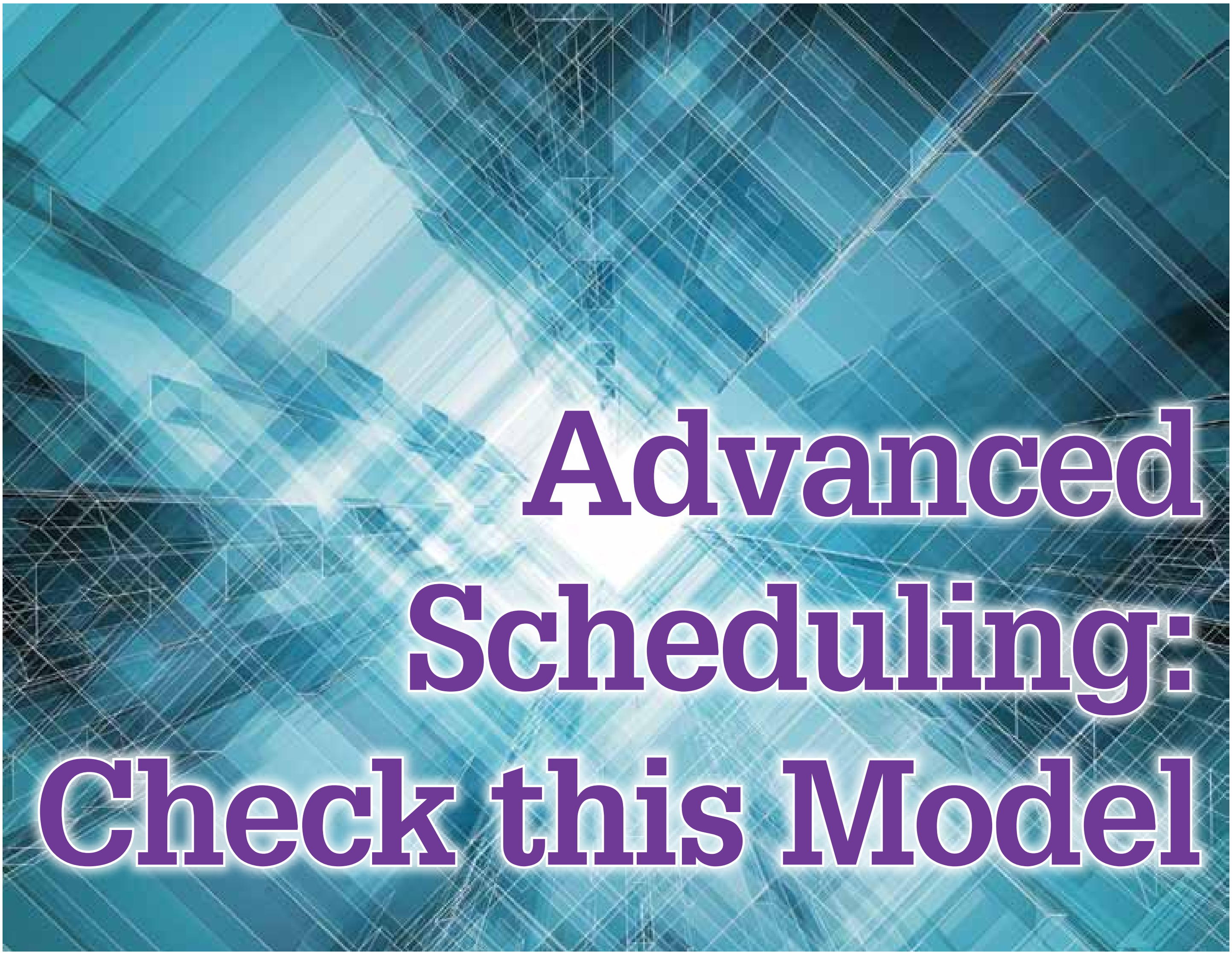
by: Martijn de Riet



Advanced Scheduling: Check this Model

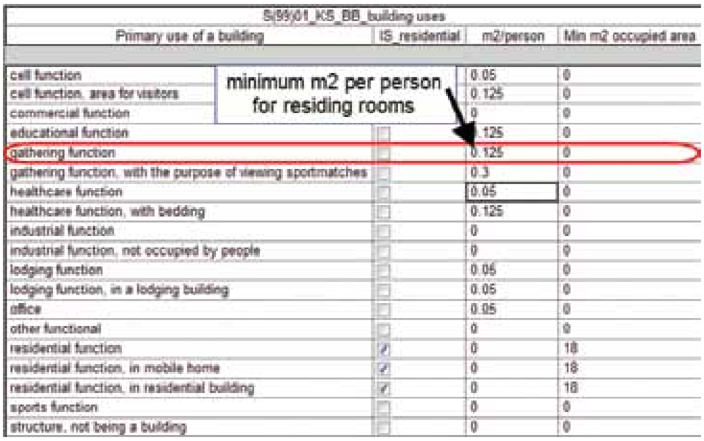
For the past few months I've been working on the Dutch Revit Standards. Currently the boring framework is all done (which, by the way, led to the IFC articles published in *AUGIWorld*), so now it's time to get on with the good stuff: schedules! Oh, and some Adaptive Components, too.

ENDLESS CAPABILITIES

Something I haven't found in any of the templates, standards, libraries, or any other resource I've seen in the last few years is Rule-Based Model Checking. And I don't mean *"why doesn't my foundation show up in a schedule?"* kind of Model Checking. I mean the real deal. Does my design meet the Building Code standards for the amount of toilets? Does my bedroom have the legally required width and area? Is the egress width of this fire door big enough compared to the size of the fire compartment(s) adjacent to it? Those are all questions Autodesk® Revit® can answer, if you know how to leverage schedules.

FRAMEWORK

Okay, even with cool stuff like schedules there is some back-end dullness you need to get through. Basically you need to set up some rules to which you can test the design. That's why Revit has Key Schedules.



S(99)01_KS_BB_building uses

Primary use of a building	IS_residential	m2/person	Min m2 occupied area
cell function		0.05	0
cell function, area for visitors		0.125	0
commercial function		0	0
educational function	□	0.125	0
gathering function	□	0.125	0
gathering function, with the purpose of viewing sportmatches	□	0.3	0
healthcare function	□	0.05	0
healthcare function, with bedding	□	0.125	0
industrial function	□	0	0
industrial function, not occupied by people	□	0	0
lodging function	□	0.05	0
lodging function, in a lodging building	□	0.05	0
office	□	0.05	0
other functional	□	0	0
residential function	☑	0	18
residential function, in mobile home	☑	0	18
residential function, in residential building	☑	0	18
sports function	□	0	0
structure, not being a building	□	0	0

minimum m2 per person for residing rooms

Figure 1: Key schedule to identify the primary use of (part of) a building.

The first thing I did was analyze the Dutch building codes. It has a very simple setup for all regulations (from fire safety to the height of a stair thread).

1. All regulations are performance based, which means there's a minimum/maximum set. This is great, because you can use those to validate your design.

2. All regulations are differentiated to the primary use of a building (part), such as Residential, Office, Medical, and so on.

3. Further differentiation depends on the primary use of a room: is it a bathroom, toilet, closet, or meant for people to reside in, and the number of people a room is intended for.

So the next thing I did was set up Key Schedules defining all possible uses of a building (see Figure 1). However, this left me with a problem: you can create only one Key Schedule for a certain Category. But the Dutch building codes are based on two distinct features: the overall usage of a building and the use of a single room. A large portion of the regulation is based upon combining those two. At first I just added more Key Names such as Office Building, toilet, Office Building, hallway, Office Building, elevator shaft, and so on. Given the fact that we have 19 or so building usages, each containing 17 different room types, that soon became a giant mess. So I decided that for the DRS I was going to cut the Key Schedule in pieces.

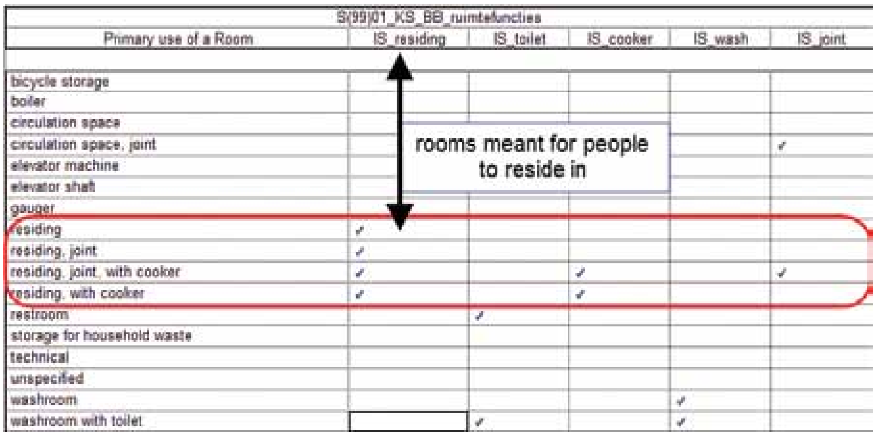
This proved to be a little hard, since you can't do that—until I found this little trick over at AUGI.

1. Create your basic Key Schedule and place it on a sheet.

2. Select it, hit Ctrl+C to copy to clipboard.

3. Close all views containing the Key Schedule.

4. Go to another sheet and hit Ctrl+V to paste. You now have a copy of your Key Schedule **which you can use to setup a new Schedule Key (see Figure 2).**



S(99)01_KS_BB_ruimtefuncties

Primary use of a Room	IS_residing	IS_toilet	IS_cooker	IS_wash	IS_joint
bicycle storage					
boiler					
circulation space					
circulation space, joint					✓
elevator machine					
elevator shaft					
gauger					
residing	✓				
residing, joint	✓				
residing, joint, with cooker	✓		✓		✓
residing, with cooker	✓		✓		
restroom		✓			
storage for household waste					
technical					
unspecified					
washroom				✓	
washroom with toilet		✓		✓	

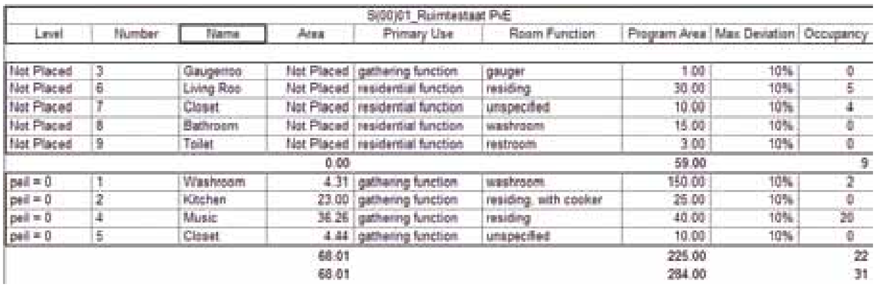
Figure 2: Second Schedule Key for rooms: use of a single room.

There is a caveat, though. These are two separate Schedule Keys, with different GUIDs. So even if you name them alike, you're still going to have two Schedule Keys in your Room Properties. If you have a Schedule Key with lots and lots of Key Values assigned to it, you're just going to have to make do.

STEP ONE: DOES YOUR DESIGN MATCH YOUR PROGRAM?

So for us, it's customary to start with a Design Program: what's the building for, what rooms do you need, and for how many occupants?

In the DRS this is covered by a schedule with a bunch of non-placed rooms (see Figure 3). This schedule contains parameters for the desired area of a room, the allowed deviation in percents, and the number of people who are going to use it. When creating rooms in your model, instead of just clicking away in Plan View, look at the Options Bar, select the desired Room from the drop-down menu, and place it where it needs to be.



S(00)01_Ruimtestaat PvE

Level	Number	Name	Area	Primary Use	Room Function	Program Area	Max Deviation	Occupancy
Not Placed	3	Gaugerroo	Not Placed	gathering function	gauger	1.00	10%	0
Not Placed	6	Living Roo	Not Placed	residential function	residing	30.00	10%	5
Not Placed	7	Closet	Not Placed	residential function	unspecified	10.00	10%	4
Not Placed	8	Bathroom	Not Placed	residential function	washroom	15.00	10%	0
Not Placed	9	Toilet	Not Placed	residential function	restroom	3.00	10%	0
			0.00			59.00		9
peil = 0	1	Washroom	4.31	gathering function	washroom	150.00	10%	2
peil = 0	2	Kitchen	23.00	gathering function	residing, with cooker	25.00	10%	0
peil = 0	4	Music	36.26	gathering function	residing	40.00	10%	20
peil = 0	5	Closet	4.44	gathering function	unspecified	10.00	10%	0
			68.01			225.00		22
			68.01			284.00		31

Figure 3: Building program schedule.

Placing these rooms fills up a second schedule (which doesn't show Not Placed Rooms by filtering for Areas > 0) that does one thing.

CHECK THE ACTUAL AREA AGAINST THE DESIRED AREA

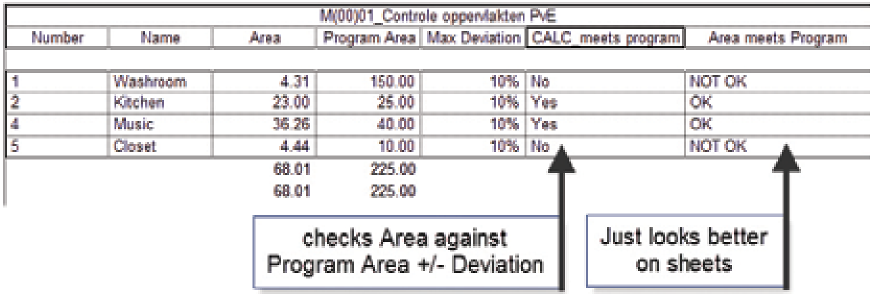
This is a Yes/No Calculated Value, named CALC_meets program, with a formula that is actually quite straightforward:

*and(Area < (Program Area + (Program Area * toegestane afwijking PvE)), Area > (Program Area - (Program Area * toegestane afwijking PvE)))*

What this does is check whether that Actual Area is within the desired range. To add a text field telling you it's OK or NOT OK, simply add another calculated value that says:

if(CALC_meets program, "OK", "NOT OK")

Revit Architecture 2013



M(00)01_Controle oppervlakten PvE						
Number	Name	Area	Program Area	Max Deviation	CALC_meets program	Area meets Program
1	Washroom	4.31	150.00	10%	No	NOT OK
2	Kitchen	23.00	25.00	10%	Yes	OK
4	Music	36.26	40.00	10%	Yes	OK
5	Closet	4.44	10.00	10%	No	NOT OK
		68.01	225.00			
		68.01	225.00			

checks Area against Program Area +/- Deviation
Just looks better on sheets

Figure 4: Checking program area against actual area.

CHECK THE OCCUPANCY AGAINST THE BUILDING CODE

The second schedule checks the desired occupancy (number of people per room) against building code regulations. Figure 5 shows the regulations of a minimum number of people per square meter of residing room space. For gathering functions, this is 0.125 person (or 8m2 per person).

To check this, we need a few things.

1. Is the room a room meant for people to reside in? This is covered in the second Key Schedule, where yes/no parameters are added to specify the kind of room (see Figure 2).

2. Room schedule with the following fields (except some "standard" properties such as name and number):

- **IS_residing** = defines which rooms are residing to base further calculations upon.

- **Program_nr of people** = number of people defined in the Design Program schedule.

- **BC_ppl per m2** = reads Key Schedule 1 for the number of people per m2 based on the Building Function.

- **BC_nr of people** = Calculated Value that has the following formula: *IF(IS_residential, roundup((Area / 1 m²) * BC_ppl per m2), 0)* This function determines whether the Building Code even applies, and if it does, which number of people a room should be designed to handle.

- **Program meets BC** = Calculated value which checks whether the parameter Program_nr of people is not smaller then the parameter BC_nr of people, using the following formula: *if(not(Program_nr of people < BC_nr of people), "OK", "NOT OK")*

STEP 2: DOING WHAT CAN'T BE DONE

Another big deal in the Dutch building code is the size of certain rooms. A residential unit needs at least 18m2 of residing rooms, each being at least 1.8m wide and 5m2 in floor area. A toilet has to have a floor area of at least 0.9x1.2m, and so on.

Now I know this can be checked manually. And it can probably be checked by tapping into the API, too. But it's not supposed to be doable in native Revit. Or is it?

Let's just see about that…

To start with: it's good to know that there are several Revit Family Categories that are aware of the room or space in which they are placed. Air Terminals, Furniture (Systems), Specialty Equipment, and Plumbing Fixtures are some of them. Another thing to know about is a special kind of families: Adaptive Components. These families have the ability to flex upon placement in a project and can be used in several Family Categories, one of which is Furniture Systems.

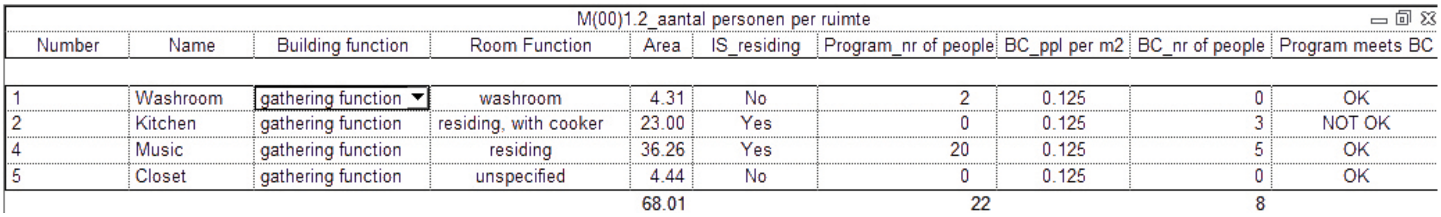
Combining those two together you can create a family that can flex upon placement, adjust to changes in room sizes, and knows which room it's placed in (and most importantly this can be scheduled, too!).

A "SIMPLE" FAMILY

I created a simple Adaptive Component quadrangle which can be placed as a rectangle, square, diamond-shaped form, and so on. Using a few shared reporting parameters that read the length of all sides and the length between the two parallel sides and some IF / AND functions, it's possible to report the smallest width and length of this shape in a schedule using Calculated Values.

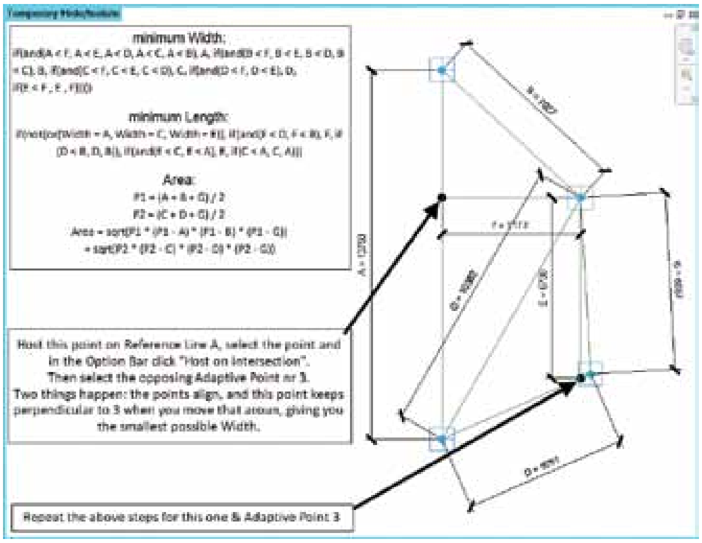
Then, you need another shared reporting parameter measuring the length of a diagonal (creating two triangles) and some basic math to calculate the area for any quadrangle shape with non-intersecting sides (which is cool since rooms usually don't have intersecting walls)—see Figure 6.

Now that we have the family reporting all the necessary stuff for size and area, it's time to create some rules. For this article, let's focus on one possible approach: A residential unit needs at least 18m2 of rooms for people to reside in, all being at least 1.8m wide and 5m2 in floor area.



M(00)1.2_aantal personen per ruimte									
Number	Name	Building function	Room Function	Area	IS_residing	Program_nr of people	BC_ppl per m2	BC_nr of people	Program meets BC
1	Washroom	gathering function	washroom	4.31	No	2	0.125	0	OK
2	Kitchen	gathering function	residing, with cooker	23.00	Yes	0	0.125	3	NOT OK
4	Music	gathering function	residing	36.26	Yes	20	0.125	5	OK
5	Closet	gathering function	unspecified	4.44	No	0	0.125	0	OK
				68.01		22		8	

Figure 5: Checking occupancy against building code.



Temporary Hide/Isolate

minimum Width:
if(and(A < F, A < E, A < D, A < C, A < B), A, if(and(B < F, B < E, B < D, B < C), B, if(and(C < F, C < E, C < D), C, if(and(D < F, D < E), D, if(E < F, E, F)))))

minimum Length:
if(not(or(Width = A, Width = C, Width = B)), if(and(F < D, F < B), F, if(D < B, D, B)), if(and(E < C, E < A), E, if(C < A, C, A)))

Area:
P1 = (A + B + G) / 2
P2 = (C + D + G) / 2
Area = sqrt(P1 * (P1 - A) * (P1 - B) * (P1 - G)) + sqrt(P2 * (P2 - C) * (P2 - D) * (P2 - G))

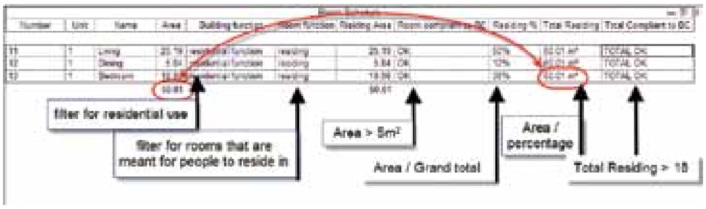
Host this point on Reference Line A, select the point and in the Option Bar click "Host on Intersection". Then select the opposing Adaptive Point nr 3. Two things happen: the points align, and this point keeps perpendicular to 3 when you move that around, giving you the smallest possible Width.

Repeat the above steps for this one & Adaptive Point 3

Figure 6: Finding smallest width and length and calculate area.

WHY DID WE WANT TO DO THIS AGAIN?

Can't I just do this with a room schedule? Well, you can come a long way (Figure 7); however, you cannot check for room width using just a room schedule. In this particular example, this means the difference between getting a permit or just having to redo the entire layout.



Room Schedule

Number	Unit	Name	Area	Building function	Room function	Residing Area	Room compliant to BC	Residing %	Total Residing	Total Compliant to BC
11	1	Living	25.79	residential function	residing	25.79	OK	52%	50.01 m²	TOTAL OK
12	1	Dining	5.84	residential function	residing	5.84	OK	12%	50.01 m²	TOTAL OK
13	1	Bedroom	18.38	residential function	residing	18.38	OK	[illegible]	50.01 m²	TOTAL OK
			50.01			50.01				

filter for residential use
filter for rooms that are meant for people to reside in
Area > 5m²
Area / Grand total
Area / percentage
Total Residing > 18

Figure 7: Using a room schedule to check for building code compliance.

CHECKING WIDTH

1. Place the family in the project, in all the rooms you want to check.

2. Create a Furniture Systems schedule with the following fields:

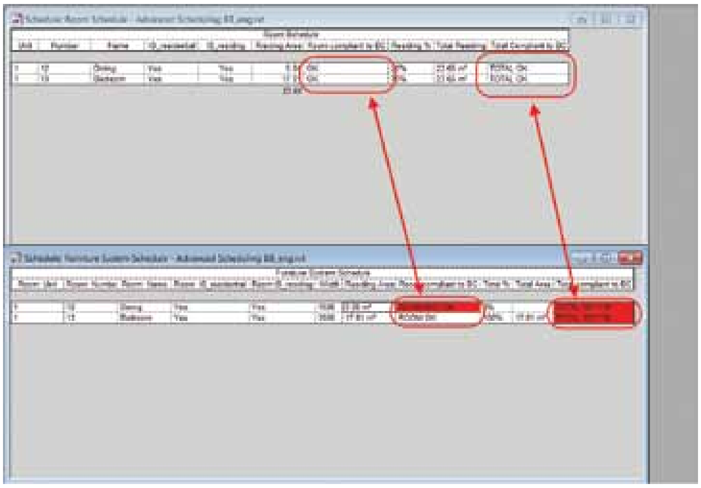
   - Room: Name

   - Room: IS_residential

   - Shared Parameters A, B, C, D, E, F and G from the Furniture Systems Families (you can hide those later on)

   - Calculated Values Width (Length) and Area (Area), using the formulas in Figure 6.

- Maybe add some calculated values to show whether or not this unit complies to the Building Code (see Figure 8).



Schedule: Room Schedule - Advanced Scheduling 03.rvt

Room Schedule

Unit	Number	Name	IS_residential	IS_residing	Residing Area	Room compliant to BC	Residing %	Total Residing	Total Compliant to BC
1	12	Dining	Yes	Yes	[illegible]	OK	[illegible]	23.65 m²	TOTAL OK
1	13	Bedroom	Yes	Yes	[illegible]	OK	[illegible]	23.65 m²	TOTAL OK
					23.65				

Schedule: Furniture System Schedule - Advanced Scheduling 03.rvt

Furniture System Schedule

Room: Unit	Room: Number	Room: Name	Room: IS_residential	Room: IS_residing	Width	Residing Area	Room compliant to BC	Time %	Total Area	Total compliant to BC
1	12	Dining	Yes	Yes	[illegible]	23.26 m²	[illegible]	[illegible]	[illegible]	[illegible]
1	13	Bedroom	Yes	Yes	2500	17.81 m²	ROOM OK	50%	17.81 m²	[illegible]

Figure 8: Same check, but now compared to using the Furniture Systems Family.

CONCLUSION

A few short examples in this article show the power of using schedules and helper families that have "special" capabilities.

If you want more examples, find the Dutch Revit Standards on the www.revitgg.nl website. This resource has a lot of different model checks incorporated, is completely documented and open for anyone to download and use.

Oh, it is in Dutch, of course. I Google Translated it for this article, so you'd probably get past that, too!

Martijn de Riet is a self-employed BIM Consultant from the Netherlands, working with Revit since version 5.1. Martijn has a bachelor degree in Building Science. After his study he started his own engineering firm working for contractors, architects and private clients. Starting 2007 his company transformed into a full-time BIM consultancy service. At the moment, Martijn's clients vary from mid-sized architectural firms to the largest Dutch general contractor and MEP engineering firms, with a focus on specific corporate solutions, design, and implementations of Revit and BIM workflows. Martijn is a member of the Dutch Revit User Group and currently working on creating a Master Template and library. He provides lectures for companies, technical universities, seminars and such on a regular basis.