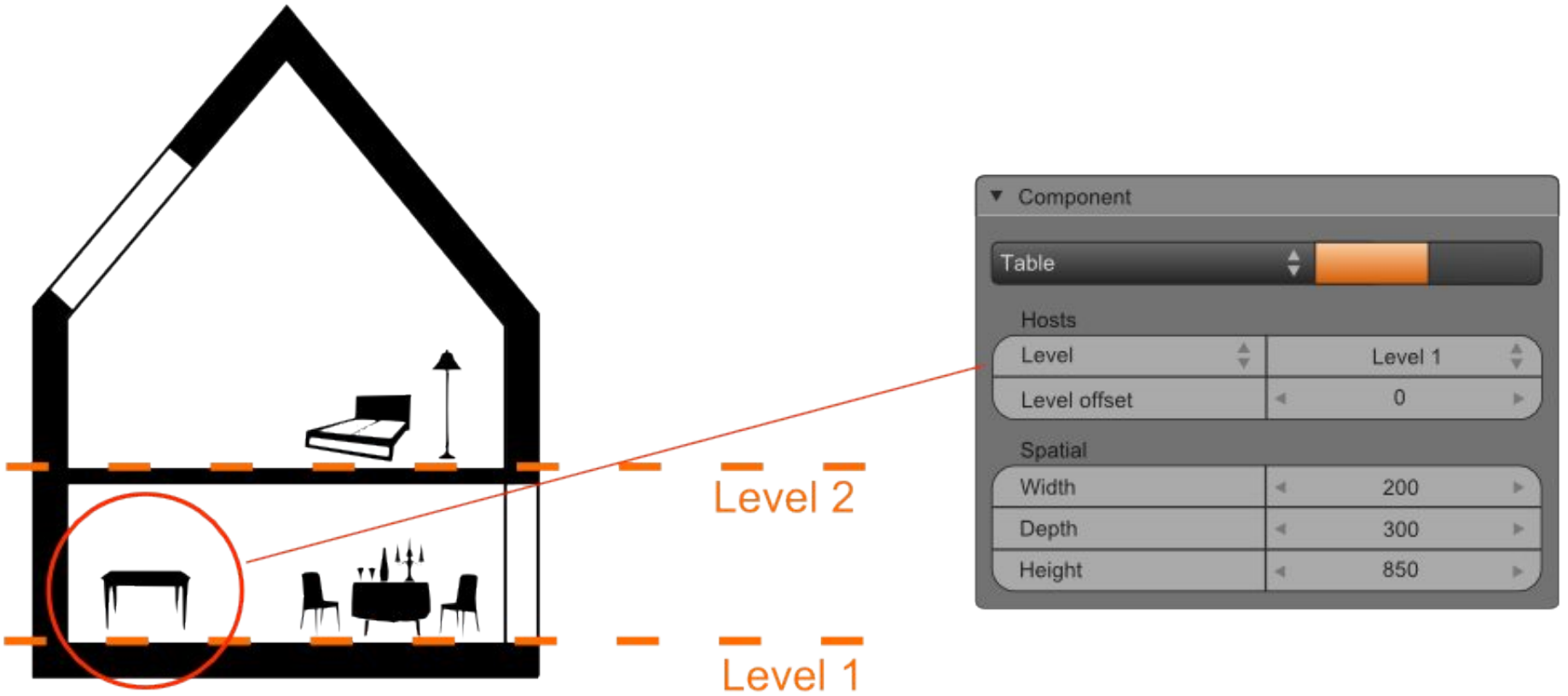


Also refplanes can be used to snap object to parent level refplane.

Refplanes and hosted components



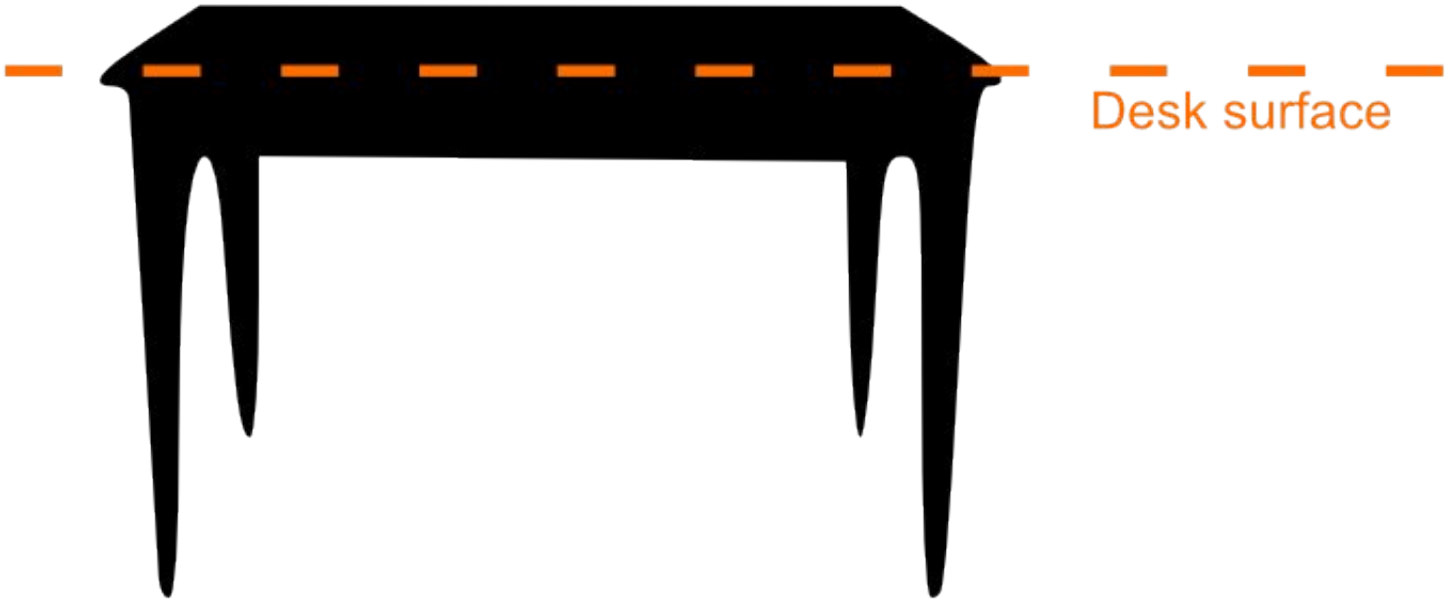
Refplanes and hosted components



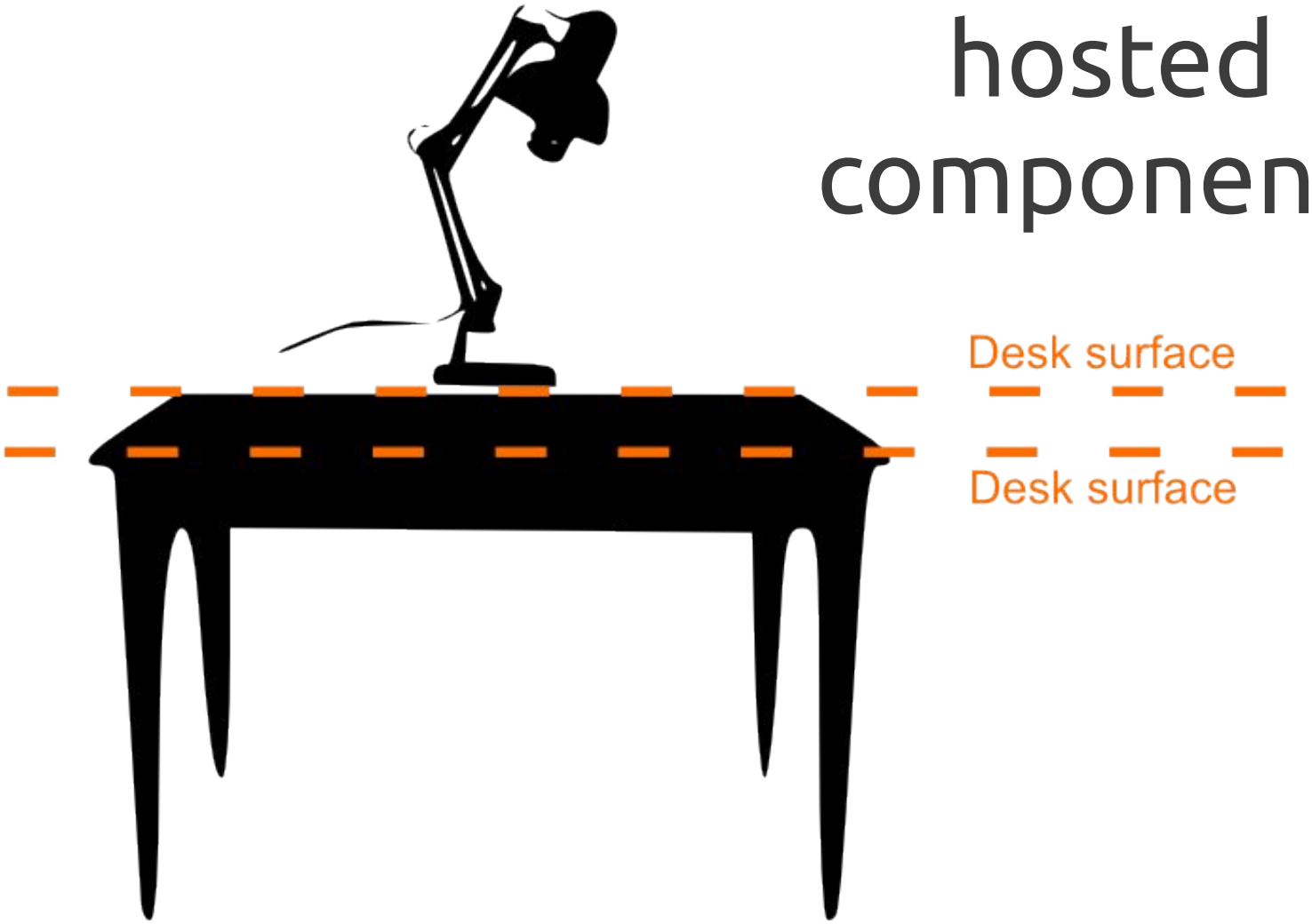
You just change hosting parameter value, and object appears in another floor
Offset allows table to levitate :-)

Refplanes and hosted components

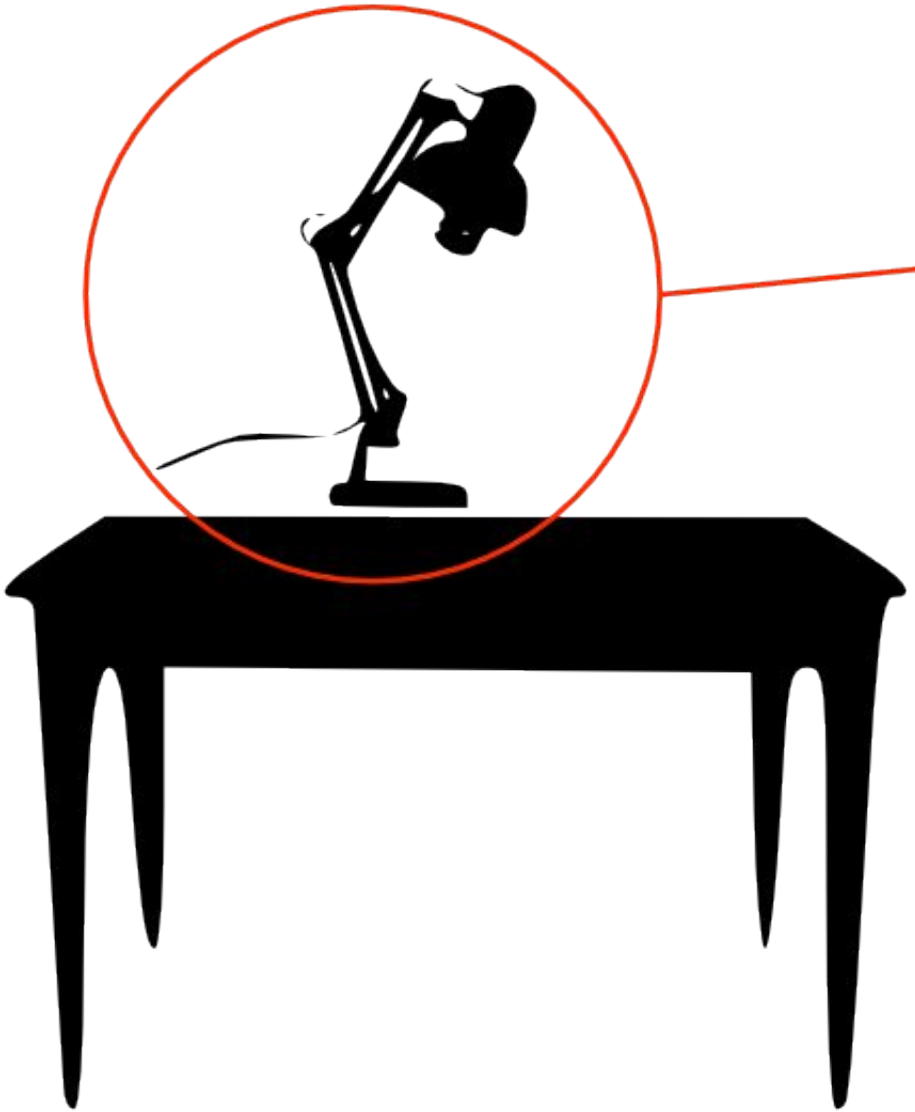
Refplanes and hosted components

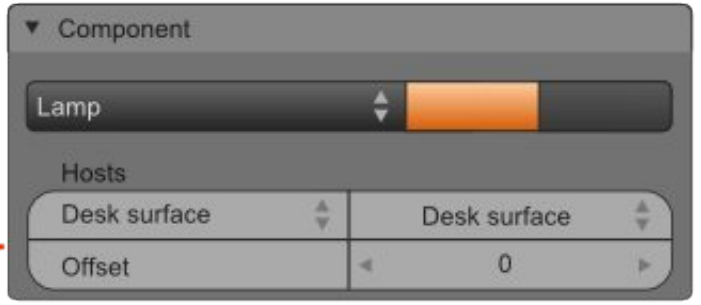


Refplanes and hosted components

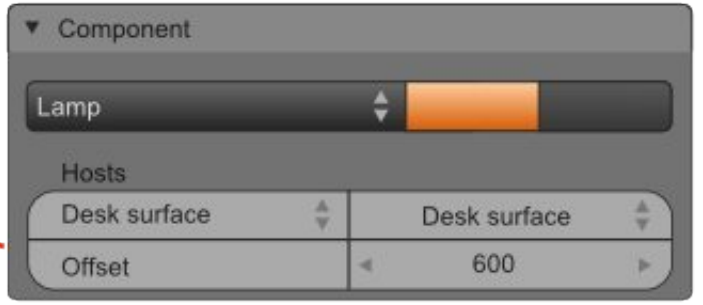


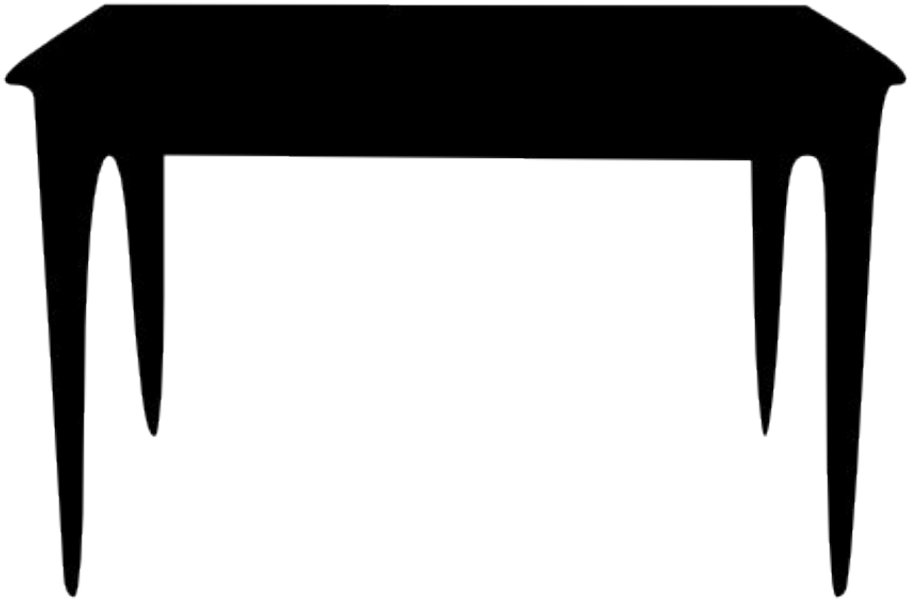
Offset from host refplane



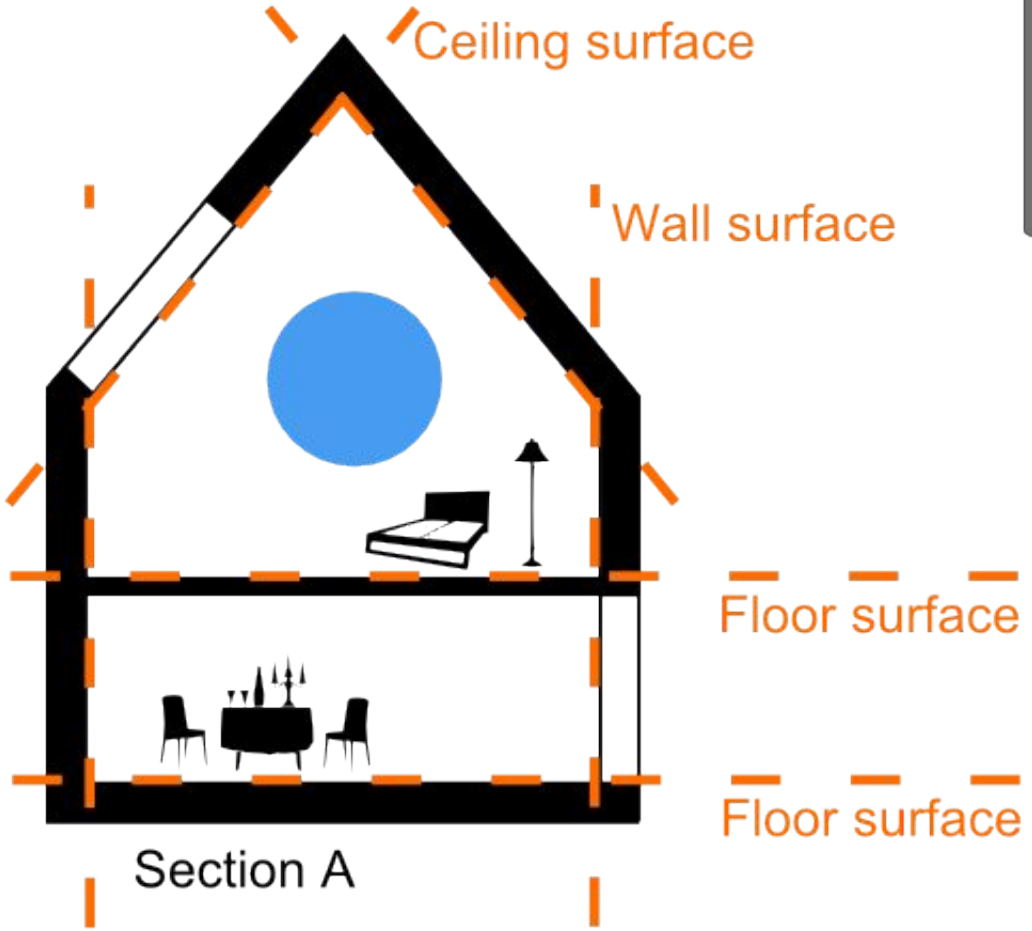


Offset from host refplane



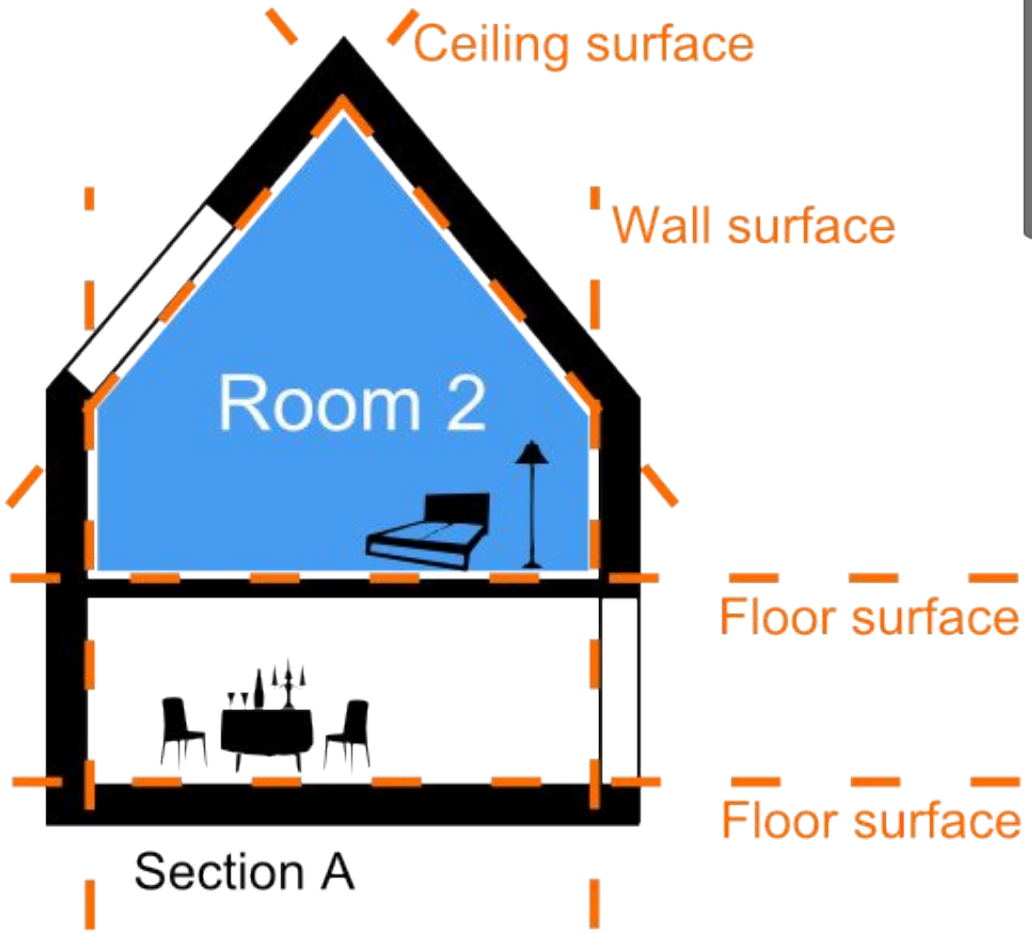


Adaptive components



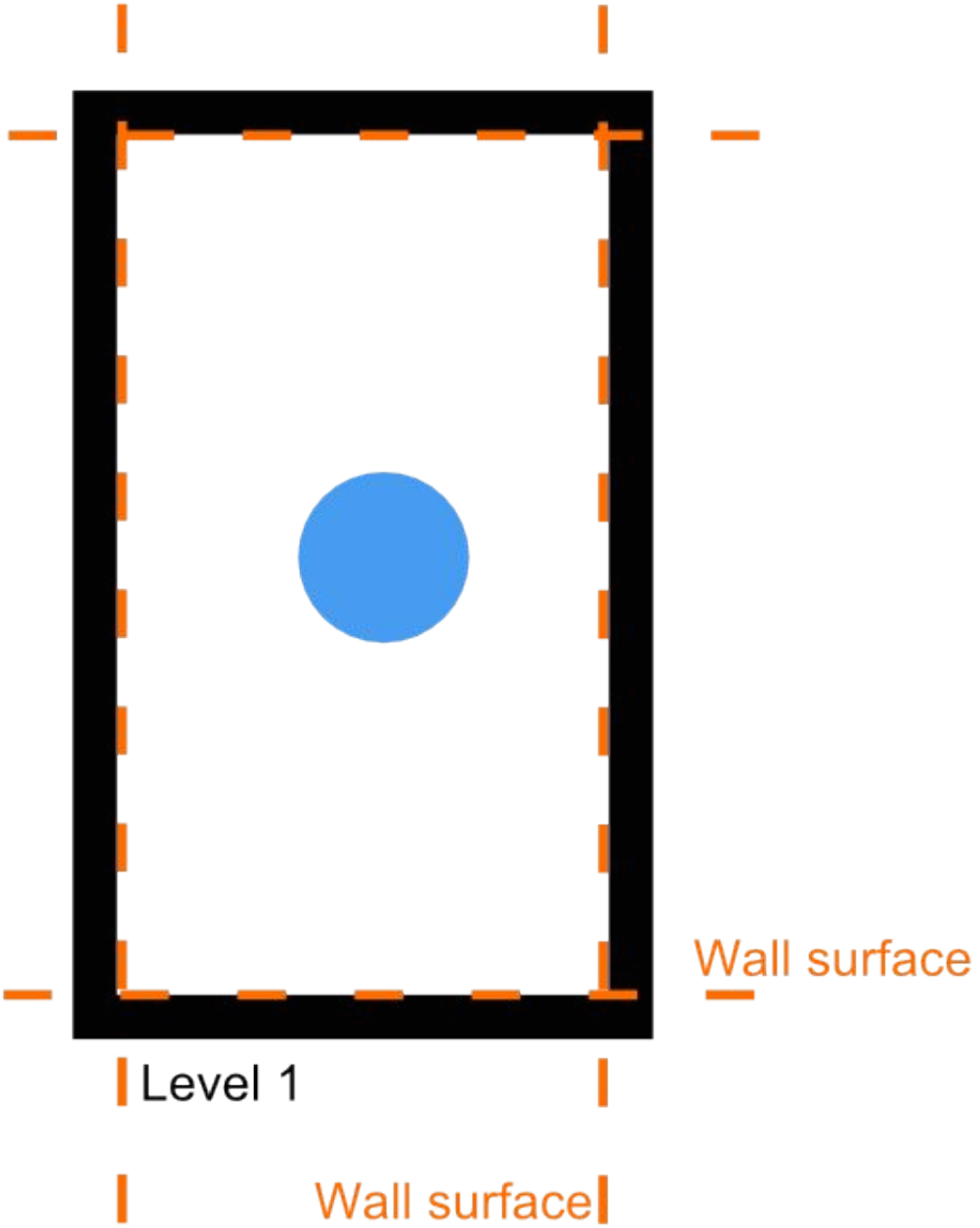
Sometimes it is required to have components that automatically expands
In desired space, delimited by desired refplanes.

Adaptive components

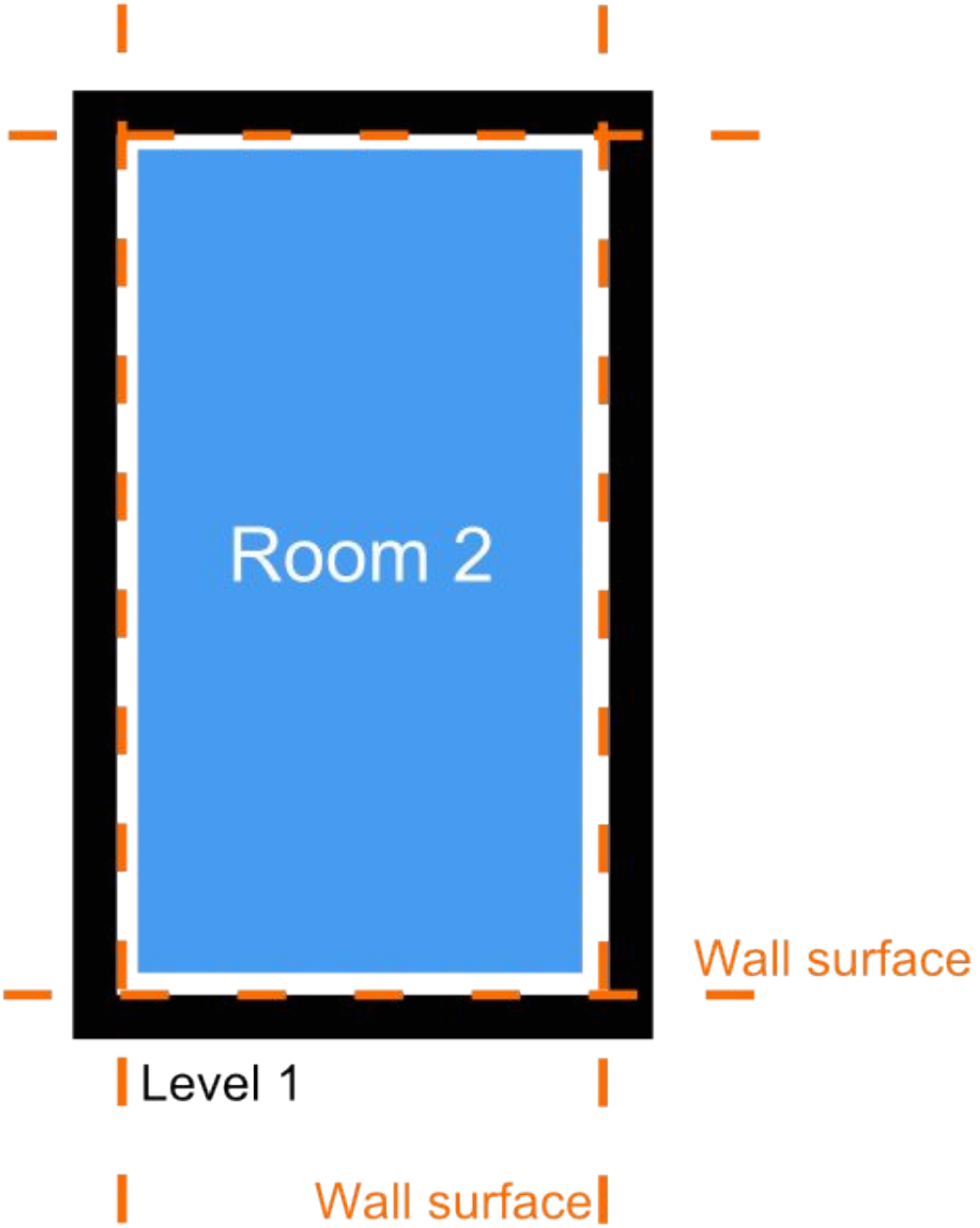


This example allows to calculate room volume. Or to specify a purpose for the area.
Living room, kitchen, lounge, etc...

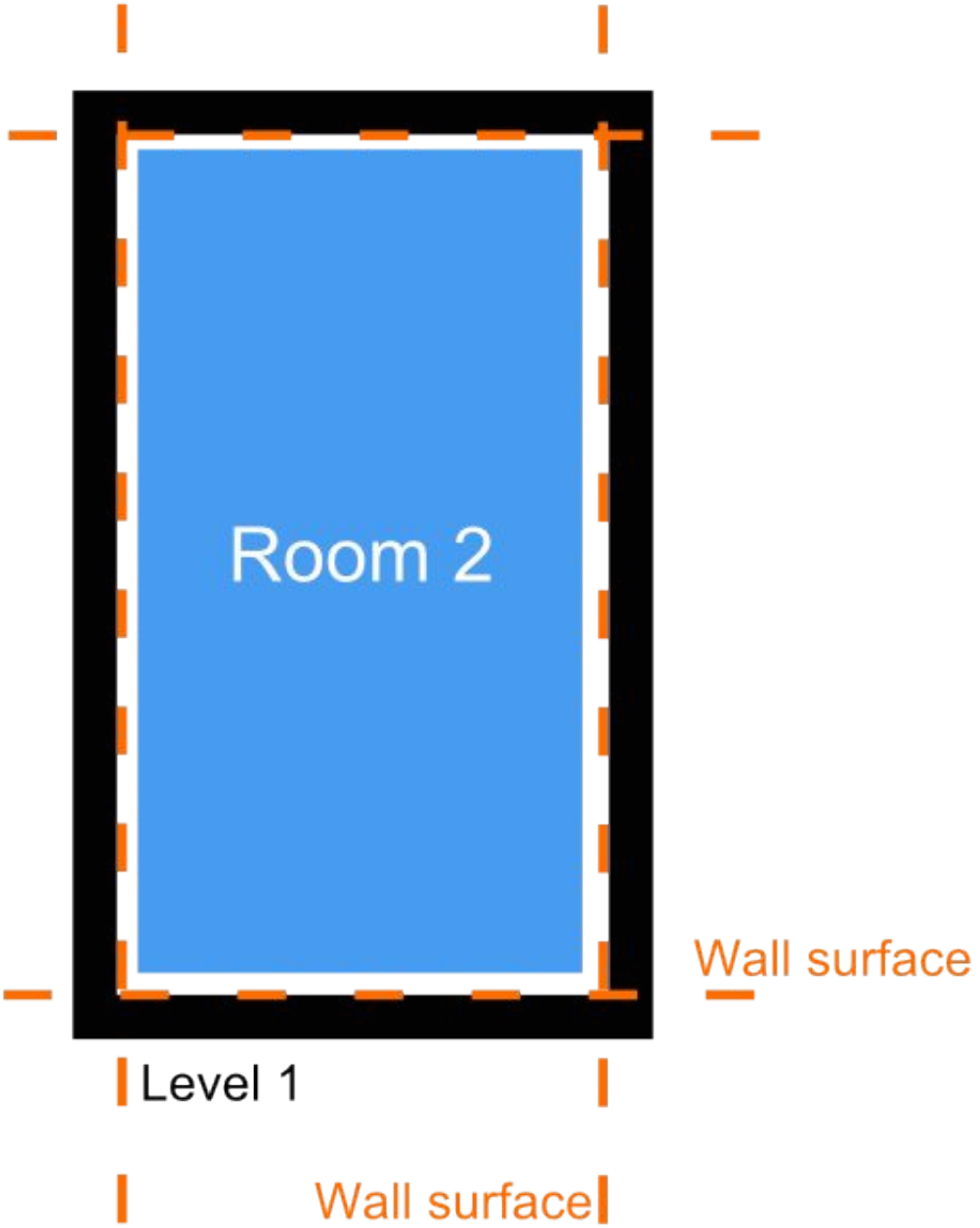
Adaptive components



Adaptive components



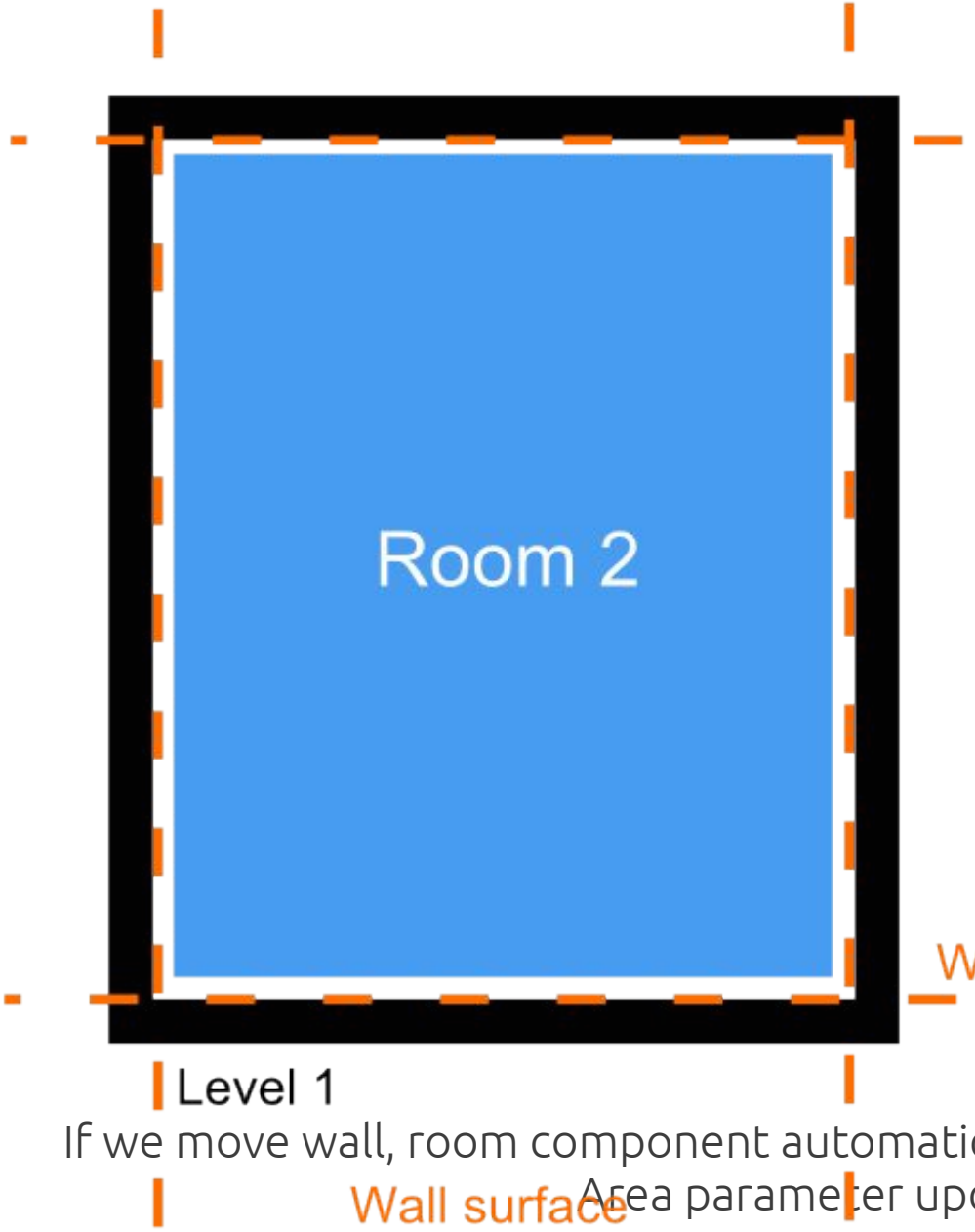
Adaptive components



Name	Area
Room 1	50 m^2
Room 2	60 m^2

In this example (plan view)
We calculate area of floors by counting room components and showing them in shedule.
Shedule column shows area parameter value of room components.

Adaptive components

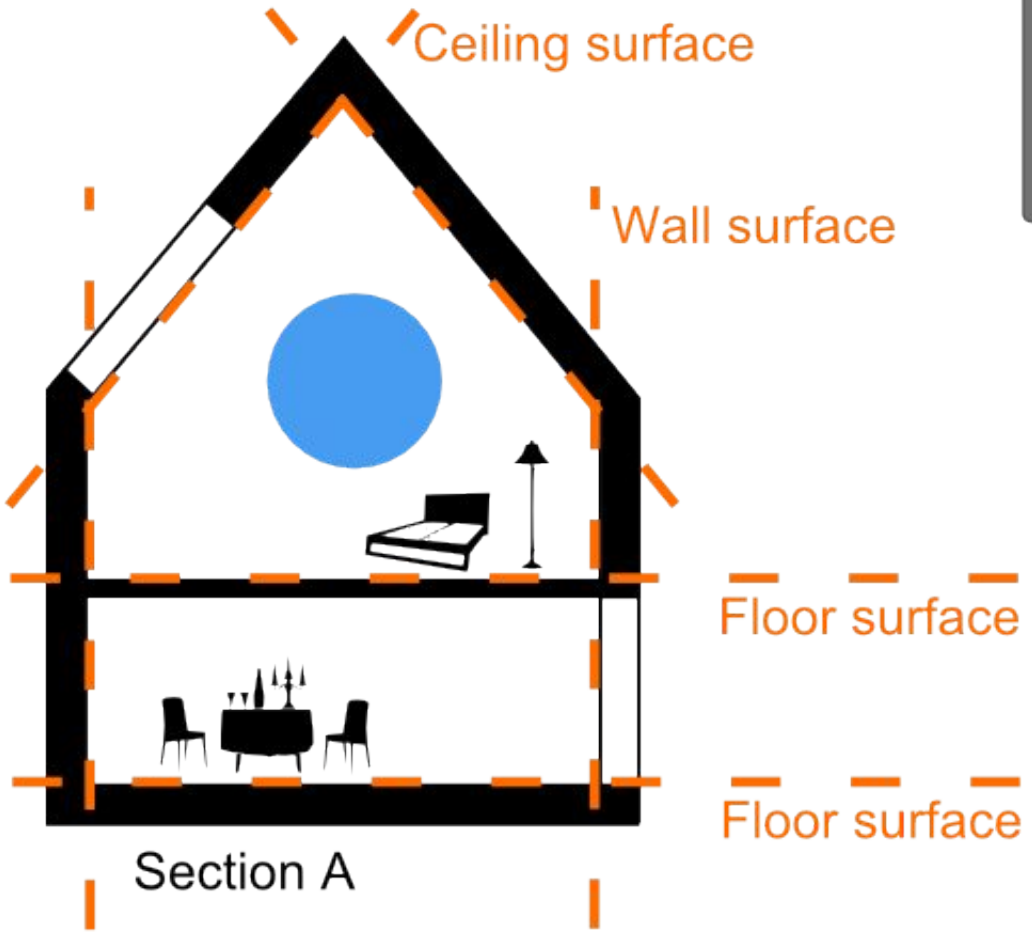


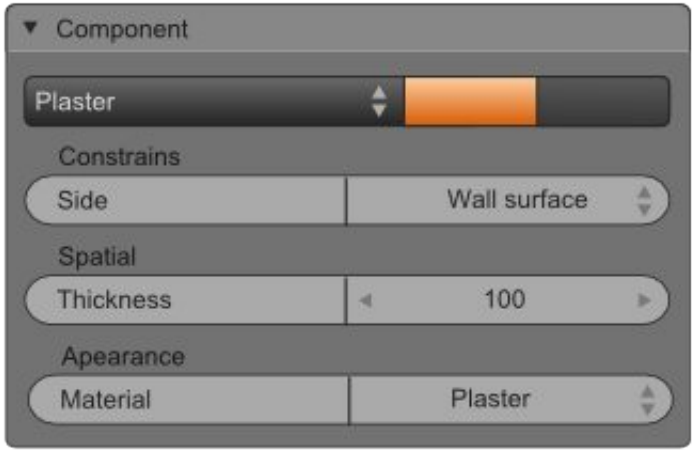
Name	Area
Room 1	50 m^2
Room 2	80 m^2

If we move wall, room component automatically extends to fill new space.

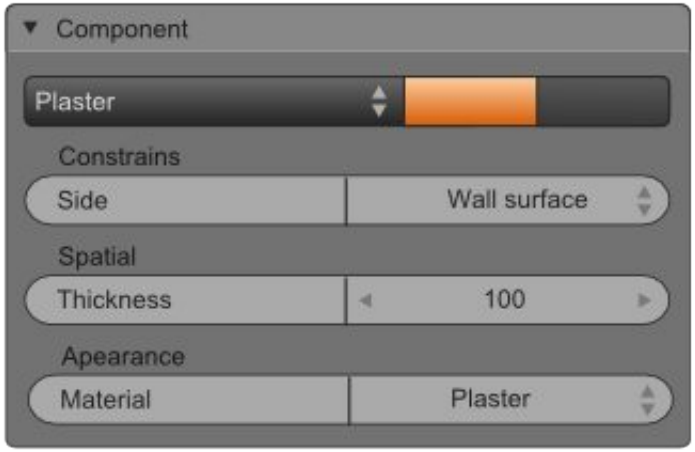
Area parameter updates

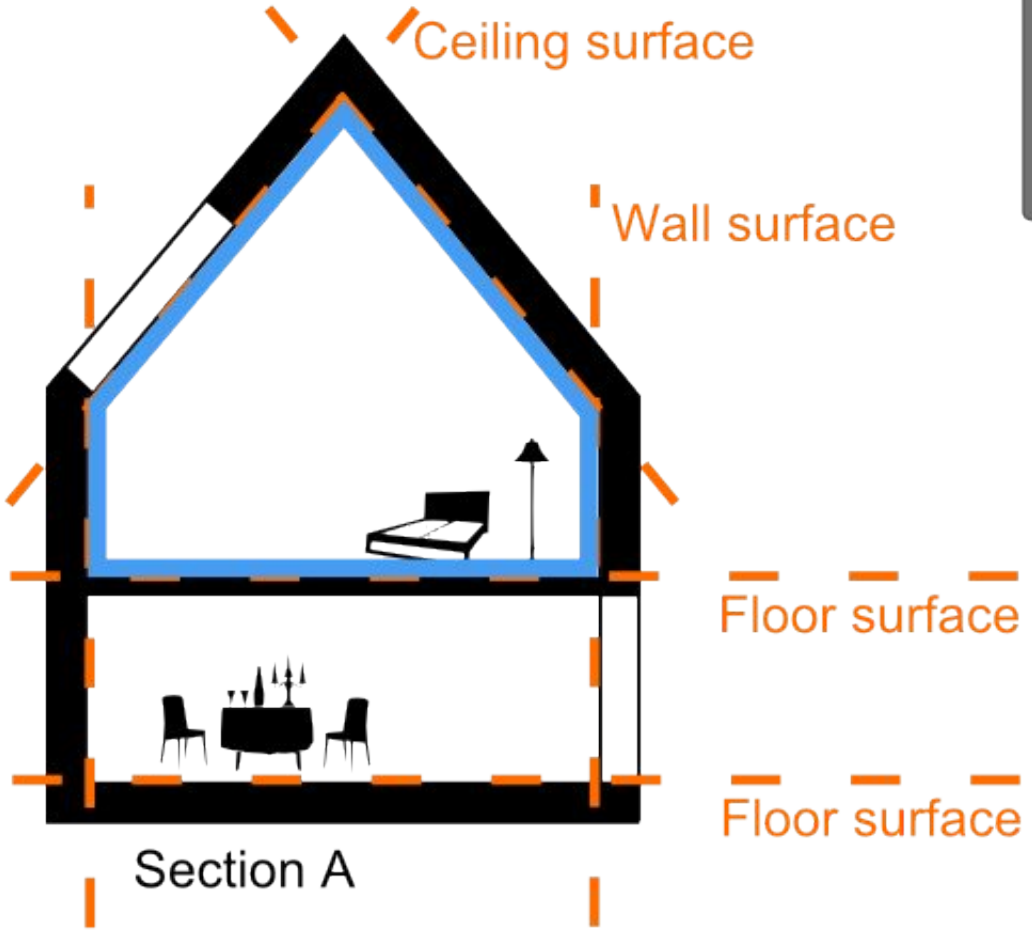
Adaptive components





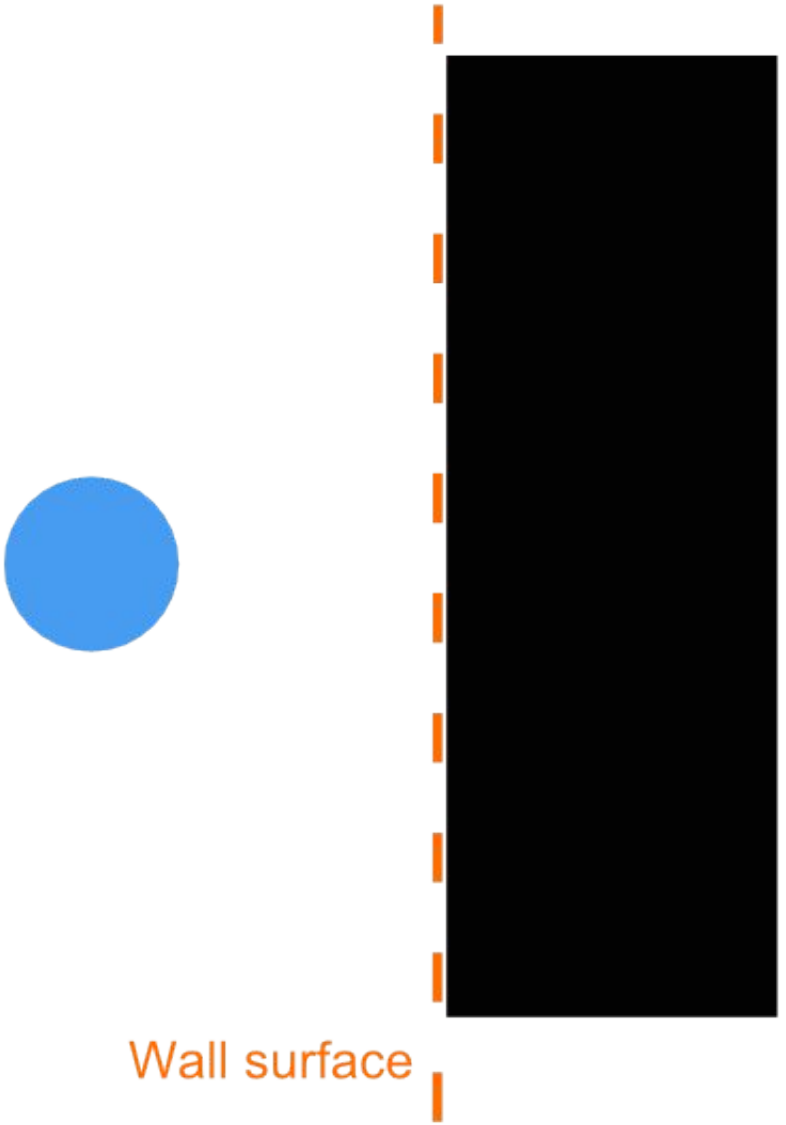
Adaptive components

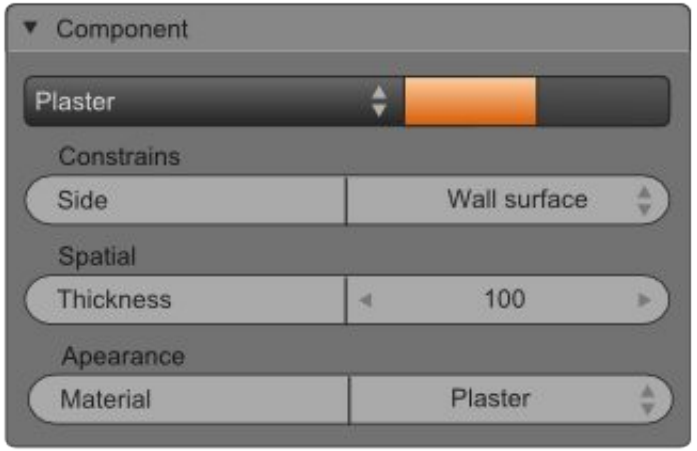




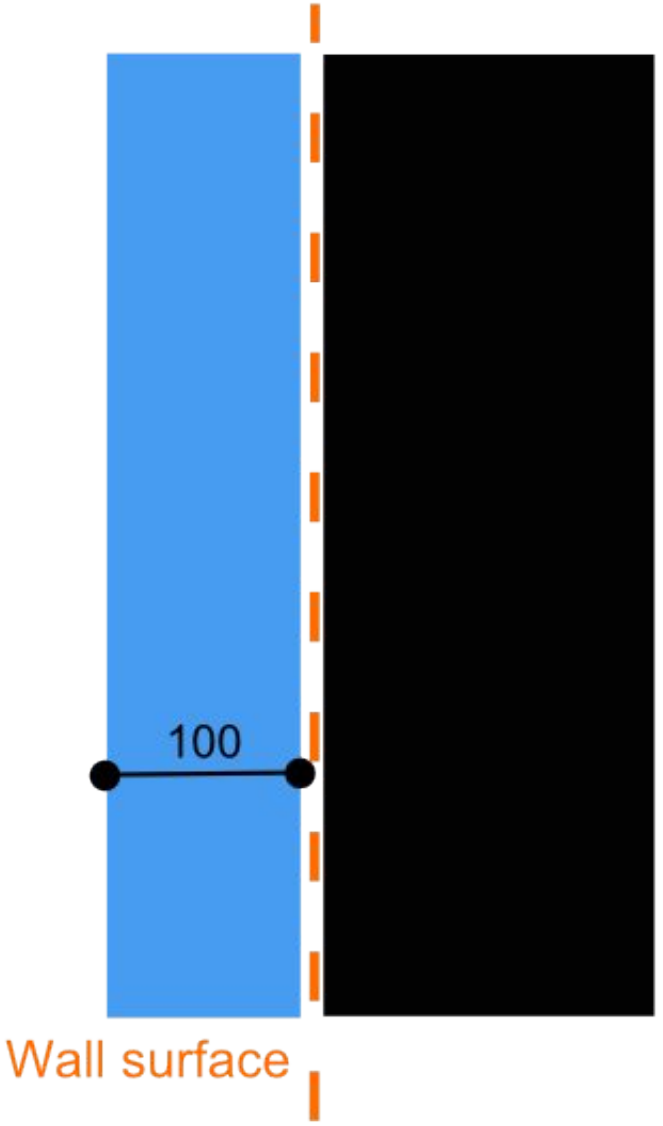
This tool can be also used to add some materials onto all walls, for instance, plaster or paint.

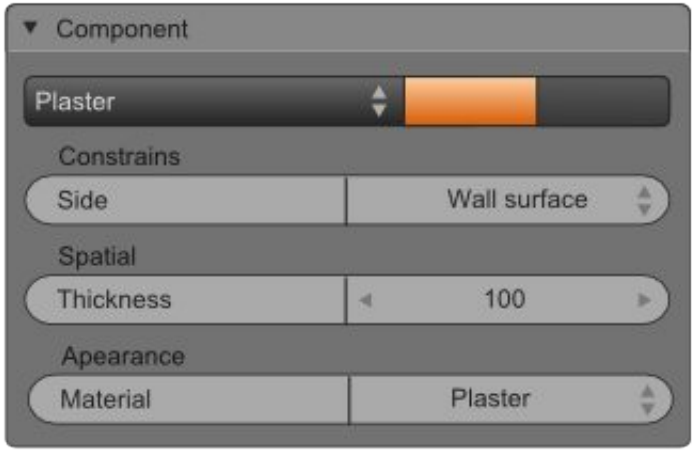
Adaptive components





Adaptive components



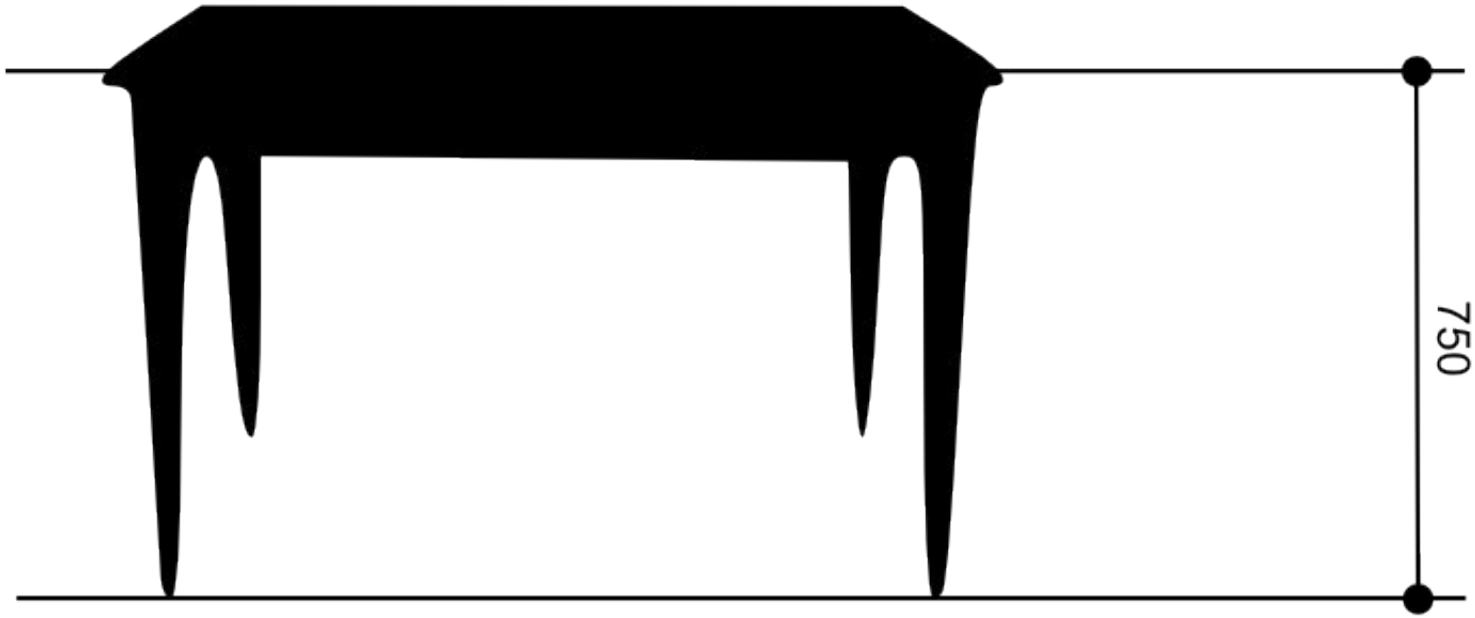


Example of applying some plaster to the wall.

Refplane and adaptive components benefits to mainstream version of Blender

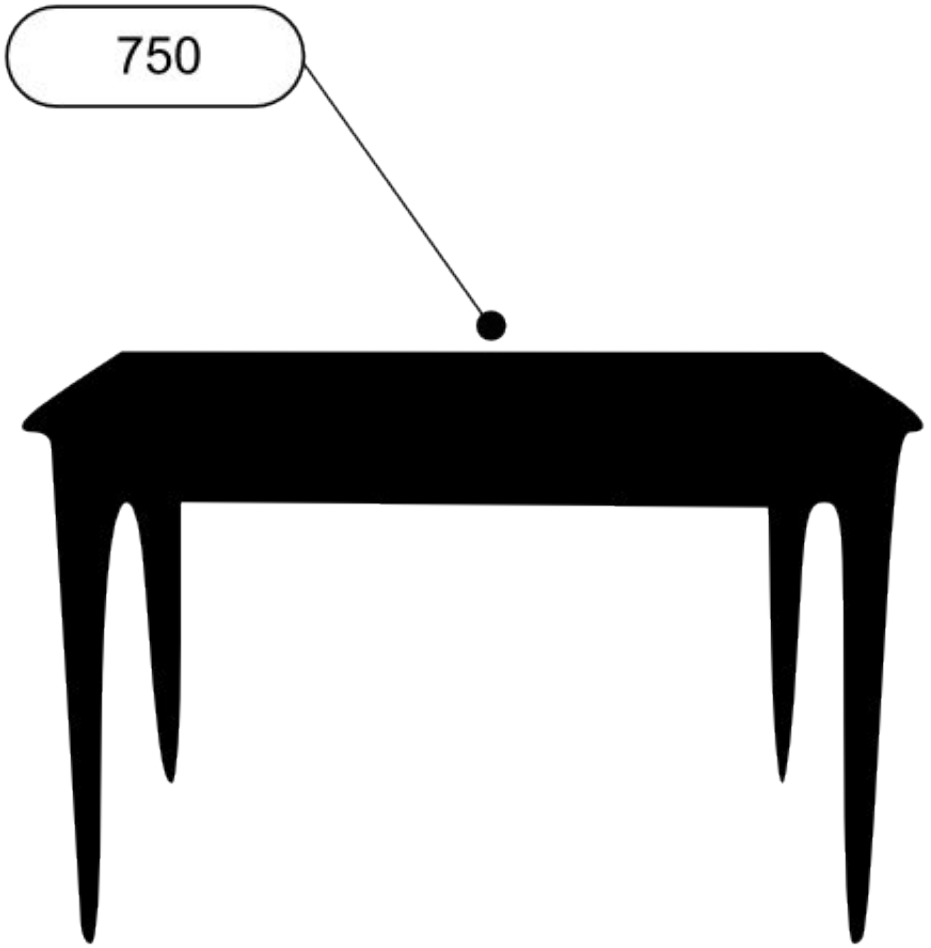
- Optimized workflow
- Increased flexibility
- Automatic object resizes, placements, etc.
- More generalized, less repeating objects
- Adaptive components saves time and hassle

Dimension tool



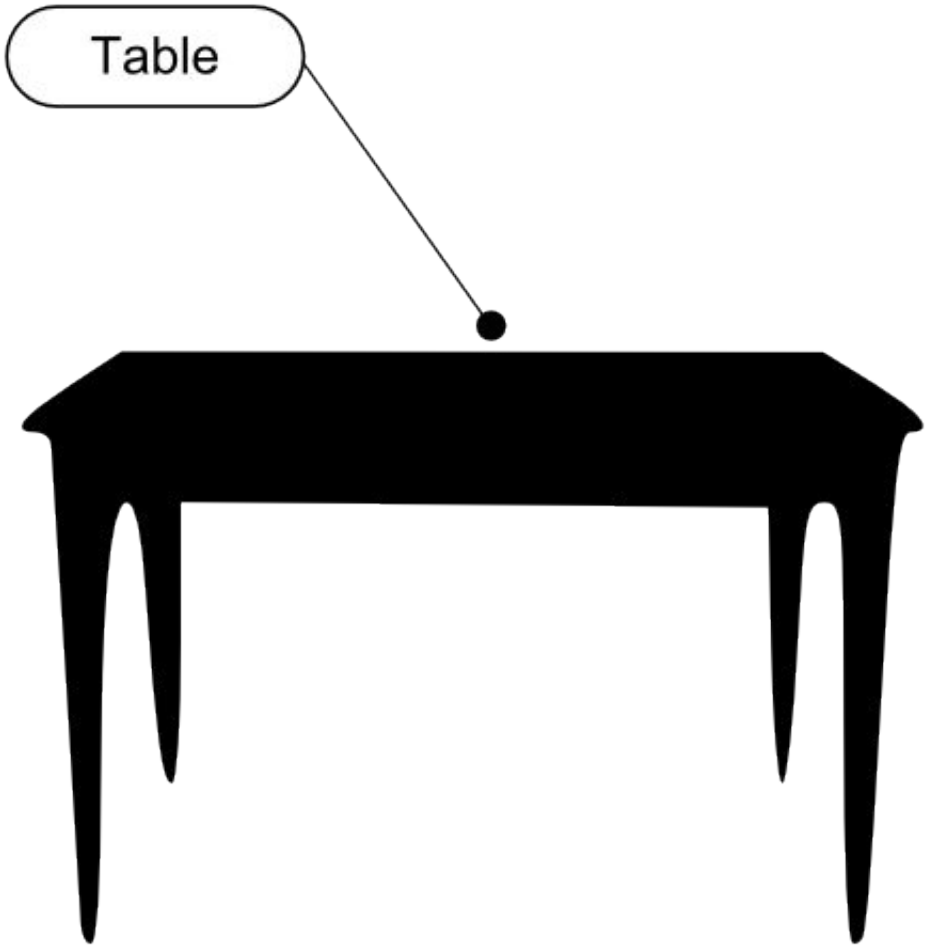
To make drawings display actual information, it is vital to have the ability to draw dimensions for some important distances, angles or altitudes

Tag



This 2D component displays particular parameter value of another component

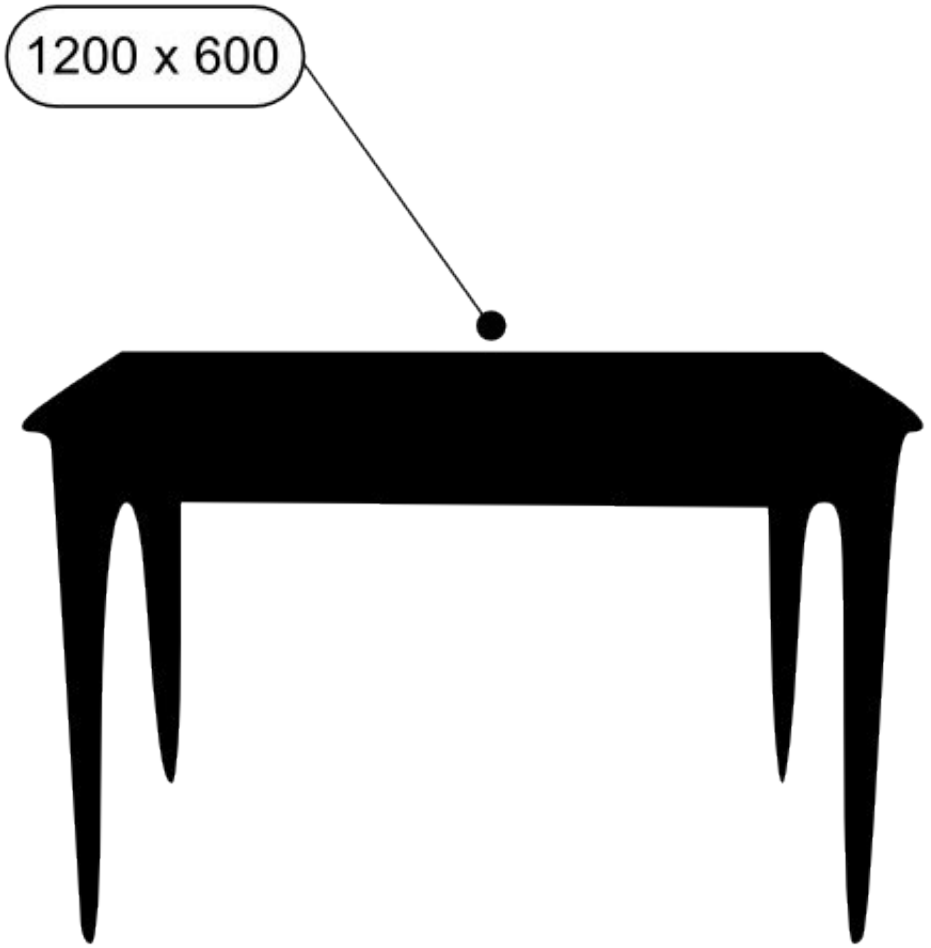
Tag



Table


Component
Table
Hosts
Level
Level 1
Level offset
0
Spatial
Width
1200
Depth
600
Height
750

Tag



Component

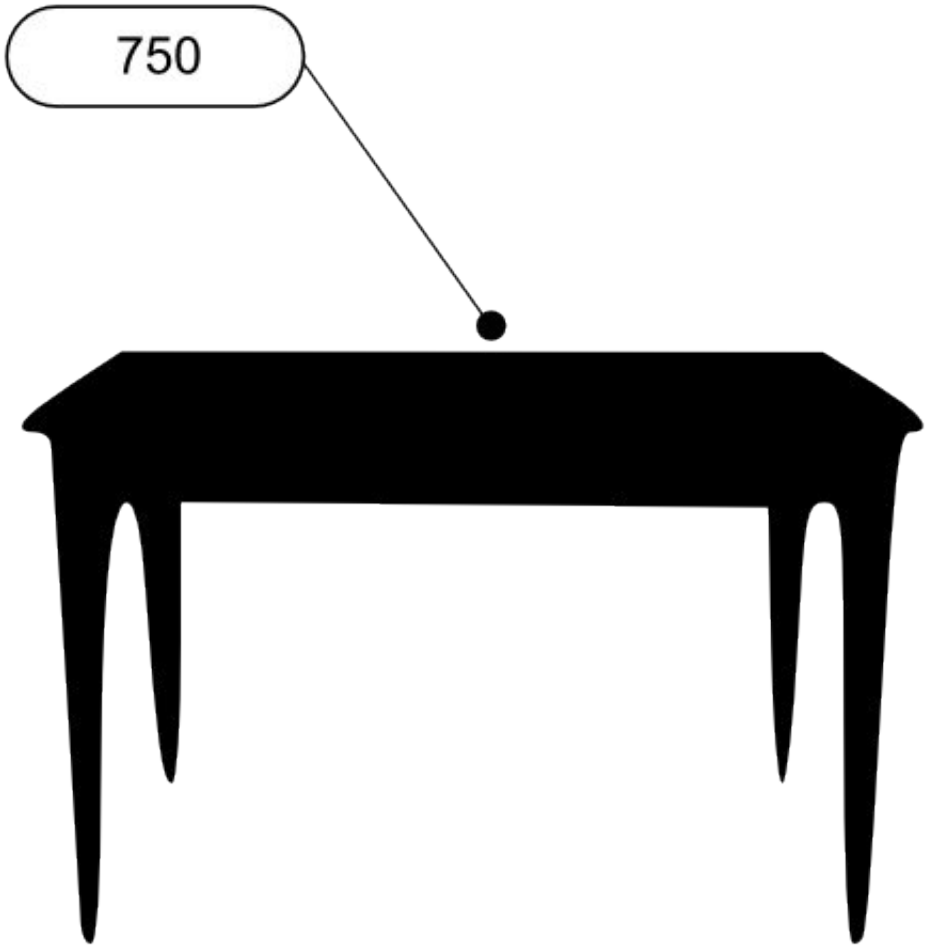
Table

Hosts

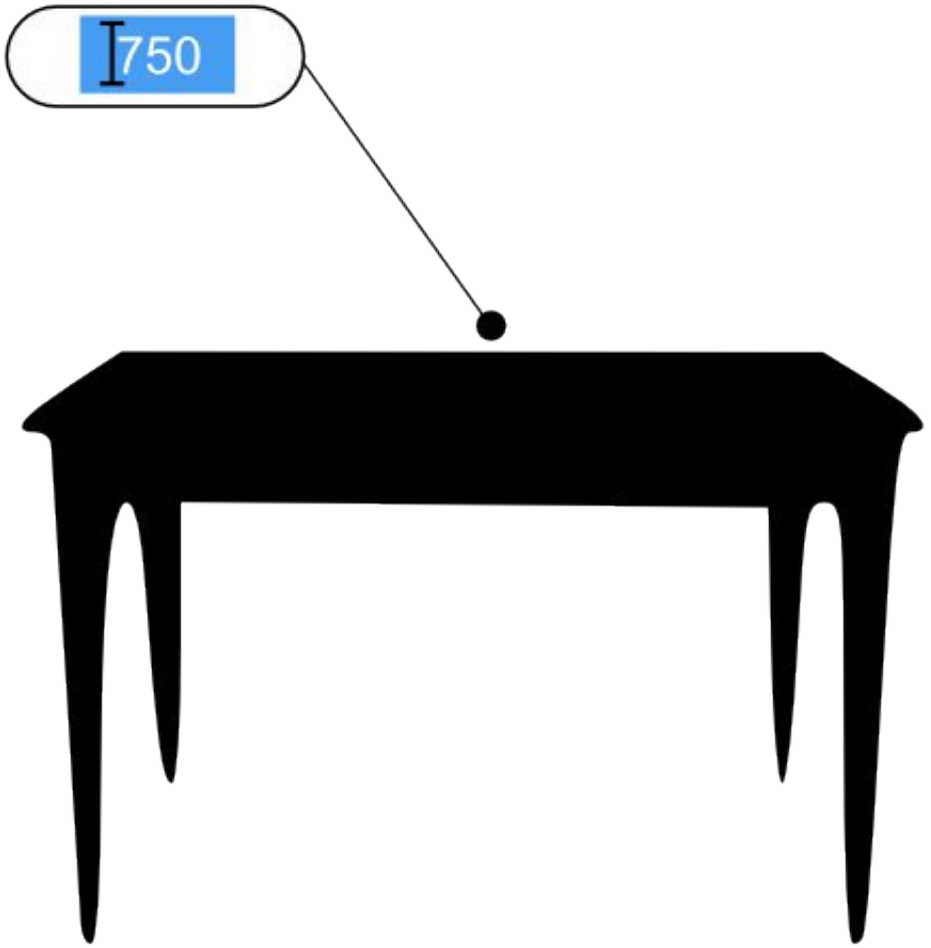
Level	Level 1
Level offset	0

Spatial

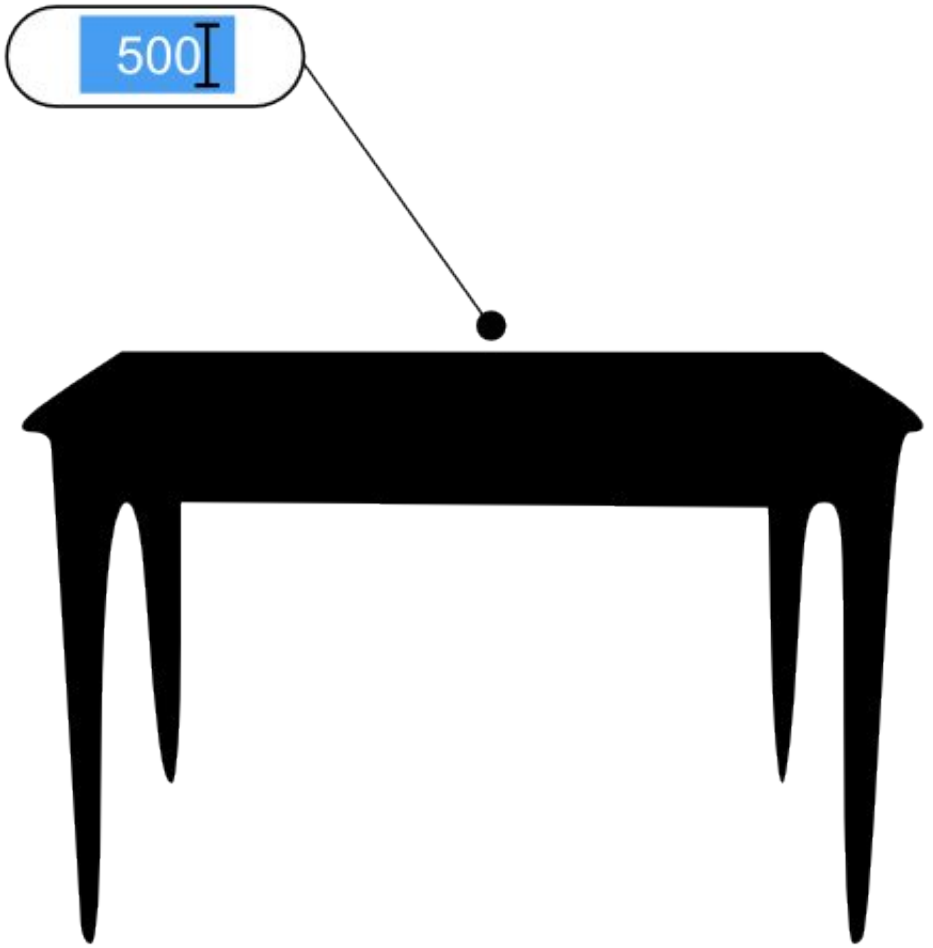
Width	1200
Depth	600
Height	750



Changing value in Tag also change taged component parameter value



Changing value in Tag also change taged component parameter value



Component

Table

Hosts

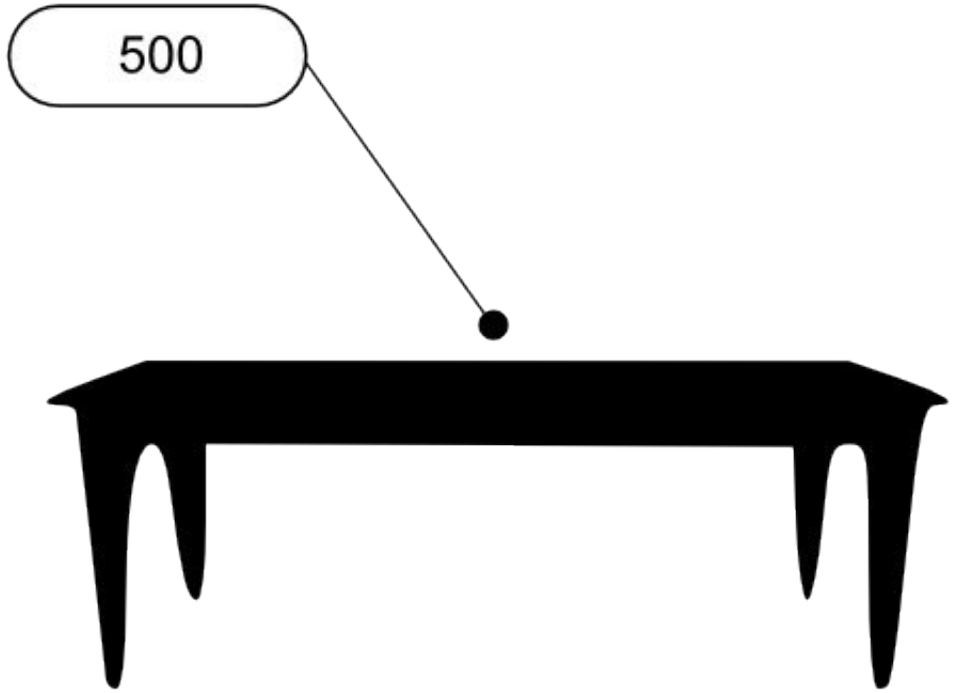
Level	Level 1
Level offset	0

Spatial

Width	1200
Depth	600
Height	750

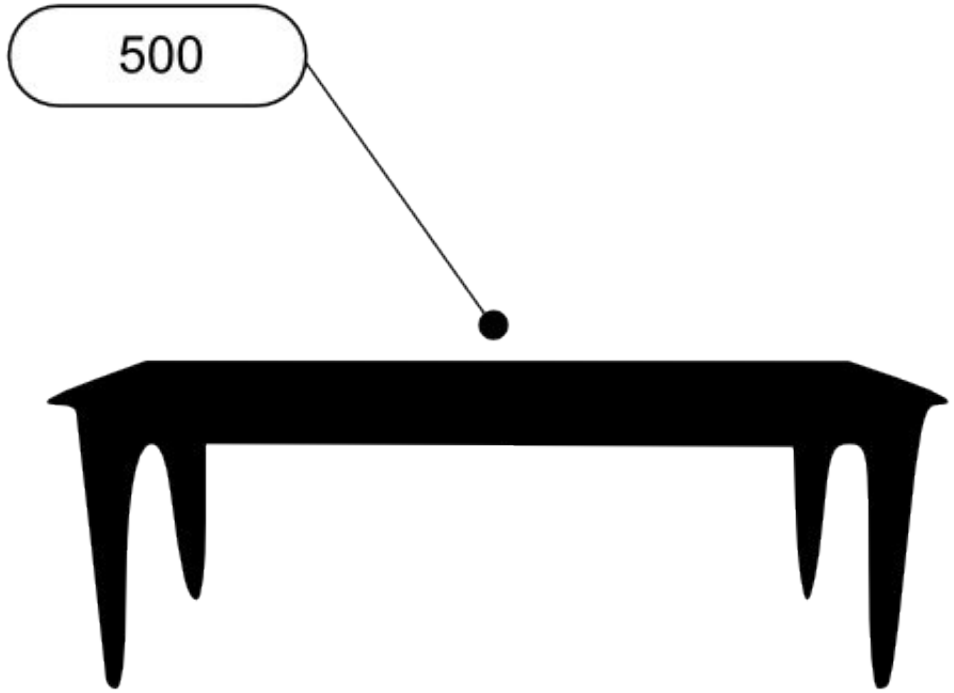
Changing value in Tag also change taged component parameter value

Name	Dimensions	Count
Chair	450x450x800	2
Table	1200x600x500	3



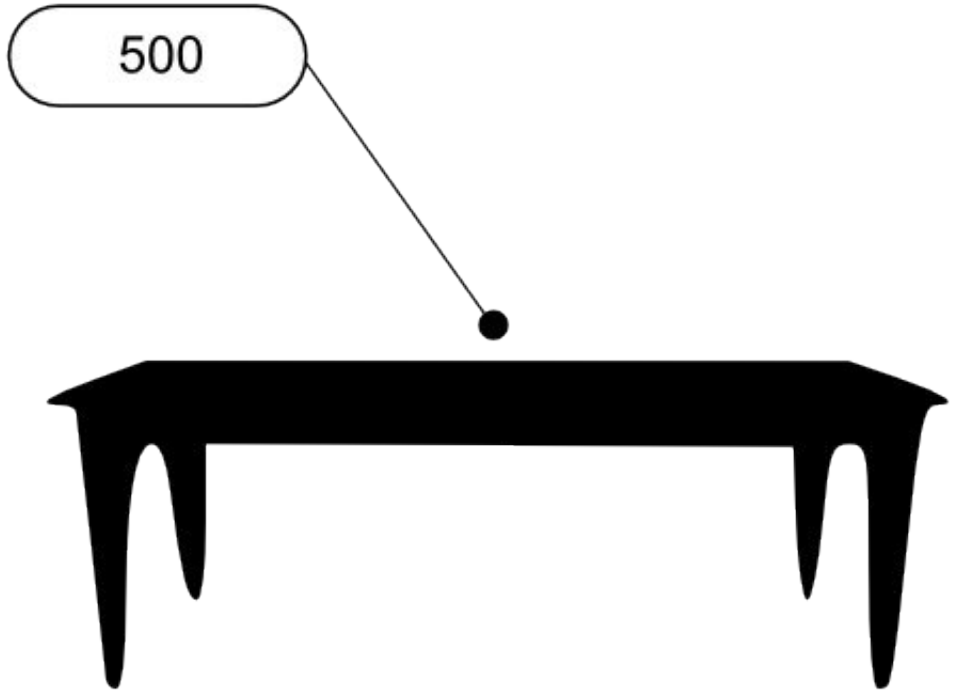
Changing value in Tag also change taged component parameter value

Name	Dimensions	Count
Chair	450x450x800	2
Table	1200x600x500	3



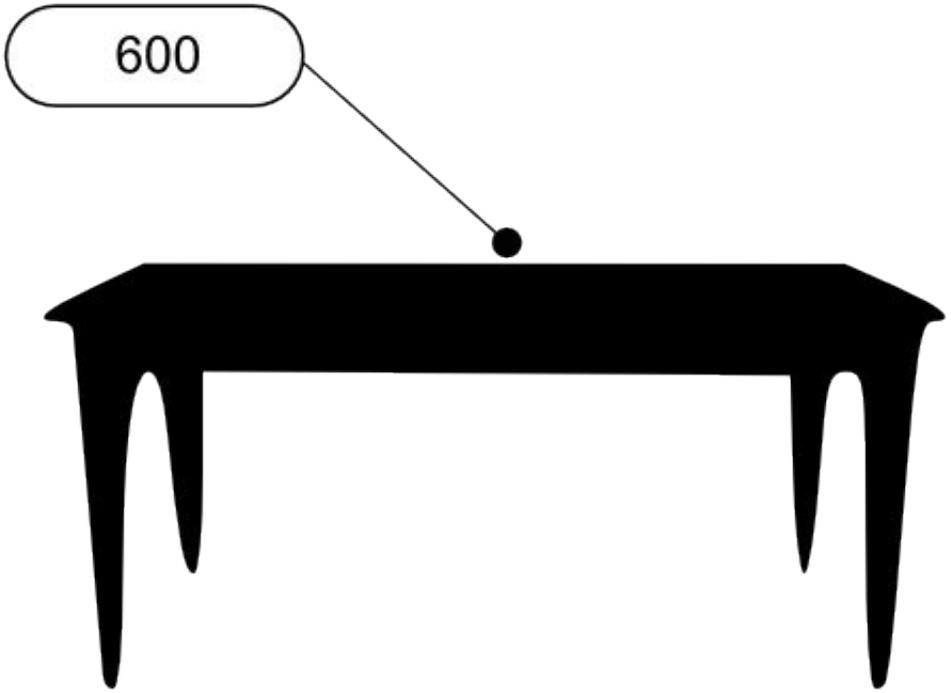
Changing value in Tag also change taged component parameter value

Name	Dimensions	Count
Chair	450x450x800	2
Table	1200x600x600	3



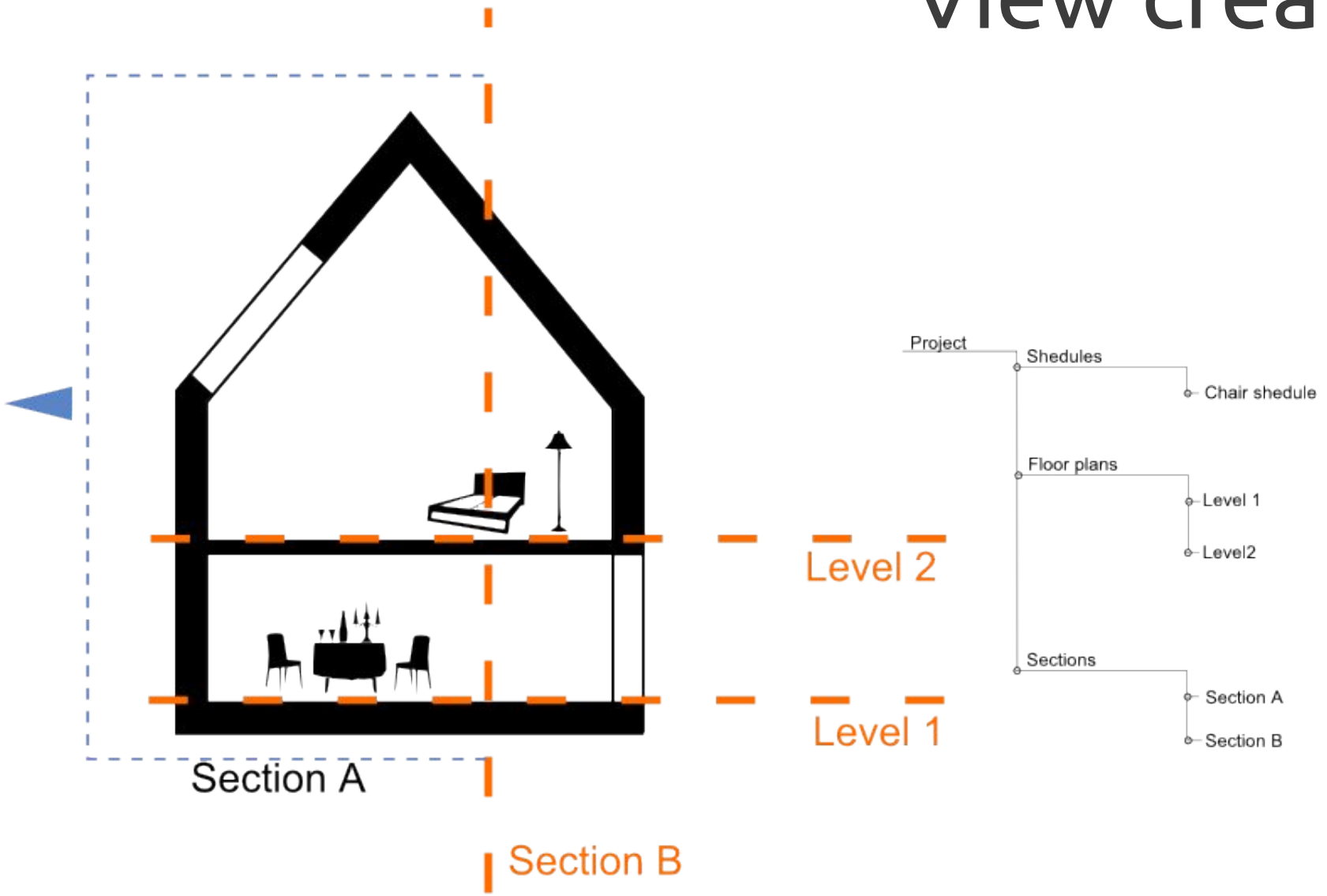
Changing value in Tag also change taged component parameter value

Name	Dimensions	Count
Chair	450x450x800	2
Table	1200x600x600	3



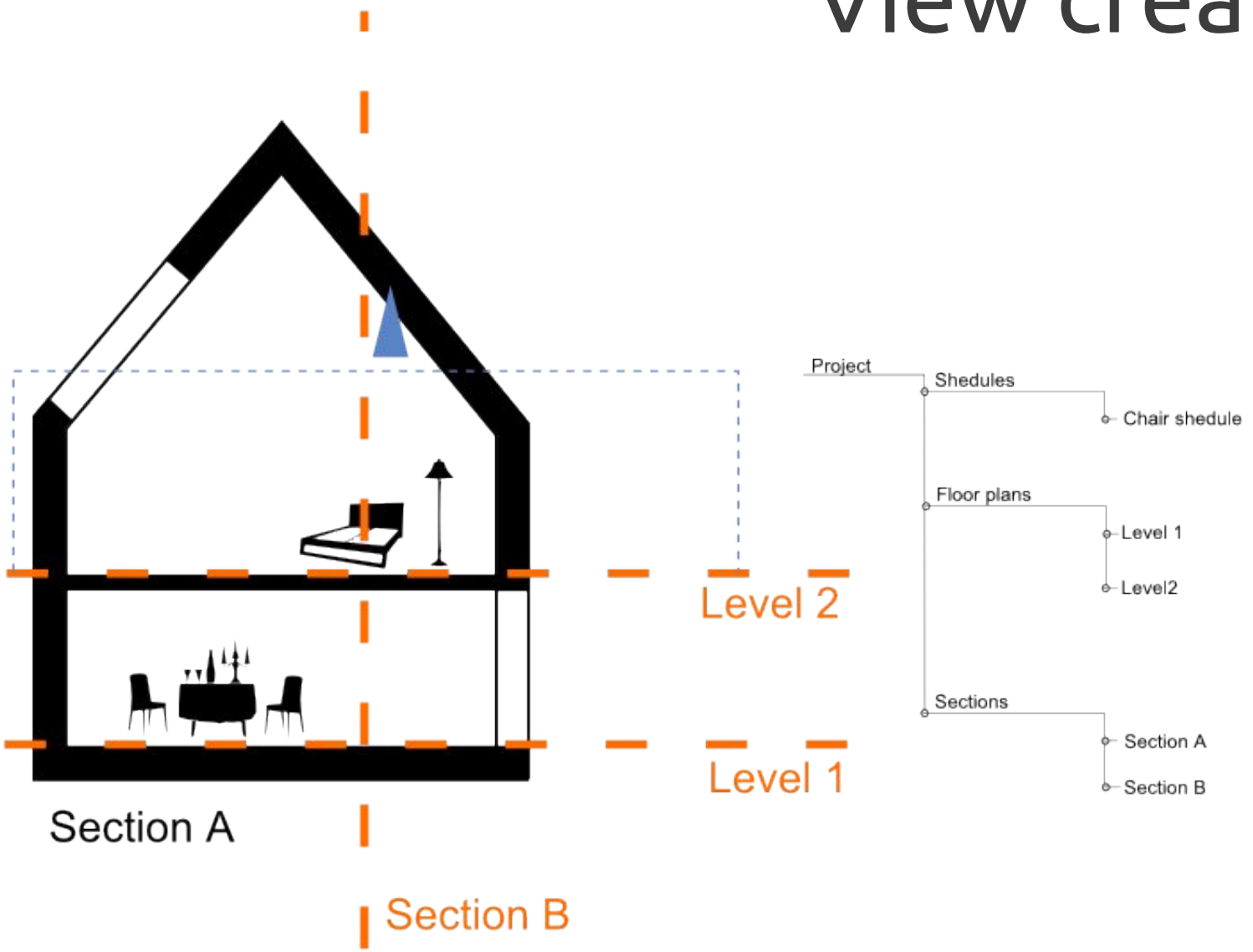
Changing value in Tag also change taged component parameter value

View creation

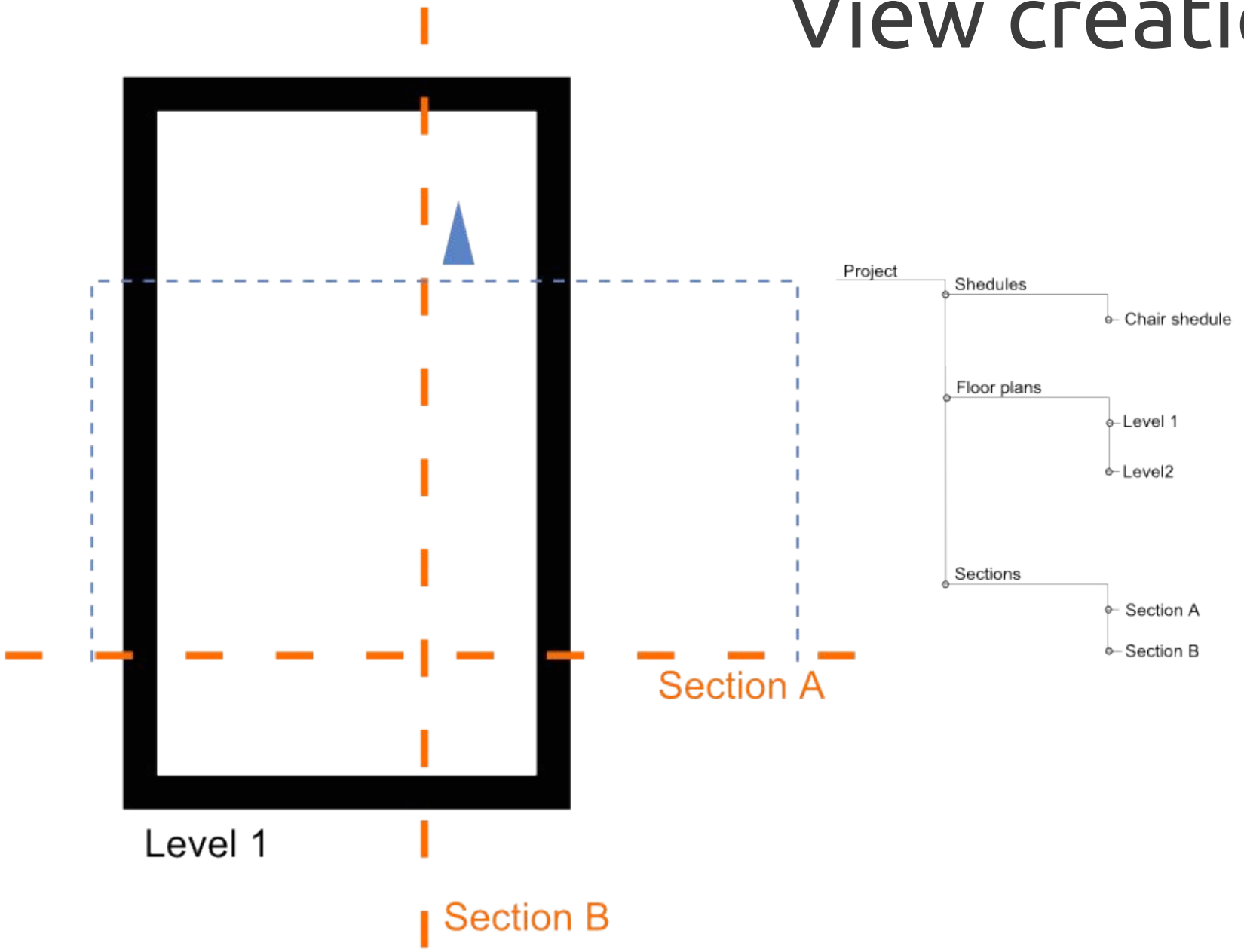


Even if we start nanolathing directly from the project model one day, still, drawings are at least interesting to print out.

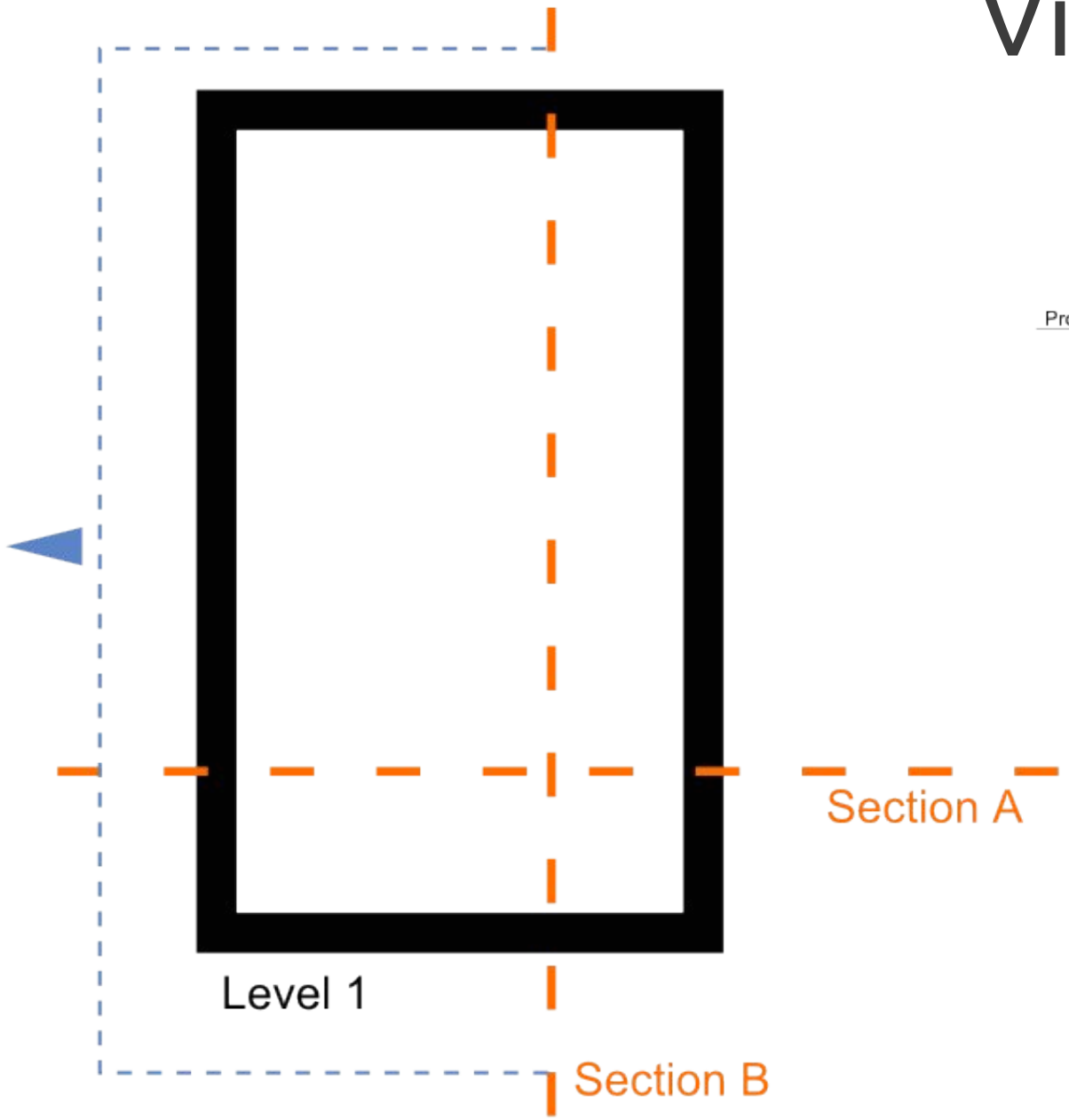
View creation

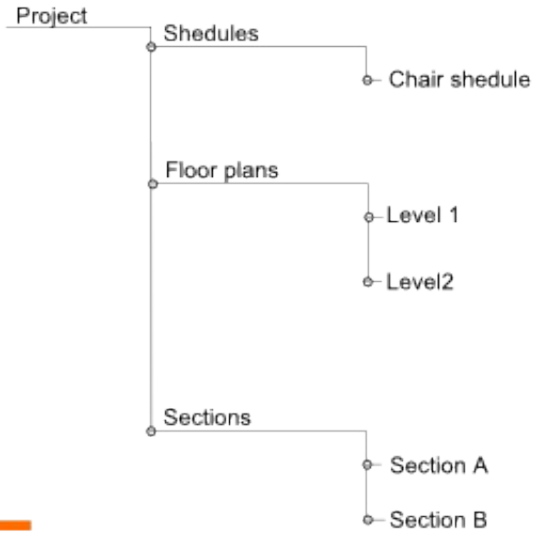


View creation

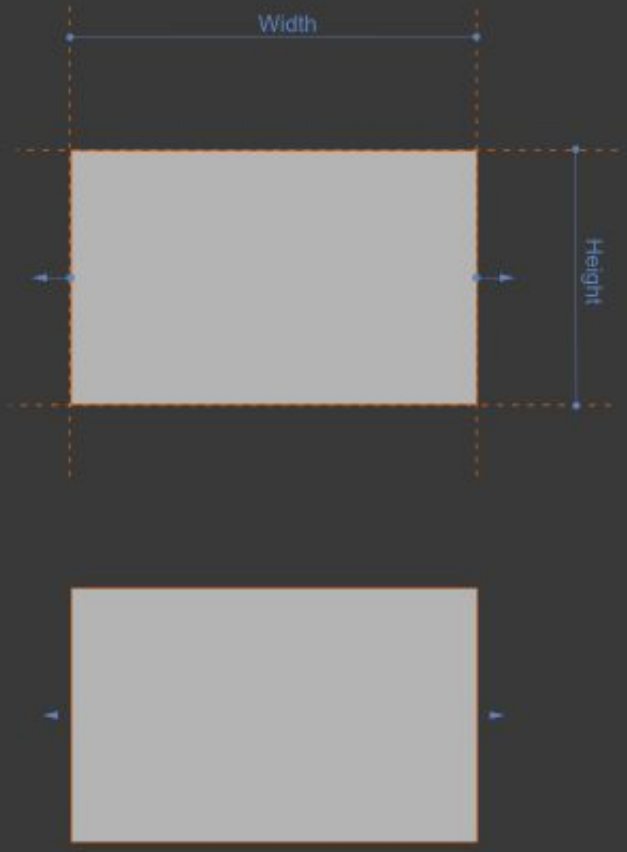


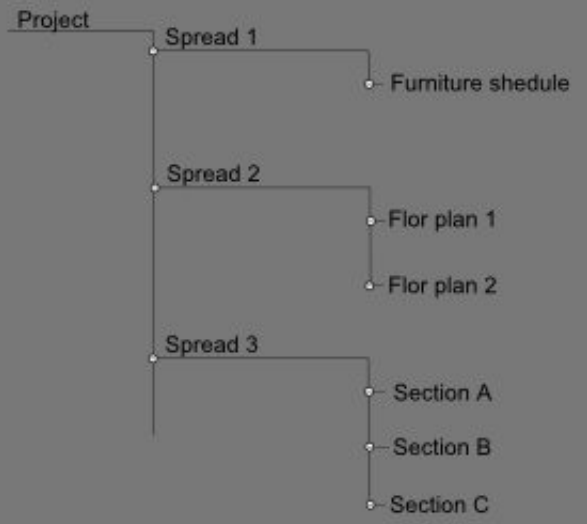
View creation

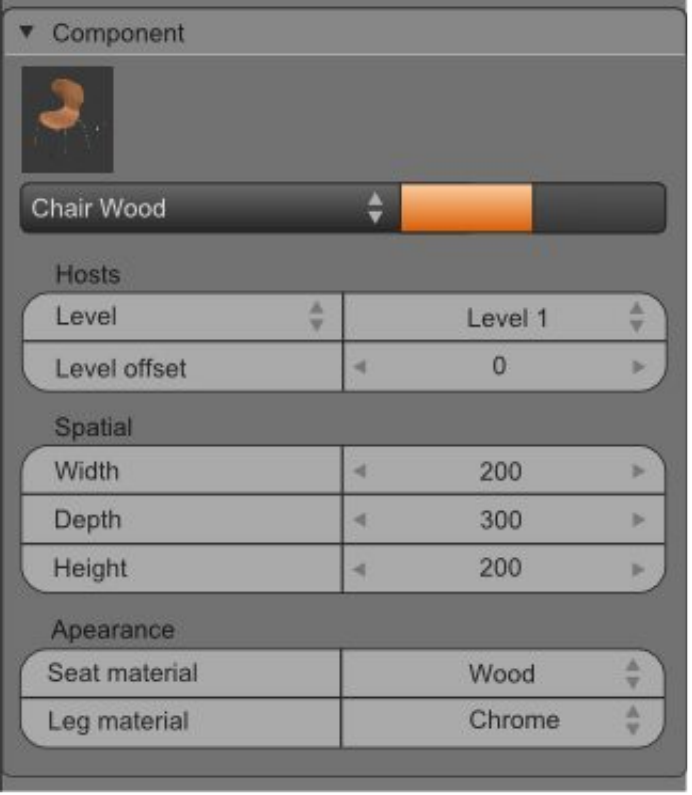




Model space

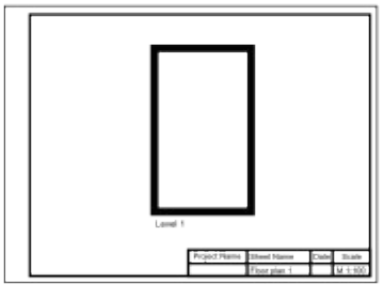


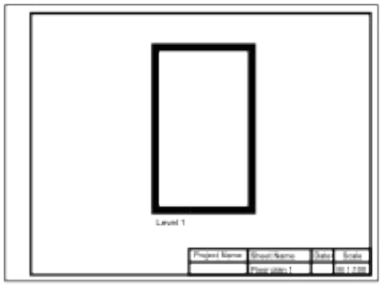


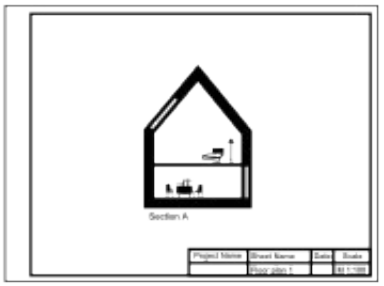


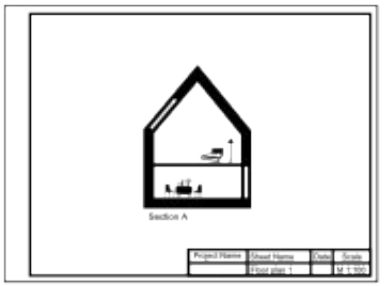
...When You click on a page or spread in project tree, you get special interface to organize the 2D drawings

Paper space

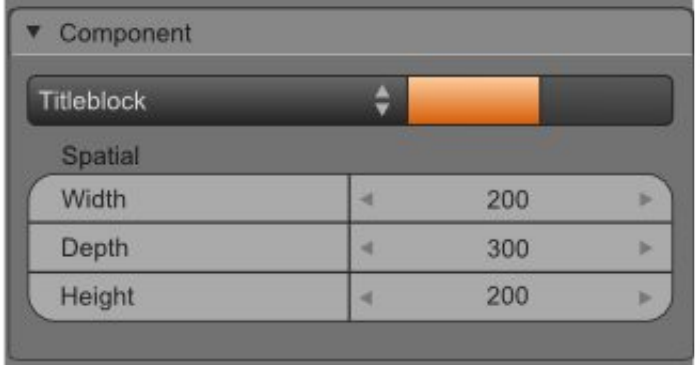




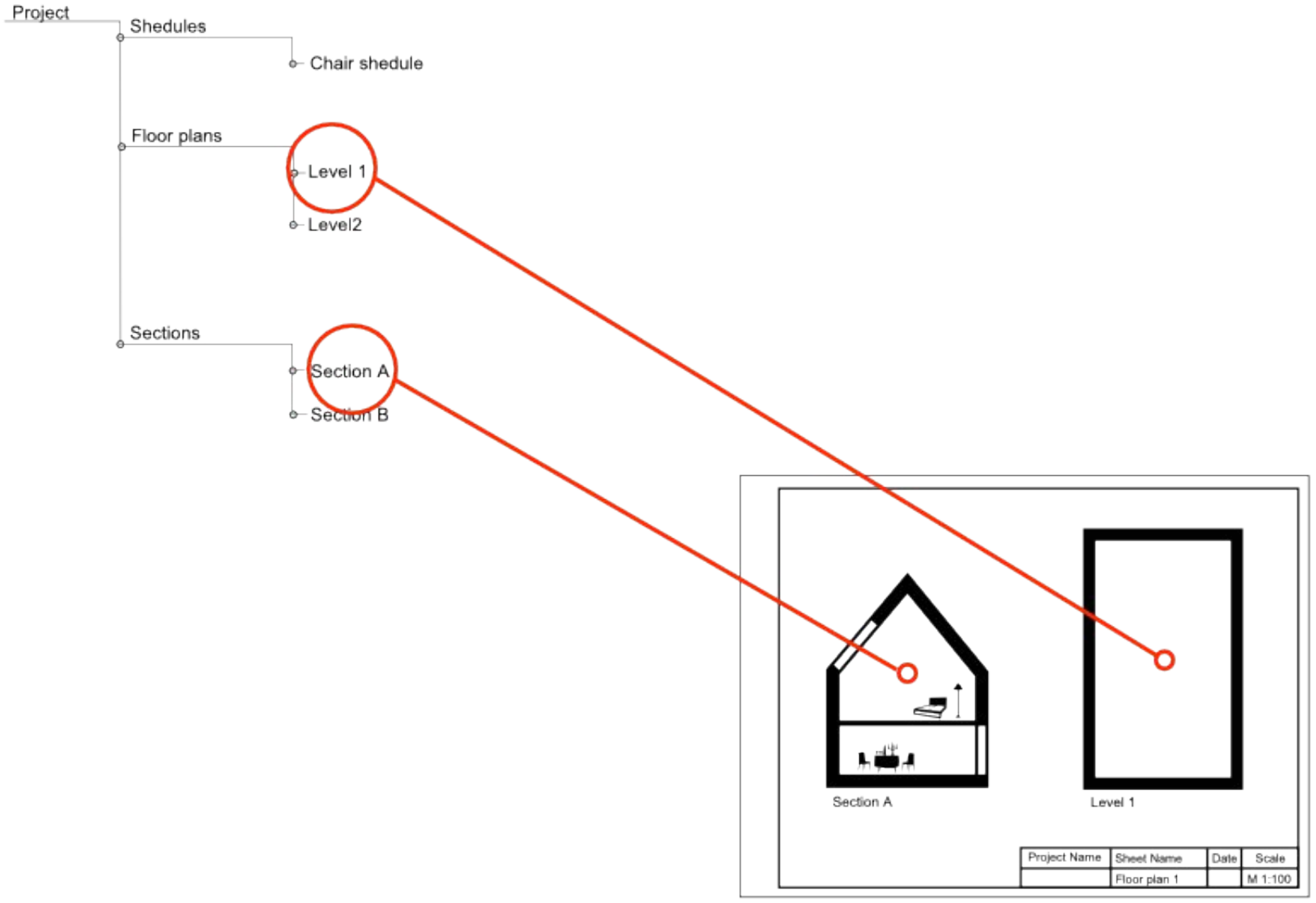




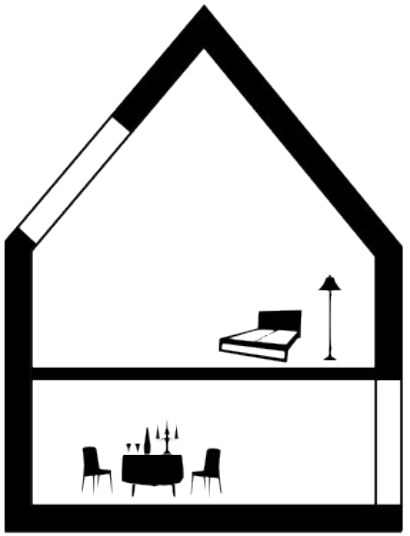

Project
Spread 1
Furniture shedule
Spread 2
Flor plan 1
Flor plan 2
Spread 3
Section A
Section B
Section C


Component
Titleblock
Spatial
Width
200
Depth
300
Height
200

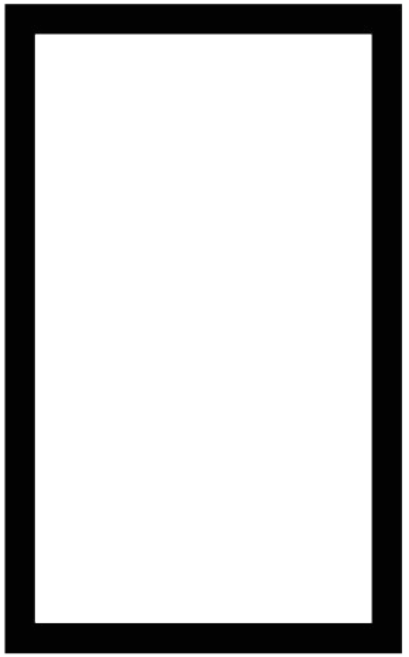
Views added to sheets



Views added to sheets



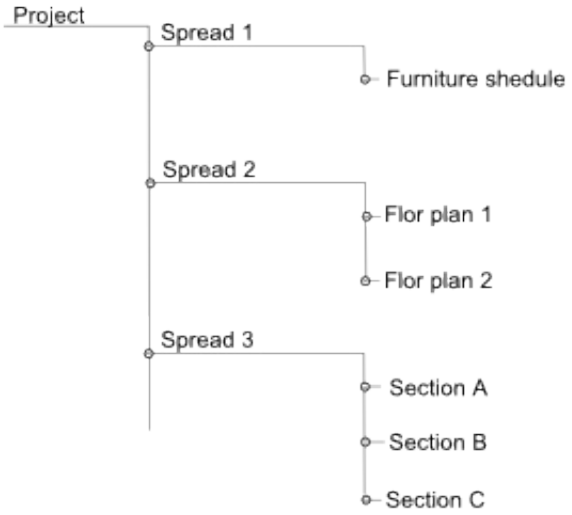
Section A

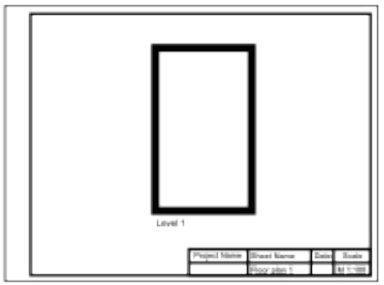


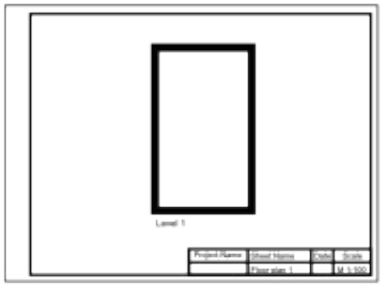
Level 1

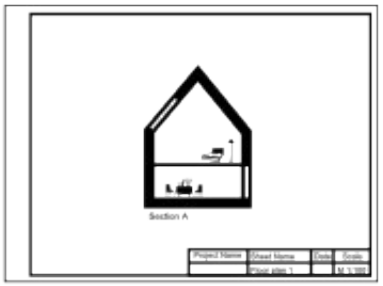
Project Name	Sheet Name	Date	Scale
	Floor plan 1		M 1:100

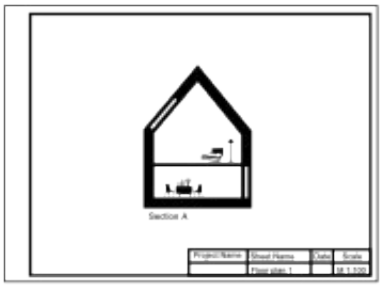
Exporting spreads to PDF

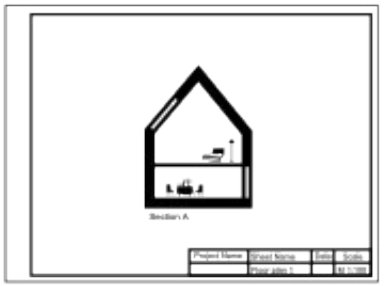




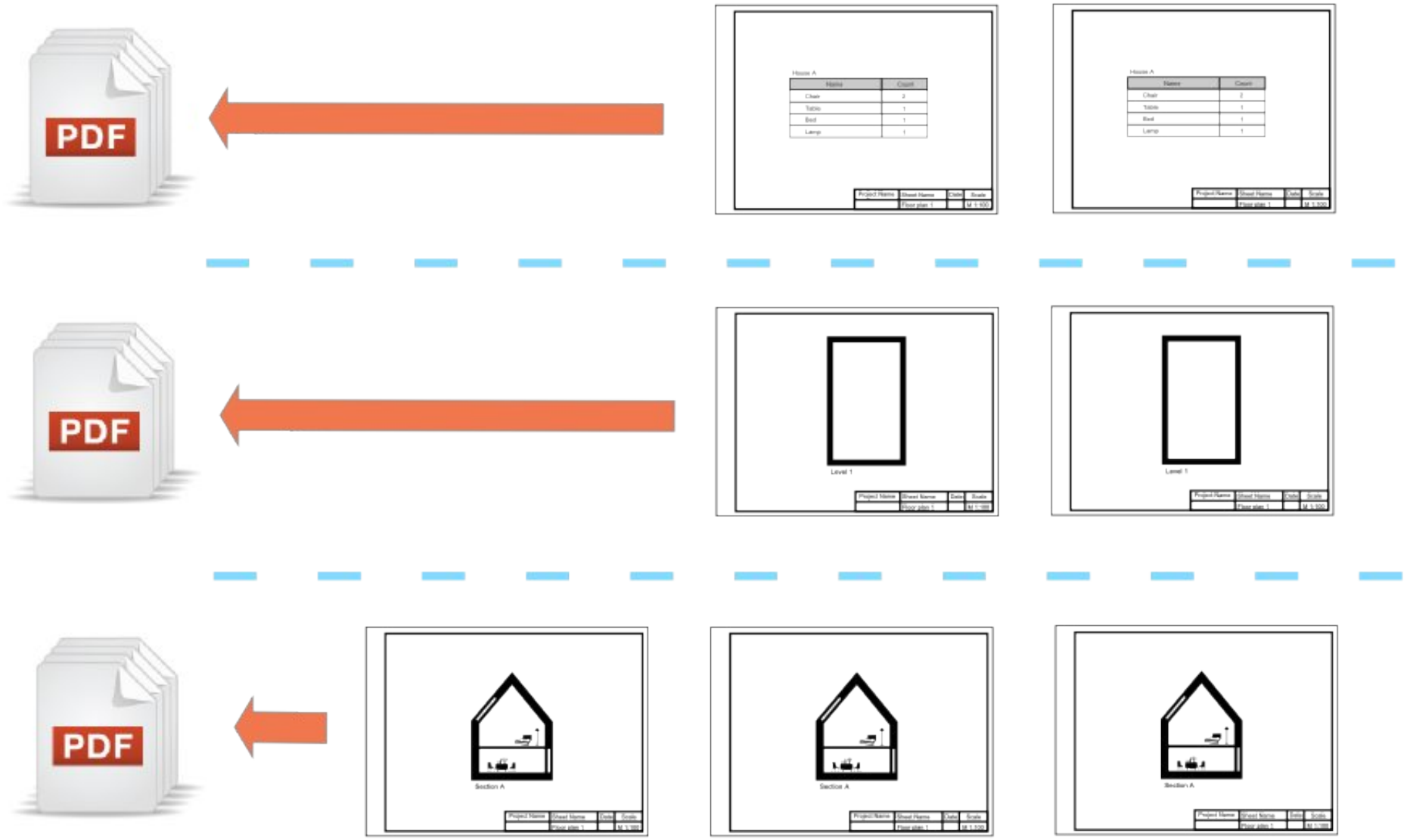








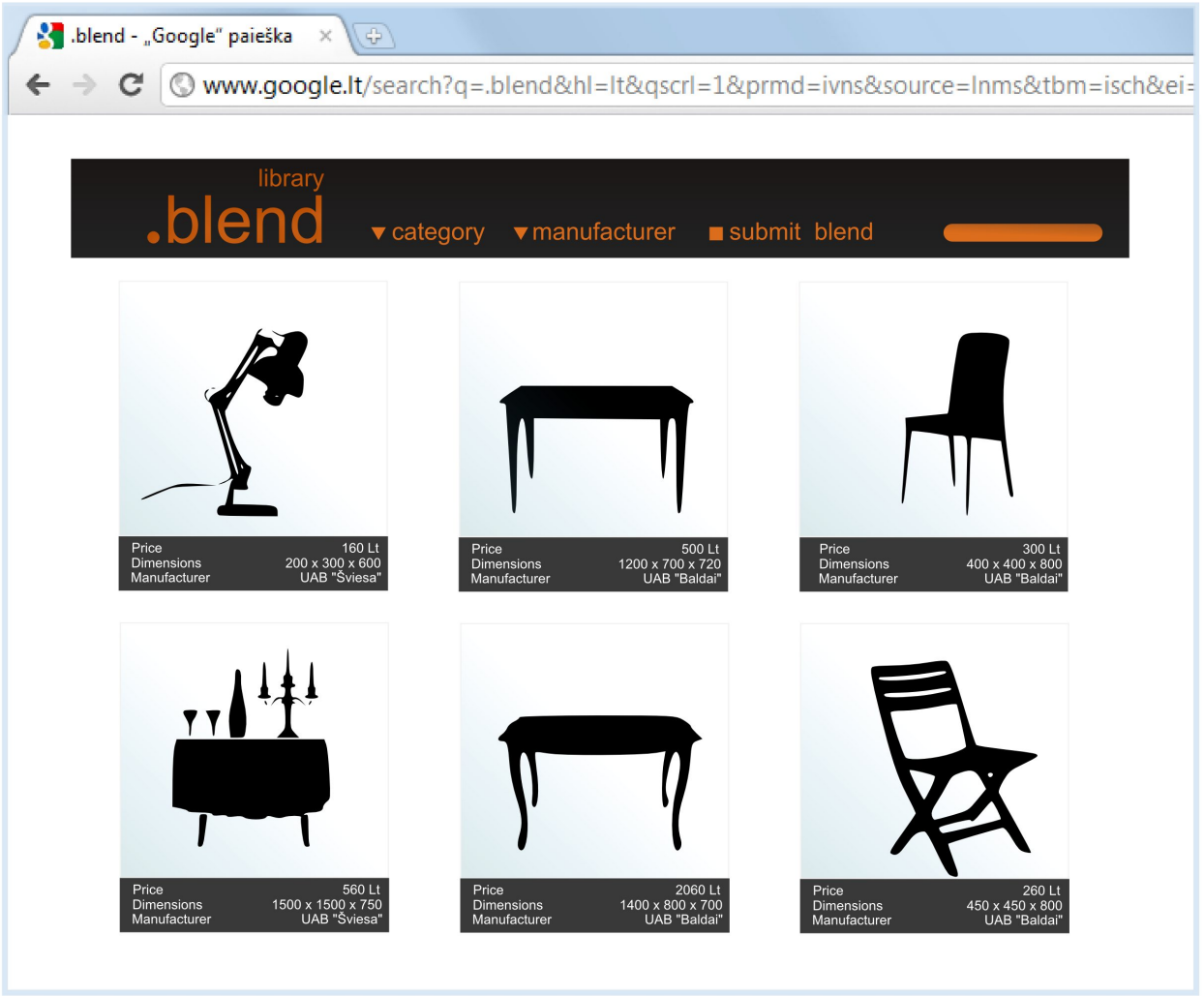
Exporting spreads to PDF



Layout engine benefits to mainstream version of Blender

- Suitable as a documentation tool.

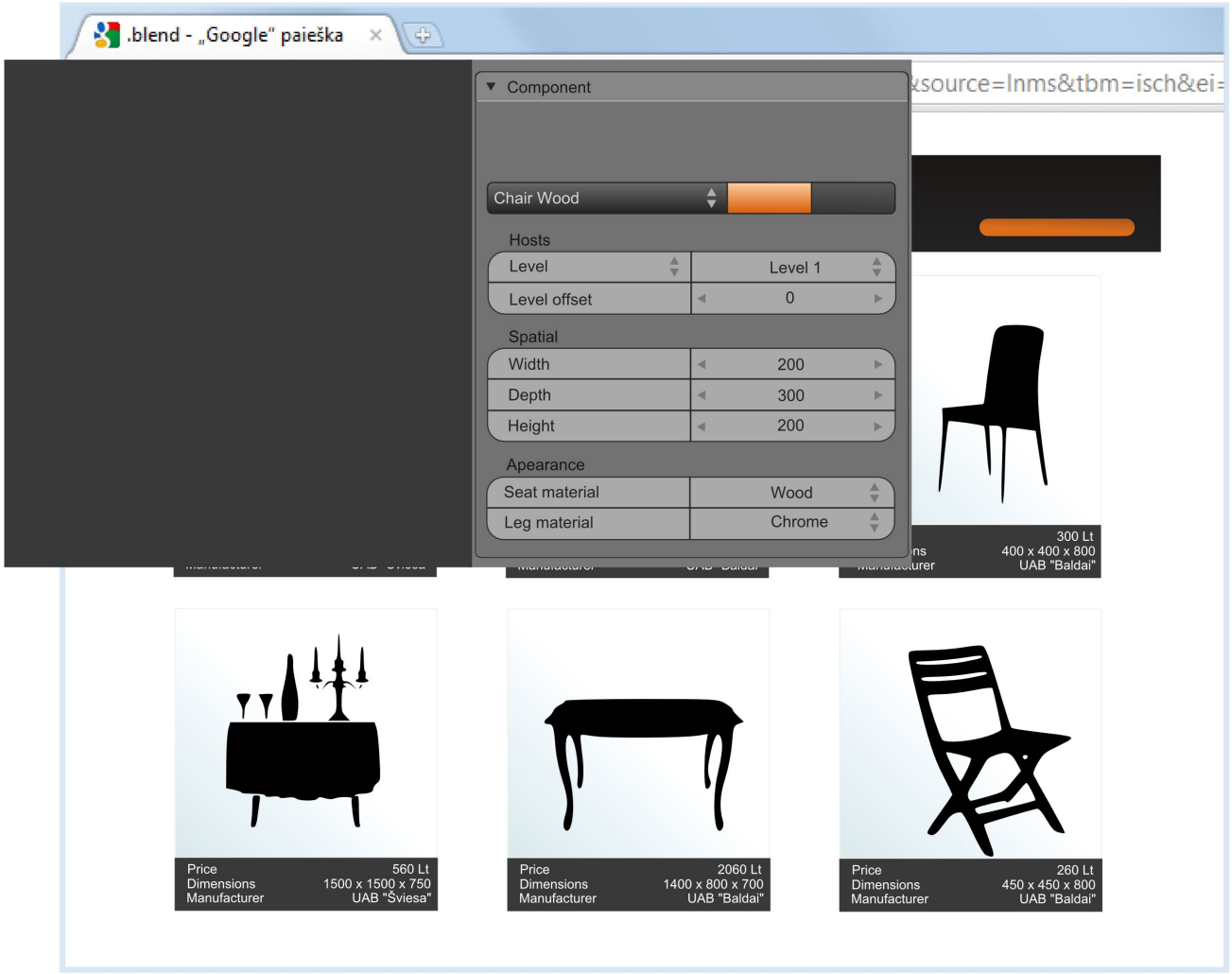
Online parametric component library



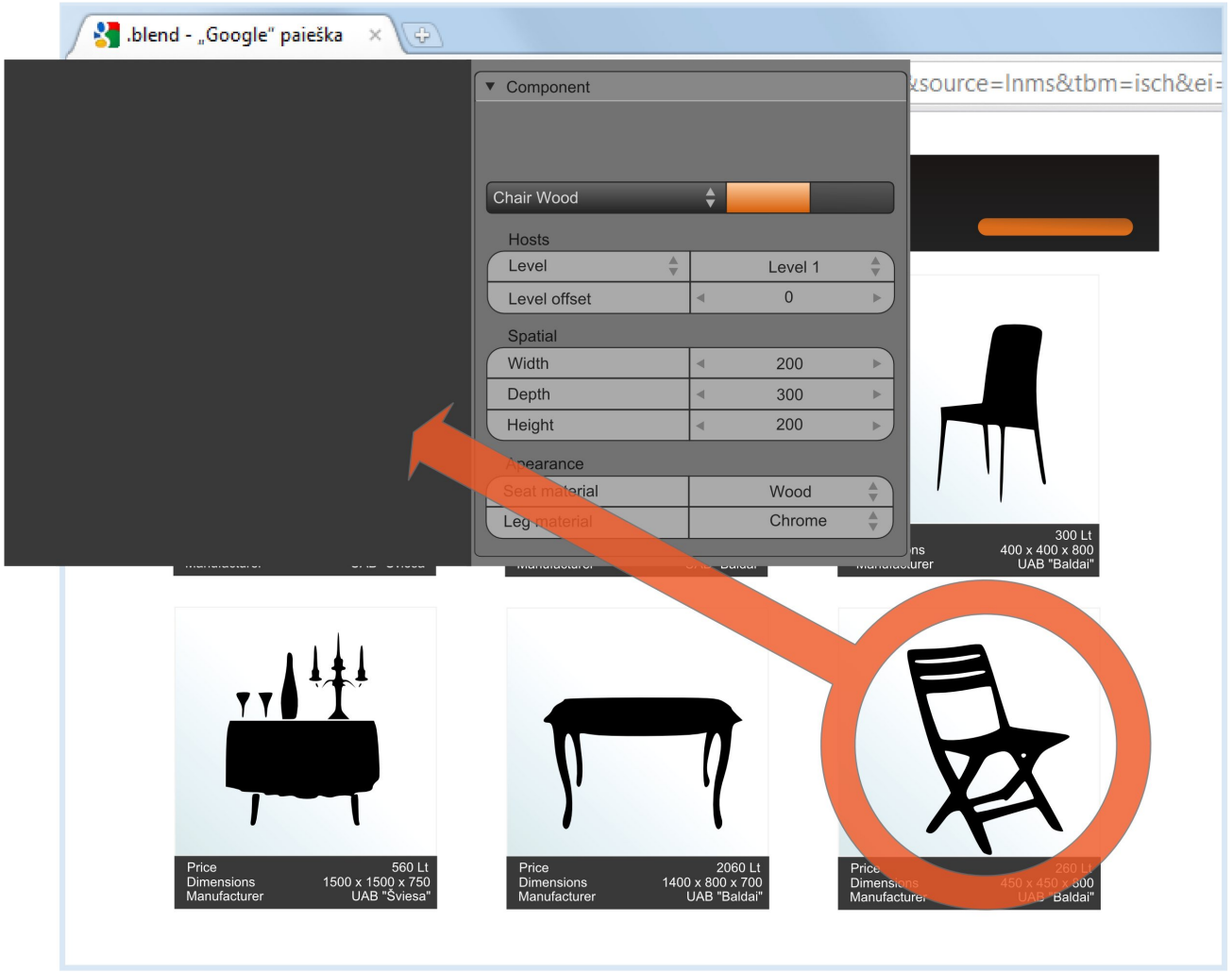
Important

- Conection between online library and local project
- Version & updates control
- Teamwork & project workflow management
- Use cases:
  - Real-life 3D model gallery/database
  - Intranet for developing complex projects

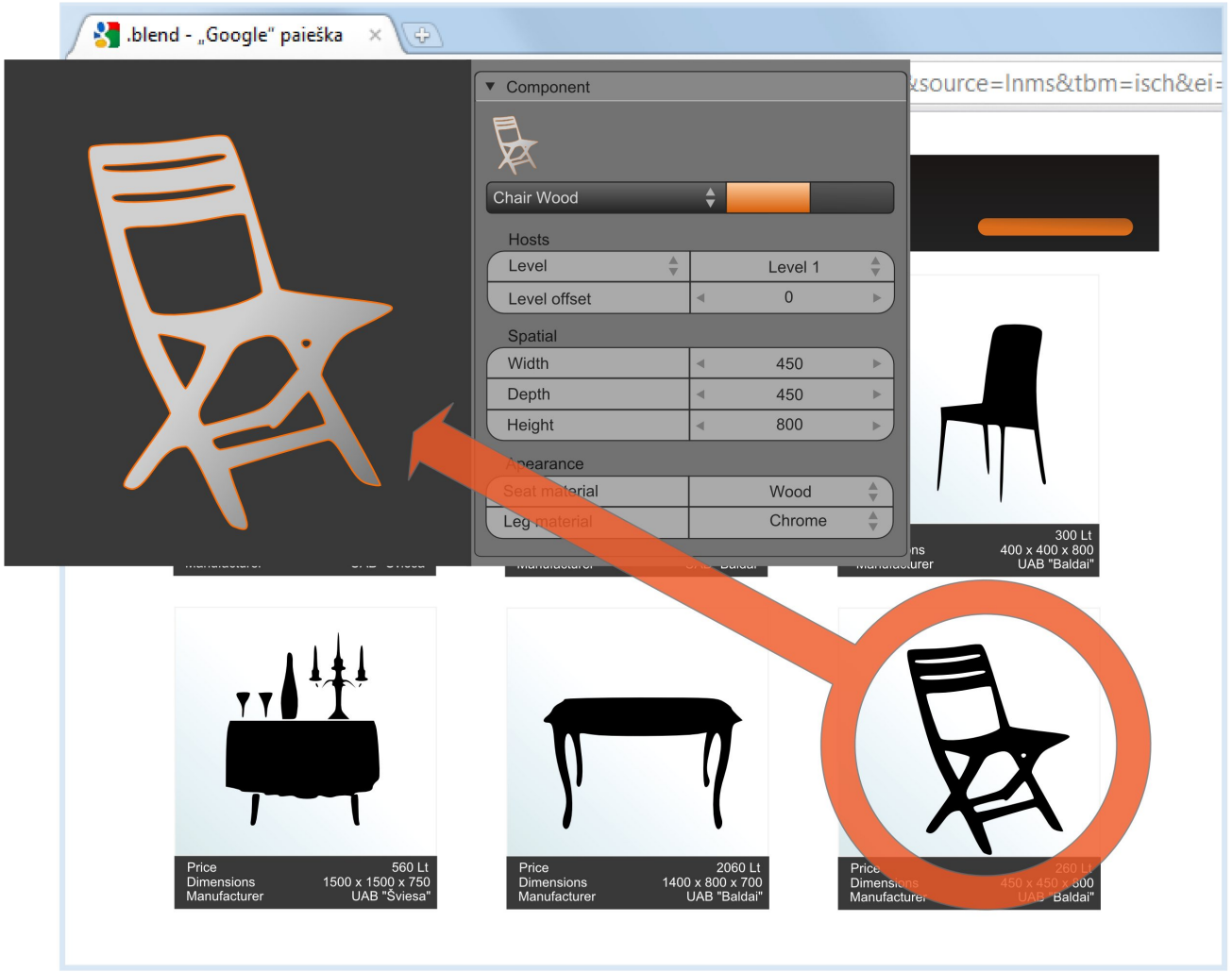
Online parametric component library



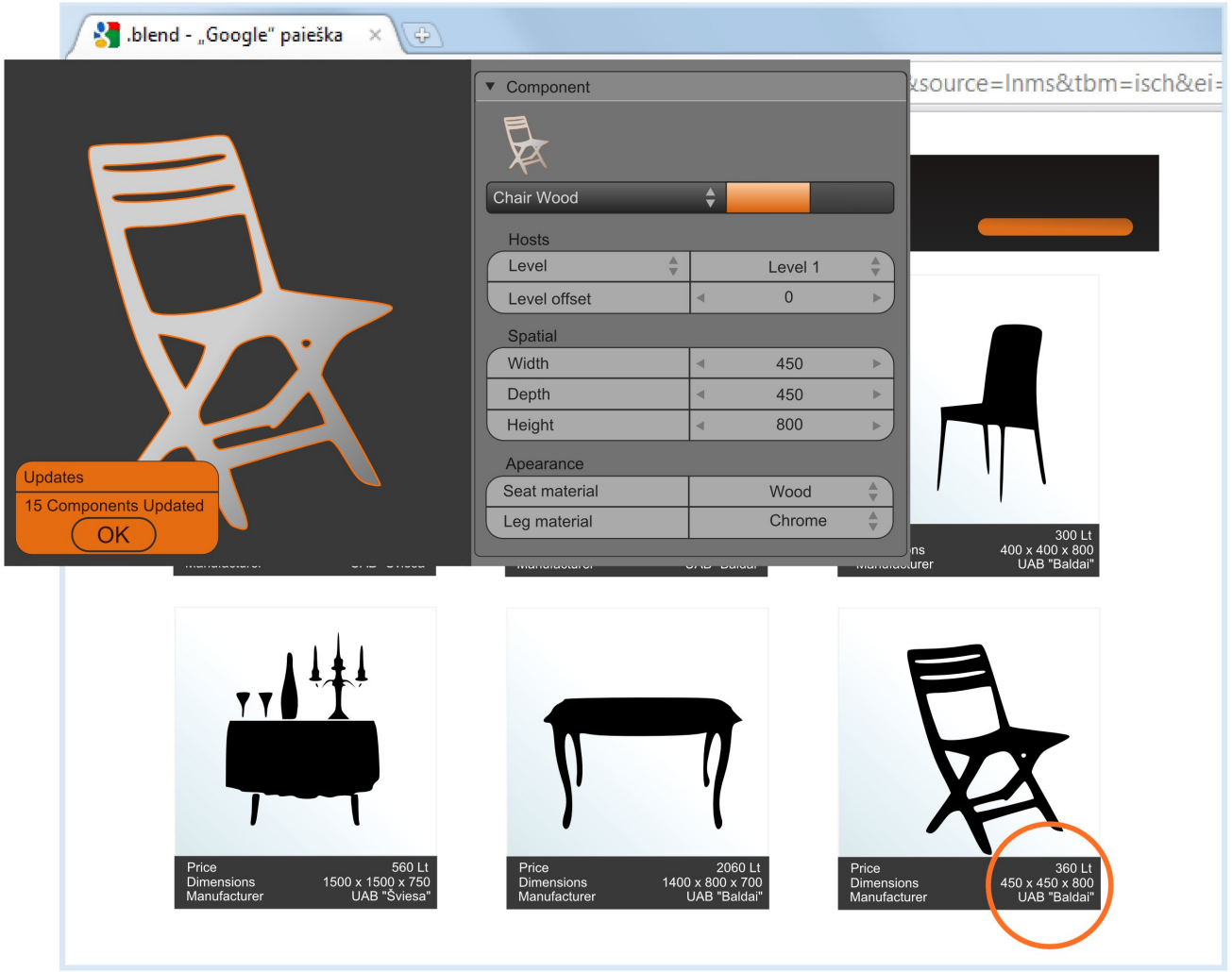
Online parametric component library



Online parametric component library



Online parametric component library



When we open old model and Blender sees updates online, it is possible to update to new version with one click.

Online parametric component library

- Library acts like Linux package repository
- Parametric components are like packages
- One module can have a loads of requirements, version dependencies etc...

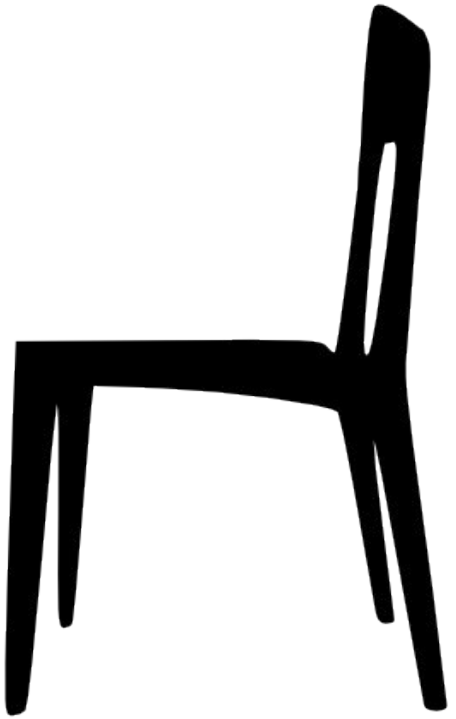
Problem of consistency

- Like in coding, it is very important, that everyone would create models and name refplanes using determined names.
- Imagine, if somene starts naming floor as „Bottom“, and other - „Level“, tables made to snap level „Floor“ will not work on any of these.
- Strict control is compusory, as like in accepting code to mainstream!

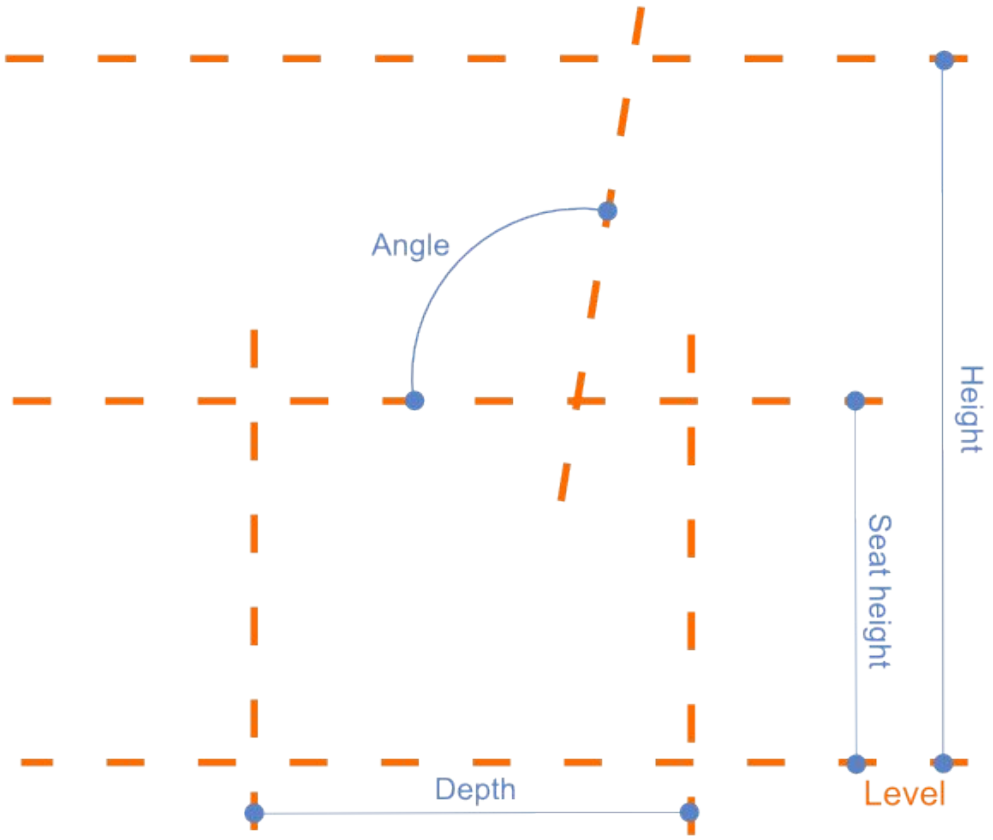
Tools to manage consistency of components

- Standard templates
- Validators

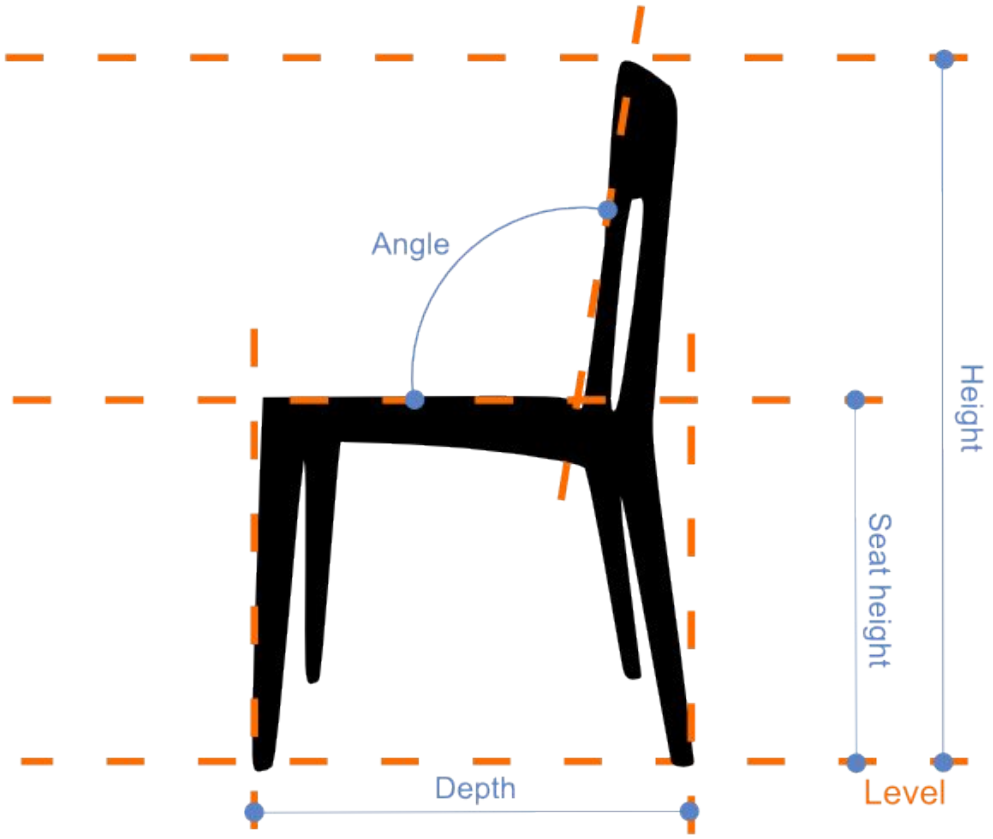
Chair geometry



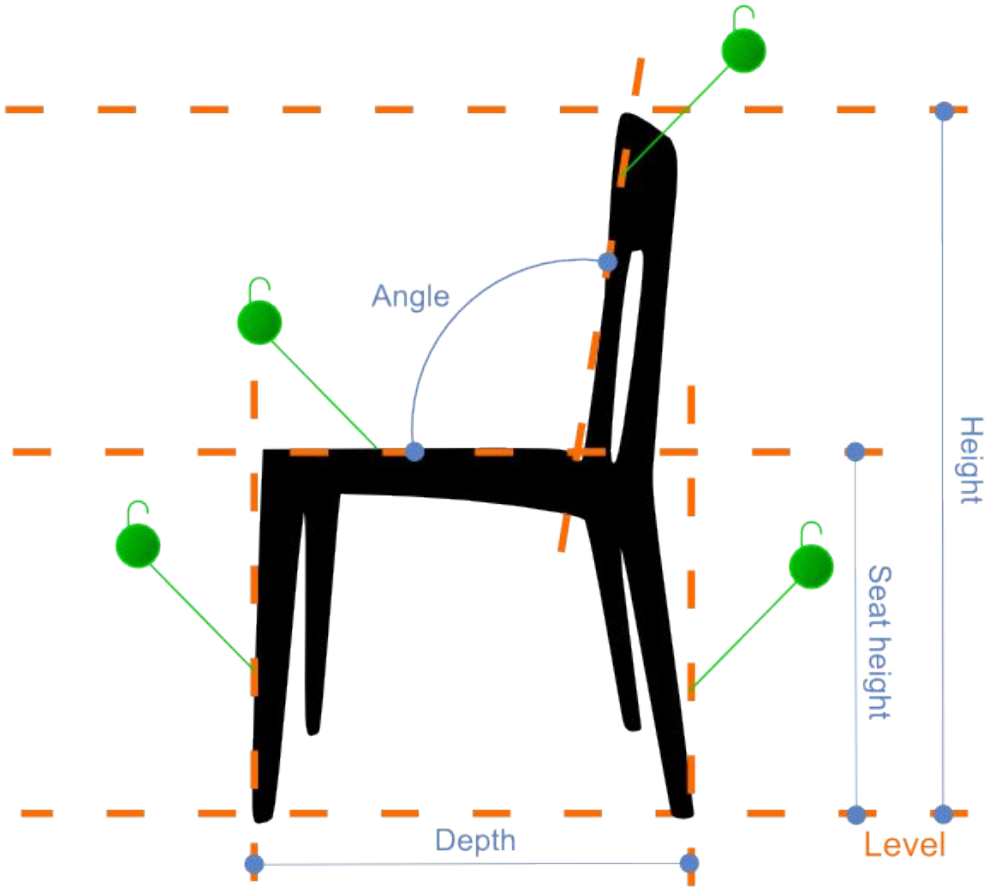
Chair template



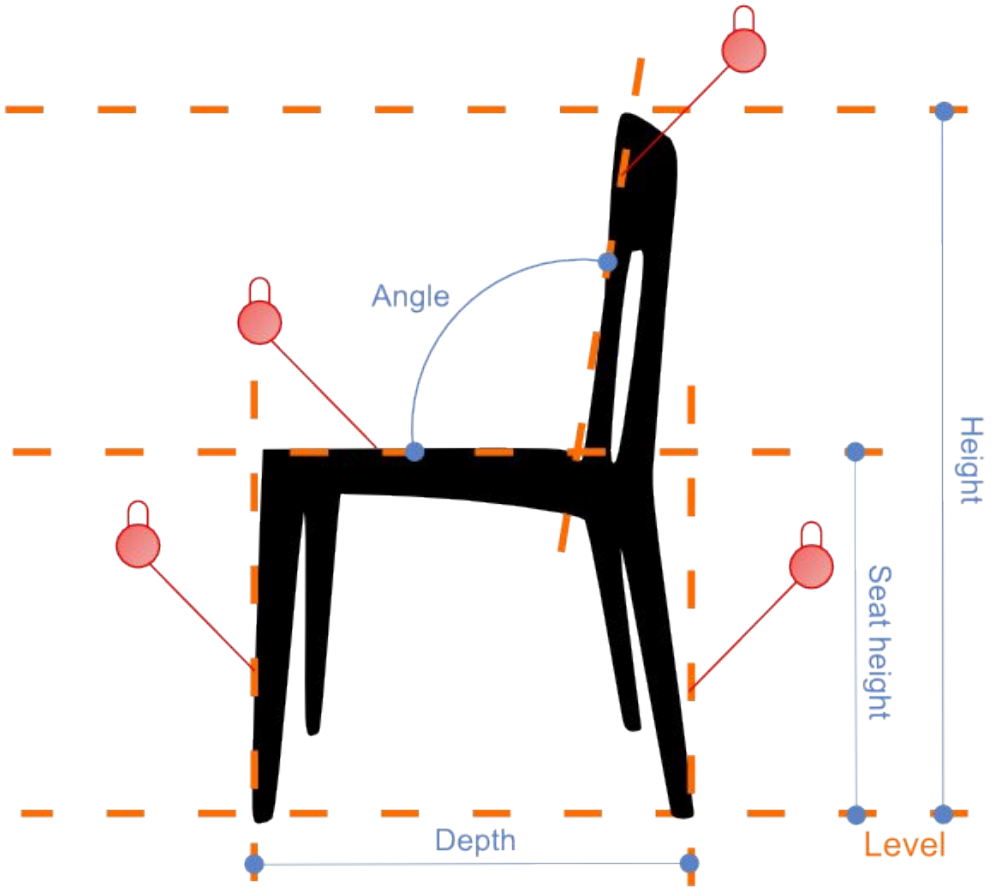
Template added to geometry



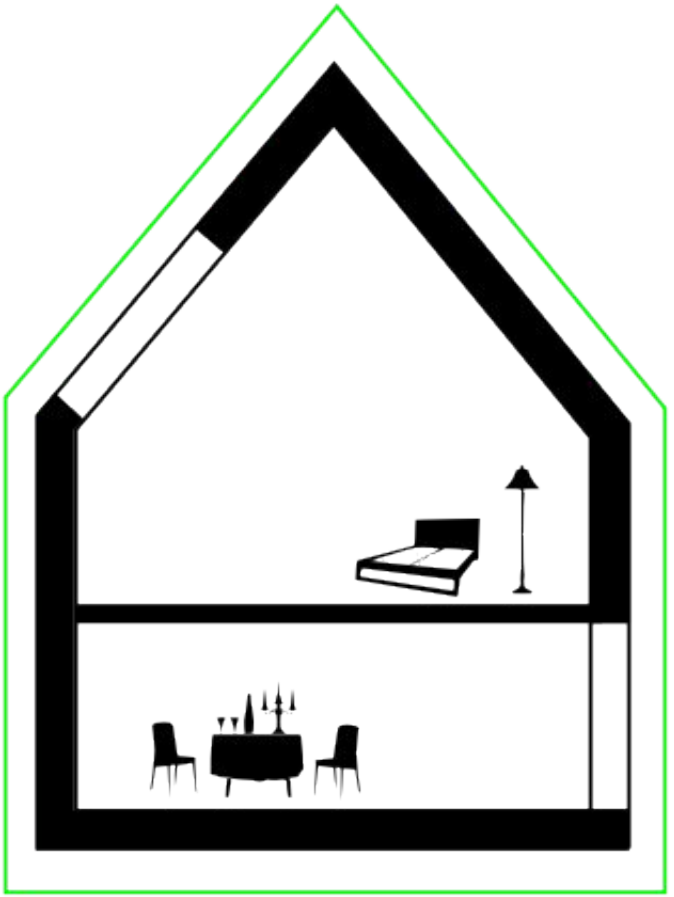
Geometry locked to refplanes



Geometry locked to refplanes

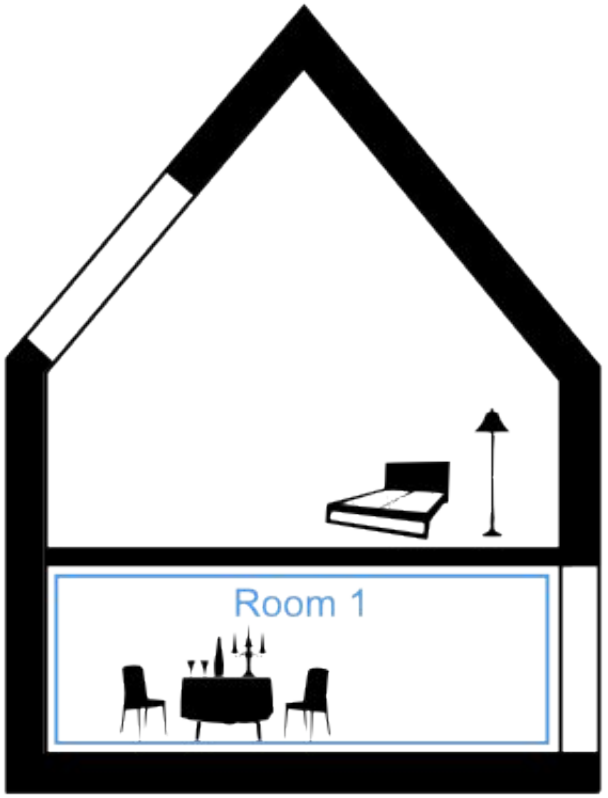


Scope boxes

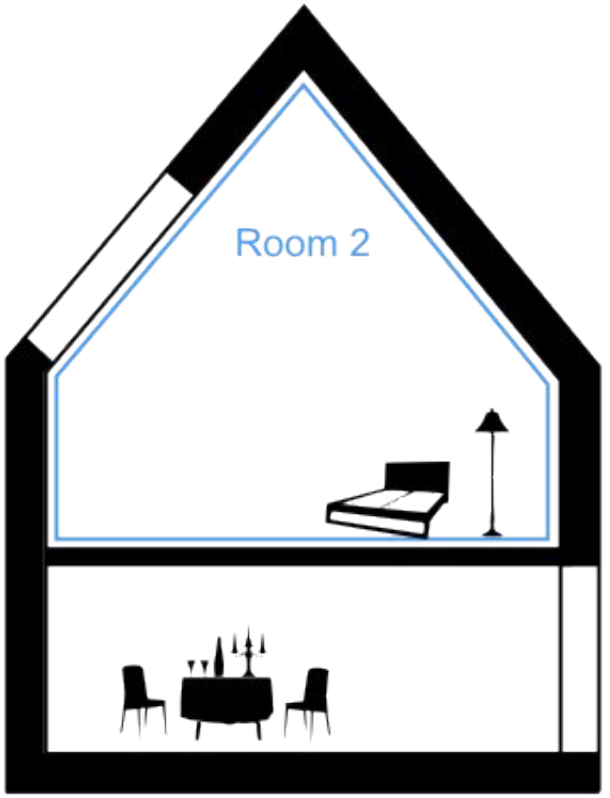


House A

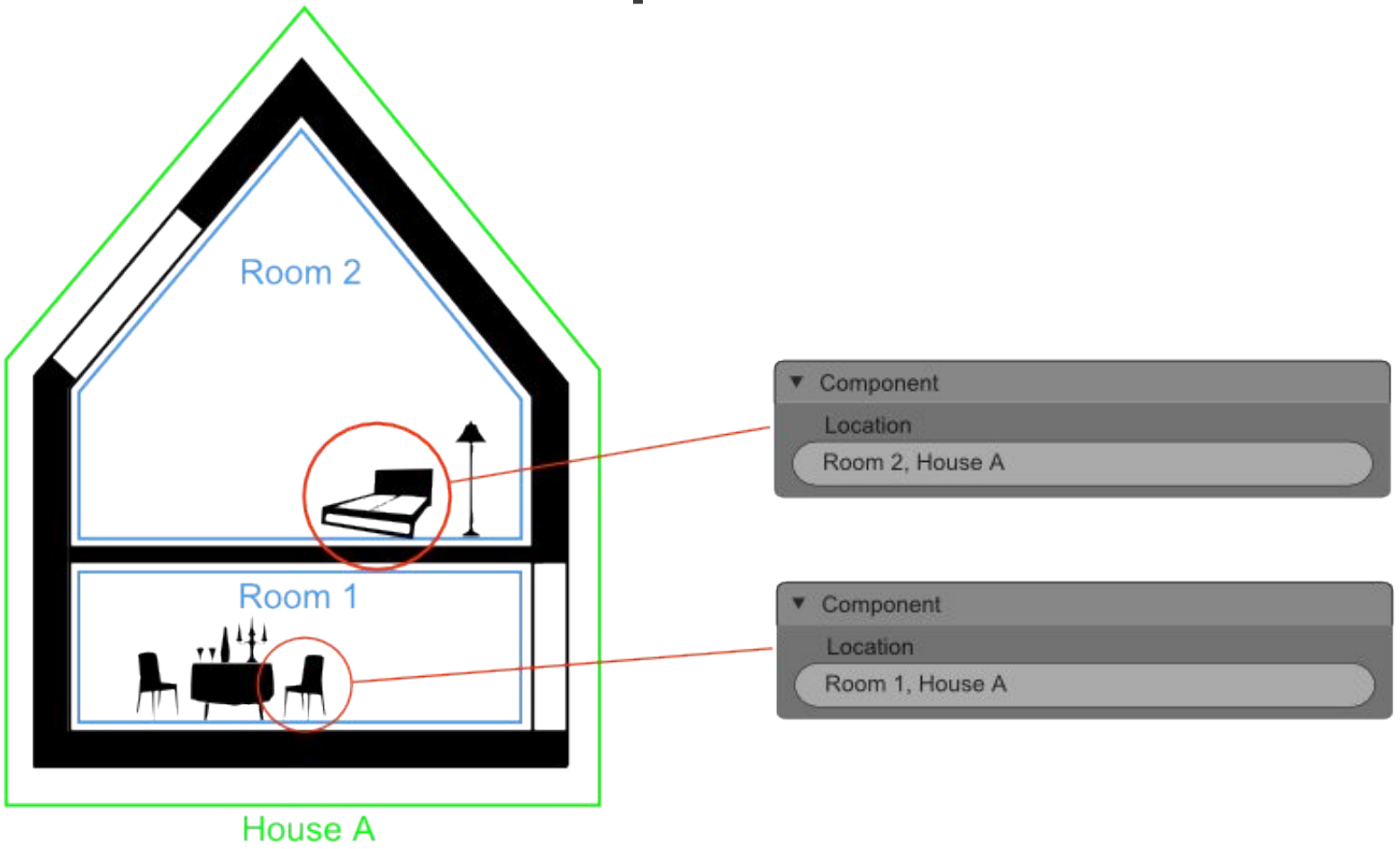
Scope boxes



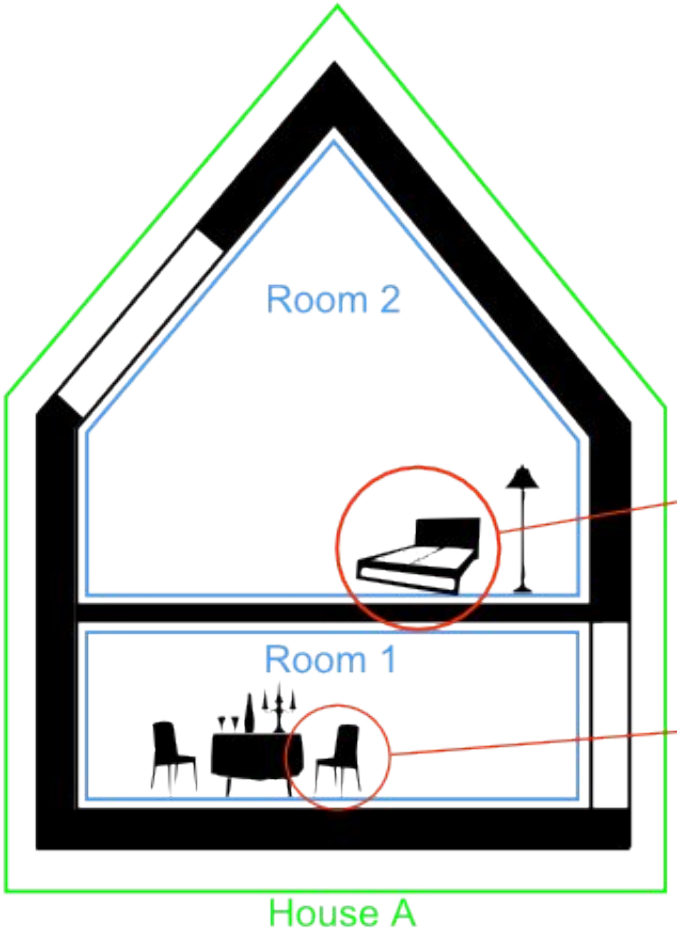
Scope boxes



Scope boxes


Room 2
Room 1
House A
Component
Location
Room 2, House A
Component
Location
Room 1, House A

Scope boxes



Room 1

Name	Count
Chair	2
Table	1

House A

Name	Count
Chair	2
Table	1
Bed	1
Lamp	1

Objects sorted by scope box parameter

Room 1

Name	Count
Chair	2
Table	1

House A

Name	Count
Chair	2
Table	1
Bed	1
Lamp	1

Beating performance bottlenecks

Complex model = slow user experience

Possible solutions:

- Use appropriate hardware :-)
  - ✗ It's expensive
- Distribute expensive calculations over the cloud
  - ✗ Won't scale in some cases; Works only for final rendering
- Strict Level of Detail
  - We don't see screws of the windows anyway :-)

Automatic Level of Detail

- Has to be automatically managed when working with models;
- Flexibility to customize component priority / rules of LoD is a must when creating PO
- Profit for the mainstream version is obvious! (faster work with complex objects)

Complex commercial formats

- No documentation
- Closed-source projects poorly support other closed-source formats
- Errors and data fidelity losses are common

.ifc (Industry Foundation Classes)

- Intended to describe building and construction industry data
- neutral and open specification
- not controlled by a single vendor or group of vendors
- well-known standard in architectural design field
- Still.. Commercial software doesn't export/import this format very well

Thank You for attention!

Justas Ingelevičius
+370 613 53 947
Justas@zvejone.lt

Justinas Jaronis
+370 601 36 912
justinas.jaronis@aksprendimai.lt

http://www.aksprendimai.lt/english/