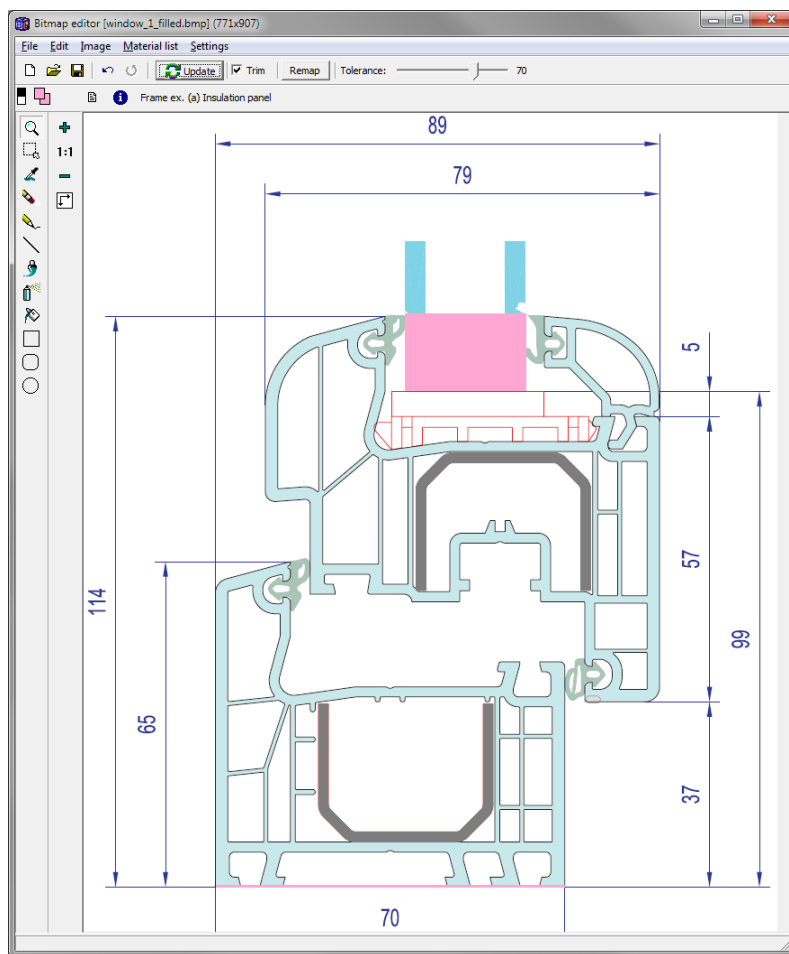


# HEAT2 8

A PC-program for heat transfer in two dimensions

Update manual



June 20, 2011

**BLOCON**

[www.buildingphysics.com](http://www.buildingphysics.com)

# Contents

<b>1. WHAT'S NEW IN HEAT2 VERSION 8.....</b>	<b>3</b>
1.1 POLYGON TOOL ADDED .....	3
1.2 BITMAP DRAWING AND FILLING.....	3
1.3 INCREASED SIZES FOR DATA FIELDS .....	3
1.4 MISCELLANEOUS .....	4
1.5 REQUIREMENTS .....	4
1.6 LIGHT VERSION.....	4
<b>2. THE POLYGON TOOL.....</b>	<b>5</b>
2.1 INTRODUCTION .....	5
2.2 POLYGON TRIMMING .....	12
<b>3. BITMAP DRAWING AND FILLING.....</b>	<b>13</b>
3.1 THE BITMAP EDITOR .....	13
3.2 USING BITMAP OBJECTS IN THE PRE-PROCESSOR.....	13
3.3 FILLING BITMAPS .....	18
3.4 HOW BITMAP COLORS ARE LINKED TO THE MATERIAL LIST IN HEAT2.....	32
3.5 REMAPPING COLORS .....	34
3.6 CREATING A NEW BITMAP.....	35
3.7 BITMAPS AND NUMERICAL MESH .....	35
3.8 BITMAPS AND FRAME CAVITIES .....	37
3.9 BITMAPS AND MODIFICATIONS .....	38
3.10 USING A CAD-PROGRAM OR A DRAWING PROGRAM AS A PRE-PROCESSOR.....	39
<b>4. MISCELLANEOUS .....</b>	<b>43</b>
4.1 LENGTH EDITOR ADDED.....	43
4.2 DIALOG TO SET DEFAULT MATERIAL FILE.....	43
4.3 DISTANCE FROM REFERENCE POINT .....	44
4.4 NOTE ON SLOPING EXTERNAL BOUNDARIES .....	44
4.5 UPDATED TEMPERATURE/HEAT FLOW CHART .....	46
4.6 ADAPTED FOR WINDOWS 7 AND VISTA.....	47

# 1. What's new in HEAT2 version 8

Blocon is proud to present a new version of HEAT2. Many new important features have been added. Up-to-date information is given on [www.buildingphysics.com](http://www.buildingphysics.com).

This update manual covers the new features that have been added since version 7. New users should also read the update manuals for version 6 and 7, and the full manual for version 5 at

[http://www.buildingphysics.com/manuals/HEAT2\\_7\\_update\\_english.pdf](http://www.buildingphysics.com/manuals/HEAT2_7_update_english.pdf)

[http://www.buildingphysics.com/manuals/Heat2\\_6\\_update.pdf](http://www.buildingphysics.com/manuals/Heat2_6_update.pdf)

[http://www.buildingphysics.com/manuals/HEAT2\\_5.pdf](http://www.buildingphysics.com/manuals/HEAT2_5.pdf)

**Tips for reading for beginners:** For a quick start read Chapter 4 (pages 23-27) in [Manual HEAT2 5.0](#). The example in chapter 8 (pages 117-121) would also give a short introduction. After this, look at the update manuals for [HEAT2 6.0](#) and [HEAT2 7.0](#). Also see the examples for the test cases: [ISO 10211 & 10077-2 validation test cases](#)

## 1.1 Polygon tool added

A new polygon tool allows easy input of sloped edges and boundaries. See Chapter 2.

## 1.2 Bitmap drawing and filling

A new bitmap editor is added where input can be made using a variety of graphical tools, such as ellipses, rounded rectangles, brushes, pencils, fill, etc.

A bitmap or jpeg can be imported and areas can easily be filled with different materials. Very complex geometries can be given in a short time using this option.

It is also possible to draw any geometry in an external cad-program or paint-program using pre-defined colors (RGB) matching the colors in the HEAT2 material list, and then export/import a bmp/jpg and scan and create the geometry in HEAT2. See Chapter 3.

## 1.3 Increased sizes for data fields

Many data fields have been increased in sizes. The maximum number of numerical nodes has been increased from 490,000 (700·700) to 4,000,000 (2,000·2,000). Other changes are shown below.

Maximum number of	HEAT2 version 7	HEAT2 version 8
<b>Nodes</b>	490,000 (700·700)	4,000,000 (2,000·2,000)
<b>Frame cavities according to ISO 10077-2</b>	250	1,000
<b>Material boxes</b>	500	10,000
<b>Boundaries</b>	500	5,000
<b>Modification areas with radiation</b>	20	50
<b>Radiative nodes in each modification area with radiation</b>	100	1,000

## 1.4 Miscellaneous

Hundreds of minor improvements are made. The most important are shown below:

- A new editor for giving lengths for drawn rectangles in the pre-processor has been added.
- Dialog added to change and set default material file.
- The distance from a reference point to the mouse cursor is now shown in the pre-processor. This may be useful for e.g. checking the width of a sloping material.
- The chart for temperatures/heat (menu item Output/Temperature along line boundaries) has been updated. The chart editor is also updated.
- The Solve-window (F9 and F10) now shows the sum of all positive boundary flows (earlier the sum of all absolute boundary flows was shown).
- Check for software updates added.
- New report generator with export to PDF, HTML, XLS, RTF, TXT, XML, QRP, QRX.
- Now also adapted for Windows 7 and Vista.

Enhancements in post-processing:

- Faster drawing/updating
- Mouse cursor sometimes “froze” for a small moment in earlier versions. This has been fixed.

*Note that the “Movie recorder/player” is not available in the release version of version 8. We plan to add this later on.*

## 1.5 Requirements

HEAT2 8 requires only about 0.2 GB RAM (a pc with 1 GB is recommended), and less than 10 MB hard disk space. HEAT2 runs on both 32-bit and 64-bit Windows 7, Vista, XP, and compatibles.

## 1.6 Light version

The light (demo) version has the following restrictions:

- Data cannot be saved.
- Materials cannot be added or changed.
- Max 1,000,000 (1,000·1,000).
- Results for heat flows are hidden for larger problems with more than 5,000 nodes and with more than 60 nodes in each direction.
- Thermal coupling coefficients (L2D) will not be calculated.
- Report preview is disabled.
- Some other restrictions apply.

## 2. The polygon tool

### 2.1 Introduction

HEAT2 now has a polygon tool which allows easy input of sloped edges and boundaries. Figure 1 shows four polygons drawn in the pre-processor. A material is chosen for each polygon (in the same way as materials are chosen for ordinary rectangles). The slopes of the polygons will be approximated by a grid step resolution as given in the upper toolbar shown in Figure 1. The grid step 1 will give the finest approximation. Figure 2 shows the results in the post-processor for some different grid steps. The default grid step is 5. A finer approximation leads to smaller numerical cells which will give longer calculation times, especially for transient calculations, but also for steady-state analysis.

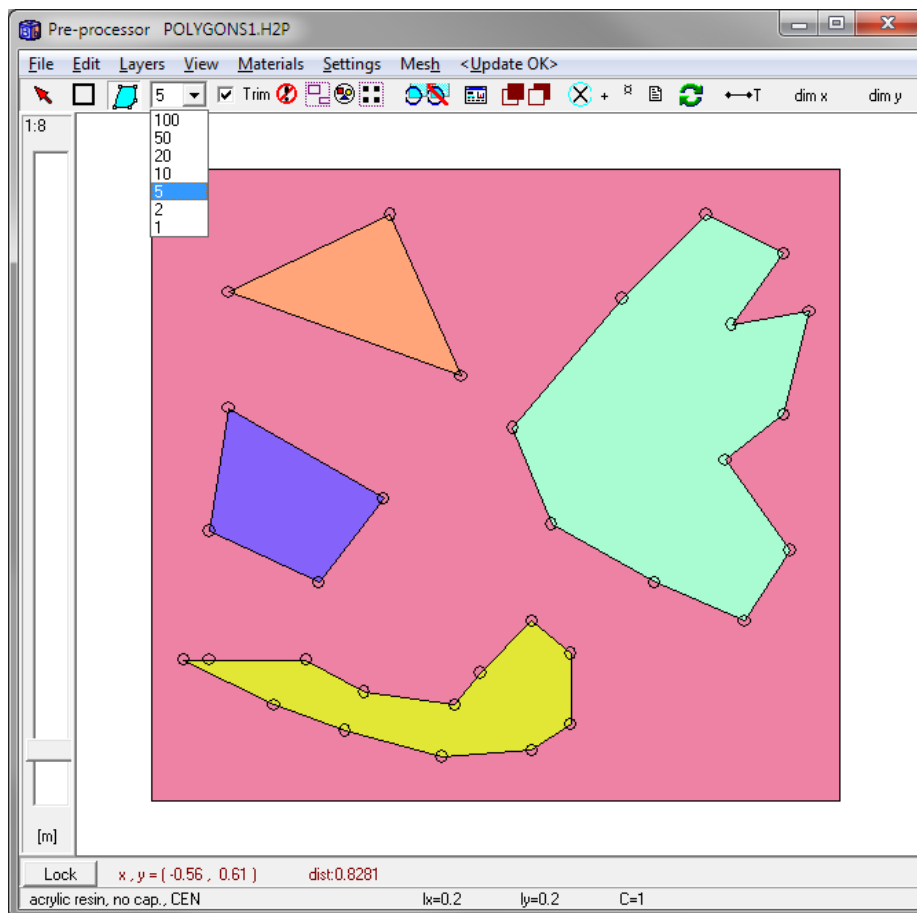
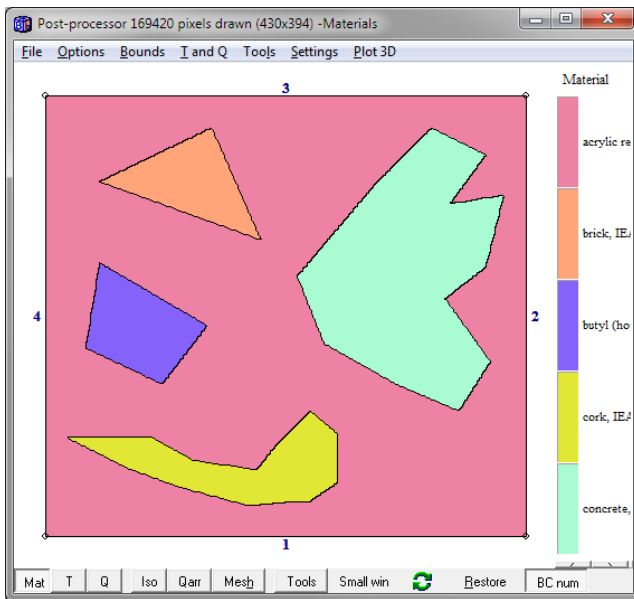
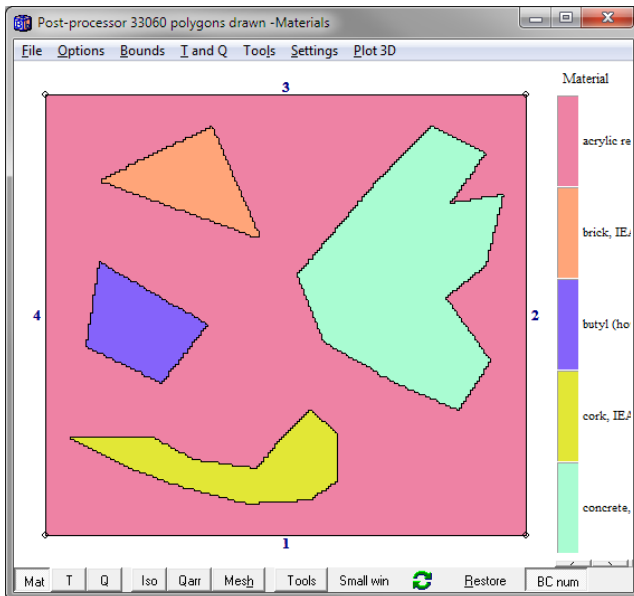


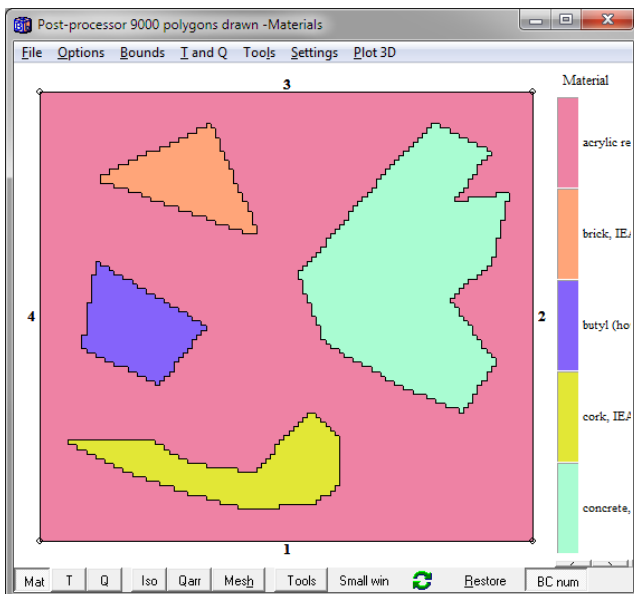
Figure 1: Example with four polygons.



Grid step 1 (finest resolution)



Grid step 5 (default)



Grid step 10

Figure 2: Different levels of grid step approximation.

Figure 3 shows an example where two polygons (green and yellow) will define external boundaries. The result for a grid step resolution of 5 is shown in Figure 4.

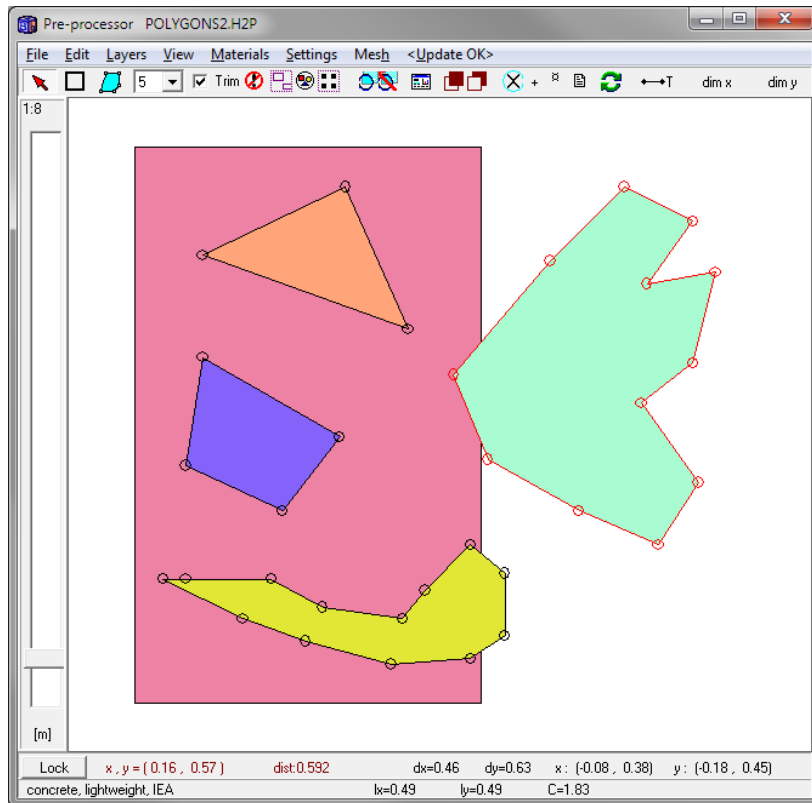


Figure 3: An example where polygons define external boundaries in pre-processor.

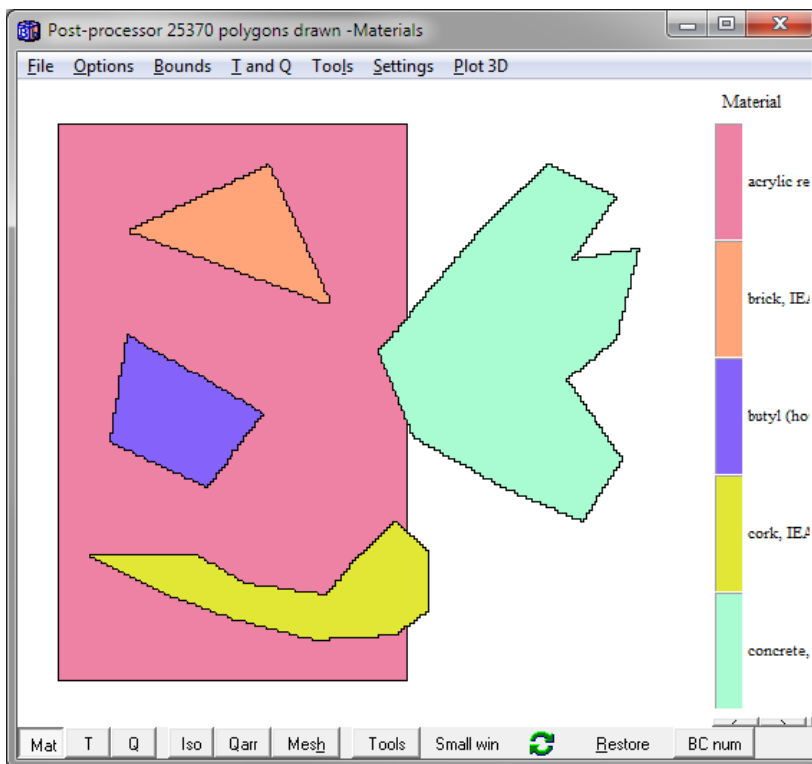


Figure 4: Result in post-processor with step-wise boundary segments.

Figure 5 shows the boundary numbers (press “BC num” on the lower tool bar to see the numbers). These numbers are used as usual to describe boundary conditions. For this example we have given  $T=1$ ,  $R=0.13$  for boundaries 2-370, and  $T=0$ ,  $R=0.04$  for boundary 372, see figure 6. The calculated temperature field is shown in Figure 7.

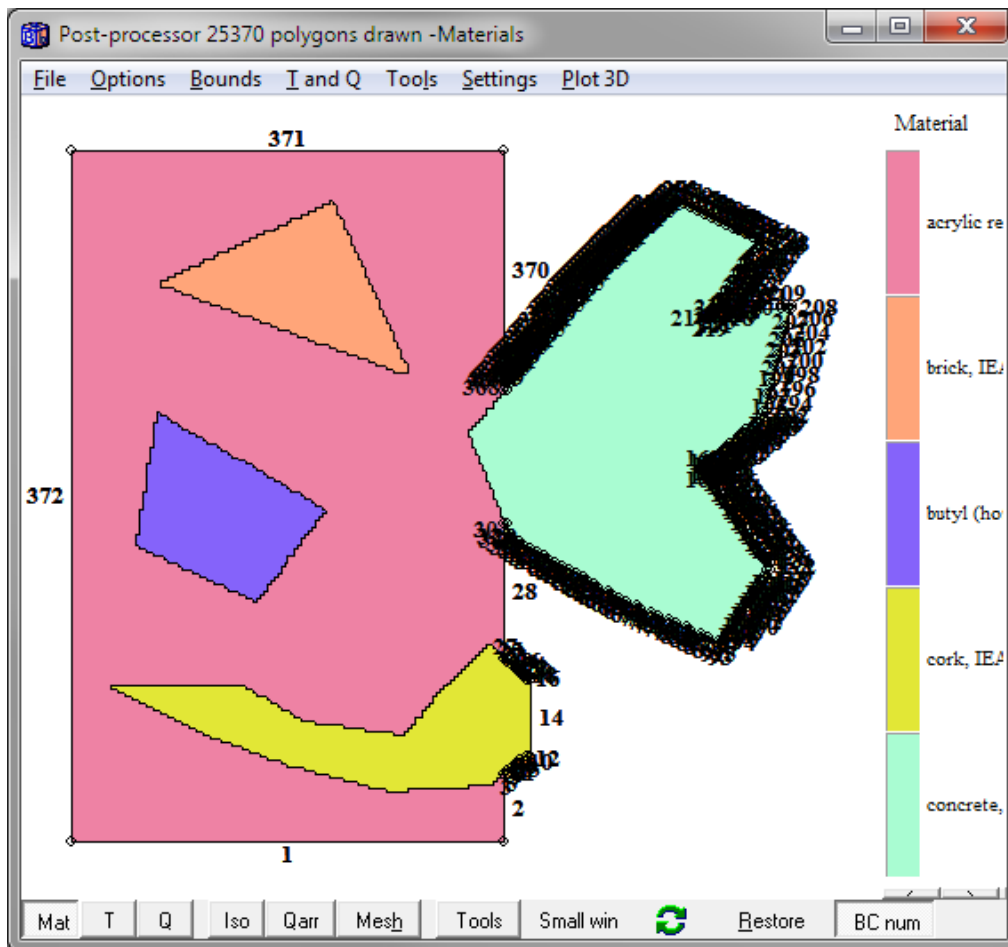


Figure 5: Boundary numbers as shown in post-processor.

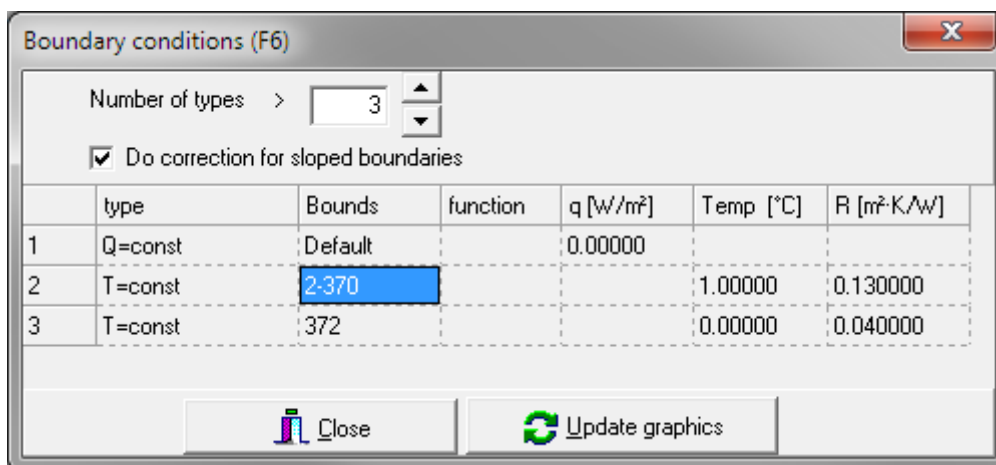


Figure 6: Boundary conditions specification window.



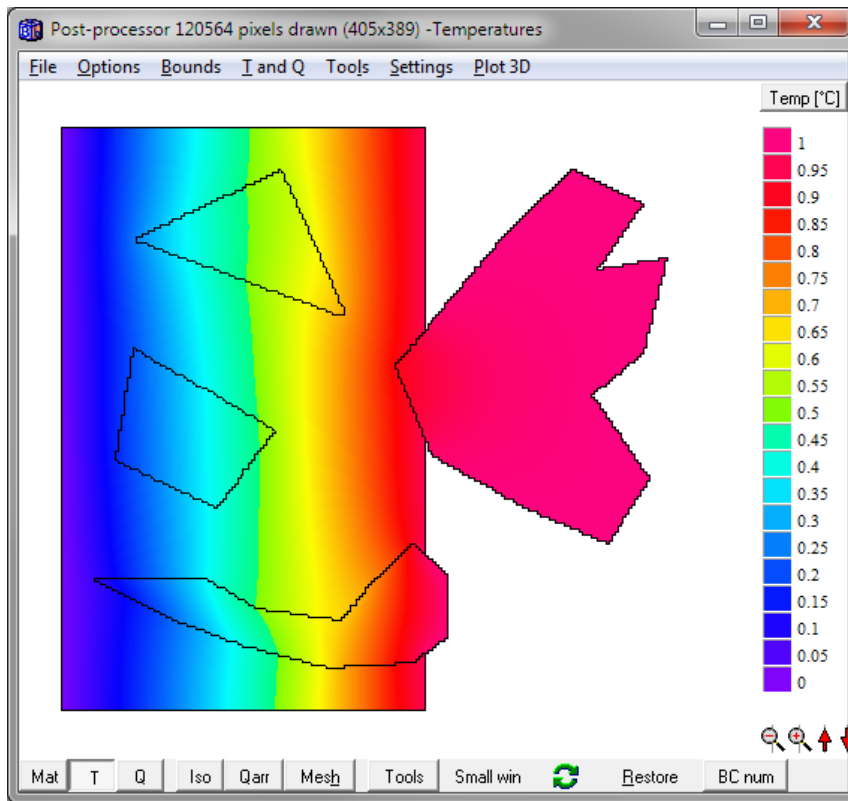


Figure 7: Calculated temperature field.

Now, assume that we want a different boundary condition ( $T=0.5$ ,  $R=0.13$ ) for boundaries 194-215, see zoomed detail in Figure 8 and Figure 9. The calculated temperature field is shown in Figure 10.

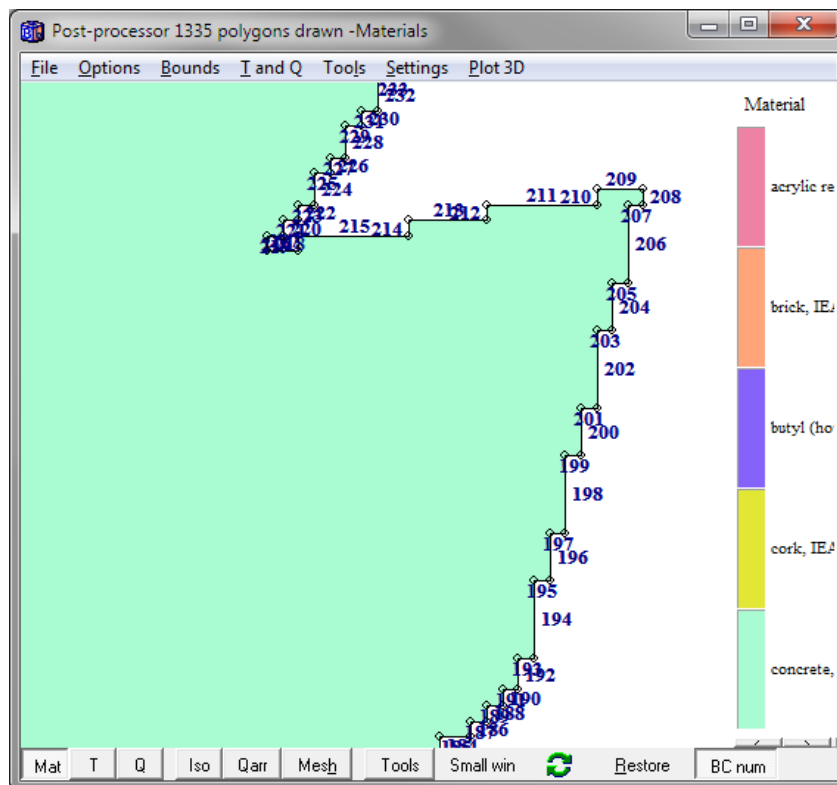


Figure 8: Detail of boundary segments.

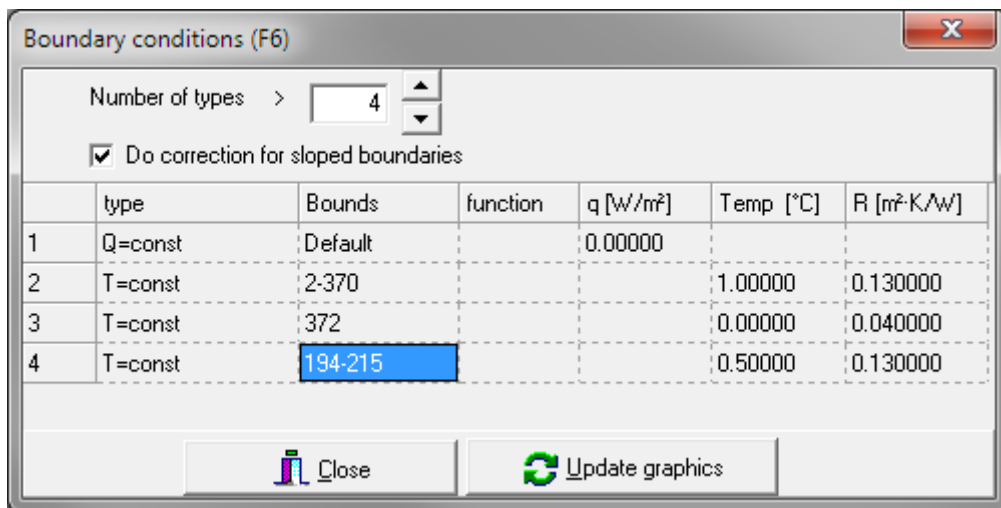


Figure 9: Boundary conditions.

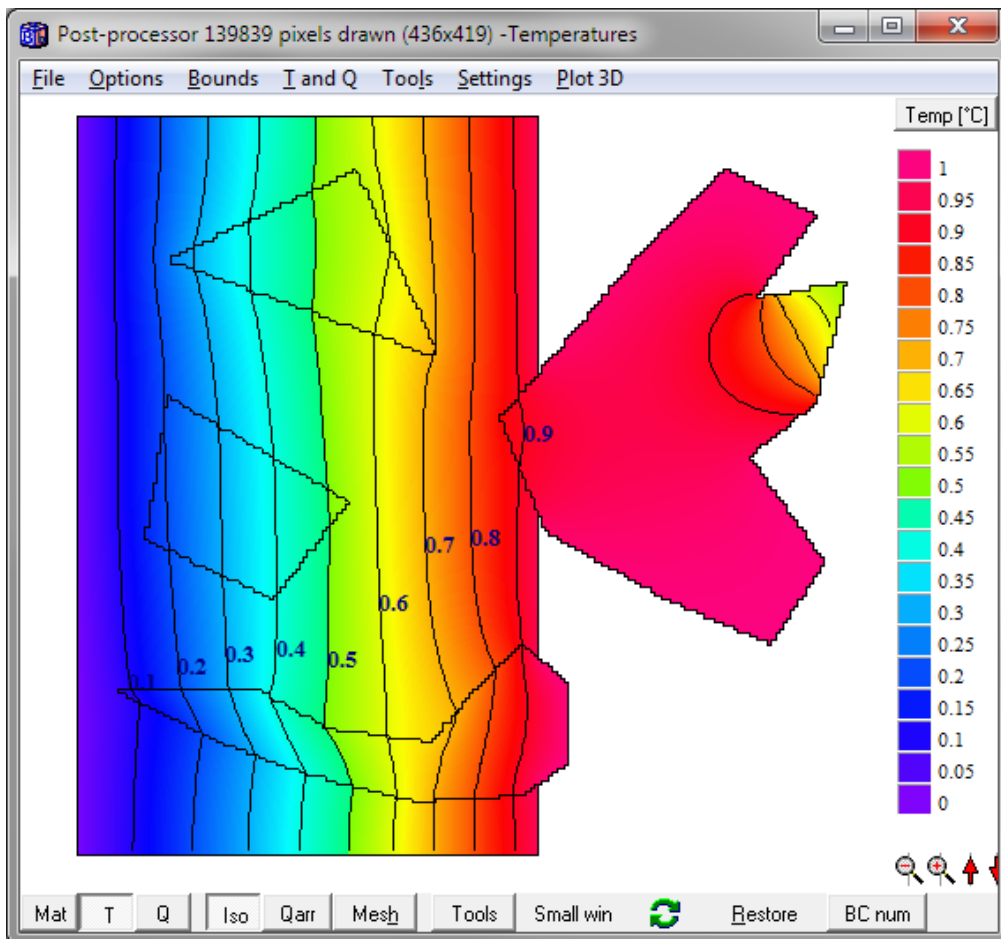


Figure 10: Calculated temperature field with isotherms.

Figure 11 and 12 show input for two ISO 10077-2 test reference cases (case D6 and D7). The left-hand side pictures show input using the old version 7 where sloped boundaries had to be created by small rectangles. The right-hand side pictures show input using the new version 8 where sloped boundaries easily are created using polygons. The input files are located in the *examples* folder.

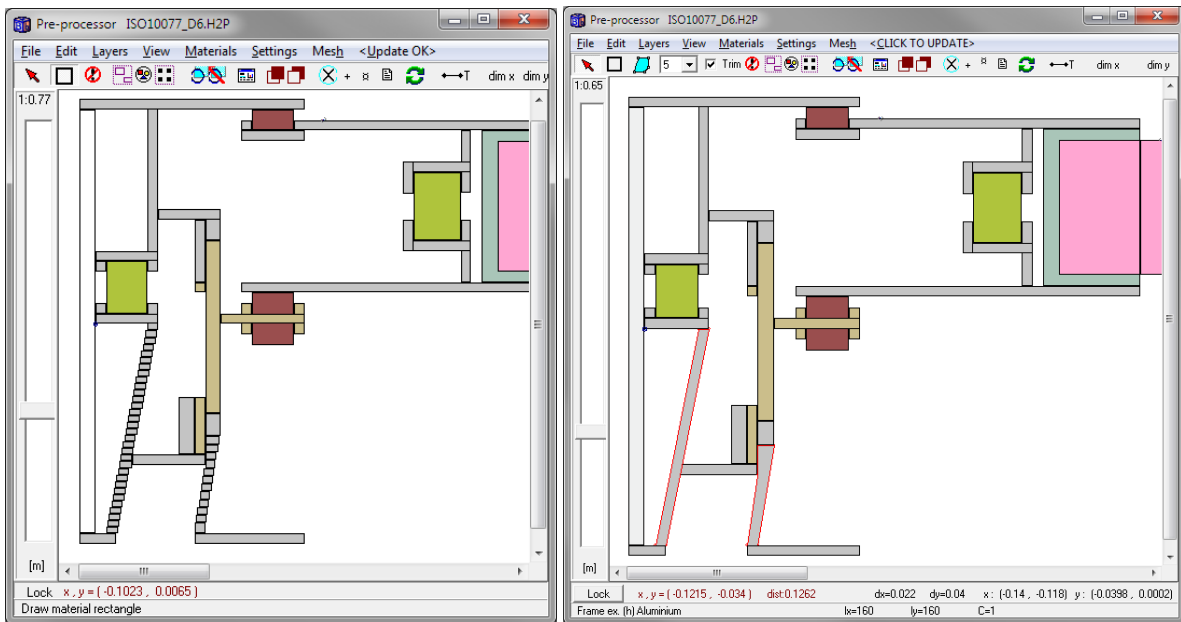


Figure 11: Input for ISO10077\_D6.dat. Left: HEAT2 7. Right: HEAT2 8 using polygons.

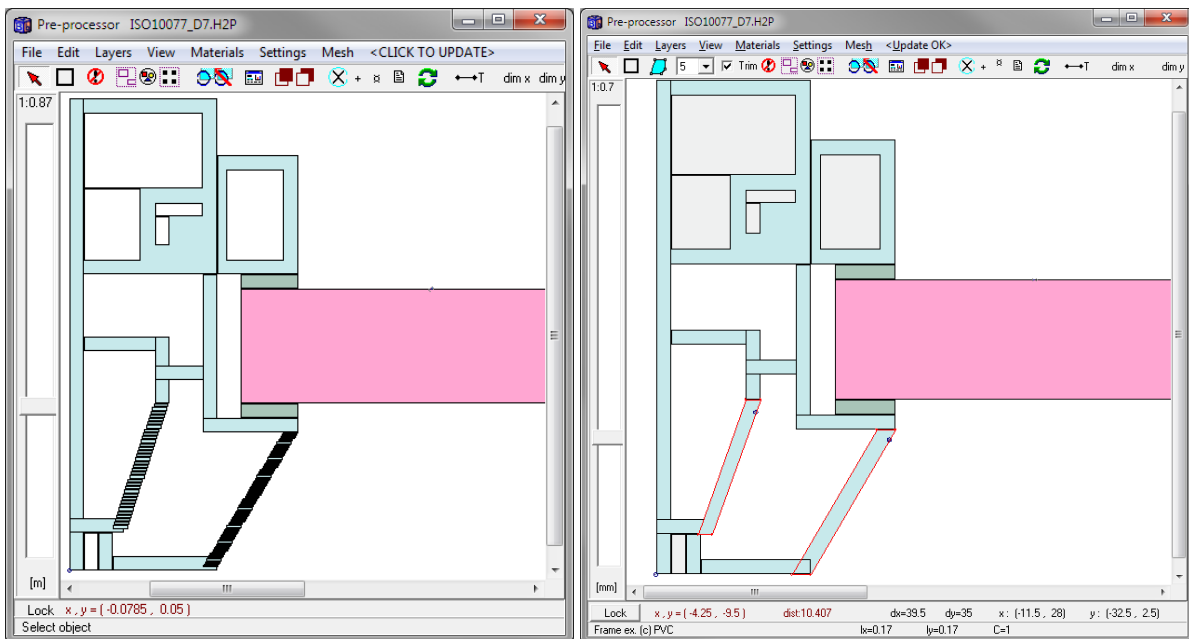
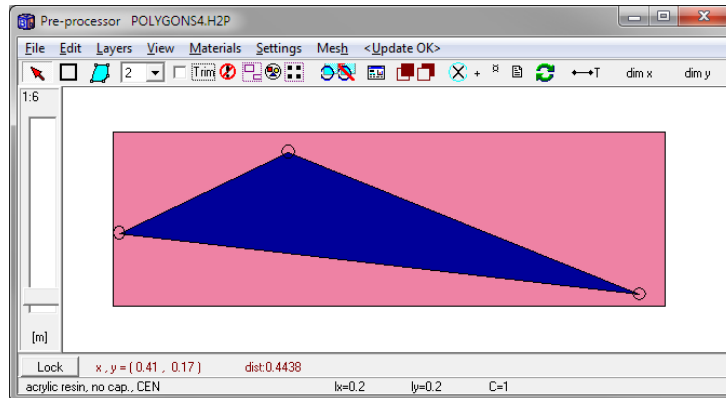


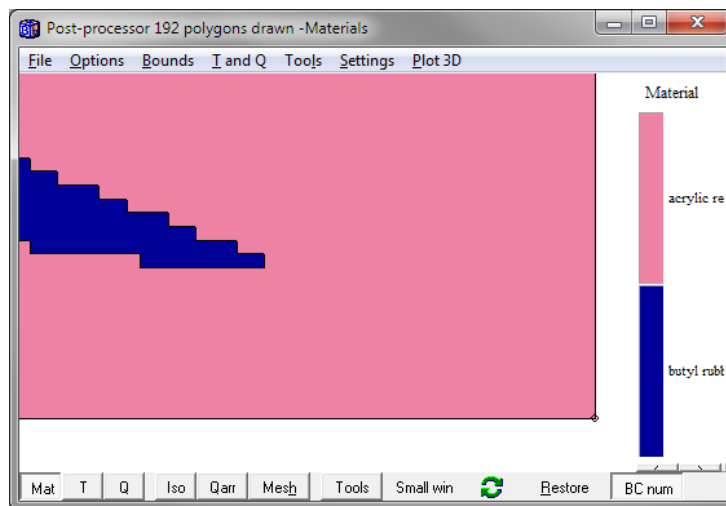
Figure 12: Input for ISO10077\_D7.dat. Left: HEAT2 7. Right: HEAT2 8 using polygons.

## 2.2 Polygon trimming

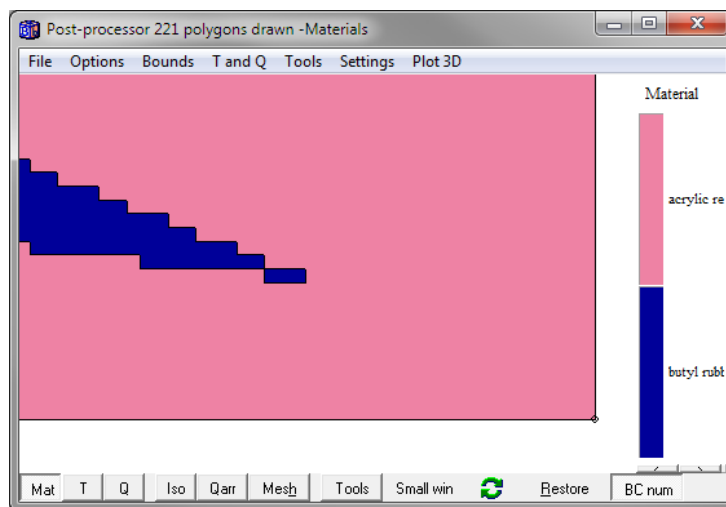
Polygons edges will be trimmed by default in order to assure a continuous area. This option can be turned off (see “Trim” checkbox on the upper tool bar row):



Original polygon.



Approximated polygon with trimmed edges.



Approximated polygon without trimmed edges. This will give a non-continuous area.

## 3. Bitmap drawing and filling

### 3.1 The bitmap editor

The bitmap editor has a variety of graphical tools, such as ellipses, rounded rectangles, brushes, pencils, fill, etc. Figure 13 shows three objects drawn in the pre-processor. The materials are chosen from the material list just in the same way as objects drawn in the pre-processor.

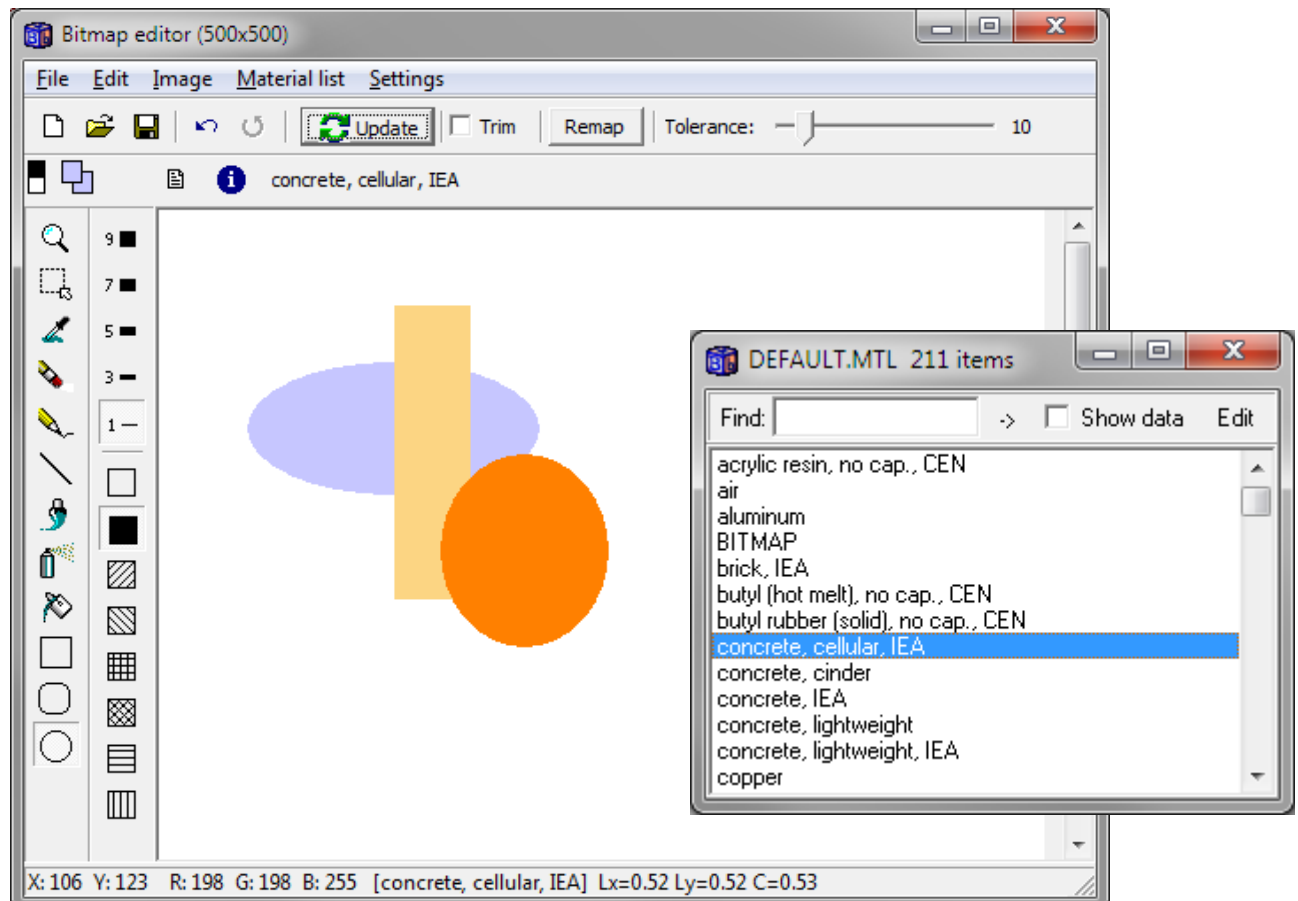


Figure 13: Three shapes drawn in the bitmap editor using materials from the material list.

### 3.2 Using bitmap objects in the pre-processor

The total width and the total height of all drawn objects in the bitmap editor will be fitted in a special rectangle drawn in the pre-processor using "BITMAP" in the material list, see Figure 14. So, the width and height of this "BITMAP" rectangle will define the sizes for the bitmap objects.

A non-rectangular "BITMAP" rectangle in the pre-processor will stretch the sizes for the bitmap objects, see Figure 15.

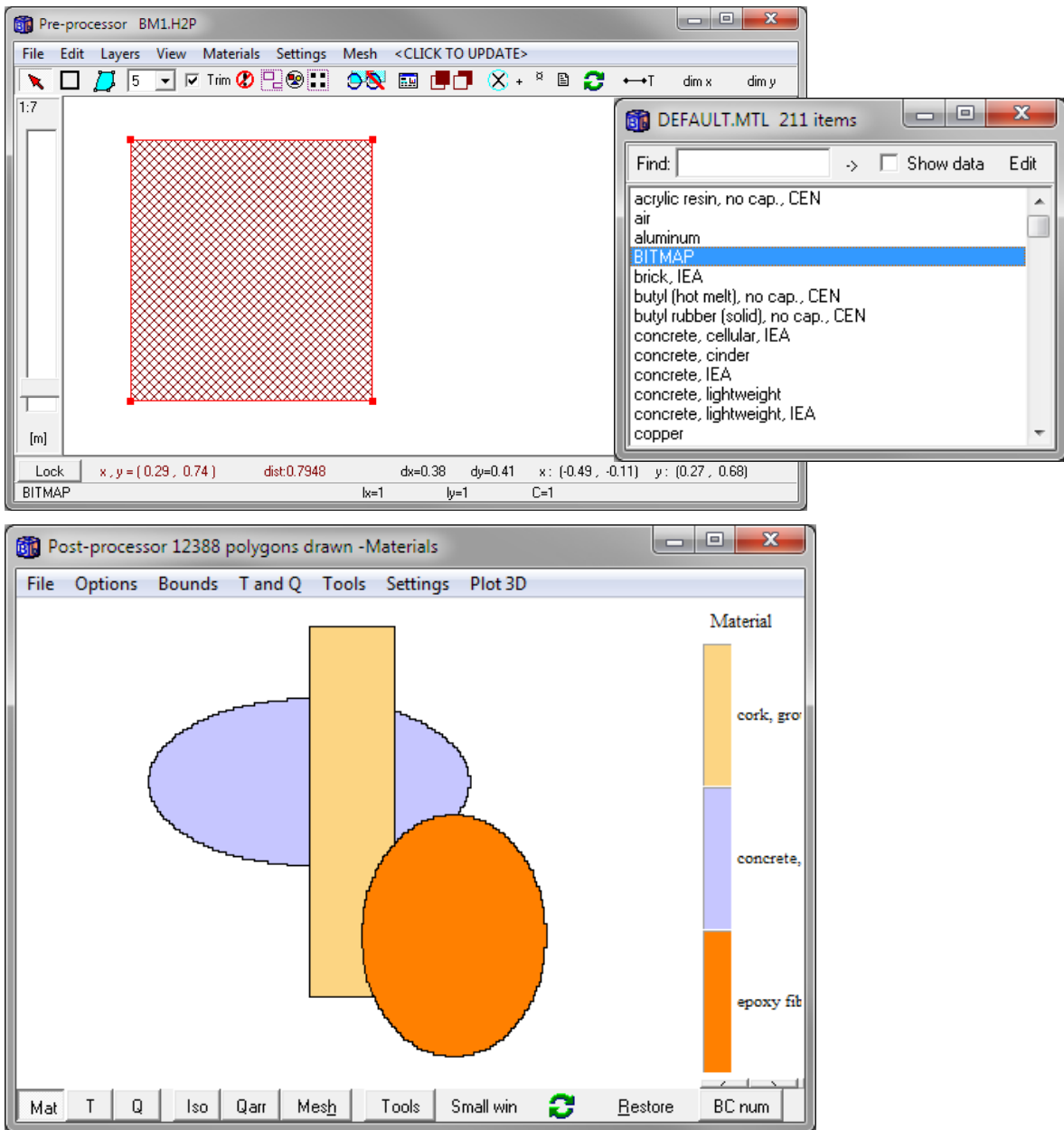


Figure 14: The “BITMAP” rectangle in the pre-processor will define the sizes for the bitmap objects.

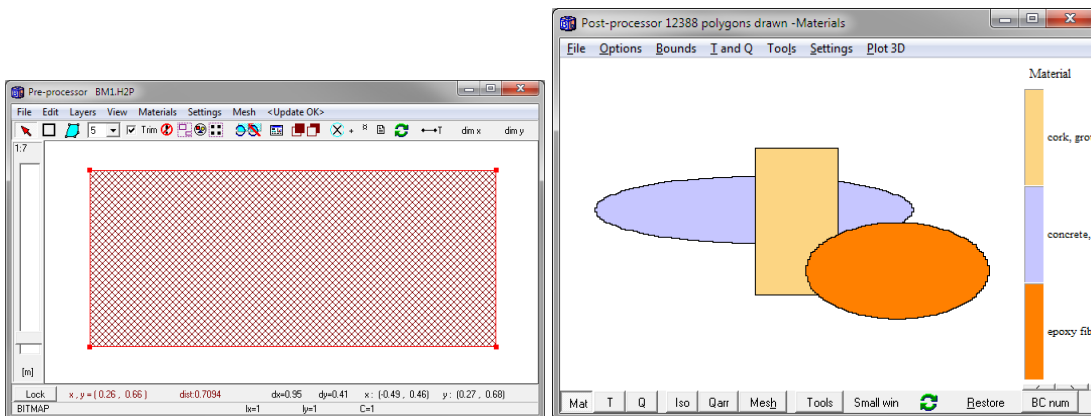


Figure 15: A non-rectangular “BITMAP” rectangle in the pre-processor will stretch the sizes for the bitmap objects.

Multiple “BITMAP” rectangles can be drawn in the pre-processor and mixed with ordinary rectangles and polygons. Figure 16 shows an example with three “BITMAP” rectangles, two ordinary rectangles and one polygon defined in the pre-processor with the result in the post-processor. Note that any object can overlap another object. E.g. in this case the polygon overlaps one “BITMAP” rectangle, and the two other “BITMAP” rectangles overlap the blue rectangle. The “BITMAP” rectangles also overlap each other.

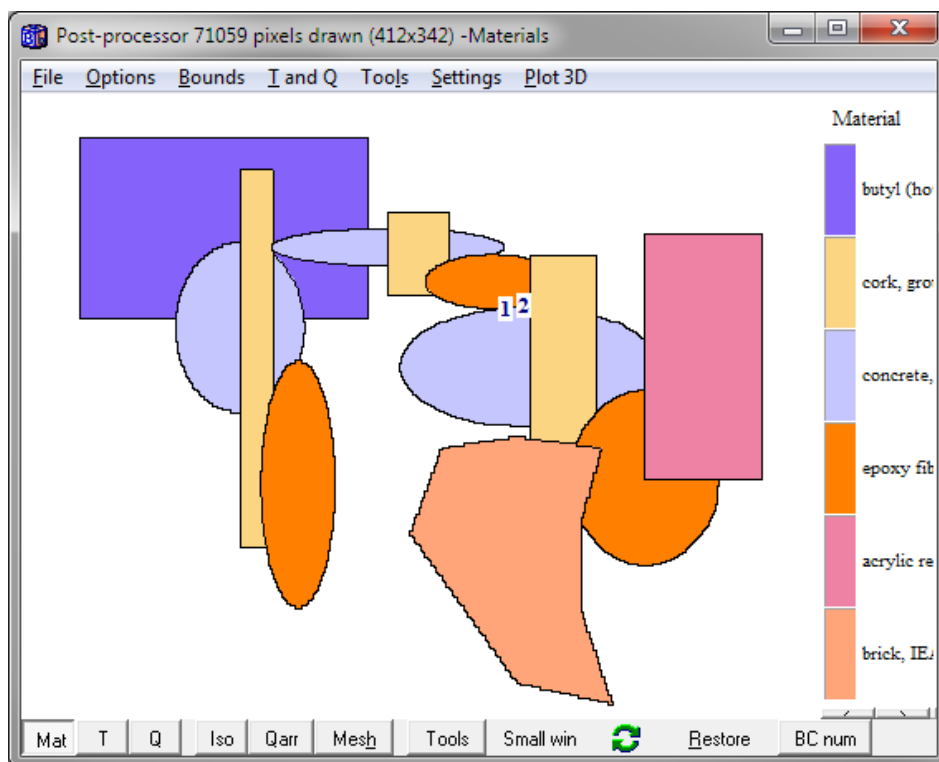
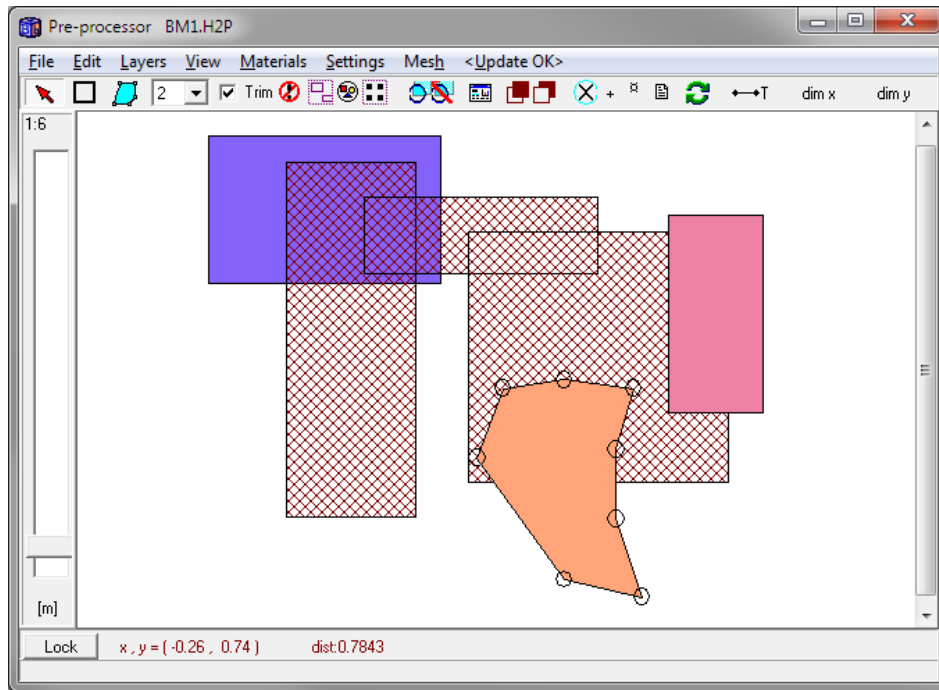
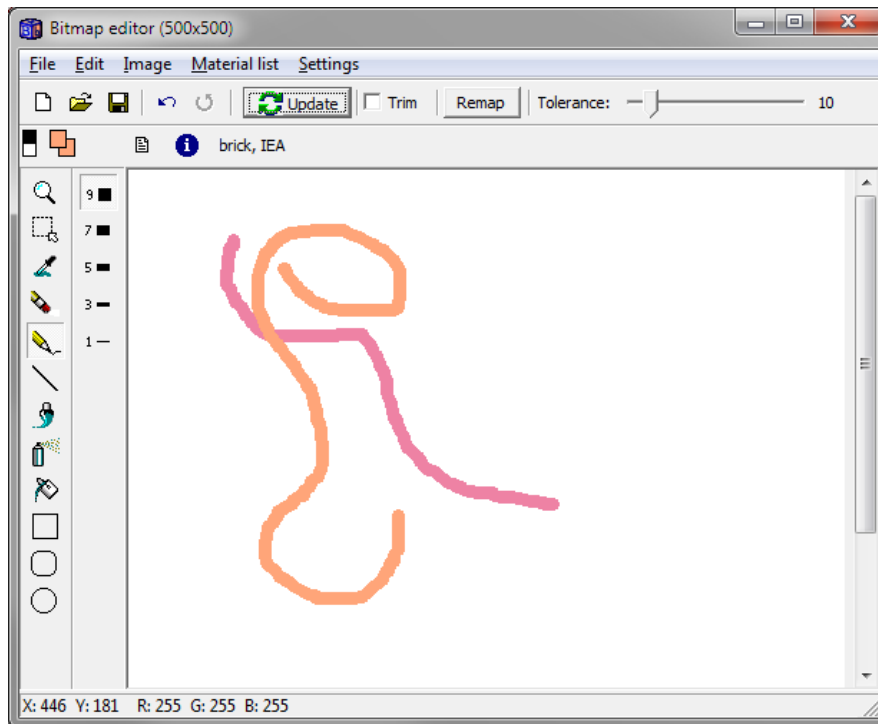
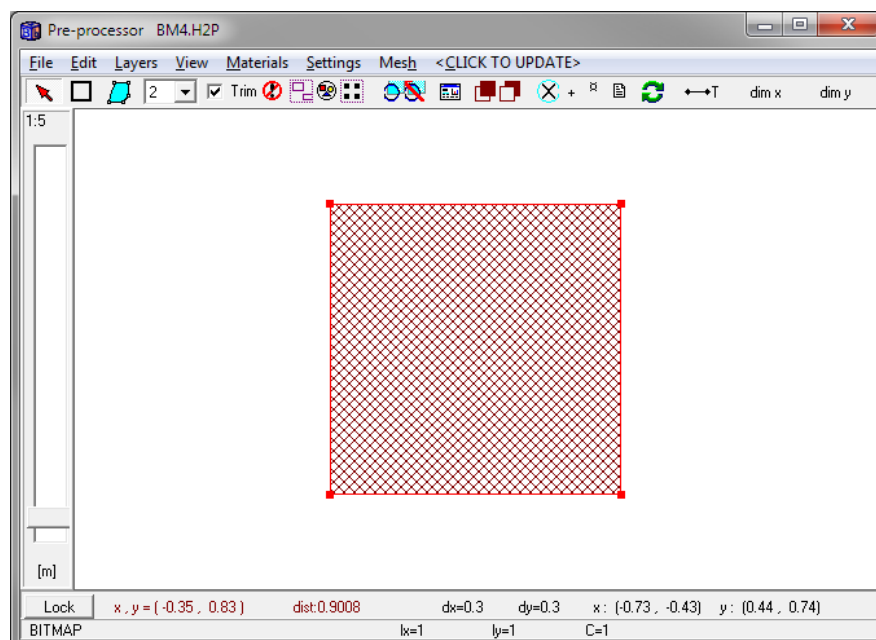


Figure 16: An example with three “BITMAP” rectangles, two ordinary rectangles and one polygon.

Figure 17 shows another example with free-hand drawing using the pencil tool of thickness 9. Note the numerical mesh for the lower detail in the post-processor. The mesh can be changed as usual in the pre-processor (menu item “Mesh”).

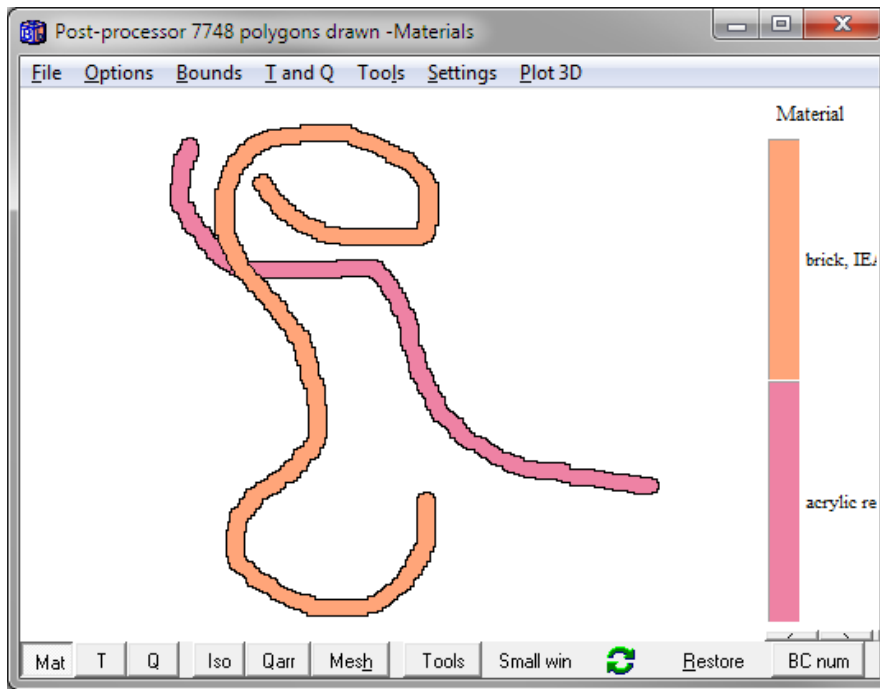


Bitmap editor.

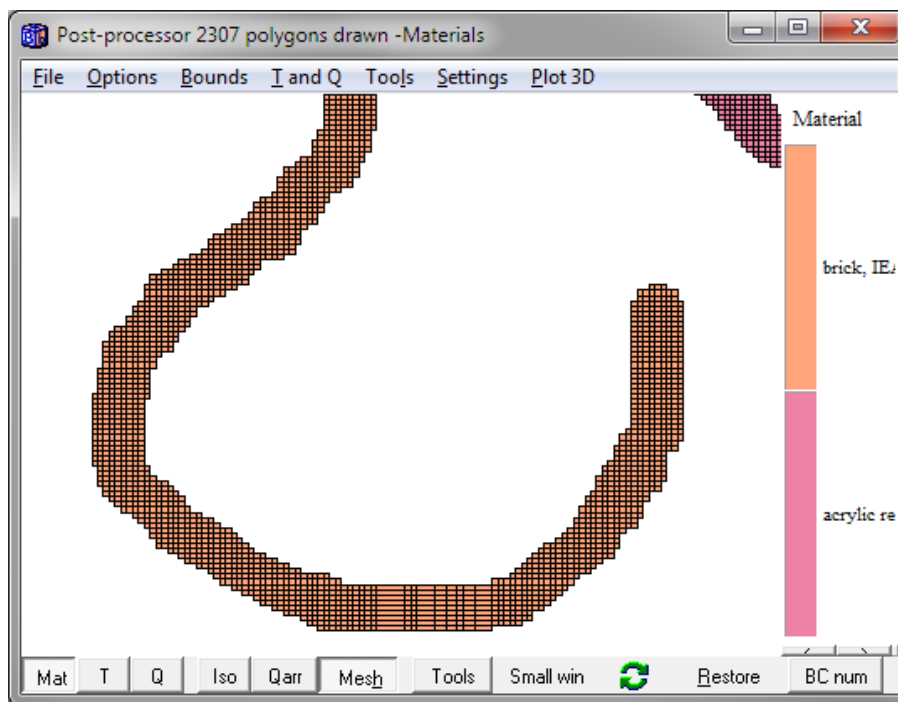


Mapping bitmap in pre-processor.





Result in post-processor.



Numerical mesh for the lower detail in post-processor.

Figure 17: An example with free-hand drawing using the pencil tool .

### 3.3 Filling bitmaps

Bitmaps and jpegs can be opened (or pasted via Windows clipboard) and manipulated in order to prepare input for a heat transfer analysis. Figure 18 shows a window frame created in a cad-program imported into HEAT2. We can use e.g. the fill-tool (on the tool-bar to the left) to quickly create the geometry. In the following example, we open a bmp-file for a window frame, generate the complete input, and solve the case in just a few mouse clicks.

*The files for the following example are located in the examples folder:*

*The clean start bitmap file: window\_1\_clean.bmp*

*The final bitmap file: window\_1\_filled.bmp*

*The final input data files window1.dat and window1.h2p*

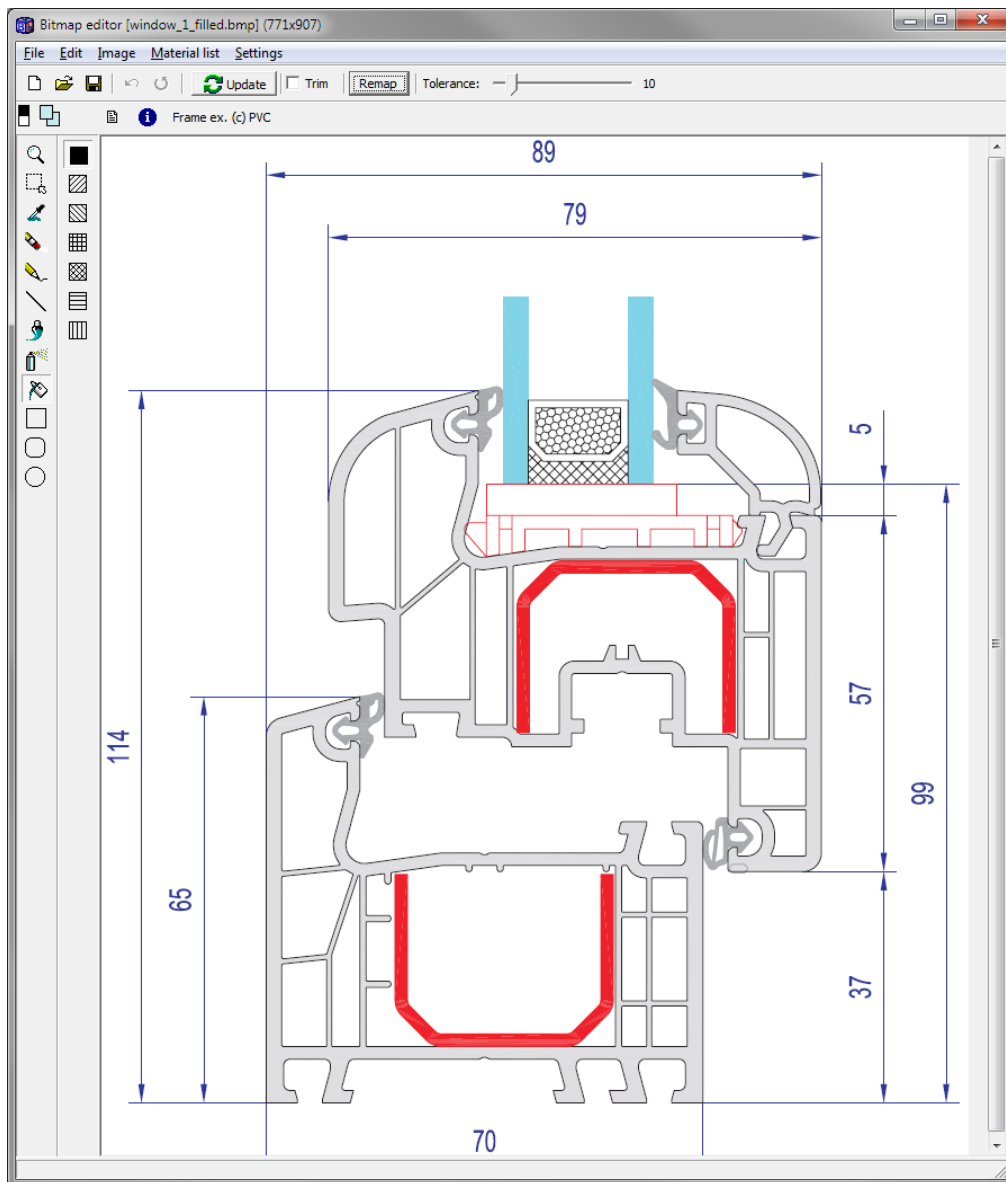


Figure 18: A window frame created in a cad-program imported into HEAT2.

Any area in the bitmap can be filled with a material using the fill tool. Figure 19 shows one part of the pvc frame filled with “Frame ex (c) PVC” chosen from the material list.

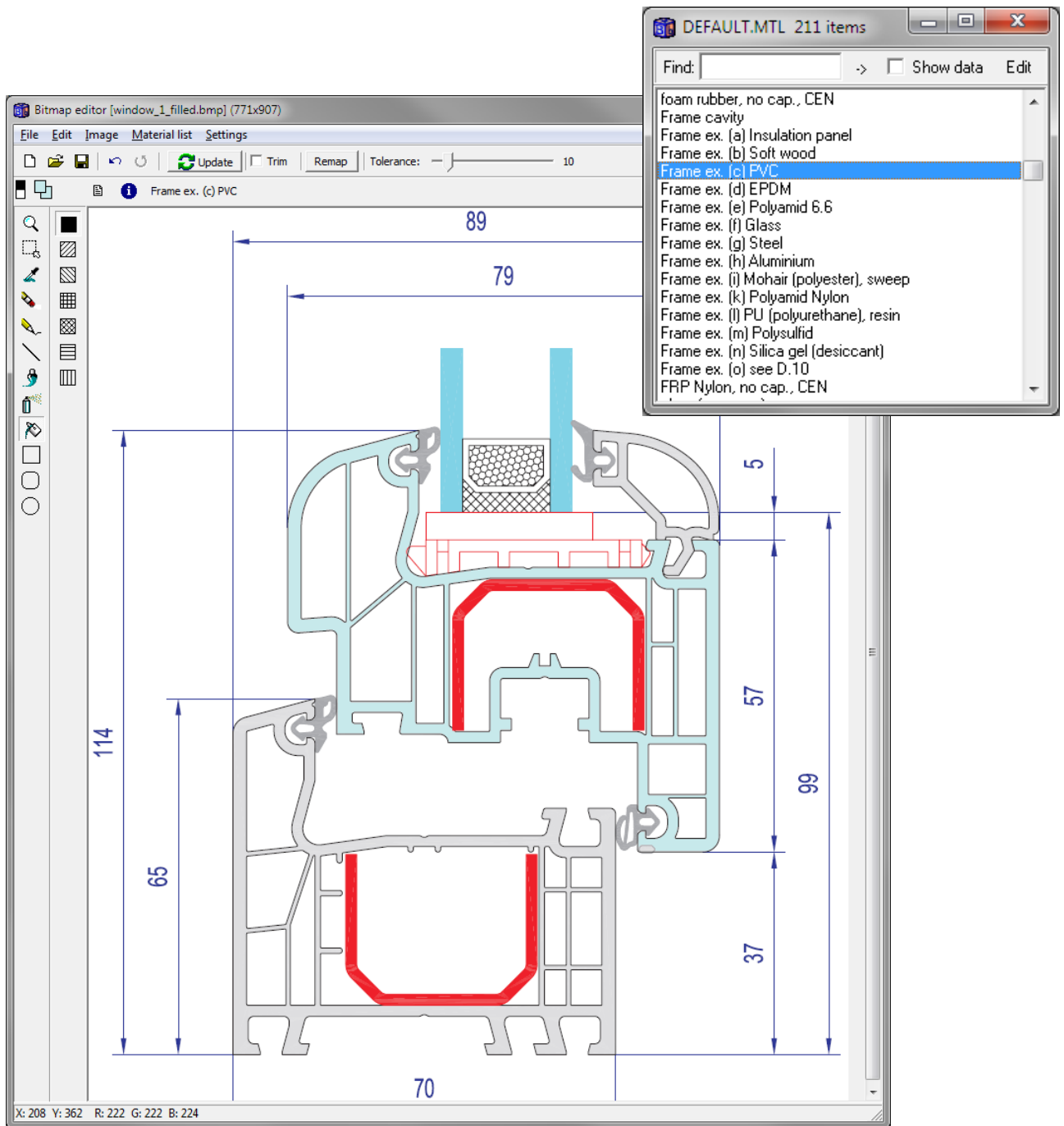
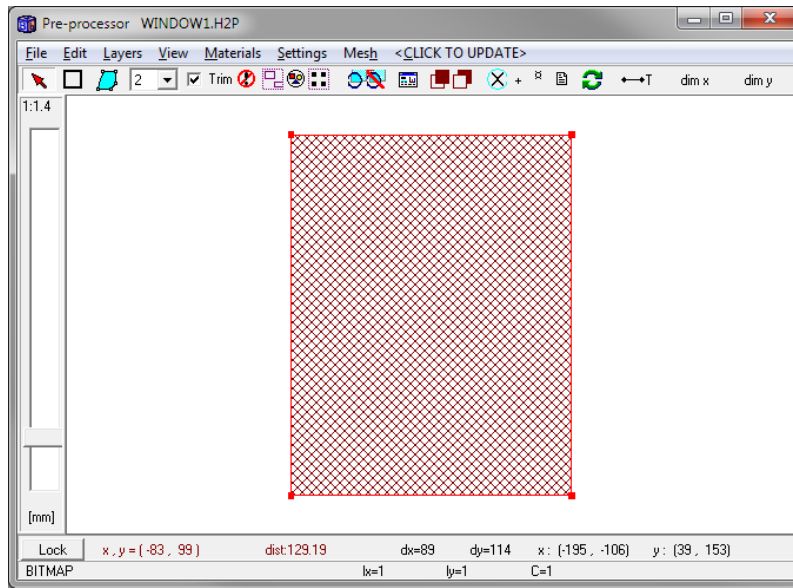
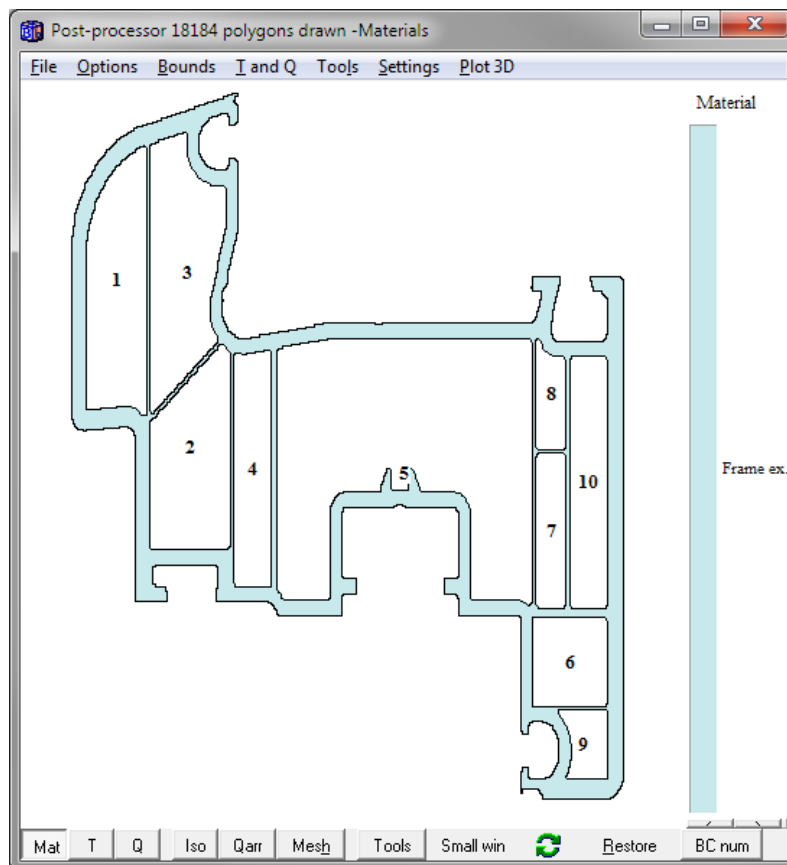


Figure 19: Part of the pvc frame (light blue color) filled with “Frame ex (c) PVC” chosen from the material list using the fill tool (left tool bar).

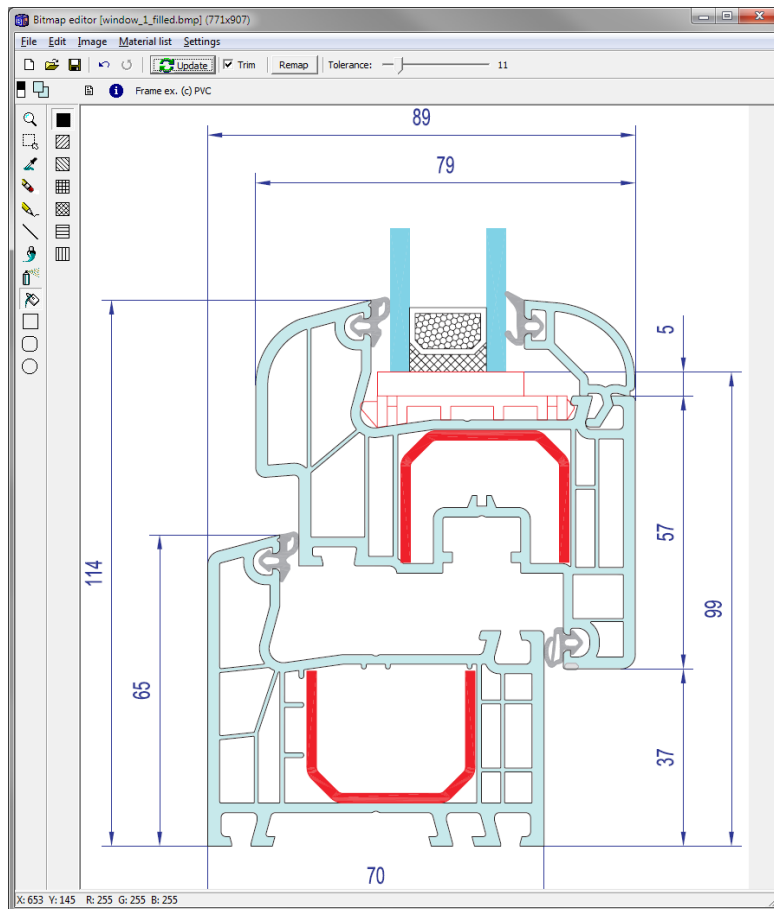
We now draw a mapping bitmap area in the pre-processor with a width of 89 mm and a height of 114 mm (we will exclude the glazing in this example):



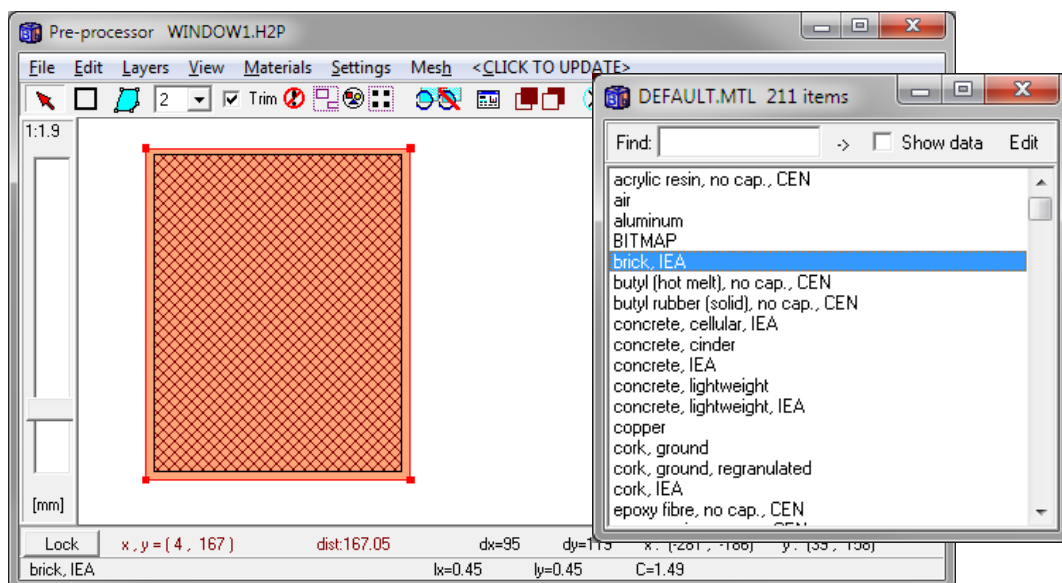
Updating the geometry (click "Update" in the pre-processor or in the bitmap editor) will generate the input we have so far. Note that HEAT2 recognizes ten frame cavities (labeled 1-10) according to ISO 10077-2:



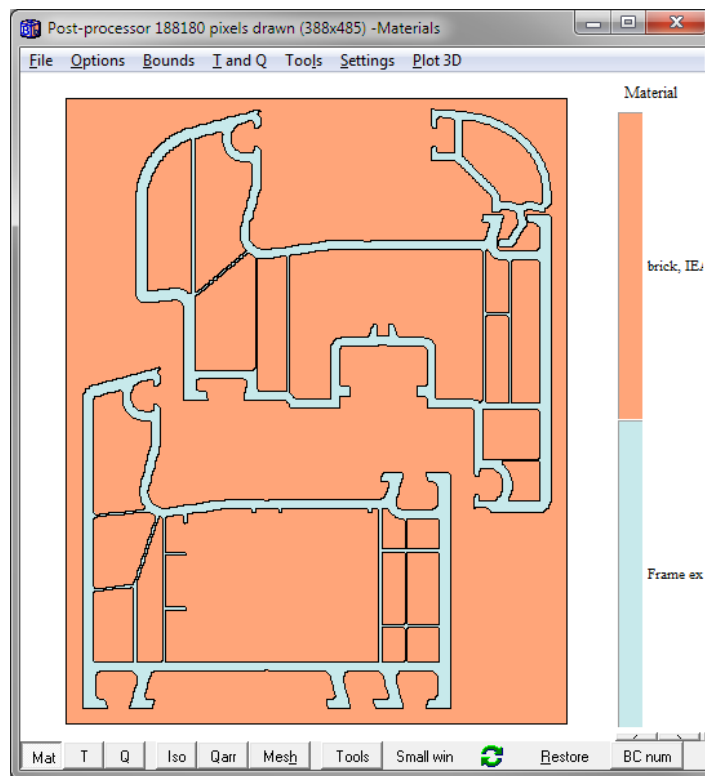
The two other areas of the PVC-frame are filled in the same way:



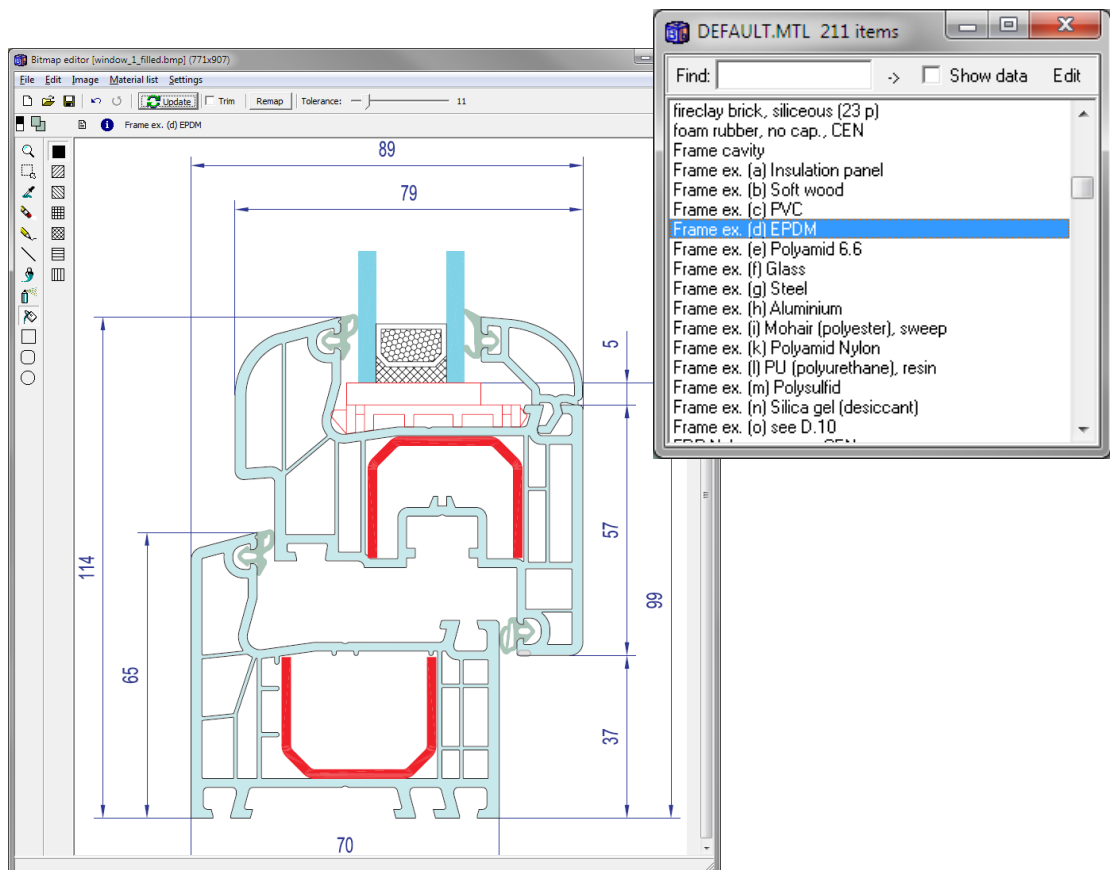
If we now try to update the geometry HEAT2 will complain due to the non-continuous area since the two frame parts are not connected to each other. One tip is to make a temporary material rectangle “below” the bitmap rectangle in the pre-processor (see yellow “brick” rectangle which later will be deleted):

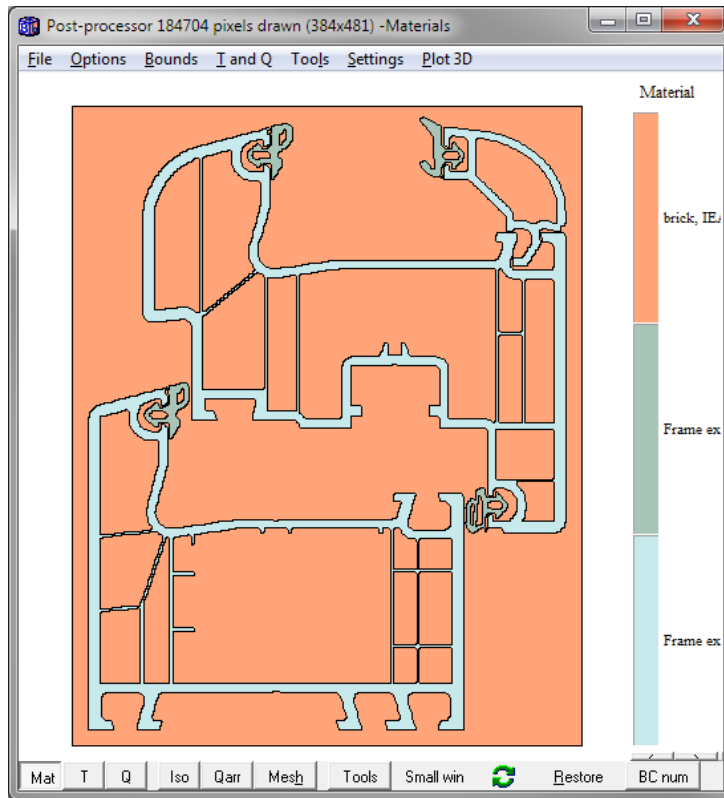


The result we have so far is shown below in the post-processor:

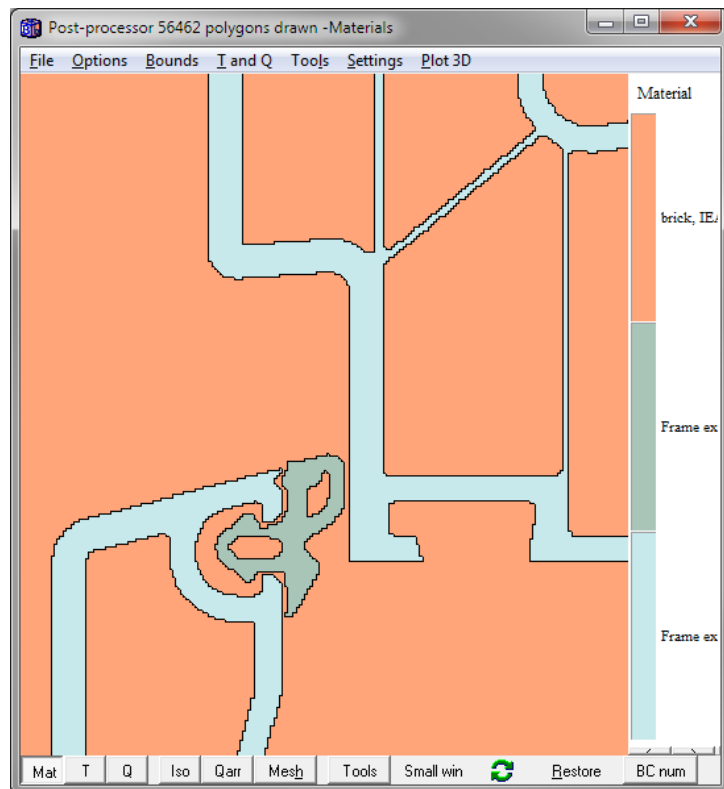


Next step is to fill the four EPDM lists:

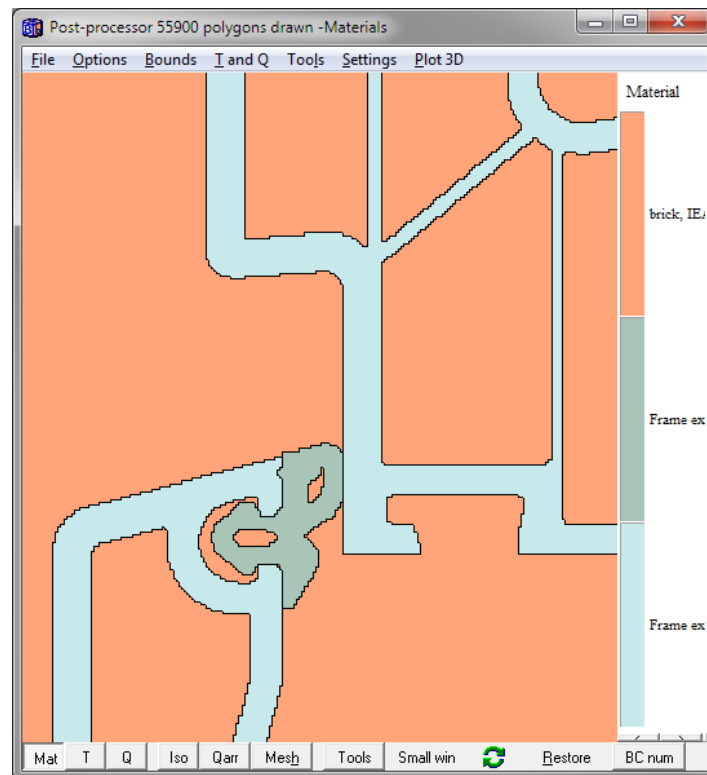




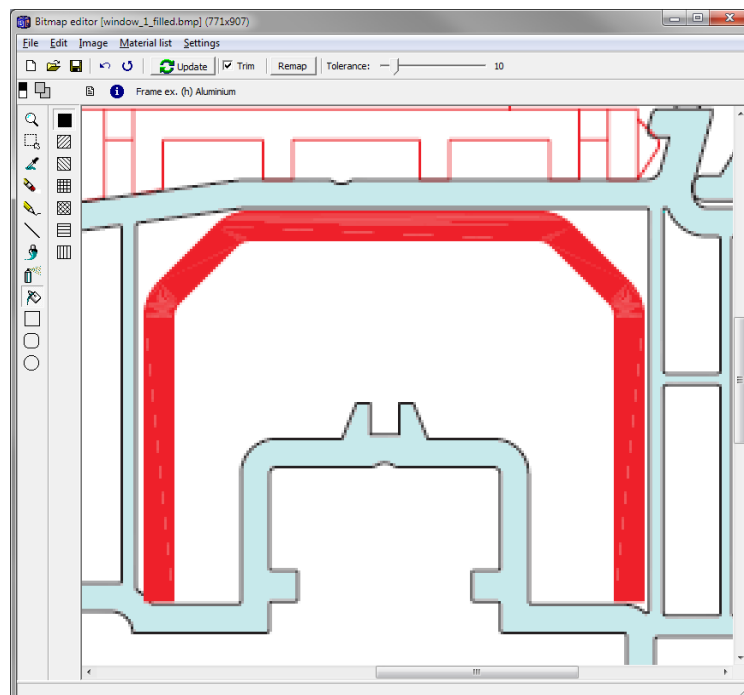
The filling tool fills an outlined area but not the line itself. We see e.g. a gap between the PVC frame and the EPDM-list:



There is a “Trim” checkbox in the bitmap editor (upper tool-bar). Check this to expand the filled area. The PVC and EPDM will now be in contact:

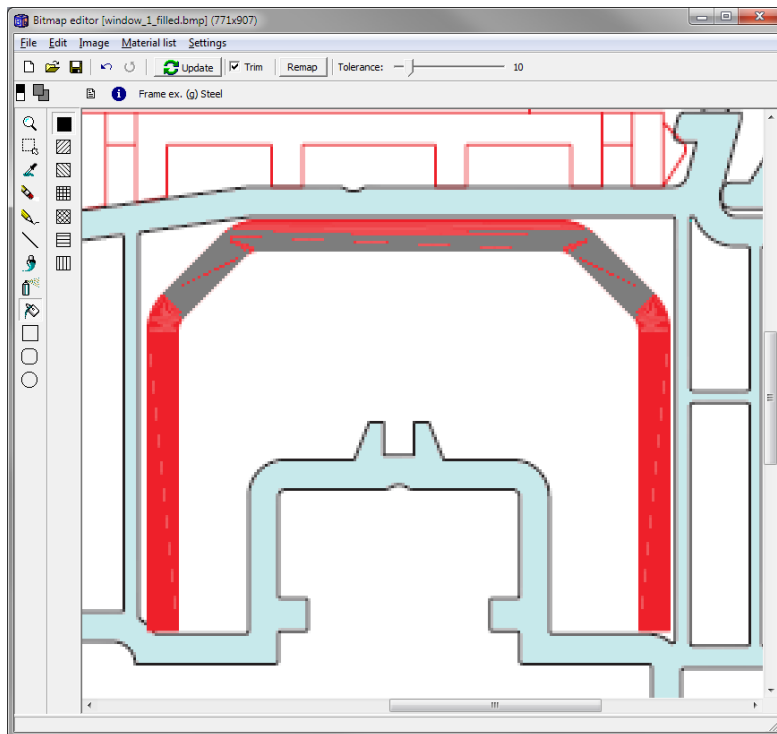


Next step is to fill the two steel bars (shown in red). Here is the first one:

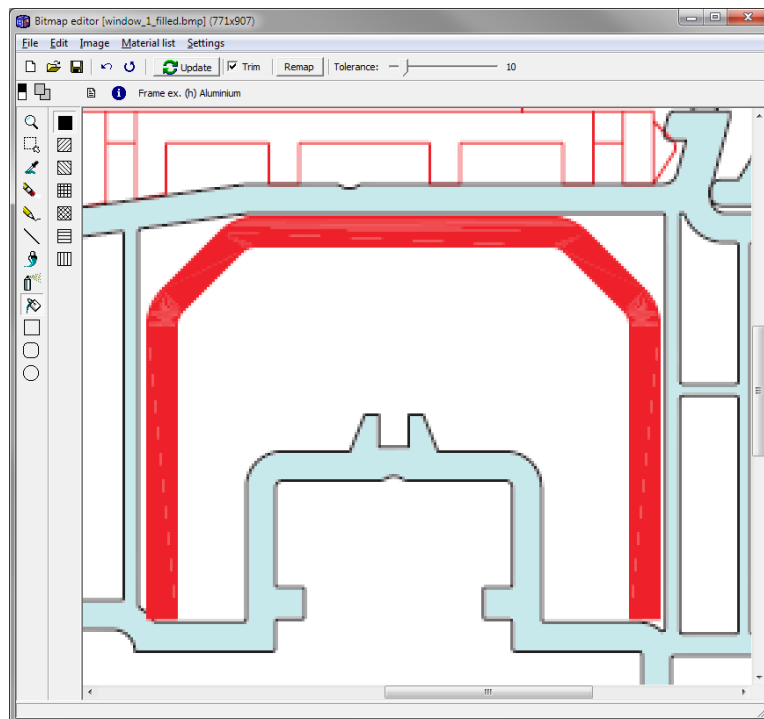




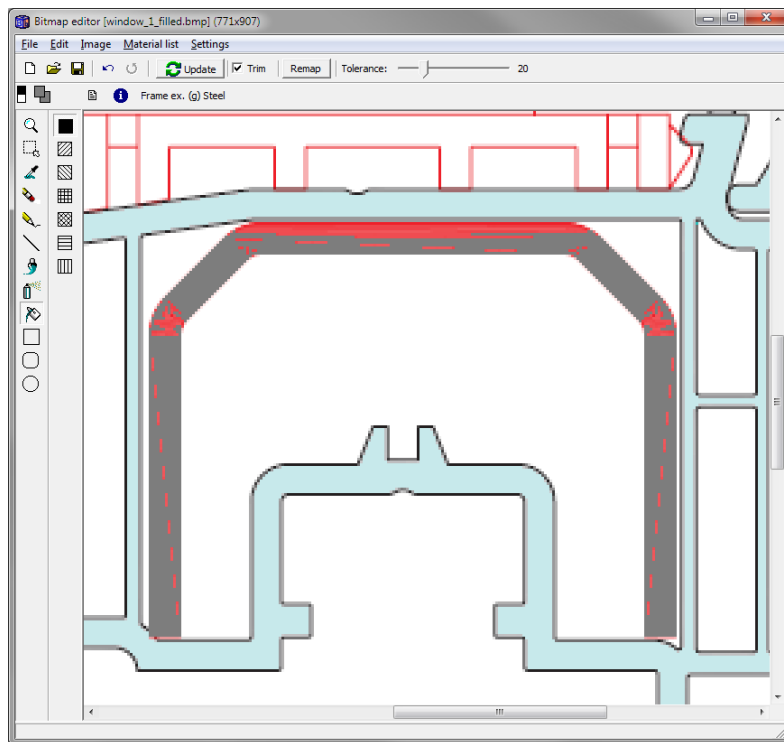
The result of filling the area with the steel (grey color) is shown in the post-processor:



Only a small area of the steel has been filled in this case. The problem is that the steel part has different color shades of red and white and only a small area is recognised and filled. To solve this, press the undo-button (there are by the way 10 levels of undo/redo) on the top tool-bar row to get the unchanged image back:

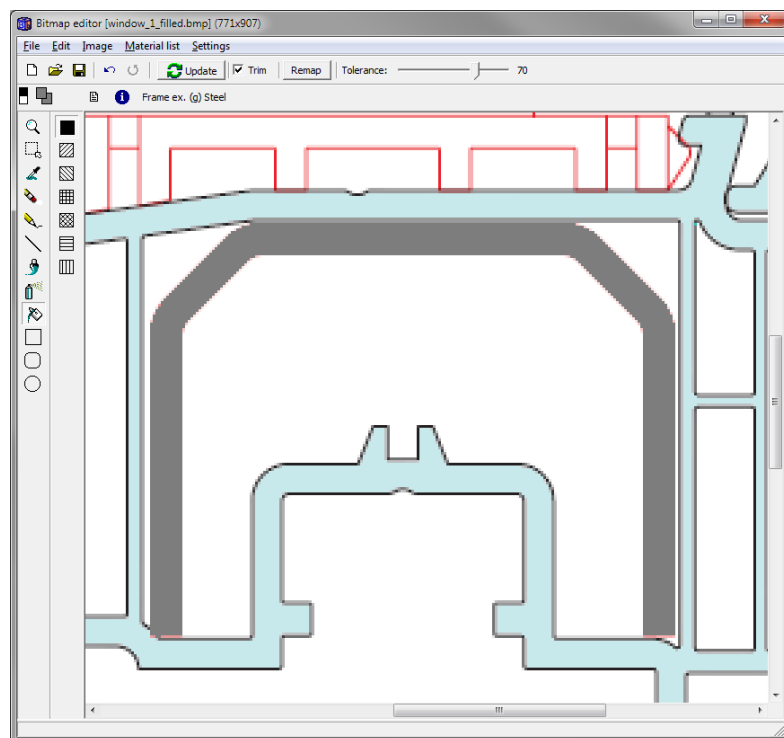


We now set the tolerance (upper tool-bar) in order to improve recognition. Changing the tolerance from the default value 10 to 20 on a scale 0-100 will produce this:



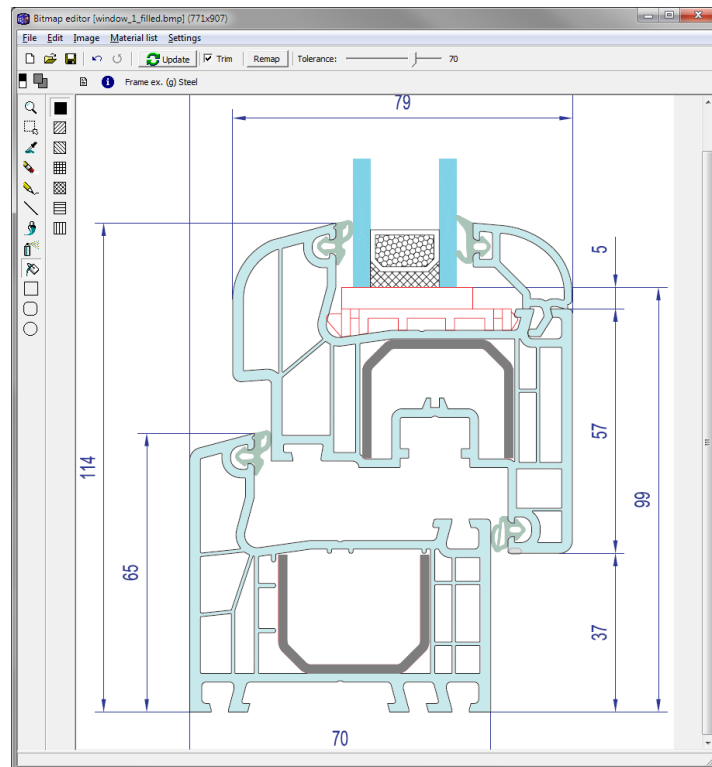
Tolerance=20

The result is better but not good enough so we press undo again and try a tolerance level of e.g. 70 which seems to produce a nice result:

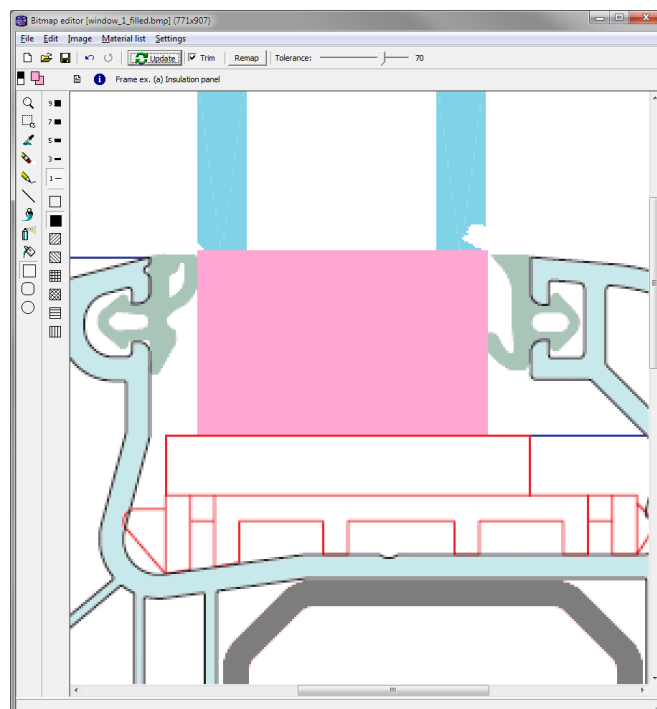


Tolerance=70

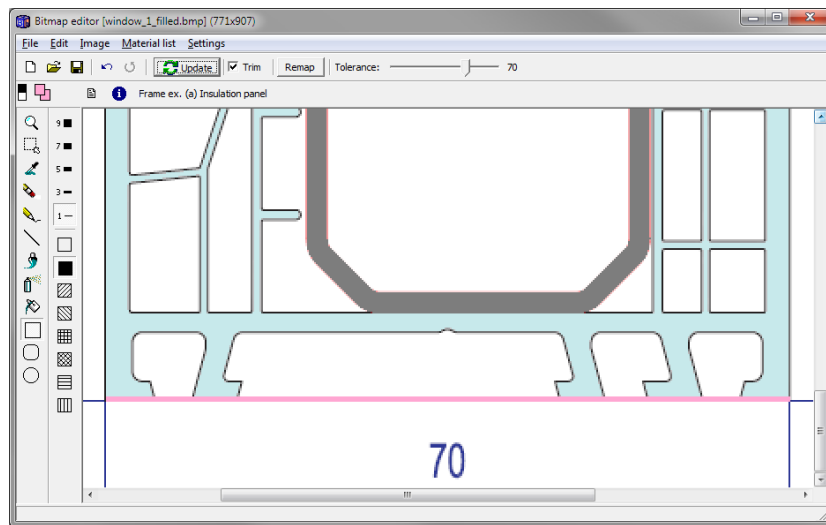
Applying the same tolerance fill on the other steel part results in this:



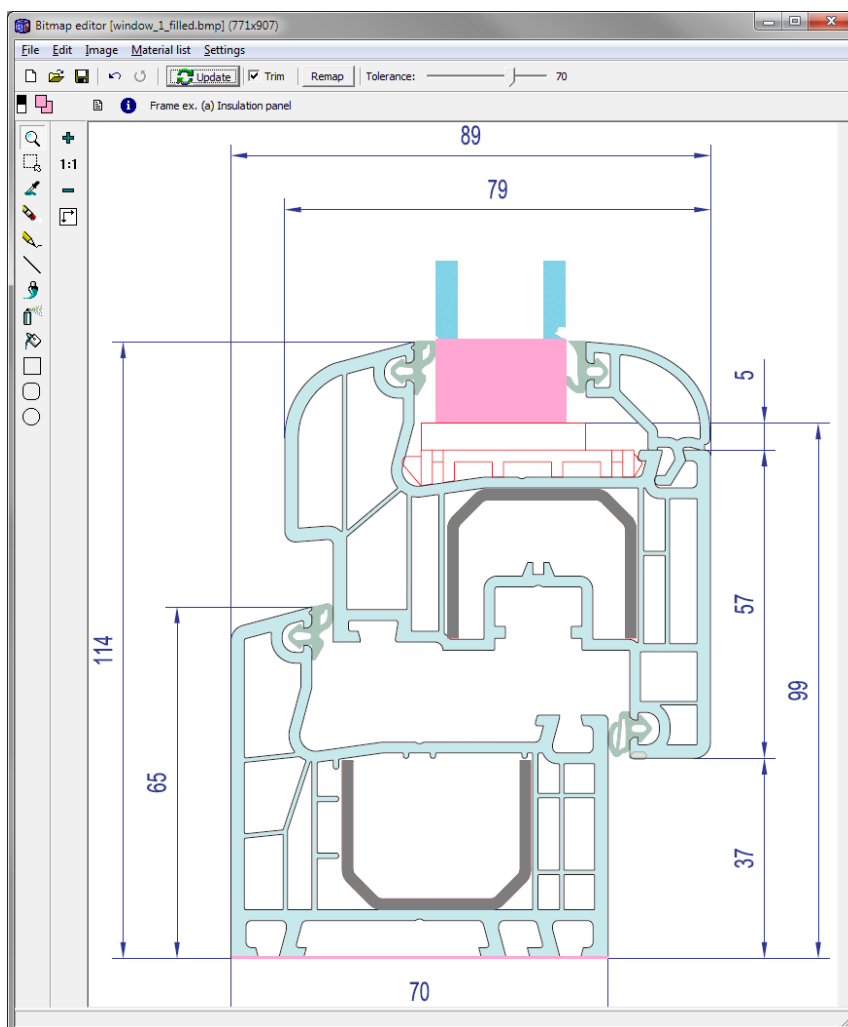
In this example we replace the glazing part with an insulation panel (we just want to evaluate the thermal performance of the frame part itself). To do this, just select a material (e.g. "Frame ex. (a) Insulation panel") and draw a rectangle using the rectangle tool in the bitmap editor. Here we also trim the EPDM using the erase tool:



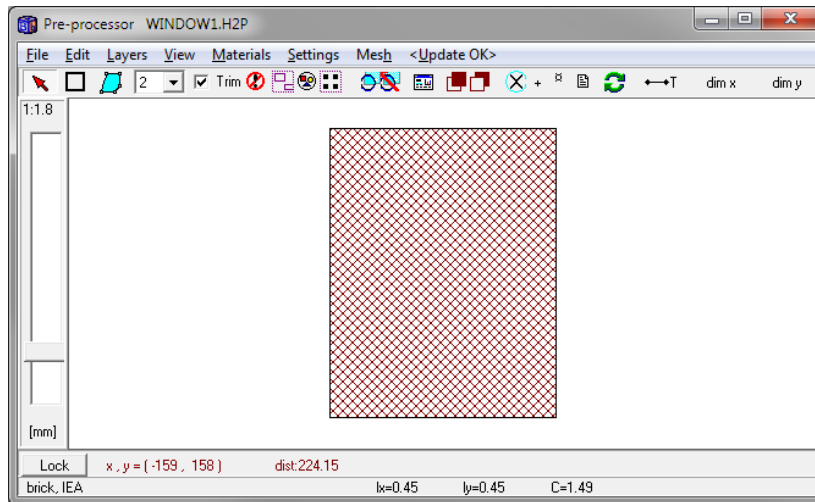
We also draw a small insulation strip at the bottom in order to later be able to define an adiabatic boundary condition there:



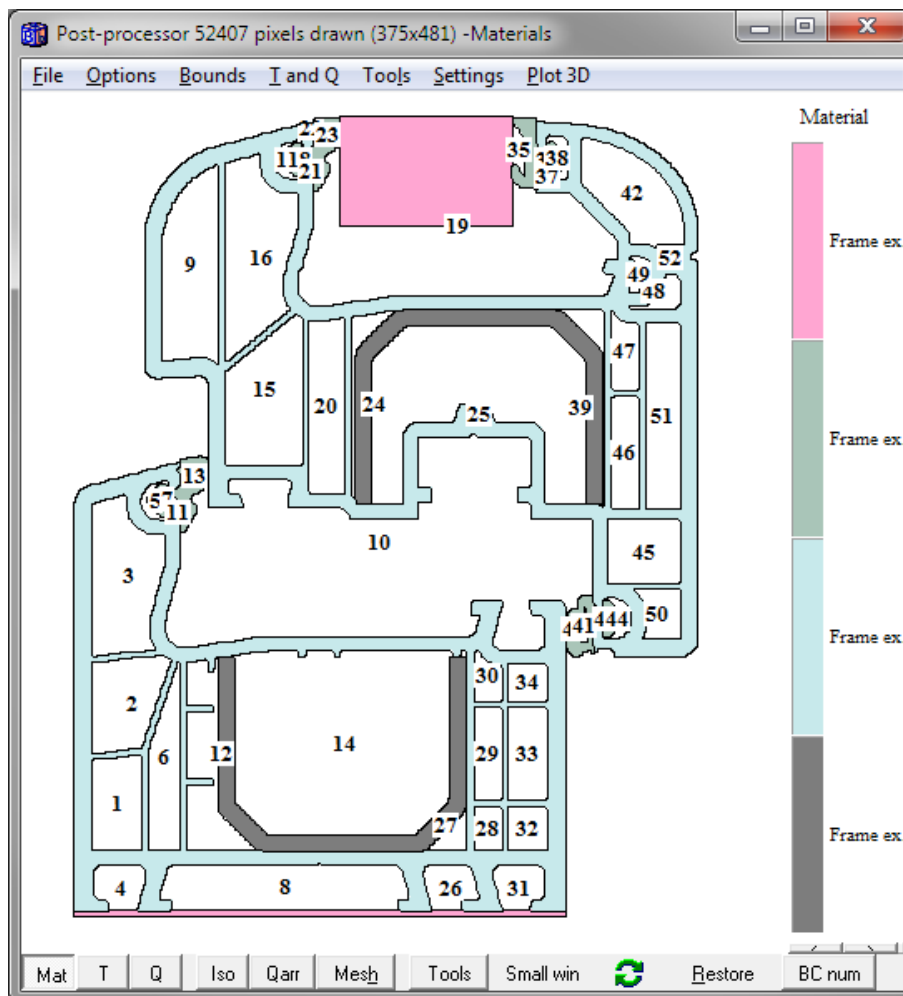
The final modified bitmap looks like this:



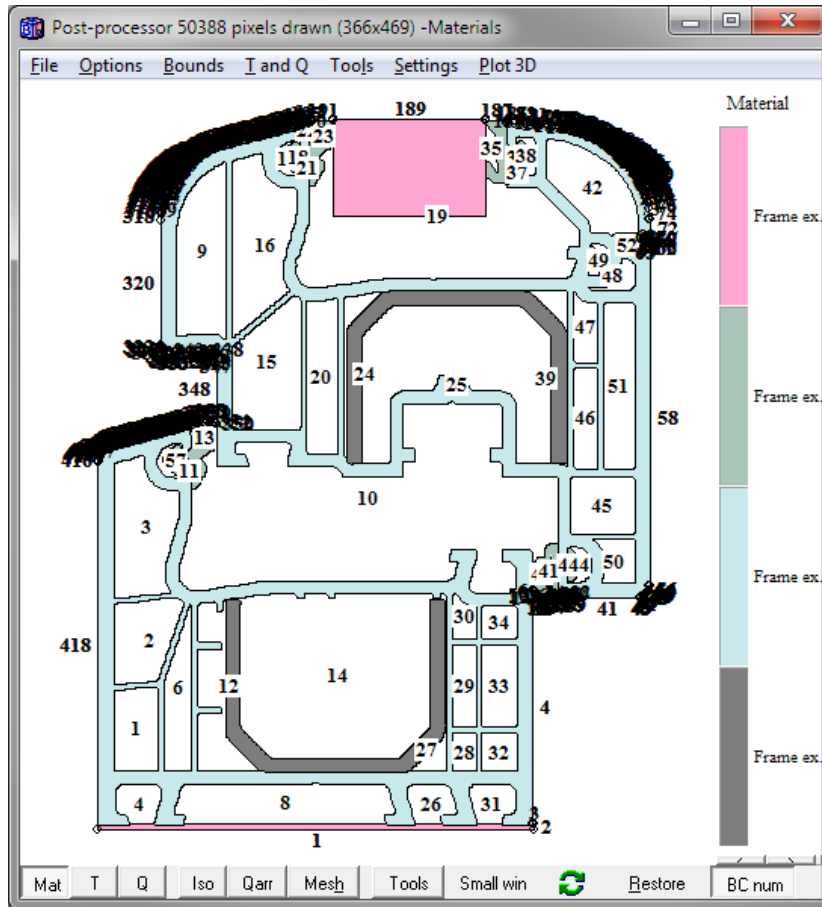
Since the area now is continuous we delete the temporary rectangle so that only the mapping bitmap rectangle is left:



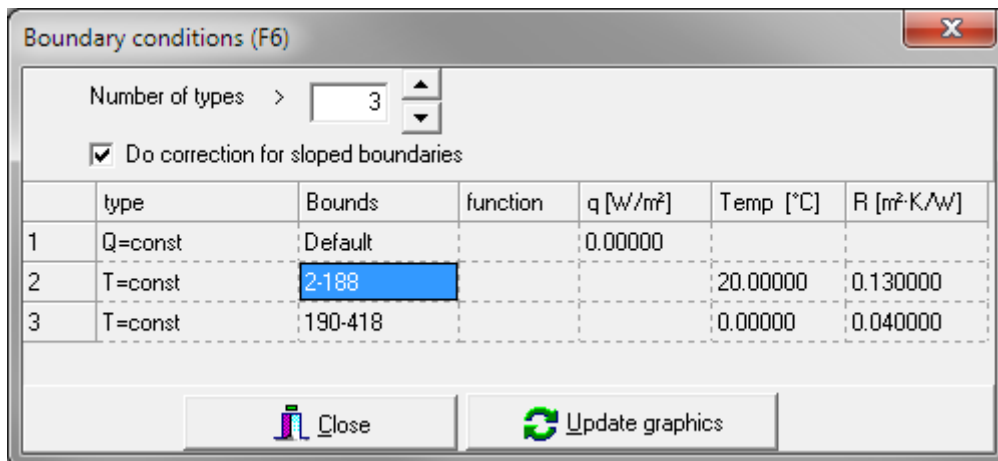
The updated geometry is shown in the post-processor. There are 52 recognized frame cavities:



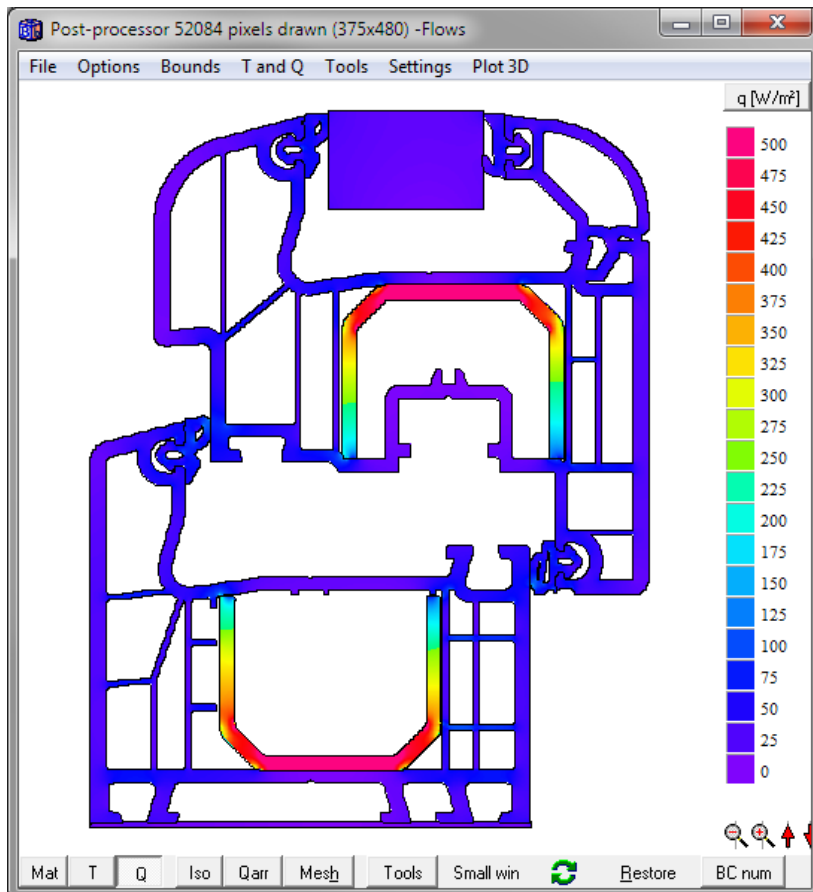
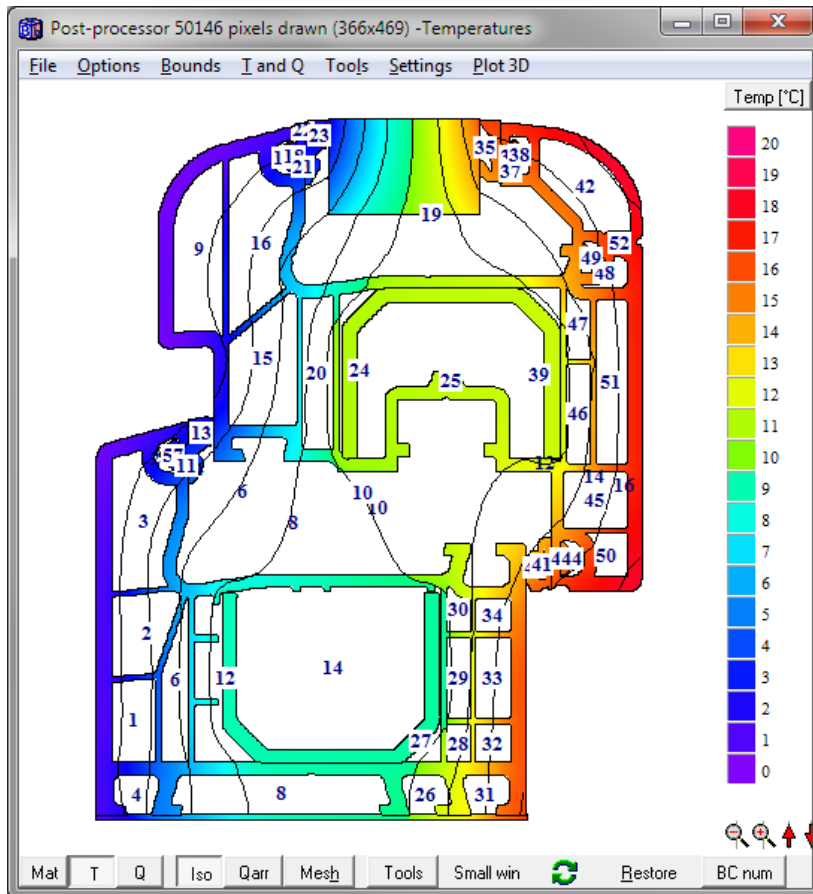
Now, press “BC num” to show the boundary numbers:



We now specify the boundary conditions and assume the warm side to the right:



Press F9 to start steady-state calculation. Below shows calculated temperatures and heat flows:



### 3.4 How bitmap colors are linked to the material list in HEAT2

Whenever an object is drawn in the bitmap editor using a material its RGB (red, green, blue) value together with the material name is saved to a special list called “bitmap material list” (BML-list). Figure 20 shows the list for the window frame example (press material list or the “Info”-button in the bitmap editor).

Note that HEAT2 also recognized that a color with RGB=196,196,196 existed in the window frame bmp-file that coincidentally matched a material in HEAT2 with name “Frame ex. (h) Aluminium” with the same RGB-values. HEAT2 will however not use this (the color will not be matched to a material upon updating the geometry) unless the box “use” is checked.

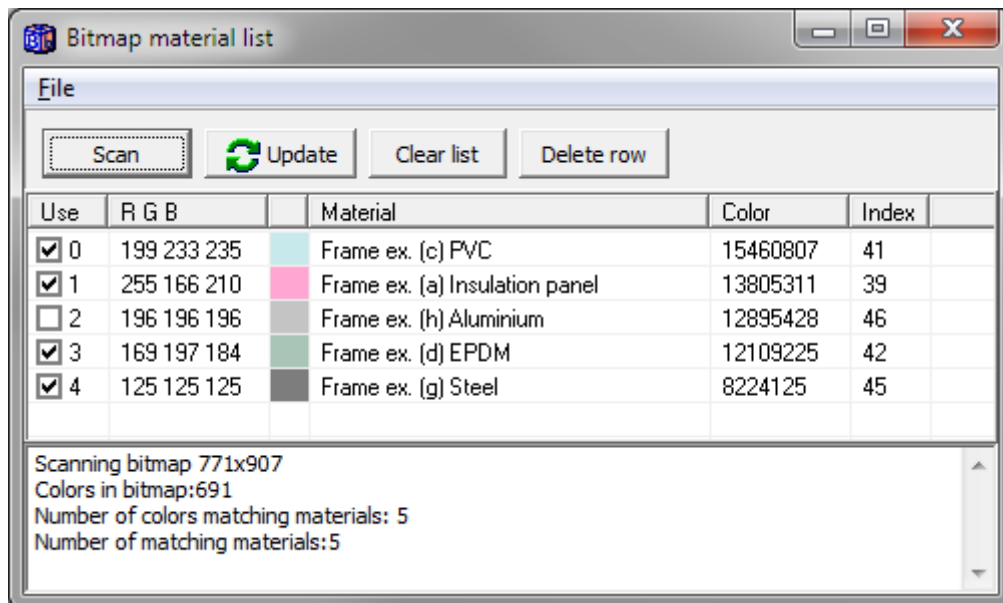


Figure 20: The “bitmap material list” (BML-list) matches the RGB colors to material names.

When a bitmap (eg. Mybitmap1.bmp) is saved by HEAT2, the BML-list will also be saved (as MyBitmap1.bml). It is an ordinary text file with material name, RGB-values, and if it should be used for matching or not (1 or 0):

```
Frame ex. (c) PVC  
199 233 235 1  
Frame ex. (a) Insulation panel  
255 166 210 1  
Frame ex. (h) Aluminium  
196 196 196 0  
Frame ex. (d) EPDM  
169 197 184 1  
Frame ex. (g) Steel  
125 125 125 1
```

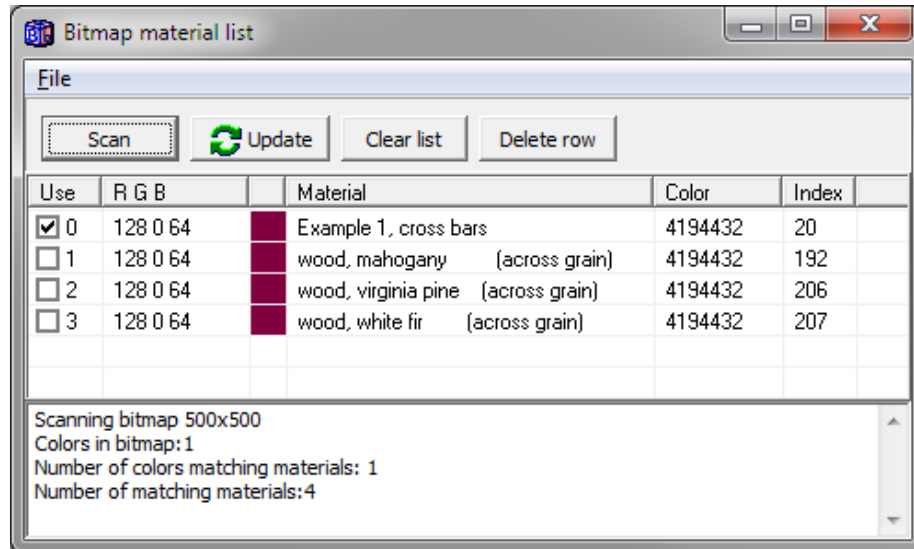
Whenever a bitmap is opened by HEAT2, the corresponding BML-file will be opened (if this exists) and HEAT2 knows what and if RGB-values should be matched with existing HEAT2 materials. If the bml-file is missing you can easily recreate the BML-list by pressing “Scan” and fill in the necessary “use” check boxes.

It is possible to draw the geometry in another drawing program or CAD-program with pre-defined colors (RGB-values) that will match the specific colors in HEAT2. Just open the bmp or jpeg file in HEAT2 and press “Scan”. HEAT2 will then identify materials and generate the necessary input (geometry, numerical mesh, etc.) for the thermal analysis. See the example in Section 3.10.

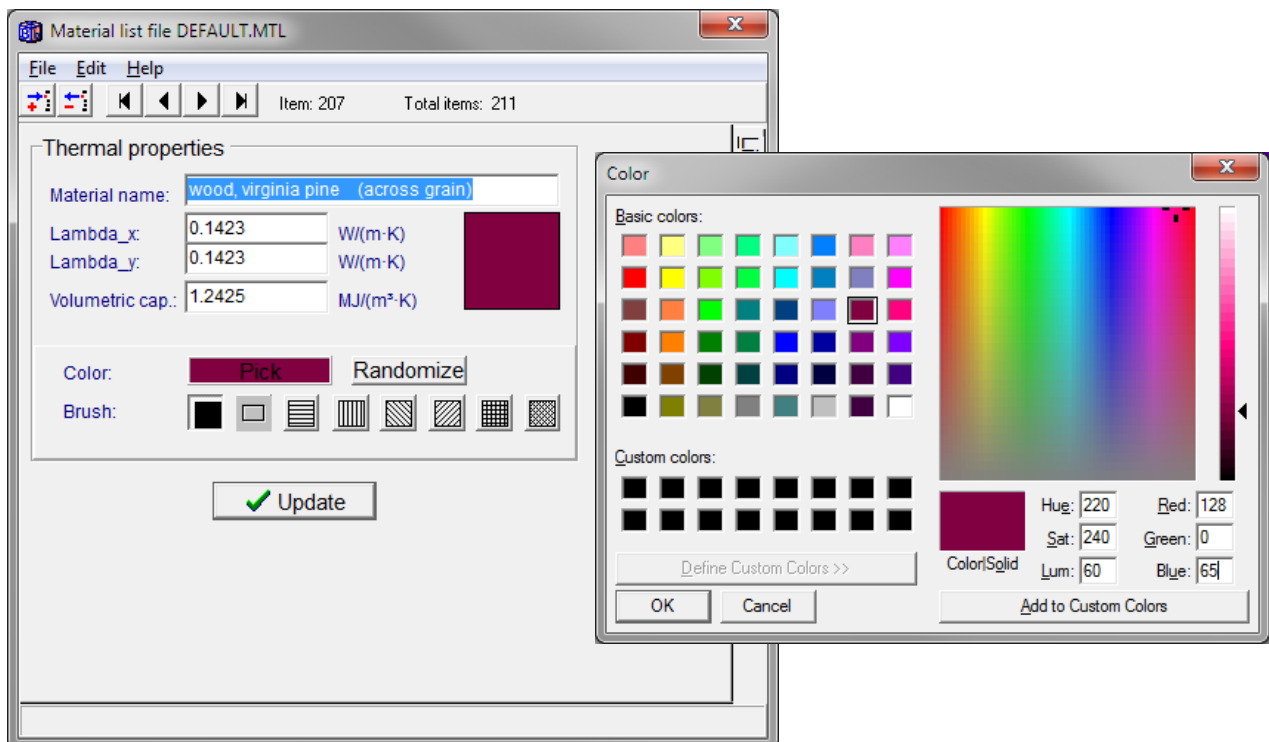


Be sure that you use the correct material file (\*.mtl) when scanning the bmp-file. Note that you can use different material files (\*.mtl) mapping the same RGB color to different materials.

Note that one color RGB-value may match several materials in HEAT2. E.g. RGB=(128,0,64) will map to the following materials in “Default.mtl”:

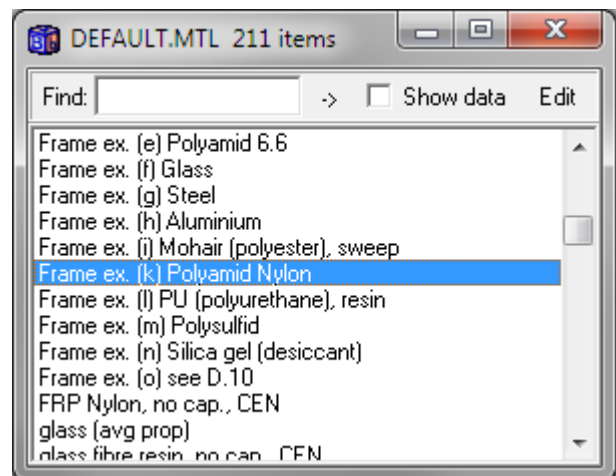
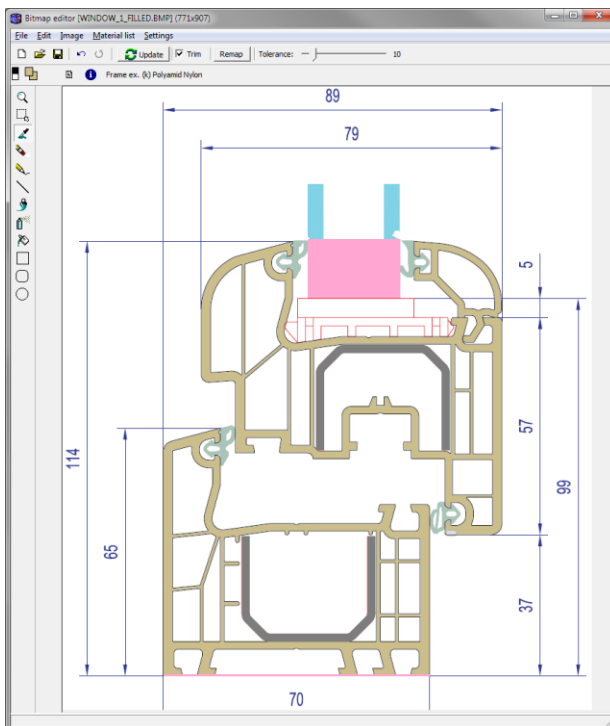
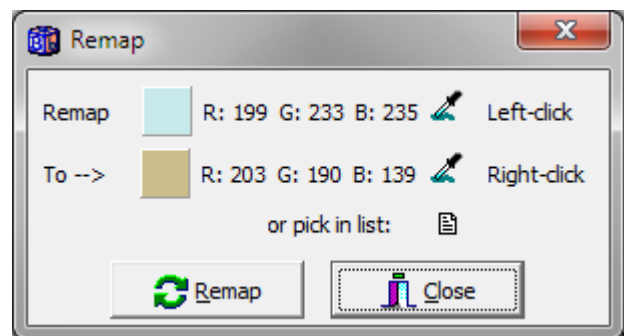
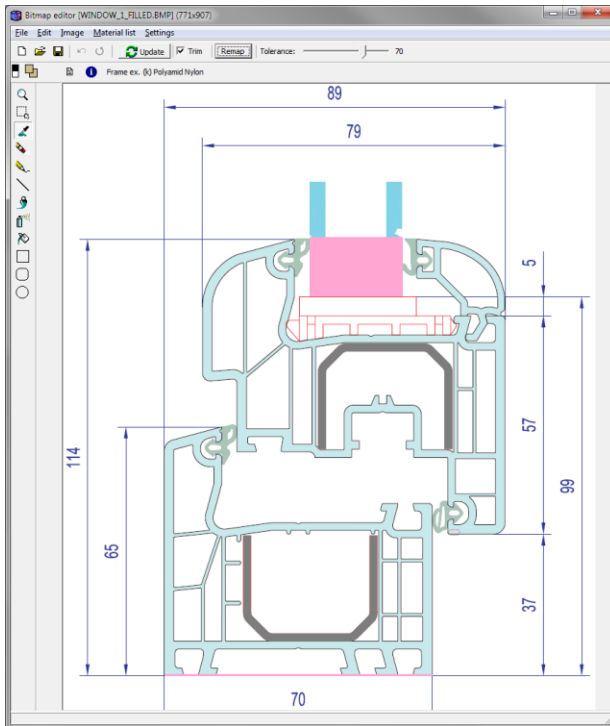


Check “Use” to match the material you want. If you need to use more two (or more) materials that coincidentally has the same color, you must change the color (i.e. change at least one of the RGB-values) to a non-conflicting one. E.g. if you want to use both mahogany and virginia pine you can change e.g. the B-value (blue) for virginia pine from 64 to e.g. 65 (or of course to any other color) in the material color editor and then use this color in the bitmap editor:



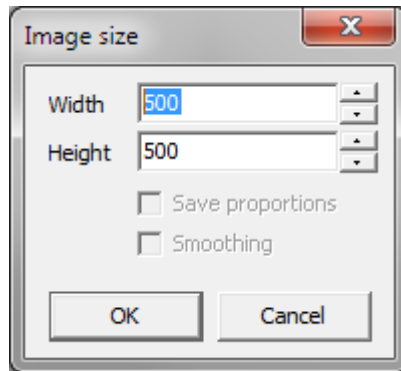
### 3.5 Remapping colors

There is a tool for remapping a color (material) to another. Assume we want to change the PVC material for our frame example. Open the “Remap” tool window (press the remap button). Select the “Dropper” tool (on the left tool bar in the bitmap editor). Place the dropper on the PVC and left-click the mouse. To choose the destination color, right-click the wanted material in the bitmap, or simply pick the material in the list as below (here we remap “PVC” to “Polyamid Nylon”). Press “Remap”. Note that you can set the tolerance for the old color that will be remapped (a higher tolerance will remap “nearby” colors of the original color to the destination color).



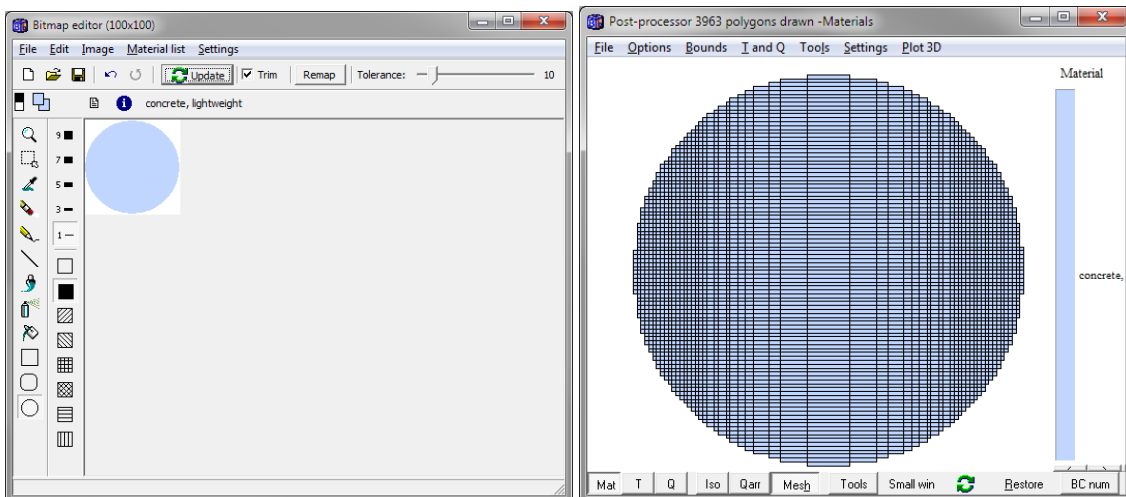
### 3.6 Creating a new bitmap

To create a new bitmap, select “New” in the bitmap editor and give the width and height. The maximum width and height is 1000.

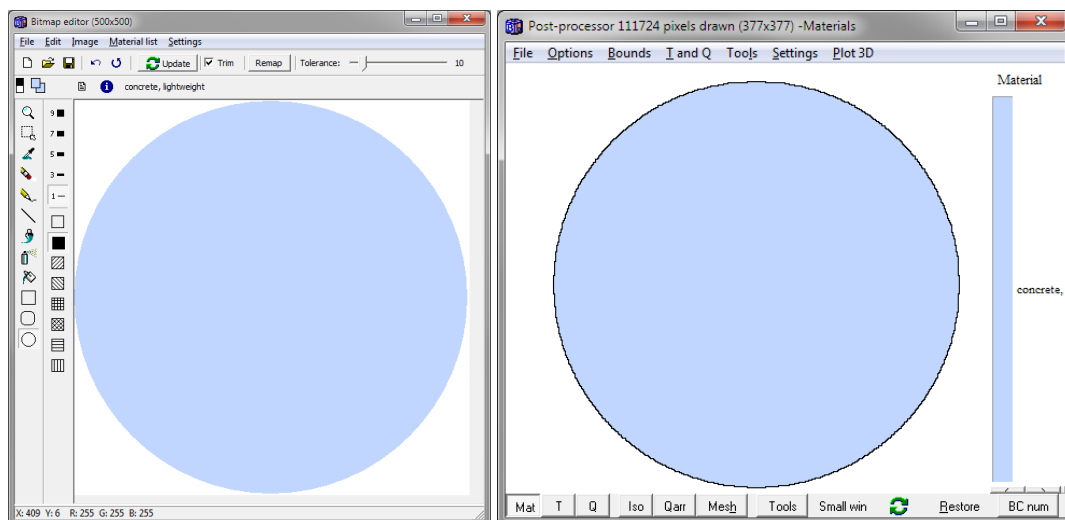


### 3.7 Bitmaps and numerical mesh

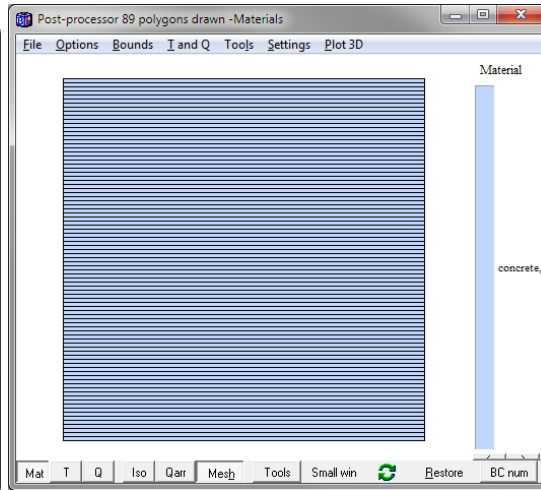
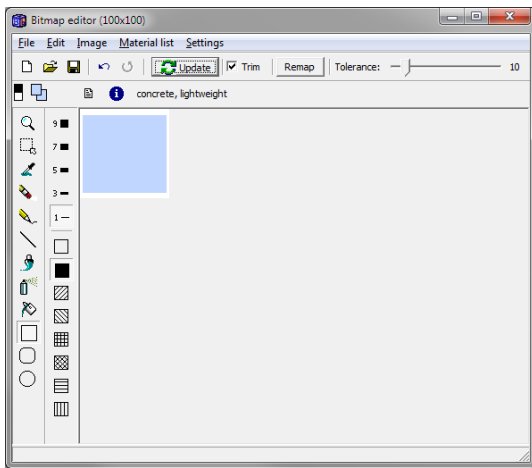
Large bitmaps will in general require more numerical cells, especially if the bitmap contains many areas with sloping boundaries. As an example consider the figures below:



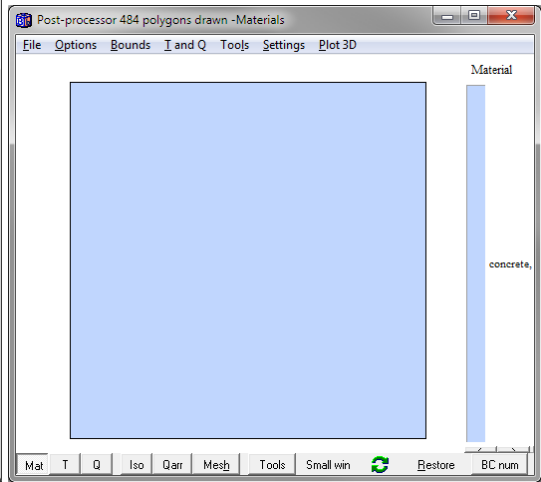
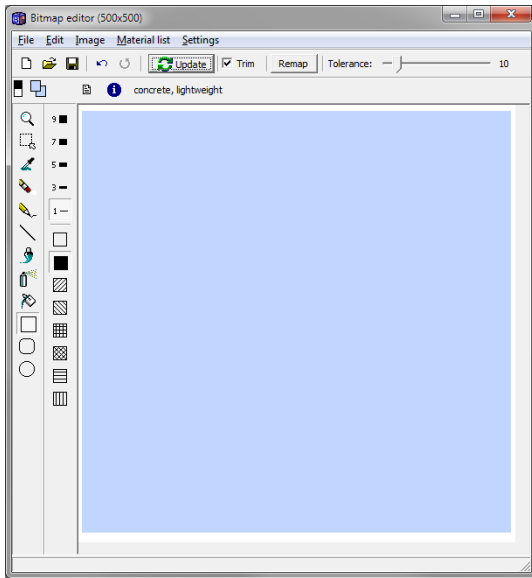
This circle drawn in a bitmap of 100x100 pixels will use a minimum of about 4,000 nodes



This circle drawn in a bitmap of 500x500 pixels will use a minimum of about 100,000 nodes



This rectangle drawn in a bitmap of 100x100 pixels will use a minimum of about 100 nodes



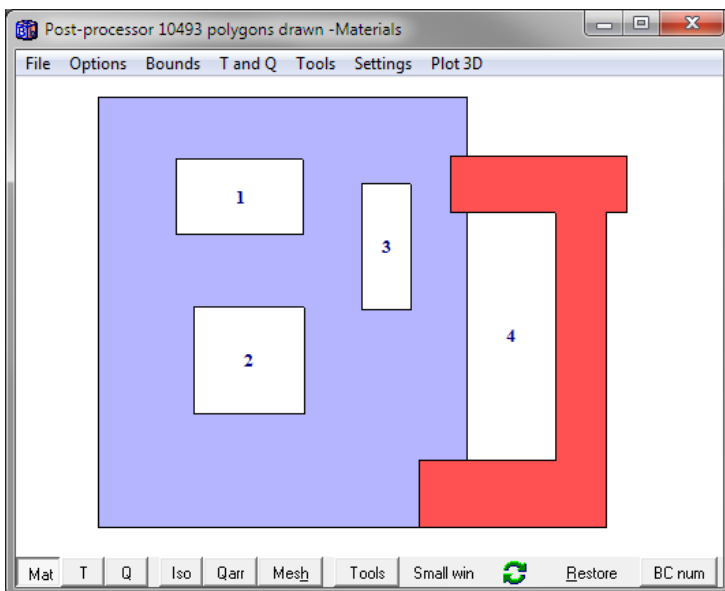
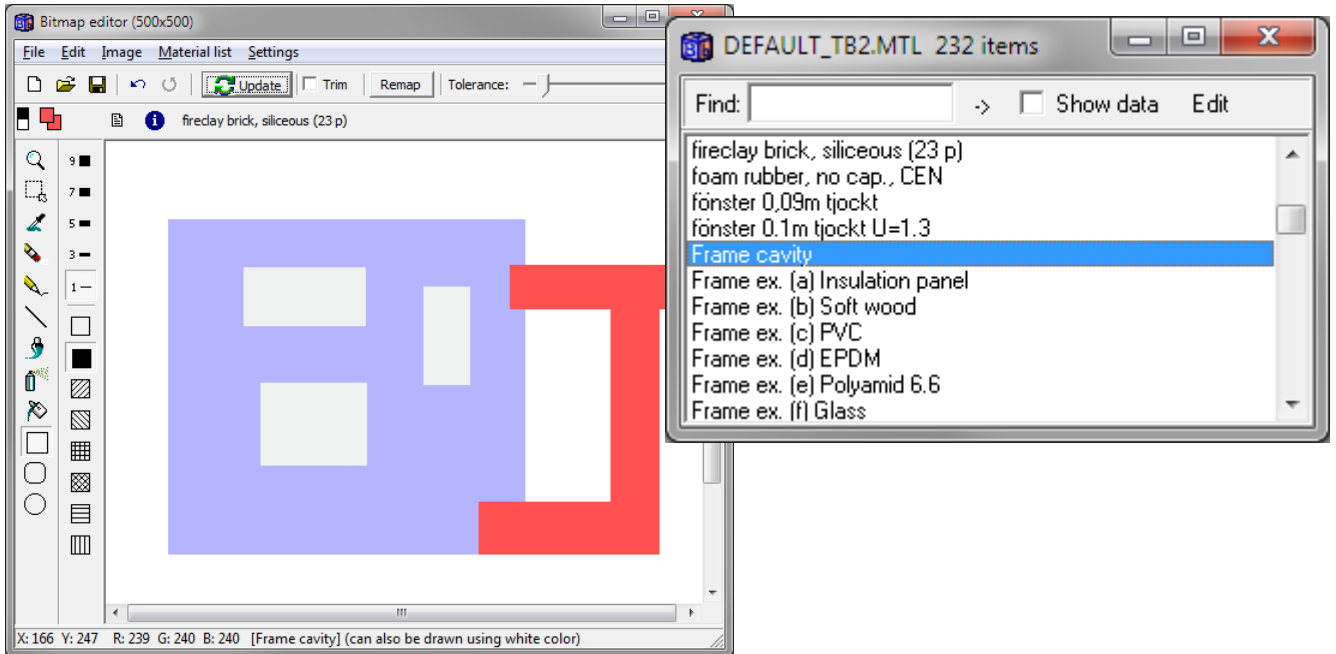
This rectangle drawn in a bitmap of 500x500 pixels will use a minimum of about 500 nodes

Note that the numerical mesh can be changed in the pre-processor mesh dialog to be even finer than the default generated.

### 3.8 Bitmaps and frame cavities

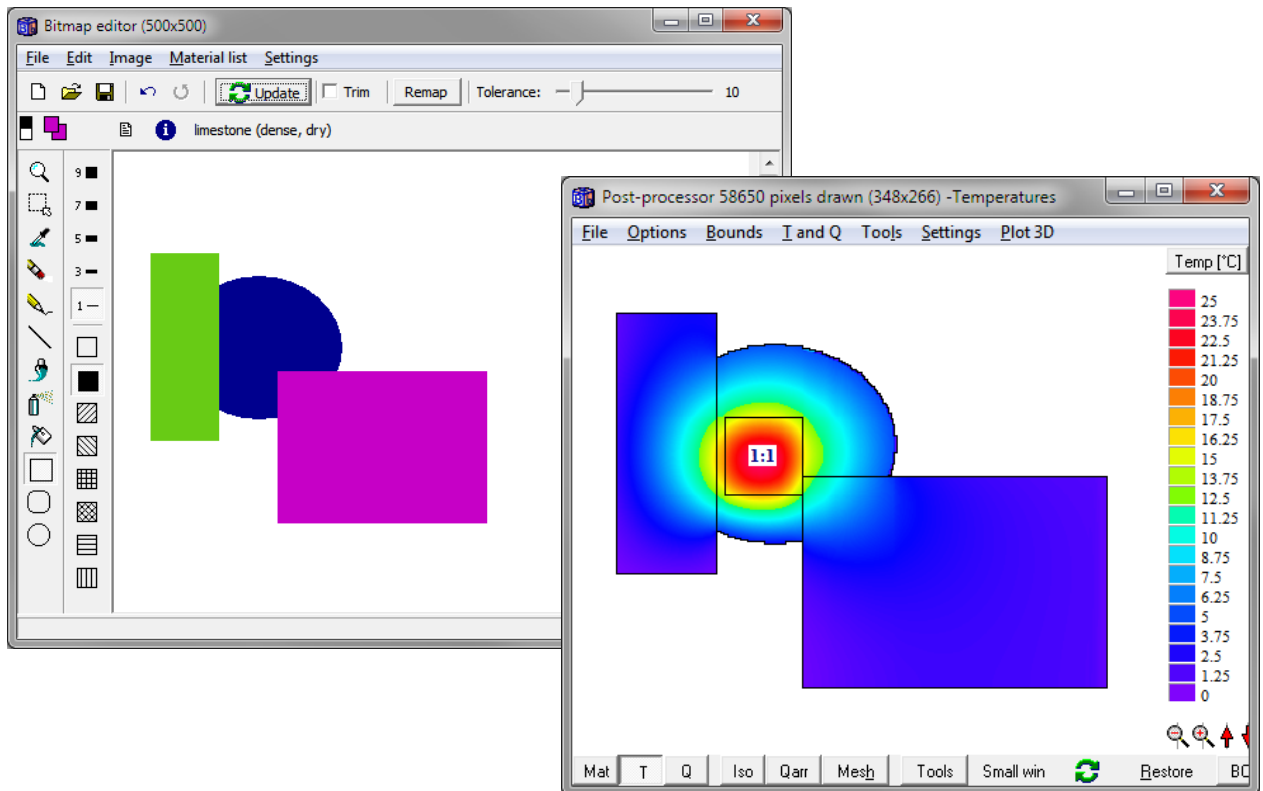
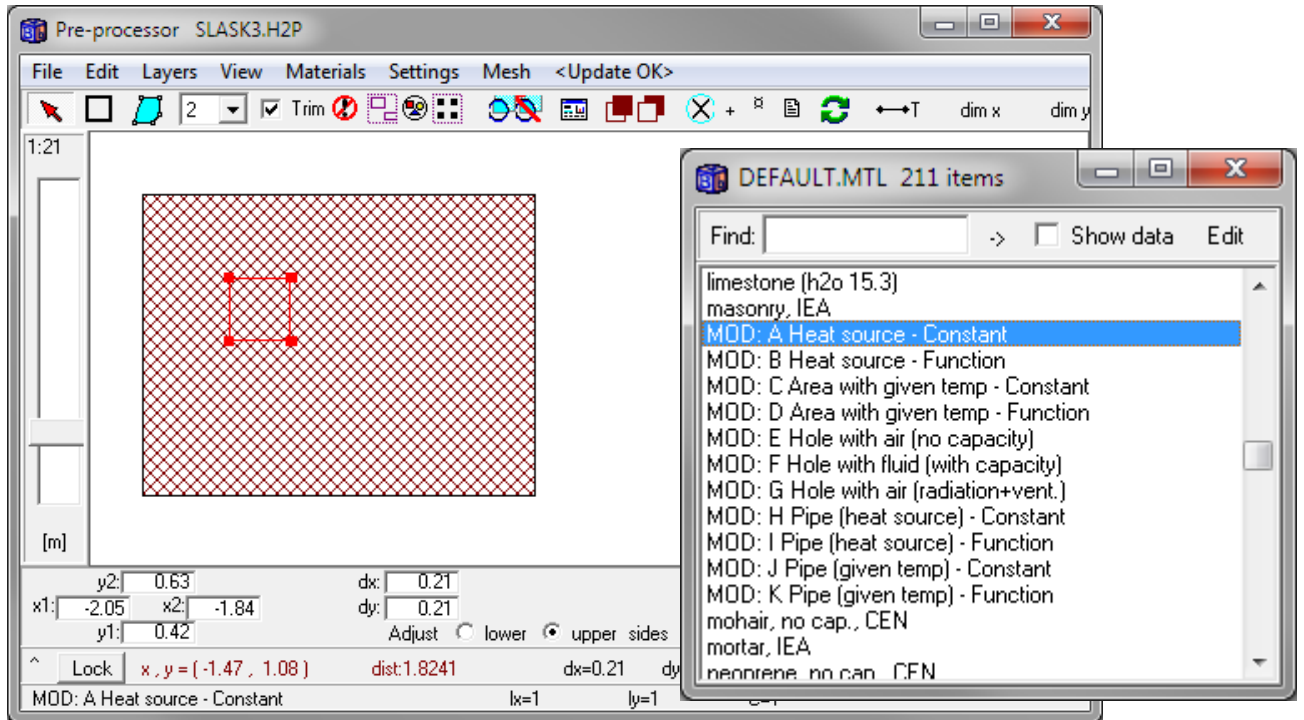
Frame cavities (according to ISO 10077-2) can be drawn in the same way as in the pre-processor:

- Using “Frame cavity” from the material list (e.g. cavities 1,2,3 below)
- Automatically identified in an enclosed space (e.g. cavity 4 below)



### 3.9 Bitmaps and modifications

Internal modifications (Mod A – Mod K in the list below) cannot be drawn in the bitmap editor. It is however possible to draw modifications in the pre-processor “on” the bitmap rectangle. In the example below, a heat source in the blue circle is defined.

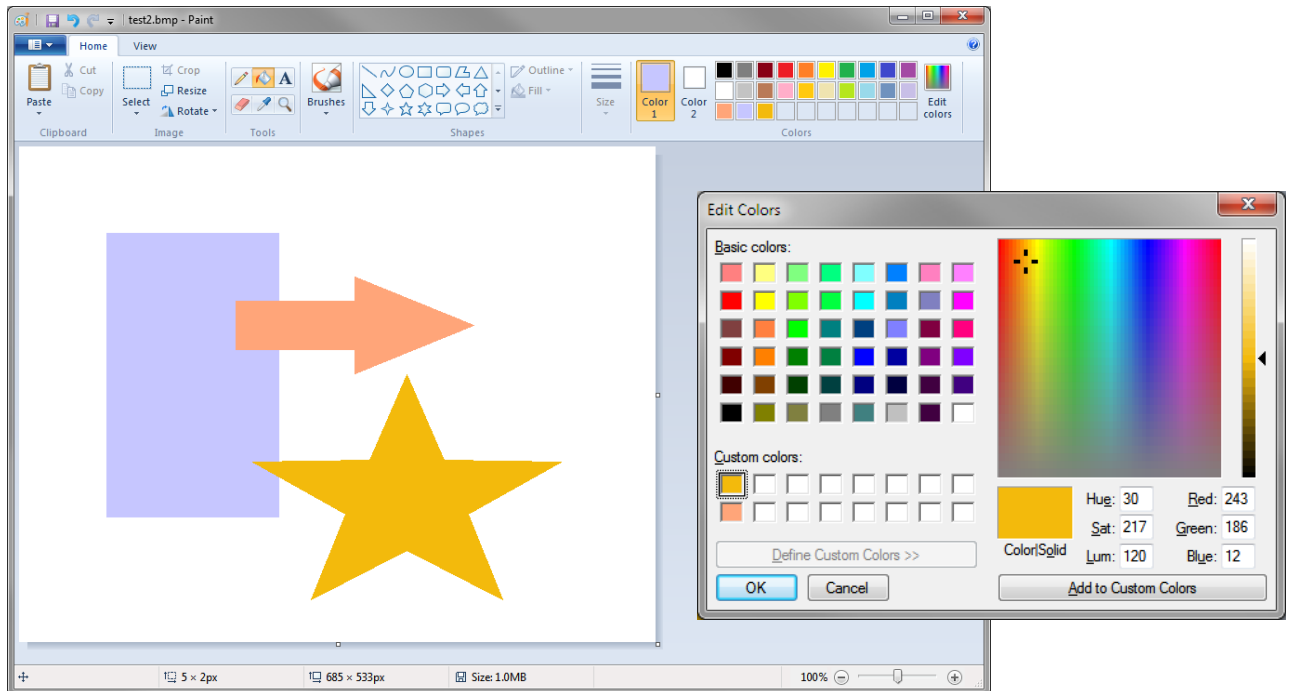


### 3.10 Using a cad-program or a drawing program as a pre-processor

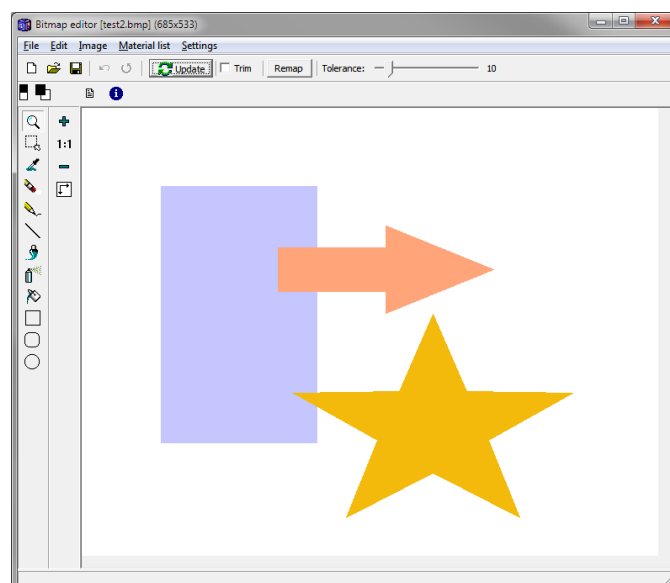
It is possible to draw the geometry in another drawing program or CAD-program with pre-defined colors (RGB-values) that match the specific colors in HEAT2. Just open the bmp or jpeg file in HEAT2 and press “Scan”. HEAT2 will then identify materials and generate the necessary input (geometry, numerical mesh, etc.) for the thermal analysis.

Below shows an example using the standard Paint program in Windows 7. First, we define three custom colors with (R,G,B)=(255,165,121), (198,198,255), and (243,186,12). This will match the materials brick, concrete, and glass-wool (open the material editor in HEAT2 to get the RGB-values).

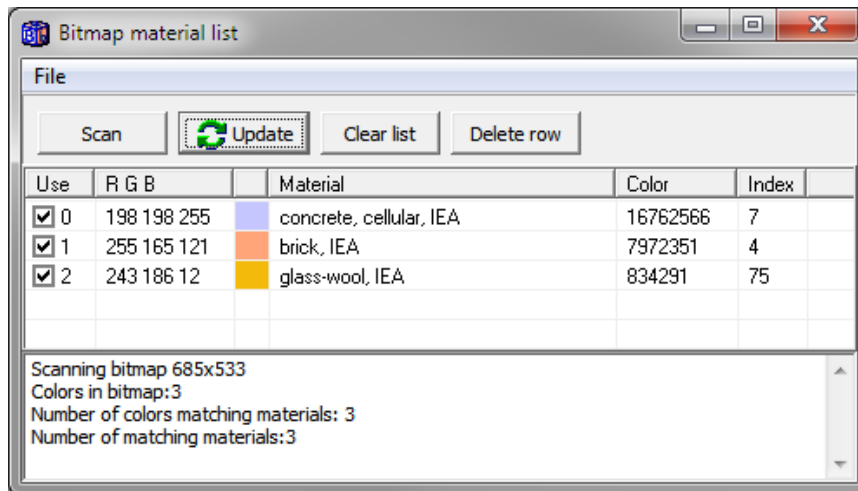
Three shapes are drawn and the bmp-filed is saved:



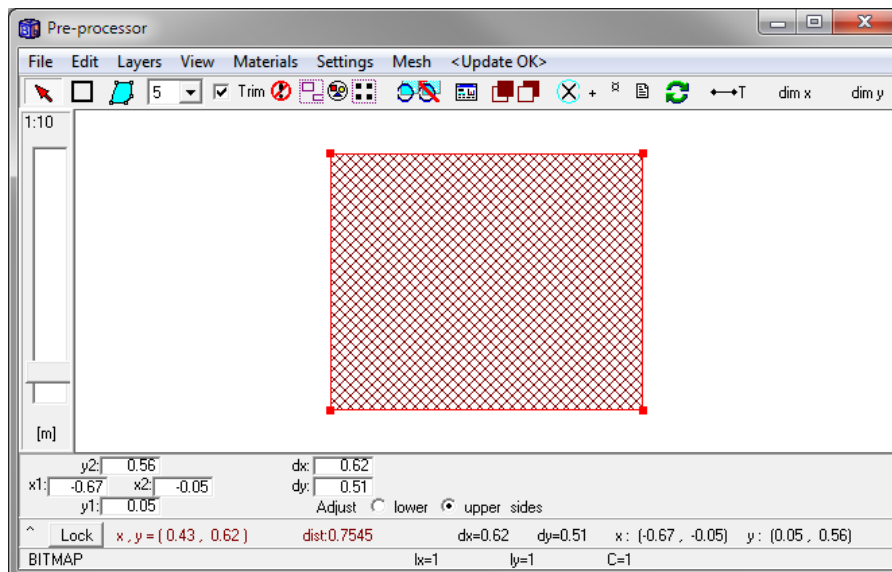
The bmp-filed is opened in HEAT2 bitmap editor:



Scan the bitmap to match the materials:

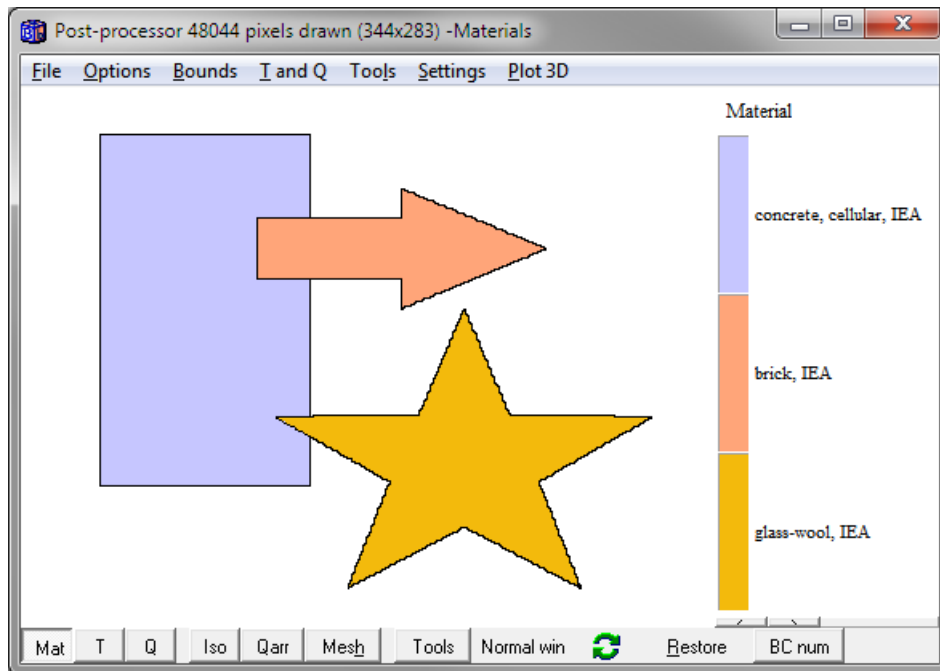


Place a “bitmap” rectangle in the pre-processor defining width and height:

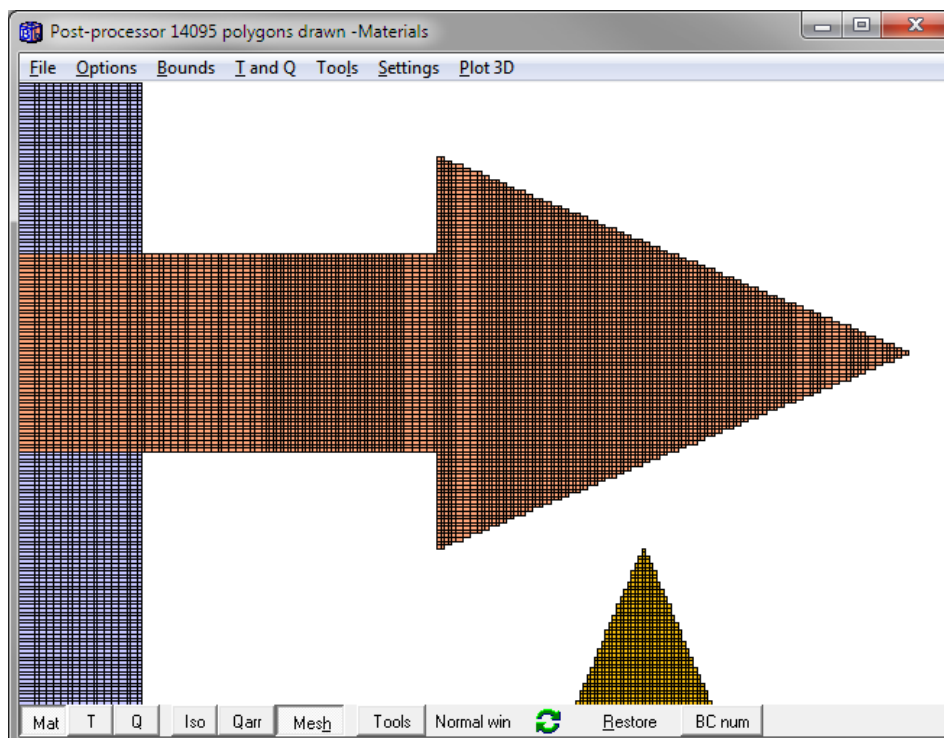




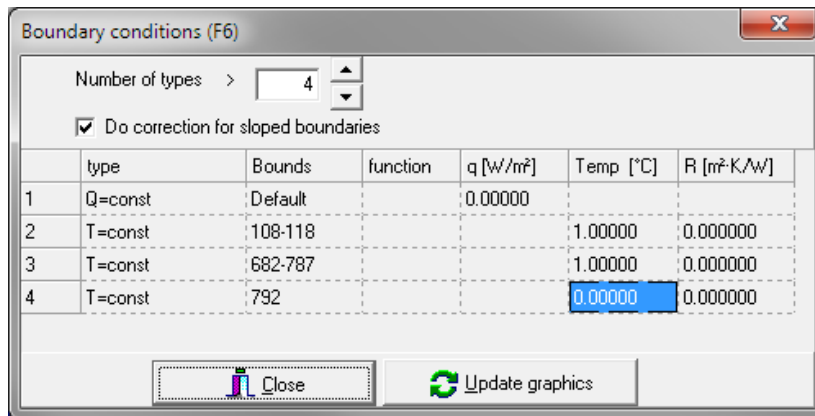
Press <Update> to see the generated geometry:



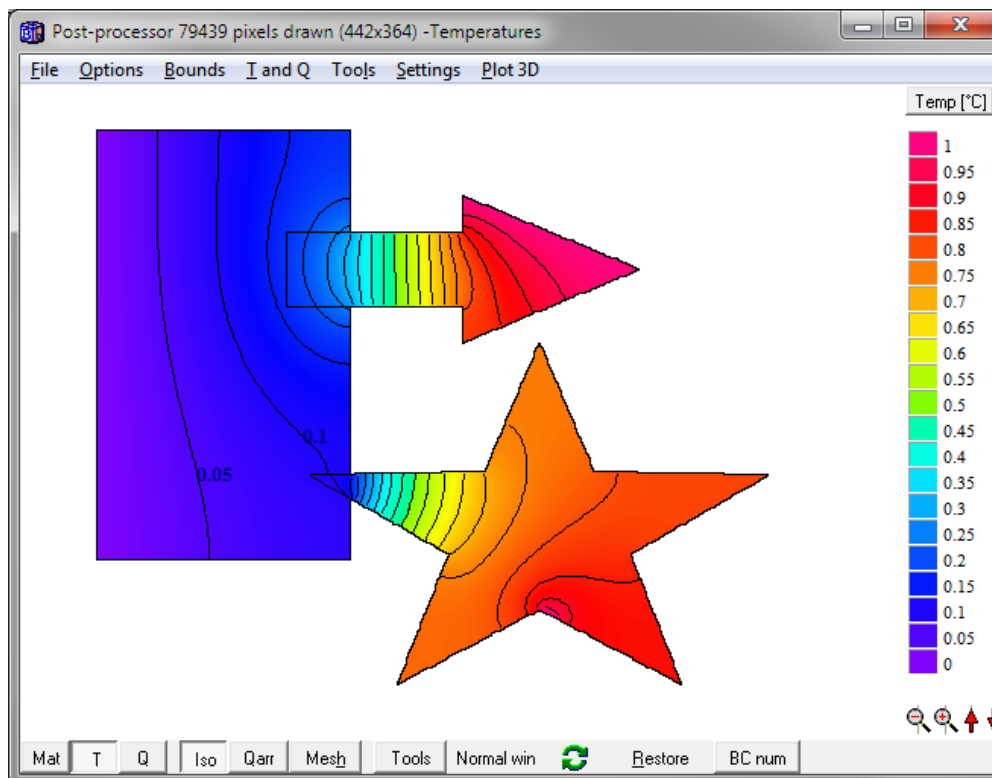
This is the default generated mesh at a detail (the mesh can be changed in the mesh editor):



Some dummy boundary conditions are assigned:



Here is the calculated temperature field and isotherms:

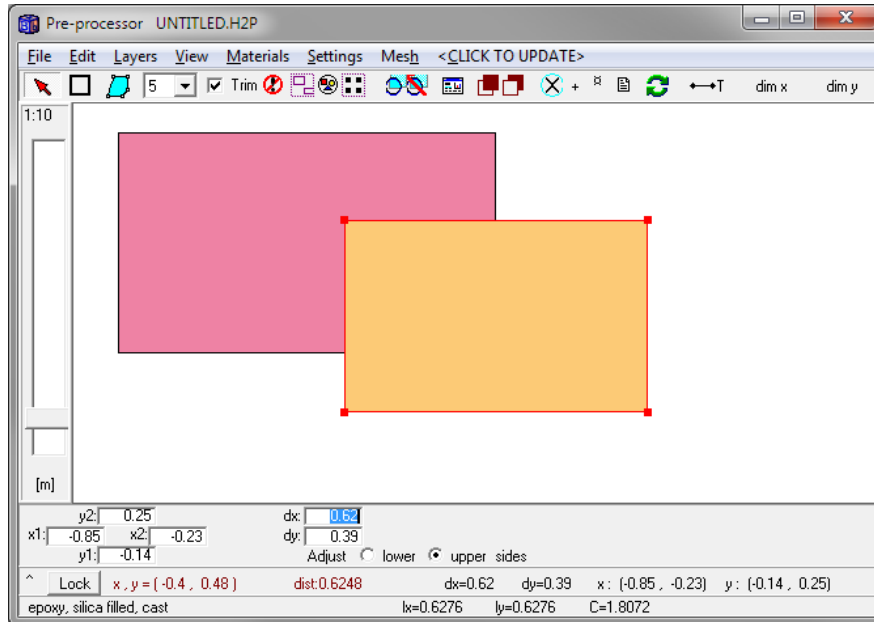


This example with 45,000 numerical cells took three seconds to solve.

## 4. Miscellaneous

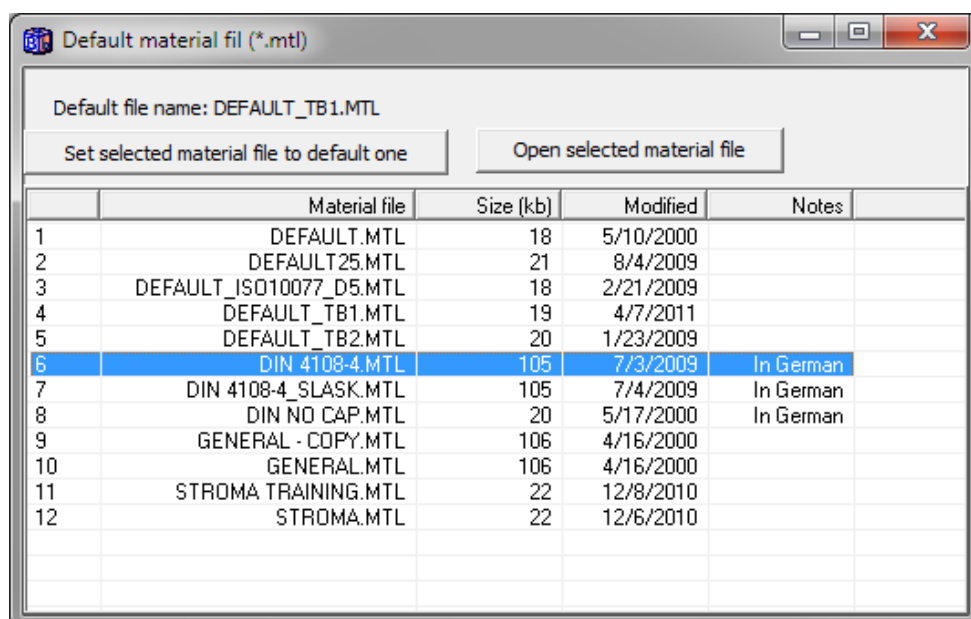
### 4.1 Length editor added

A new editor for giving lengths for drawn rectangles in the pre-processor has been added:



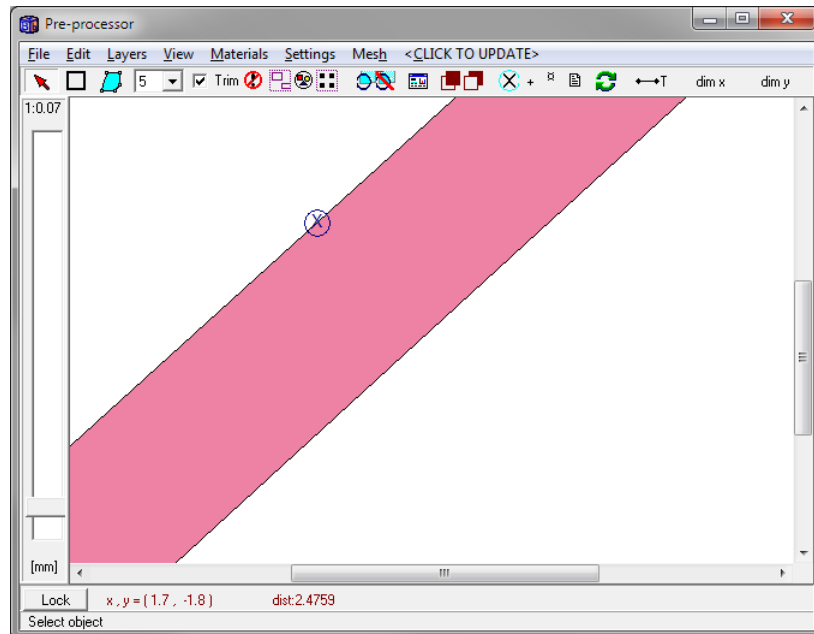
### 4.2 Dialog to set default material file

The default material file that will be used when HEAT2 starts can be changed in menu item "Options/Default material file". All material files (\*.mtl) available in the root directory (e.g. My Documents\Blocon\HEAT2 8) will be listed.



### 4.3 Distance from reference point

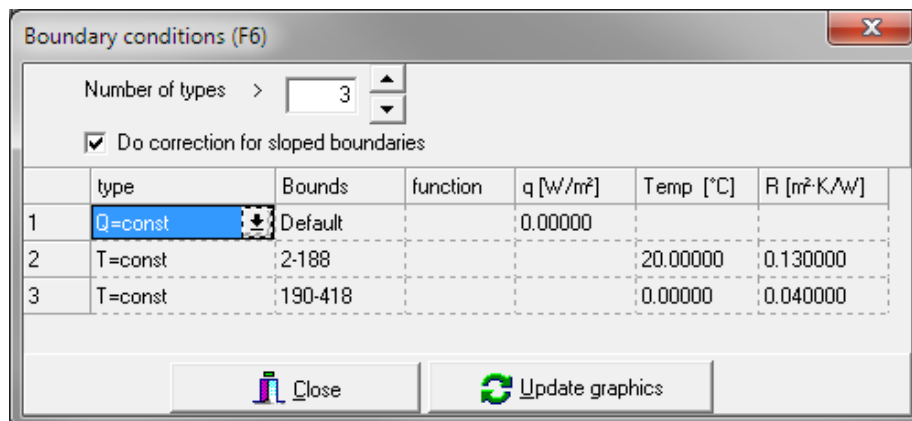
The distance from a reference point (or from origo if there is no reference point) to the mouse cursor is now shown on the second lower row in the pre-processor. This may be useful for e.g. checking the width of a sloping material.



### 4.4 Note on sloping external boundaries

There is a need to adjust the thermal resistance of a numerical cell lying at a boundary when a sloping boundary is approximated by steps in order to simulate a boundary condition with a temperature and a surface resistance, or a given heat flux, correctly.

This adjustment is made automatically by HEAT2 when “Do correction for sloped boundaries” is checked (it is on by default):

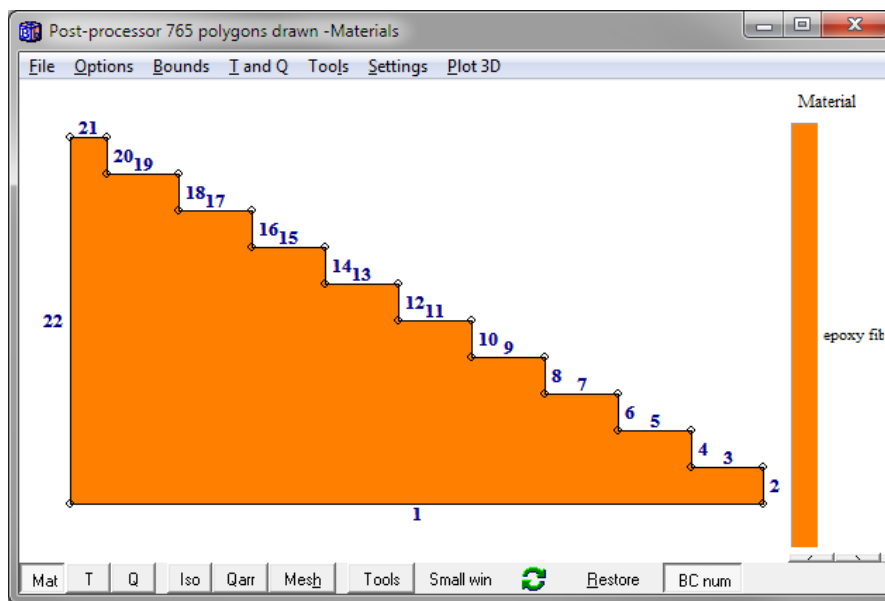


So, if you want e.g. a surface resistance of 0.13 on a sloping boundary, give R=0.13 and check “Do correction for sloped boundaries”. If you want 0.13 on the actual steps (parallel to the x,y-axes), uncheck this option.

### Case with a given heat flux as a boundary condition:

A similar correction is also needed for a boundary condition with a given heat flux. Consider e.g. a triangle. The length of the slope is  $L = \sqrt{w^2 + h^2}$ . The total length of the steps for the approximation is larger:  $w+h$ . This means that if you specify a heat flux ( $\text{W}/\text{m}^2$ ) for the slope, HEAT2 will scale the value  $q$  with  $w+h / \sqrt{w^2 + h^2}$  (if “Do correction for sloped boundaries” is checked).

E.g. if  $w=2$  and  $h=1$ , the length of the slope is  $L = \sqrt{5}$ . The total length of the steps becomes  $2+1=3$  regardless of number of steps. If you specify a heat flux of  $10 \text{ (W}/\text{m}^2)$  for the slope, HEAT2 will scale this automatically by  $\sqrt{5}/3=0.745$ , i.e. use  $7.45 \text{ (W}/\text{m}^2)$  for the steps. If correction is off,  $10 \text{ (W}/\text{m}^2)$  will be used for each step.



Boundary conditions (F6)

Number of types > 3

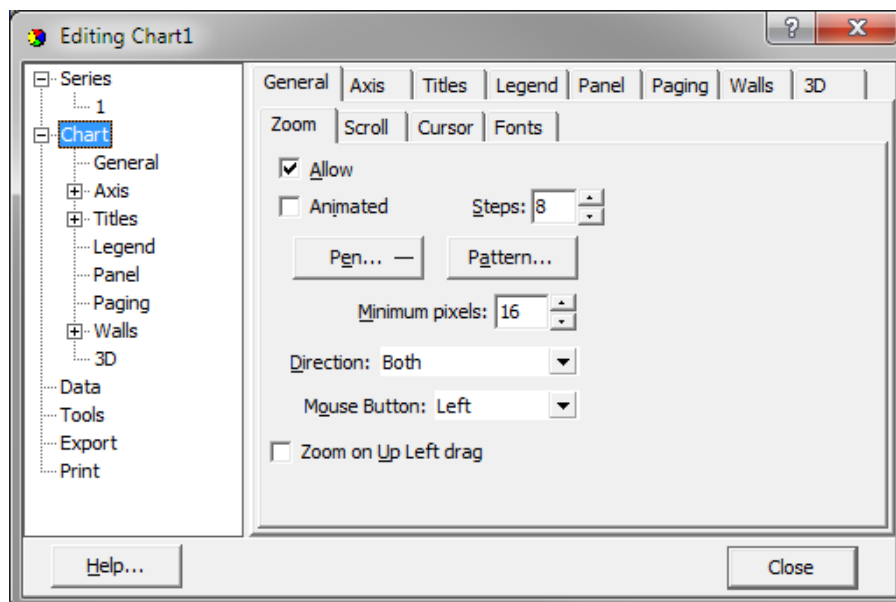
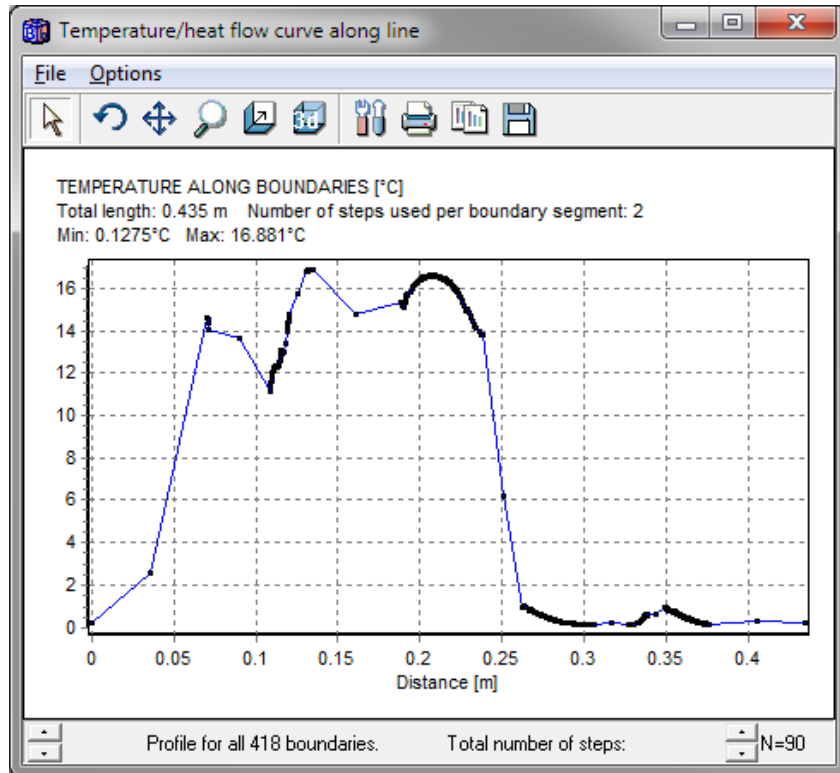
Do correction for sloped boundaries

	type	Bounds	function	q [W/m²]	Temp [°C]	R [m²K/W]
1	Q=const	Default		0.00000		
2	Q=const	2-21		10.00000		
3	T=const	1 22			0.00000	0.040000

Close Update graphics

## 4.5 Updated temperature/heat flow chart

The chart for temperatures/heat flows (menu item Output/Temperature along line boundaries and Plot 3D in the post-processor) has been updated. The chart editor is also updated:



## 4.6 Adapted for Windows 7 and Vista

HEAT2 is now fully adapted for windows Vista and Windows 7. HEAT2.exe is placed in folder %ProgramFiles(x86)” during installation.

One important change from previous versions is that the startup folder now is “My Documents\Blocon\HEAT2 8”. This means that the path in Windows Vista/7 would be ”C:\Users\[user name] \Blocon\HEAT2 8”. The materials files (\*.mtl) should be placed in this folder.

