NaviCell

Web tool for navigation, curation and maintenance of molecular interactions maps

Guide for map creator/manager and system administrator

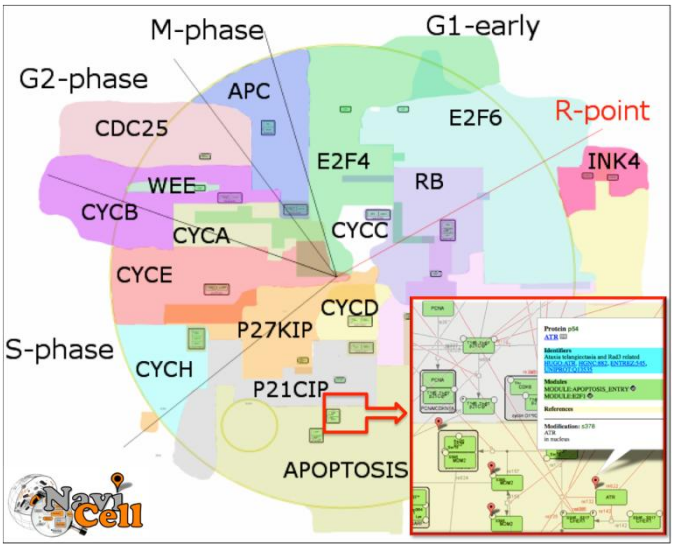


Table of Contents

1. Introduction

1.1 Description of NaviCell

NaviCell is a web tool for exploring large maps of molecular interactions created in CellDesigner (http://celldesigner.org). The tool is characterized by a unique combination of three essential features: efficient map navigation based on Google maps engine, semantic zooming for viewing different levels of details on the map and an integrated blog for collecting the community curation feedbacks.

For general description of NaviCell and instructions for users and map curators see NaviCell guide available at:
http://navicell.curie.fr/doc/NaviCellUserCuratorGuide.pdf

1.2 Guide content

This technical manual contains recommendations for map construction and NaviCell format for entities and species annotation in CellDesigner (Section 2). Once the map is built, a procedure for converting the map into NaviCell is available (Section 3). Instructions for installation of the NaviCell application are found in Section 4.

2. Map construction and annotation in CellDesigner

2.1 Entities and complexes naming in CellDesigner

For detailed guide about CellDesigner tool features and map construction tutorial please see http://www.celldesigner.org.
Here are a number of map construction rules to ensure compatibility of CellDesigner maps with the NaviCell format:
The dimensions of the map should be scaled to ×256 pixels.
-It is highly recommended to create square maps for achieving the best representation in NaviCell, but this is not a requirement.
-When a new entity is created in CellDesigner, the name dialog window appears. The common rules for naming entities are recommended for conserving the consistency through all maps.
1) Usage of HUGO IDs for gene, protein, RNA, asRNA, RNAs. In the case of using common name, or synonym, add a star mark (*) after the name (Figure 1A and Box 1).
2) Use common name or synonym for ion, simple molecule, drug and add star mark (*) after the name (Figure 1B and Box 1).
3) Use common names for phenotype and unknown, no need to add star mark (*) in the name, because these categories of entities are unique and not included in any of existing nomenclatures for molecules naming (Figure 1B and Box 1).
4) For names of complexes, list all names of all components in alphabetic order, separate names by a slash (/) (Figure 1C and Box 1). In the case of common name of complex, add star mark (*). In the case of using common name, the complex name by listing components in alphabetic order should appear in the annotation under ALT_NAME:NAME in the 'Identifiers' section (Figure 1D and Box 1).
5) Place the name of the module(s) in which the entity or species is included (see Section 2.4 for modular map description). Use preferably short names for modules and write module names in capital letters and without space (example MODULE:APOPTOSIS, MODULE:CHECKPOINT_G1_S, MODULE:CC_PHASE_S).
6) Multiple representation of the same entity on the map is allowed, this entity should be created by copy-pasting the original entity to preserve the CellDesigner ID of the entity.

NOTE: It is recommended to pre-define modules in the map and to create list of module names before initiation of map construction. Modular map preparation for NaviCell is simplified if the map can be pruned using modules names attribute (see Section 2.4).

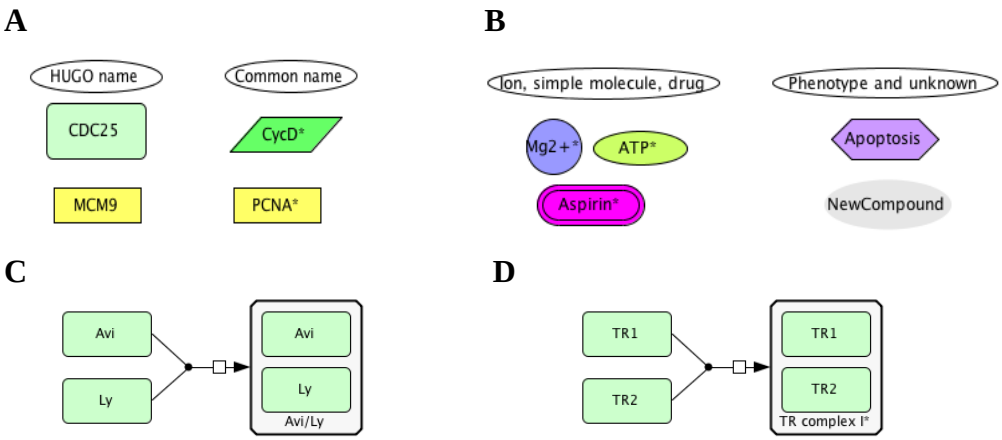


Fig1: Naming entities and complexes.
(A) *Naming of gene, protein (and receptor, ion channel, truncated protein), RNA, asRNA ,* ***(B)*** *Naming of ion, simple molecule, drug, phenotype and unknown,* ***(C)*** *Naming of complex without common name,* ***(D)*** *Naming of complex with common name*

BOX 1: Examples of naming entities and complexes

Naming Protein:
- By HUGO name (CDK2, HUGO name is repeated in the annotation in the section 'Identifiers'),
- By common name (CycE1*, HUGO name CCNE1 will appear in the annotation in the section 'Identifiers')
Ion naming:
Ca2+* (Official identifiers as CAS, CHEBI etc. appear in the in the annotation in the section 'Identifiers')
Complex naming:
- If common name does not exist, by listing of all components in the alphabetic order (RPA1/RPA2/RPA3).
- If common name exists: RPA complex. The name by listing all components appears in the annotation in the section 'Identifiers' as ALT_NAME

2.2 Avoiding representation errors

Complex with one component inside is recognized by CellDesigner as an error.
Any regulation on component of complex should be shown before complex formation.

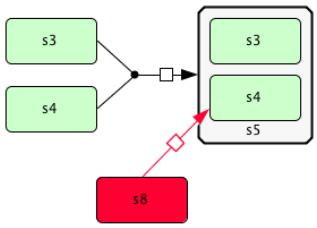


Fig2: Avoiding error.
Entity cannot be regulated inside the complex (red=error)

2.3 NaviCell map and entities annotation format in CellDesigner

The annotation of the map or modular maps (see Section 2.4 for modular map) is added into the 'Component-Model Notes' in CellDesigner. The annotation has to include the title. This title will appear in the NaviCell map representation.

The annotation of all entities and species of the map in CellDesigner should be done according to the common NaviCell annotation format during constructing of the map in CellDesigner. There are two groups of entities in the CellDesigner. Group 1 may have general annotation (entity annotation) and species annotation. In majority of the cases, the annotation is done for the entity. Species annotation is done only if there is need for a specific comment related to particular species. Species annotations are added after the general one in the post in NaviCell. Group 2 is not represented by entities, but by species only and annotation is done for species. There is no general annotation for group 2 (Box 2).

BOX 2: Groups of entities and species divided per type of annotation

Group1: Entities that contain general annotation and species annotation: proteins, receptor, ion channel, truncated protein, genes, RNA, asRNA. General annotation appears in the annotation in pop-up bubble of entities on NaviCell, Species annotations are added after the general annotation and appear only in the annotation post of entities in the blog of NaviCell.
Group2: Entities that contain only species annotation: complexes, simple molecules, ions, phenotypes, unknowns, reactions

The annotation is added in the 'Edit (entity type) note' (left) for group 1 and in 'Edit note' (right) for group 2 in CellDesigner.
The NaviCell template for entity and species annotation should be used for all annotations. The template contains 'Entity name', 'Identifiers', 'Modules' and 'References' sections. The sections of annotations should be encapsulated in a bigger section starting with 'Sectionname_begin' and ending with 'Sectionname_end' (example: 'Identifiers_begin', 'Identifiers_end'- Box 3). The individual sections are automatically color-marked while formatting in NaviCell (Figure 3). The information from the annotations is automatically retrieved from the CellDesigner file (xml) and transferred to the annotation in the pop-up bubble and to the annotation post in the blog by NaviCell (Figure 3).

For annotation of entities (group 1) and species (group 2), copy-paste the whole corresponding templates from Box 3 into the annotation field in CellDesigner (button 'Edit (entity name) Notes' for group 1, button 'Edit Notes' for group 2).

For annotation of individual species from group 1, copy-paste only 'Modules' and 'References' sections (marked by blue in Box 3) to the corresponding templates into the species annotation field in CellDesigner. (button 'Edit Notes').

BOX 3: NaviCell format templates for annotating entities and species in CellDesigner

Group1

Annotation format of protein, receptor, ion channel, truncated protein Identifiers_begin: Full name of the protein HUGO:NAME, HGNC:NAME, ENTREZ:NAME, UNIPROT:NAME, GENECARDS:NAME Identifiers_end Modules_begin: MODULE:NAME Modules_end References_begin: Text of annotation for corresponding PMID(s) PMID:NUMBER, PMID:NUMBER References_end	Annotation format of gene Identifiers_begin: Full name of the gene HUGO:NAME, HGNC:NAME, ENTREZ:NAME, GENECARDS:NAME Identifiers_end Modules_begin: MODULE:NAME Modules_end References_begin: Text of annotation for corresponding PMID(s) PMID:NUMBER, PMID:NUMBER References_end
Annotation format of RNA Identifiers_begin: Full name of the RNA HUGO:NAME, HGNC:NAME, ENTREZ:NAME, GENECARDS:NAME Identifiers_end Modules_begin: MODULE:NAME Modules_end References_begin: Text of annotation for corresponding PMID(s) PMID:NUMBER, PMID:NUMBER References_end	Annotation format of asRNA Identifiers_begin: Full name of the asRNA HUGO:NAME, HGNC:NAME, ENTREZ:NAME, GENECARDS:NAME Identifiers_end Modules_begin: MODULE:NAME Modules_end References_begin: Text of annotation for corresponding PMID(s) PMID:NUMBER, PMID:NUMBER References_end

Group2

Annotation format of complex Identifiers_begin: NAME: name of complex as list of components or common name, if exisits ALT_NAME: name of complex as list of components in the case the NAME is common name. Identifiers_end Modules_begin: MODULE:NAME	Annotation format of simple molecule Identifiers_begin: Full name of the simple molecule CAS:NAME, PUBCHEM:NAME, CHEBI:NAME, KEGGCOMPOUND:NAME Identifiers_end Modules_begin:

Modules_end References_begin: Text of annotation for corresponding PMID(s) PMID:NUMBER, PMID:NUMBER References_end	MODULE:NAME Modules_end References_begin: Text of annotation for corresponding PMID(s) PMID:NUMBER, PMID:NUMBER References_end
Annotation format of ion Identifiers_begin: Full name of the ion CAS:NAME, PUBCHEM:NAME, CHEBI:NAME, KEGGCOMPOUND:NAME Identifiers_end Modules_begin: MODULE:NAME Modules_end References_begin: Text of annotation for corresponding PMID(s) PMID:NUMBER, PMID:NUMBER References_end	Annotation format of drug Identifiers_begin: Full name of the drug CAS:NAME, PUBCHEM:NAME, KEGGDRUG:NAME Identifiers_end Modules_begin: MODULE:NAME Modules_end References_begin: Text of annotation for corresponding PMID(s) PMID:NUMBER, PMID:NUMBER References_end
Annotation format of phenotype *Identifiers_begin:* *Text of description if needed* *Identifiers_end* *Modules_begin:* *MODULE:NAME* *Modules_end* *References_begin:* *Text of annotation for corresponding PMID(s)* *PMID:NUMBER, PMID:NUMBER* *References_end*	*Annotation format of unknown* *Identifiers_begin:* *Text of description* *Identifiers_end* *Modules_begin:* *MODULE:NAME* *Modules_end* *References_begin:* *Text of annotation for corresponding PMID(s)* *PMID:NUMBER, PMID:NUMBER* *References_end*
Annotation format of reaction *References_begin:* *Text of annotation for corresponding PMID(s)* *PMID:NUMBER, PMID:NUMBER* *References_end*	

Official identifiers for annotating entities and species in CellDesigner can be retrieved from databases (Box 4).

BOX 4: Sources for official identifiers

To obtain **HUGO, HGNC, ENTREZ**	http://www.genenames.org
To obtain **GENECARDS**	http://www.genecards.org
To obtain **CAS**	http://www.genome.jp
To obtain **PUBCHEM**	http://pubchem.ncbi.nlm.nih.gov
To obtain **CHEBI**	http://www.ebi.ac.uk/chebi
To obtain **KEGGCOMPOUND**	http://www.genome.jp/kegg
To obtain **KEGGDRUG**	http://www.genome.jp/kegg

Notes added by the map creator should be preferably very short, represented by 1-2 sentences. Each entity or reaction on the map should be annotated by 3-5 references, when possible. References are listed using PMID:NUMBER (without space). All papers that are used in the annotation of entities of maps should be saved is a separate folder in the PDF format using format of name that contains PMID:NUMBER for verification of references by NaviCell (arbitrary text_PMID:NUMBER_arbitrary text).

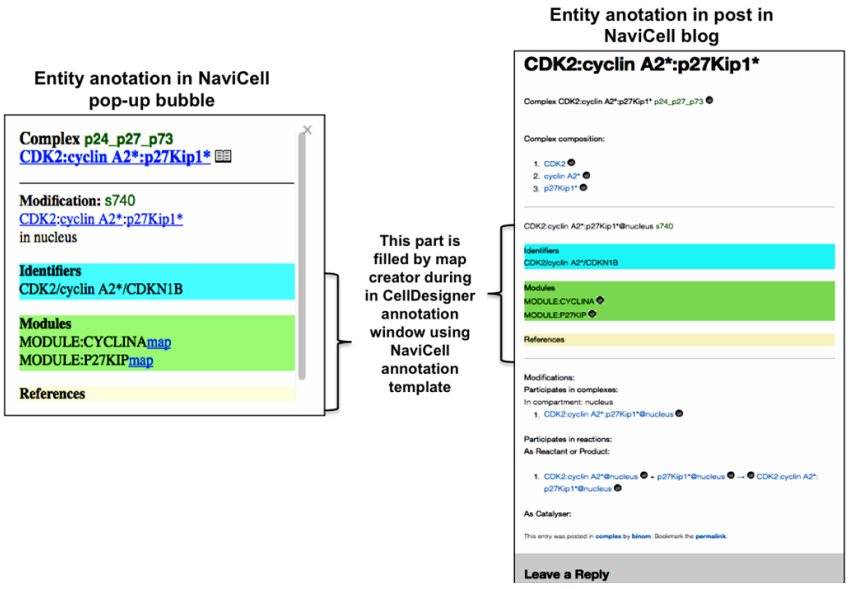


Fig3: NaviCell entity annotation format.

Annotation in the NaviCell pop-up bubble (left) and part of annotation in the entity post in the NaviCell blog (right) are generated from the CellDesigner's entity annotation. For the annotation in CellDesigner, the NaviCell template has been used. The rest of the information in the NaviCell annotation is generated automatically by NaviCell.

NOTE: NaviCell entities and species annotation template plugin is under construction, the manual procedure of inserting annotation template will se replaced by automatic one soon.

2.4 Modular maps

Map modules can be prepared in two ways: (1) each module can be represented as a separate map or (2) each module can be isolated by frame in the context of the whole map in CellDesigner that gives a continuous modular representation. In the first case, edges that are connected to the rest of the map are not kept. In the second case, entities of the module are emphasized whereas the rest of the map is partially hidden (brighten) and the edges connecting entities of the module to the rest of the map are kept. In both cases, the tags for modules must be indicated on the initial map (master map) by introducing a layer for each module with the module name in the CellDesigner (xml format).

The number of zoom levels for each module is similar to the number of zooms of the whole map. However, modular maps might undergo a simple size reduction without the creation of semantic zoom levels as it has been done for the whole map. Similar to the whole map zoom levels, the sizes of the images representing each zoom level is scaled to two times smaller than the preceding zoom level see Section 3.2 for Semantic zooming).

2.4.1 Separate modules with master map

In the case of deriving a separate map for each module from the initial (master) map, modules are created by pruning the master map. Edges connecting each module to the rest of the map are not preserved. The layout of modular maps can be changed for a more compact representation of module, when components of the module are dispersed on the master map. When creating a new layout, the location of the entities on the module map will change compared to the master map. The pruning of master map to derive modules and layout adjustment of resulting modules maps can be performed manually in CellDesigner or in Cytoscape. A semi-automatic procedure of pruning in Cytoscape using BiNoM plugin is applicable if lists of components in the module are pre-generated (Box 5).

BOX 5: Pruning maps using Cytoscape plugin BiNoM

Import the RB network in Cytoscape using BiNoM
Plugins => BiNoM I/O => Import CellDesigner document from file…
Select nodes from the network using the ID list (.txt) for each of the predefined modules
Select => Nodes => From ID list…
Create a new subnetwork for each of the modules
File => New => Network => Selected nodes, all edges
Update the connections of the newly-created network to insure that no interaction was lost in the process
Plugins=> BiNoM Utilities => Update connections from other networks
Rename the modules: Right click on the network in the 'network' panel, choose "Edit Network Title" and rename them
Verify that the modules are complete:
Merge all subnetworks or modules into one network
Plugins => Advanced Network Merge
Choose "union" and select networks to merge.
Compare the merged network to the initial RB network (network difference)
Plugins => Advanced Network Merge
Choose "difference"
If nodes are different, re-assign them to one of the appropriate modules
Create modular view of the network
Plugins => BiNoM modules => Create Nest networks
Plugins => BiNoM modules => Update connections between nests
Move modules on the graph according to the pre-defined place

2.4.2 Continuous modules with master map

If a continuous modular representation is chosen, isolating modules by frame in the context of the master map is performed in CellDesigner. In this case, module layout and localization on the master map should be carefully pre-defined, localization of modules components should be compact enough to enable framing of all components of each module into frames in a visually acceptable way. Layout of modules cannot be changed compared to the master map. For this type of modular representation, the entities within the module are emphasized whereas the rest of the map is partially hidden (brighten) and the edges connecting entities of the module to the rest of the map are kept.

NOTE: Procedure for preparation of modular maps for continuous modular representation will be updated soon.

2.4.3 Separate modules as independent maps

In the case of separate modules that are not derived from the common master map, each module is created as a separate map. Each entity is annotated by the name of the corresponding module. These separate modules can be merged into a common (master) map via common players. The layout of the new master map can be adjusted in CellDesigner or Cytoscape.
Creation of maps using this approach is recommended in the case of very big maps with many modules, multiple nodes and edges for simplicity of map manipulation in CellDesigner

NOTE: Tool for maps merging and procedure for visual module isolation on newly created master map are under construction and will be updated soon.

3. Map preparation for NaviCell representation

3.1 Map visual representation preparation

3.1.1 Creating pictures with colored modules background

For simplification of navigation through large maps of molecular interactions, the 'territory' of each module on the master map is indicated by different background color (Box 6). These module's background colors are preserved through all zoom levels. The top-level view is created using the image of the colored master map with 'territories' of modules with indication of the module names only (Figure 8).

BOX 6: Preparing modules background coloring

Export the map from CellDesigner as PNG
File => Export image => format PNG
Background layer in Gimp
Open (CTRL + O) the image in Gimp as background layer
Add new layer in Gimp to add background color for individual module
CTRL + SHIFT + N
Select area for coloring, can be larger than the area that needs to be colored (CTRL + R)
Choose color
Click on black square in the toolbox, then choose color in the pop-up window
Click on OK
Fill the selected area with the chosen color (SHIFT + B) or use the brush (P)
For another module, add new layer (repeat step 3 to 6)
Adjust opacity for each layer (around 20%) otherwise layers are not visible on top of each other.
To add lines (e.g. to indicate cell cycle phases), add new layer
Select pencil tool (N), click on layer where you want to start the new line, keep SHIFT pressed and click again to end the line
To add names choose Text Tool (T); text will be added as new layer
When satisfied with coloring of layers or adding additional text or lines save as PNGCTRL+SHIFT+S

Choose the appropriate file format (xcf as for Gimp file or otherwise choose PNG)
In case you have chosen PNG, select merge visible layers. In this case on the layers that are visible will be seen on the PNG file (so you can hide layers that won't use for now)
Also save the file as .xcf, this fill will be the master file in case you have to make adjustments e.g. change pictures, etc.

3.1.2 Edges coloring adjustment

Edges on the network created in CellDesigner are black and there is no distinction between reaction edges and reaction regulating edges (catalysis, inhibition, modulation, etc.). To simplify the vizualisation of the differences between edges depending on the function represented by the edge, we propose a color code of edges (Box 7).

BOX 7: Color codes of edges

Catalysis – Red
Unknown Catalysis – Dark Red
Inhibition – Green
Unknown Inhibition – Dark Green
Physical Stimulation – Orange
Modulation – Blue
Trigger – Violet

Positive influence – Red
Unknown Positive Influence – Dark Red
Negative Influence – Green
Unknown Negative Influence – Dark Green
Unknown Transition – Grey

The reaction can be recolored by importing the xml file into Cytoscape. With the Vizmapper, each reaction type can be designated a separate color (Box 8).

BOX 8: Assigning colors to edges

Import the xml file into Cytoscape using the BiNoM-plugin.
Once the map is opened in Cytoscape, click on the tab called Vizmapper in the control panel
Click on "+" sign next to "Edge Color"
Select Discrete mapping for mapping type
Then below the cell containing "Mapping Type" all the types of reactions are listed that are present on your map
Click on the empty cell next to Catalysis
A button showing "..." will show up
Click on that button
Select the desired color
Repeat the last four steps for each other reaction type

The network of edges on the maps is often dense, and parts of the nodes on the maps are obscured by the crossing edges. To avoid this, edges on the map can be converted to transparent ones when they cross/obscure nodes. This is a two-step procedure where we first make a xml file that does not show reactions, then export this file as PNG which will be imported in the Gimp master-file (Box 9).

BOX 9: Preparing map pictures with semi-transparent edges

Open the xml-file of your map in a text-editor that can import xml files.
Look for the following tags
<listOfReactions>
</listOfReactions>
Delete all text between those two tags, they are separated far from each other as in between is the description of all reactions that are present on the map
Save file as .xml type with a new name

Open the newly saved .xml file in CellDesigner
You will see the same map as before BUT without any reactions
Export the file as PNG
Open the file in Gimp as new layer in the master-file (CTRL+ALT+O)
Re-order the newly opened layer by clicking on either the up- or down-arrow in the Layers, Channels, Path window. The newly opened layer should be above the background layer i.e. the layer that contains the original CellDesigner map will all reactions and below the other layers (background coloring of modules)
The newly opened layer is covering the original map as the newly opened layer is opaque.
Make the white background of the newly opened layer transparent:
Select the layer that contains the map without reactions in the Layers window
Than click on Layers in the main window (at top of the window).
Choose Transparency.
Click on "Color to Alpha".
On the pop-up window, there is a preview image how the layer will look like.
On the preview image there is a button for selecting which color should be transparent, by default the white color is selected.
Click on OK, and the new layer is without the white background color.
This layer goes on top of the layer containing the map with all reactions.
Save file as xcf-type and PNG

3.2 Creating semantic zoom levels

The development of semantic zooming in NaviCell simplifies the navigation through large maps of molecular interactions by providing several levels of details, resembling navigation through geographical maps. Exploring the map from a detailed toward a top-level view is achieved by gradual exclusion and modification (simplification and abstraction) of details while zooming out. One of the main principles of semantic zooming is in that every detail which is shown on the map at a current zoom level, should be ***readable***.

If a user does not want to implement semantic zooming, the map images will be mechanistically and automatically scaled by NaviCell. However, we highly recommend investing time in carefully designing levels of semantic zoom, starting from the top-level, which illustrates the idea of the global map layout. BiNoM Cytoscape plugin provides important technical support, allowing manipulation of CellDesigner font sizes, scale of the map, and width of the edges.

In exploring comprehensive maps of molecular interactions we suggest to use four semantic zoom levels: (A) detailed, (B) hidden details, (C) pruned and (D) top-level (Figure 4).

A. The **detailed zoom level** is the original version of the map that contains all details as created in the CellDesigner map.
B. The **hidden-details zoom level** hides entity modifications, complex names and reaction numbers.
C. The **pruned zoom level** represents only major routes on the map (canonical pathways).
D. The **top-level view zoom level** is a territory map.

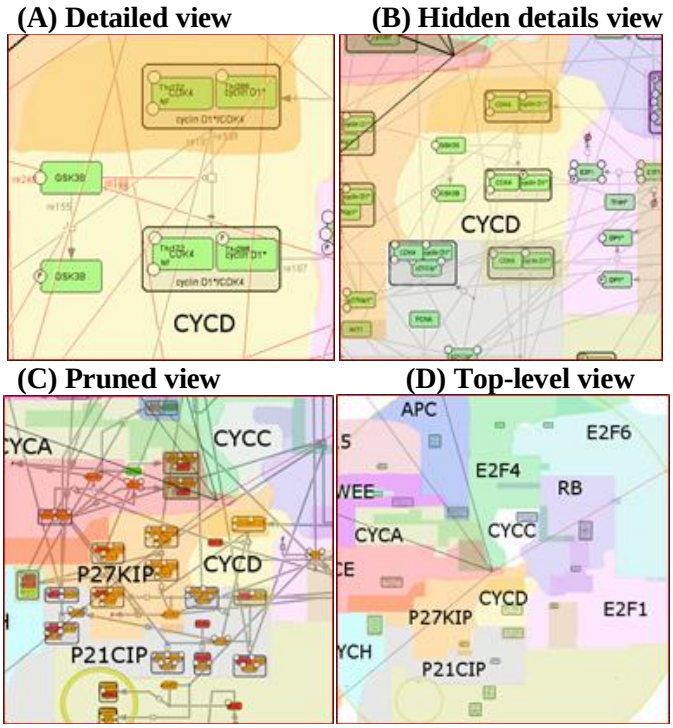


Fig 4: NaviCell semantic zooming.
The same area of the map is visualized using 4 zooms; each image is twice smaller that in the preceding zoom level. ***(A)*** *In the detailed view, entity names, modifications and reaction IDs are visible.* ***(B)*** *In the hidden details view, part of textual information is removed.* ***(C)*** *In the pruned view, only major routes are visible.* ***(D)*** *In the top-level view, boundaries of map modules are visible.*

The size of the images (PNG format) representing each semantic zoom level is scaled to two times smaller than the preceding semantic zoom level.

3.2.1 Detailed zoom level

The detailed zoom level is the original version of the map that contains all details as created in the CellDesigner.

3.2.2 Hidden-details zoom level

In the hidden-details zoom level entity's modifications, complex names and reaction numbers are hidden (Figure 5). To remove the reaction numbers one must unselect "Show reaction ID" under the menu option "View" in the CellDesigner program. In addition to remove entity's modifications, complex names, resizing font is done on Cytoscape using the BiNoM plugin, which can be found on the toolbar under the option Plugins in the Cytoscape program. Follow procedure in the Box10.

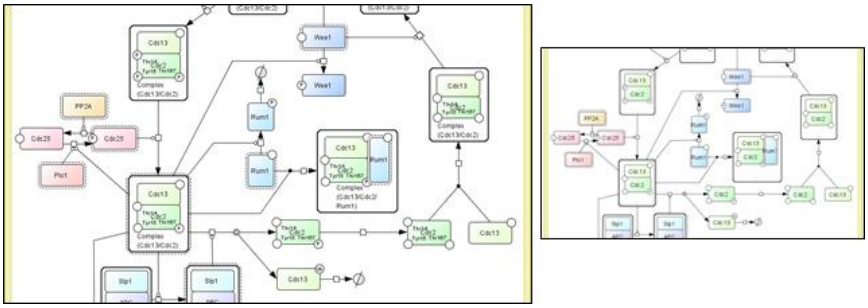


Fig 5: Hidden-details semantic zoom level.
Hiding some of the details (can produced by BiNoM). Left: Detailed zoom level. Right: Hidden details zoom level: pay attention at that complex names, residue names and reaction ids are hidden and the font sizes are enlarged.

BOX 10: Procedure for hidden details semantic zoom level preparation

Open Cytoscape that has the plugin BiNoM
Import CellDesigner file into Cytoscape using the BiNoM plugin (see section "isolated modules")
Export the map as CellDesigner file, using the BiNoM plugin (see section "isolated modules")
A pop-up window will appear
Check the following options for in- or decreasing the size of an entity: Scale only shapes
Adjust scaling factor: <1 for decreasing and >1 increasing the size of the entity
Check the option: Remove complex names
Check the option: Remove residue names
If desired font size can be adjusted as well
When satisfied click on OK, the file will be exported as xml-file
Open the exported file in CellDesigner
Export the file in CellDesigner to PNG-file

3.2.3 Pruned zoom level

The pruned zoom level represents only major routes on the map (canonical pathways), see example in Figure 7. We suggest several solutions to produce the pruned zoom level.

- *Solution 1 (Manual curation, selecting "canonical" players)*

Pruning of the map is done manually using CellDesigner and relies/is based on the description of the pathways in commonly accepted databases such as KEGG, Spike, WikiPathway, Reactome, Cell Signalling or other molecular pathways databases. Based on those databases, a selection of entities were selected and considered as canonical entities that can be found in those databases. The selected entities will be kept on the pruned zoom level. Other entities are to be deleted using the CellDesigner tool. Some of the edges connecting entities in the pruned zoom level have to be redrawn in case there are intermediate entities between them (see an example in Figure 6).

- *Solution 2 (Using structural analysis options of BiNoM to prune CellDesigner graph)*

This solution represents a completely automatic way to produce the pruned zoom level. For this, the CellDesigner map is first imported into BiNoM Cytoscape plugin and then manipulations on the map are performed. For example, the function "Prune graph" can be applied and the SCC part of the graph can be used for the Pruned semantic zoom (see more details in BiNoM manual), see an example in Figure 7.

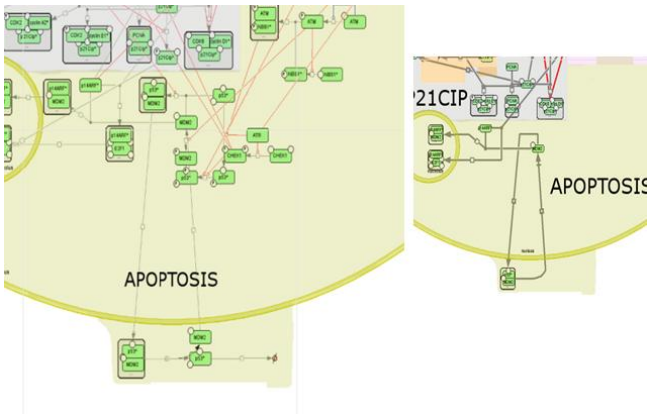


Fig 6: Hidden-details zoom level view.
Left: the hidden-details zoom level showing all entities. Right: The pruned-zoom level showing only canonical entities and newly drawn edge that follows the shape and direction of all edges in the hidden-details zoom level

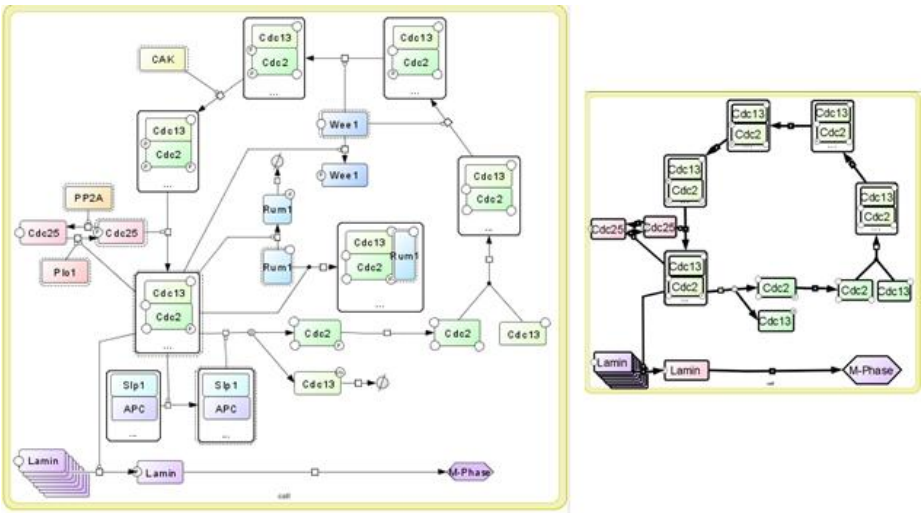


***Fig 7:* Example of a *pruned semantic zoom level* produced by graph pruning in BiNoM Cytoscape plugin.**
Left: hidden-details zoom level. *Right*: pruned zoom level, showing the main structural "skeleton" of the pathway.

3.2.4 Top-level view zoom level

The top-level view zoom level is a territory map. This zoom level does not contain map entities and edges, but represents localization of different map parts (modules). The top-level view zoom image is generated manually using any image editing program (e.g. Gimp, Adobe Photoshop). Each module is indicated by a semi-transparent background color. The names of modules are enlarged proportionally to this zoom. This color-code is recommended to maintain through all semantic zoom levels. See Section 3.1 for procedure of background coloring.

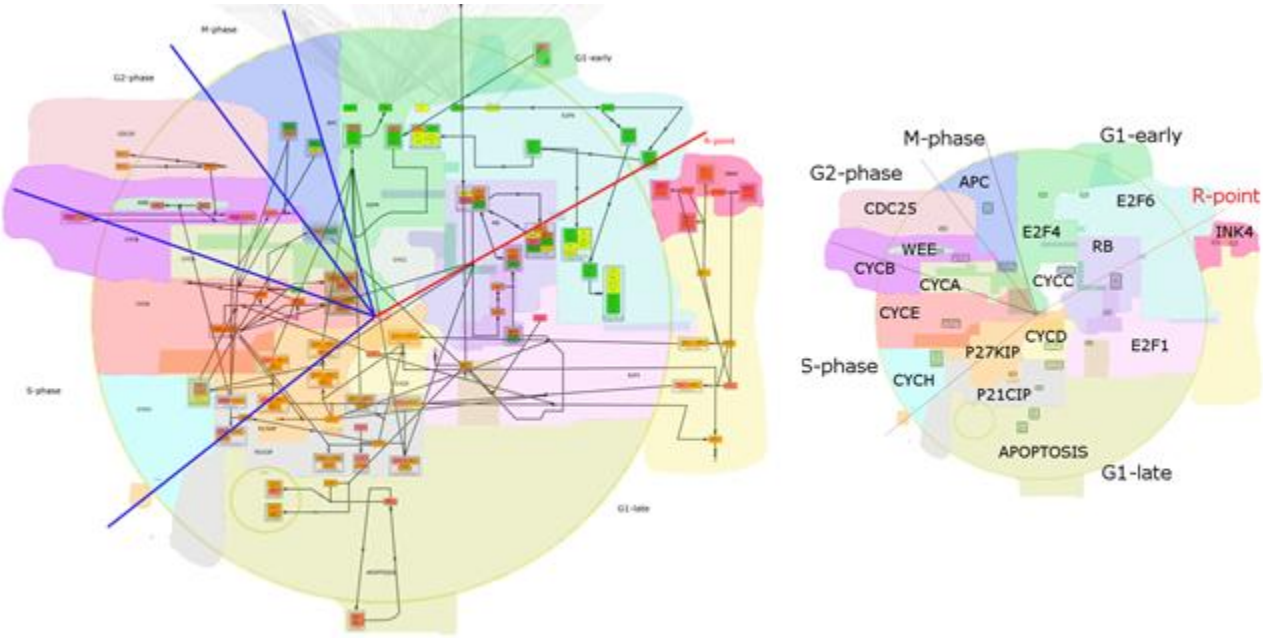


Fig 8: Example of a top-level semantic zoom level.
Left: pruned semantic zoom level. Right: top-level view, depicting only colored modules territories ("country borders") and modules names ("country capitals").

NOTE: In the near future, the process of generating semantic zoom levels will be (semi)-automated.

4. NaviCell Map Installation

4.1 General requirements

Creating NaviCell map requires:
1) A collection of map images for the map itself and (maybe) for modules of the map
2) Installed and pre-configured WordPress server for the map's blog (not needed in the light "browse-only" version of NaviCell installation)
3) Created WordPress blog for the map (not needed in the light "browse-only" version of NaviCell installation)
4) A collection of html pages with JavaScript implementing Google Maps API and other NaviCell's functionalities. Functioning of NaviCell requires permanent access to Internet because of binding with Google Maps engine.

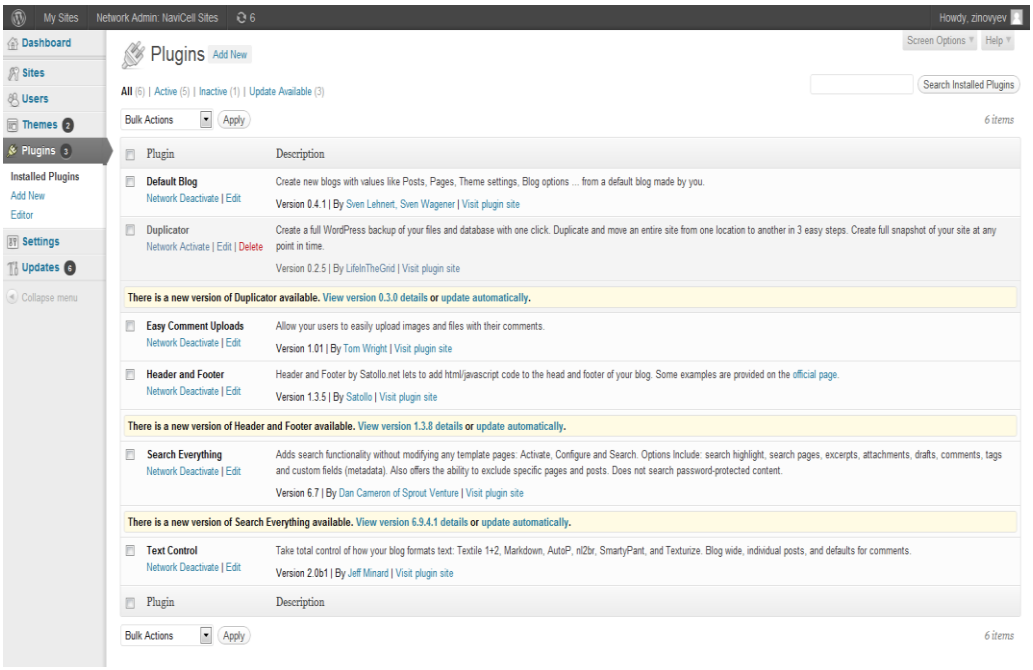
Full NaviCell installation requires access to functional and properly configured WordPress Server and creation of a web-blog. If this is not possible for technical reasons, NaviCell can be installed in light (browse-only) version, in which commenting maps will not be possible (see instructions below on how to generate both versions of NaviCell).

This procedure explains in details 1) How to configure WordPress Server for NaviCell; 2) How to create a new blog for a new NaviCell map; 3) How to prepare source files for a new NaviCell map; 4) How to generate html pages needed for launching NaviCell (locally or on a web-server).

4.2 Configuring WordPress Server

We assume that a WordPress Server is installed and set up. For instructions, refer to WordPress guide at http://wordpress.com.

For NaviCell functioning, several WordPress plugins should be installed and activated: "Default Blog", "Easy Comment Uploads", "Header and Footer", "Search Everything", "Text Control". The "Plugins" page of WordPress should look like this:



In the "Settings/Network Settings" we recommend choosing the option "User accounts may be registered" and "Max upload file size" to 400K.

4.3 Installing a new NaviCell map in NaviCell

Installing a new map in NaviCell is composed of 3 sequential steps:

1) preparing a folder with a set of files describing the main map and the map modules, the images that will be used to browse the map and a configuration file called "config" (Section 4.3.1)

2) creating and configuring a new WordPress blog (*or not, in the "browse-only" version*) (Section 4.3.2)

3) populating the WordPress blog (*if not the "browse-only" version*) and creating a set of html pages implementing NaviCell's functionality.

4.3.1 Preparing the xml files for map and modules

Currently NaviCell supports only maps created in CellDesigner 4.* software. If a map was created in an older version of CellDesigner, it should be converted and saved using CellDesigner 4.* version.

First of all, the ***master map*** xml file should be created. After this, several ***module maps*** can be created by 1) removing part of the entities and chemical species from the master map and 2) changing the map layout. Module maps should have the same entity and chemical species identifiers as in the master map, and not others. Thus, the purpose of the modular maps is to show a subset of master map entities and species, using a new layout. See Section 2.4 for map modules preparation and Section 3.1 for Preparing maps for NaviCell representation.

NOTE: If there are modules defined, the master map should contain a list of layers one per each module with the names corresponding to the module names. Each layer should contain at least one graphical primitive (e.g., rectangle) which marks the positioning of the module marker on the map.

All xml files should be named according to the template: [MAP_BASE]master.xml for the master map, [MAP_BASE][MODULE_NAME].xml for the modules map, where [MAP_BASE] is the prefix specified in the *config* file (see below) and [MODULE_NAME] is the name of the module.

For each xml file, a set of images in PNG format should be generated, numbered according to the zoom level. The top-level map (containing the most top-level abstract map representation) is assigned a number "0", all further zoomed-in map representations have numbers from "1" to some *N*, where *N* is the maximum level of zoom (containing the most details).

All PNG files should be named according to the template: [MAP_BASE]master-[ZOOM_LEVEL].png for the master map, [MAP_BASE][MODULE_NAME]-[ZOOM_LEVEL].png for the modules map, where [MAP_BASE] is the prefix specified in the *config* file (see below) and [MODULE_NAME] is the name of the module, and [ZOOM_LEVEL] is the level of the semantic zoom (0-for the most abstract, *N* – for the most detailed), see Box 11.

BOX 11: Example of map and modules files naming

A test example files describing a map with the prefix "test_", separated into modules "MITOCHONDRIA", "NUCLEUS", "OVAL", "SQUARE", with 4 levels of zoom ("0" – for the most abstract, and "3" – for the most detailed) a set of files should look like the following:

test_master-0.png
test_master-1.png
test_master-2.png
test_master-3.png
test_master.xml
test_MITOCHONDRIA-0.png
test_MITOCHONDRIA-1.png
test_MITOCHONDRIA-2.png
test_MITOCHONDRIA-3.png
test_MITOCHONDRIA.xml

```
test_NUCLEUS-0.png
test_NUCLEUS-1.png
test_NUCLEUS-2.png
test_NUCLEUS-3.png
test_NUCLEUS.xml
test_OVAL-0.png
test_OVAL-1.png
test_OVAL-2.png
test_OVAL-3.png
test_OVAL.xml
test_SQUARE-0.png
test_SQUARE-1.png
test_SQUARE-2.png
test_SQUARE-3.png
test_SQUARE.xml
```

4.3.2 Preparing map and modules images

General requirements

The images are named *X*Y-*i*.png where *X* is the base as defined in the *config* file, Y is the module's name and *i* is the number of the image. The images are numbered starting from 0 and there must not be any gaps in the numbering. The image numbered *i*+1 must be exactly 2_n times bigger than the image *i* (for some positive integer value of *n*). As each final image must be exactly twice as big as the preceding image, missing images will be created by de-zooming the next existing image. All modules must have the same number of images at the same size. Objects should not move between zoom levels.

Preparing images for semantic zooming

In principle, the PNG images provided for map browsing can contain any decorations and representations of chemical species and reactions provided that their coordinates in the xml file describe their positions correctly. This means also that a part of chemical species can be hidden or redrawn with fewer details at more abstract "zoomed-out" images.

We developed a set of recommendations for making four levels of semantic zooming for large maps, with the help of BiNoM plugin. For procedure, see Section 3.2. Instruction for preparing decoration of map images see Section 3.1.

4.3.3 Preparing the "config" file

Together with xml and png files in the folder containing the sources of the NaviCell map, one has to put a file named "config". This file should contain three lines. The first line describes the prefix (base name) for the set of files. The second line describes the name of the map (a name that will appear in the URL address and in the name of the blog). The third line describes a Boolean option specifying whether the names of the species contained in the largest compartment will contain a suffix with the compartment name (setting it up to "true" might be convenient if the map contains only one compartment or one big and several small compartments).

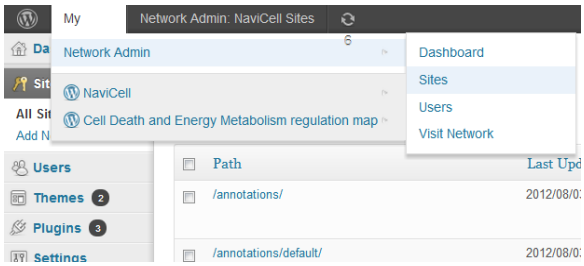
BOX 12: Example of config file

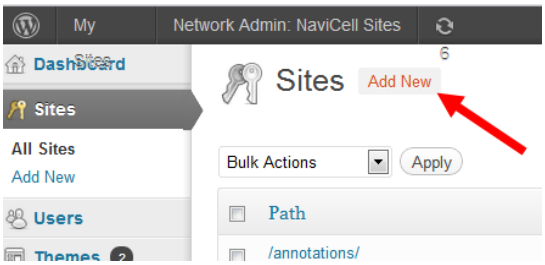
```
base: test_
name: testmap
showDefaultCompartmentName: true
```

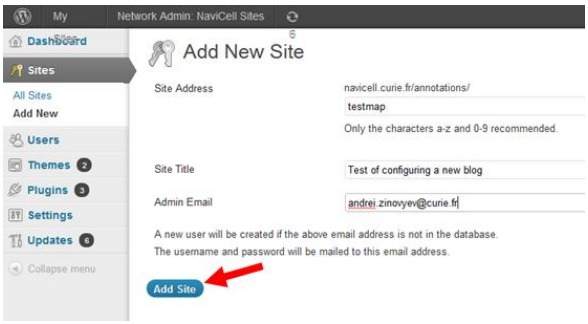
4.4 Configuring the map's blog in WordPress

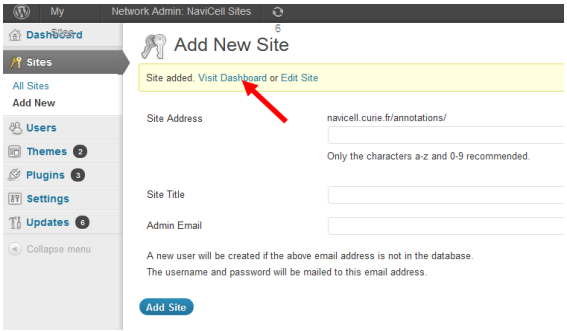
For a new NaviCell map, a new WordPress blog ("site") should be created. It must be named with the same name specified in the second line ("name") of the "config" file.

The blog itself is created through clicking at "My/Network Admin/Sites" and "Add new" as shown below:

1) 

2) 

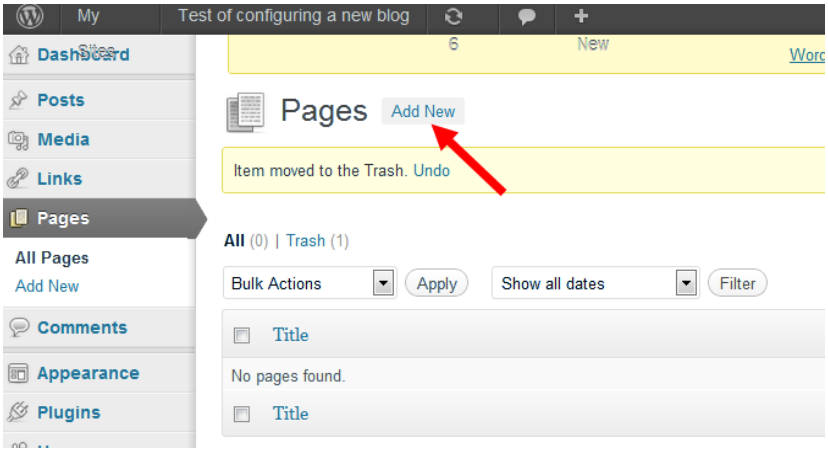
3) 

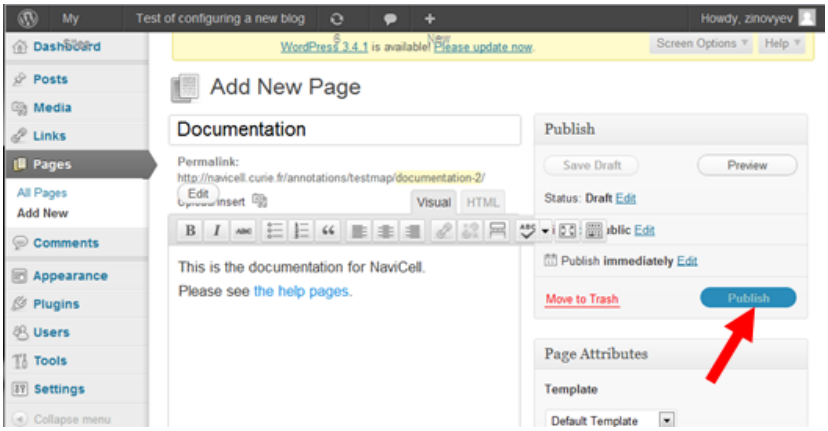
4) 

In the Dashboard for the site, the following 13-steps site configuration must be made:

Step 1)

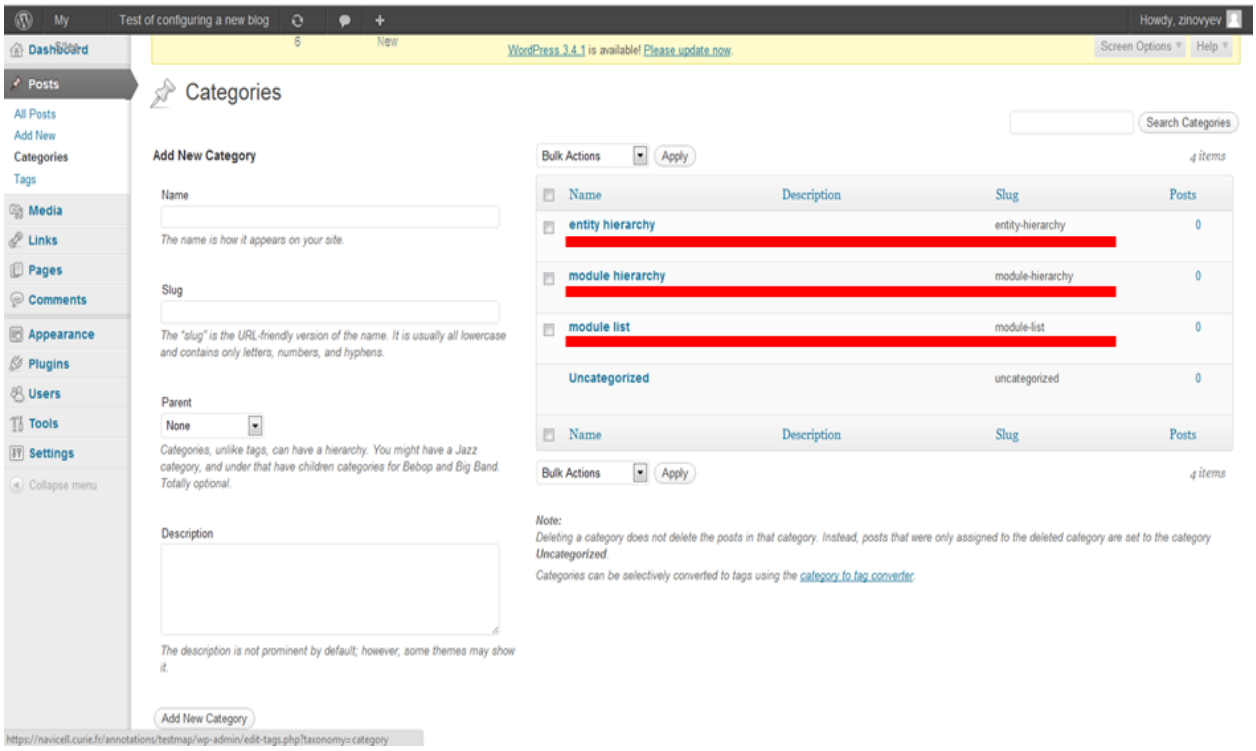
Create a new "**Documentation**" page if it is not yet created (*one can create it in the default blog and reuse it after*) as shown below:

1) 

2) 

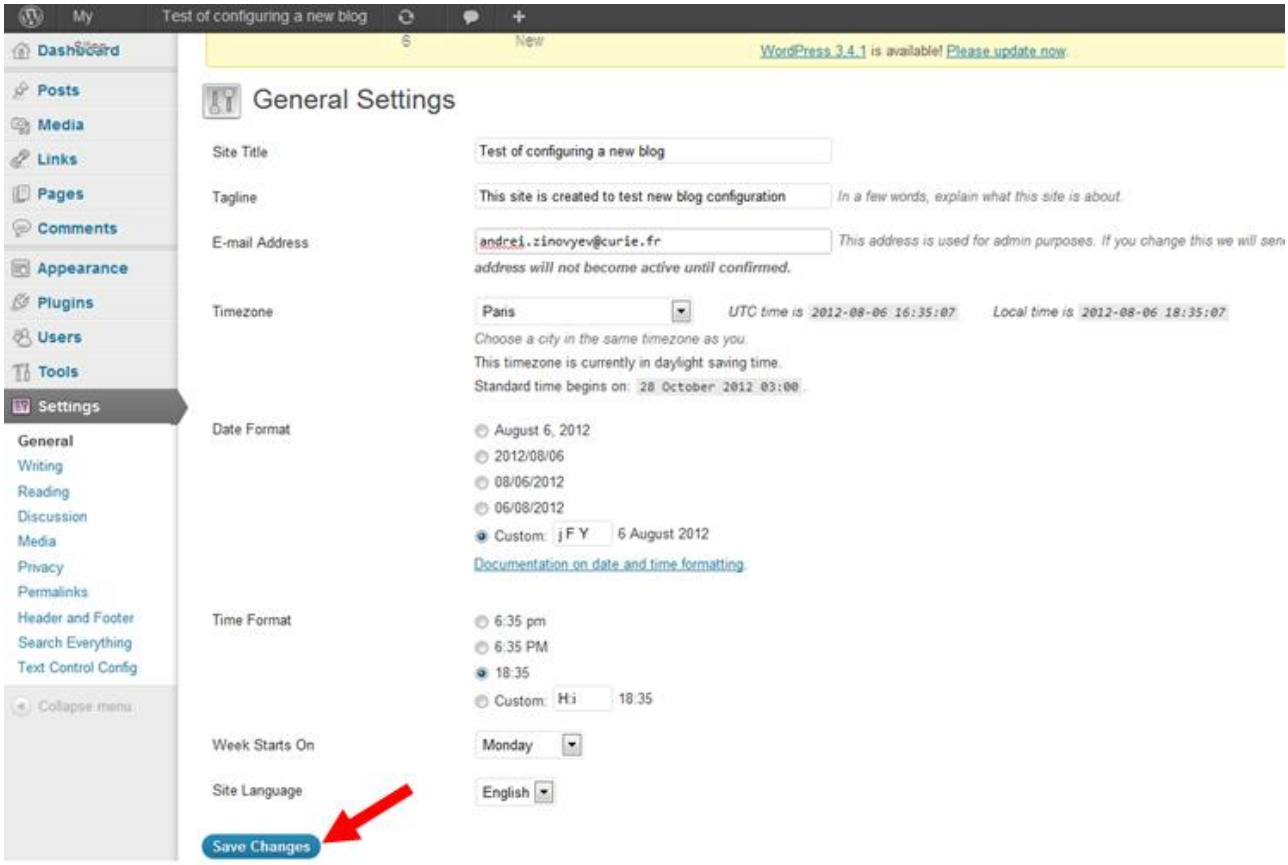
Step 2)

Create three categories **entity hierarchy** (slug entity-hierarchy), **module hierarchy** (slug module-hierarchy), **module list** (slug module-list), if they are not created, in **Posts/Categories** (*one can create them in the default blog and reuse it after*) as shown below

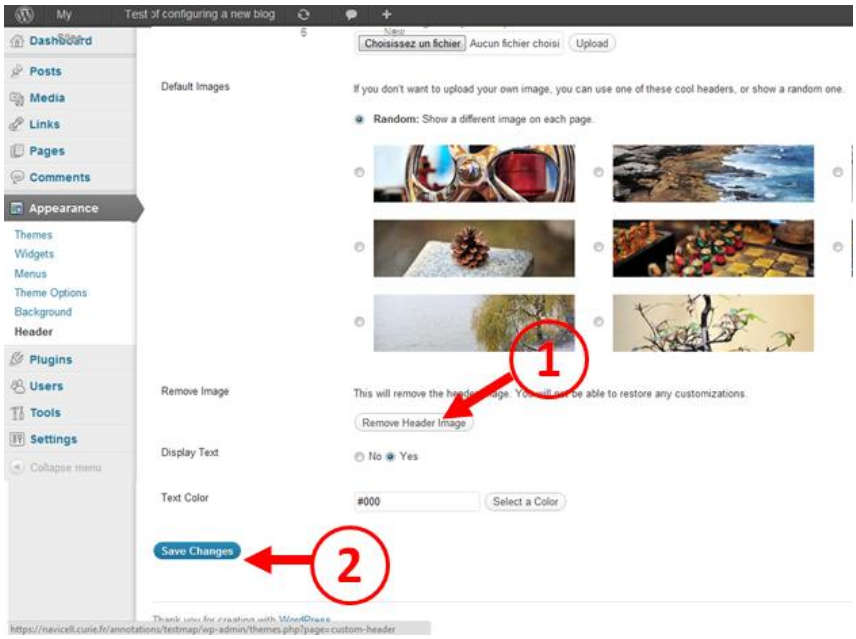


Step 3)

Configure **General settings** as shown below (modify accordingly):

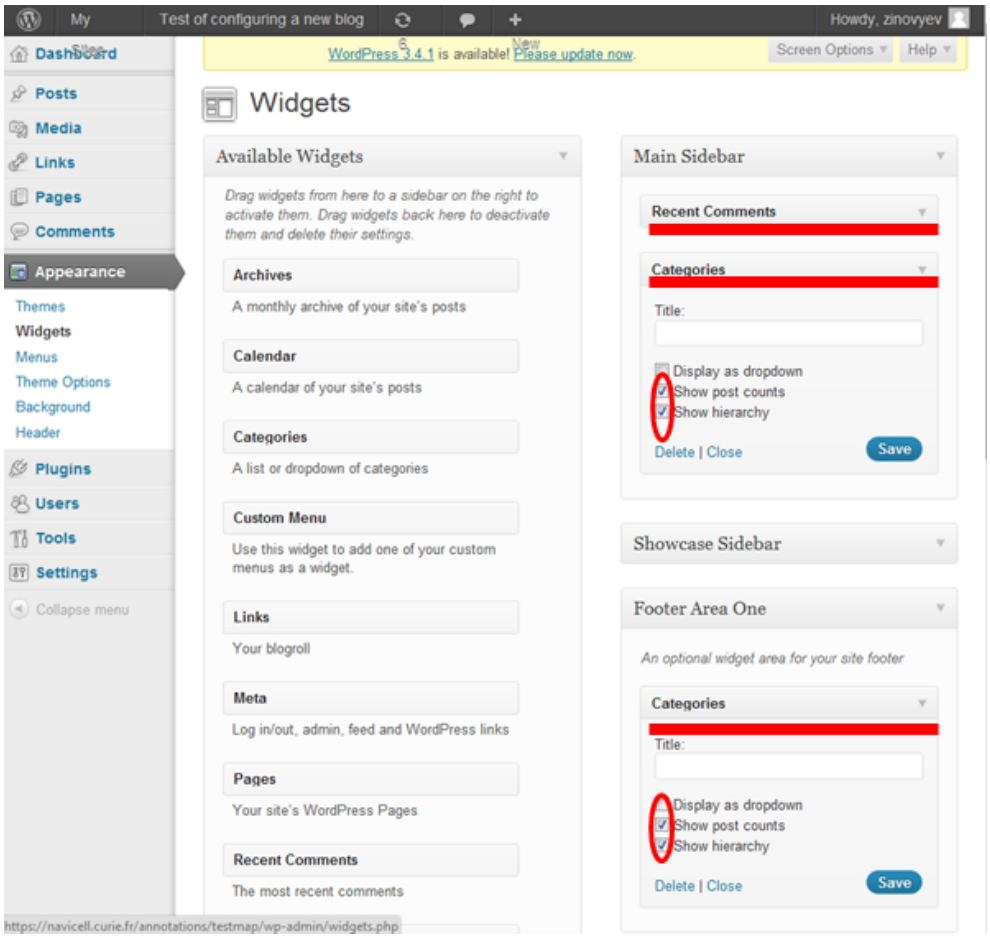


Step 4)
Configure **Appearance/Header** as shown below (the purpose is to remove header image):



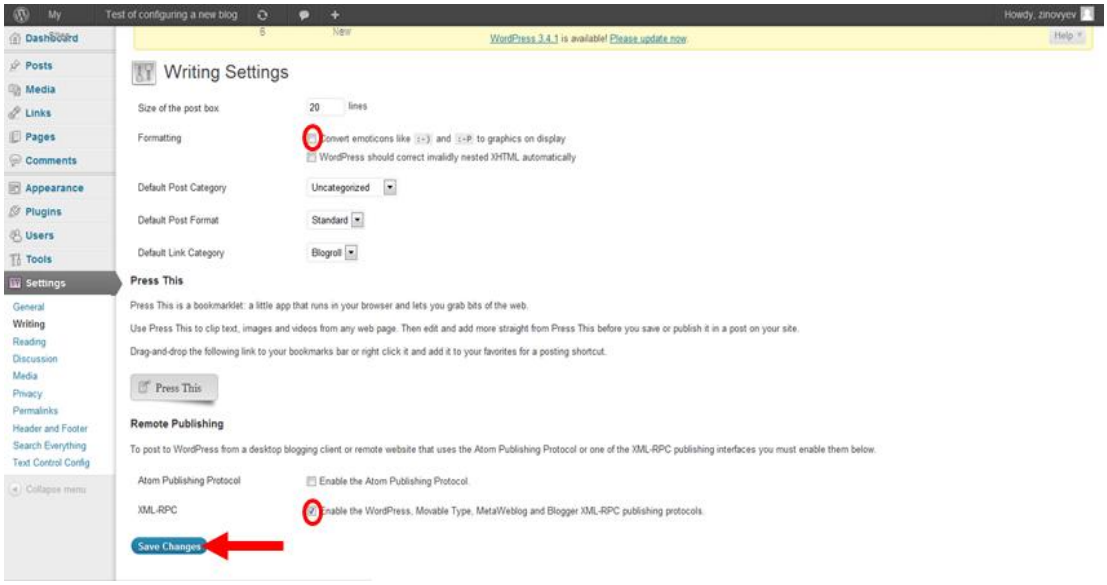
Step 5)

Configure **Appearance/Widgets** as shown below (the purpose is to leave only "Categories" and "Recent Comments" in the Main Sidebar):



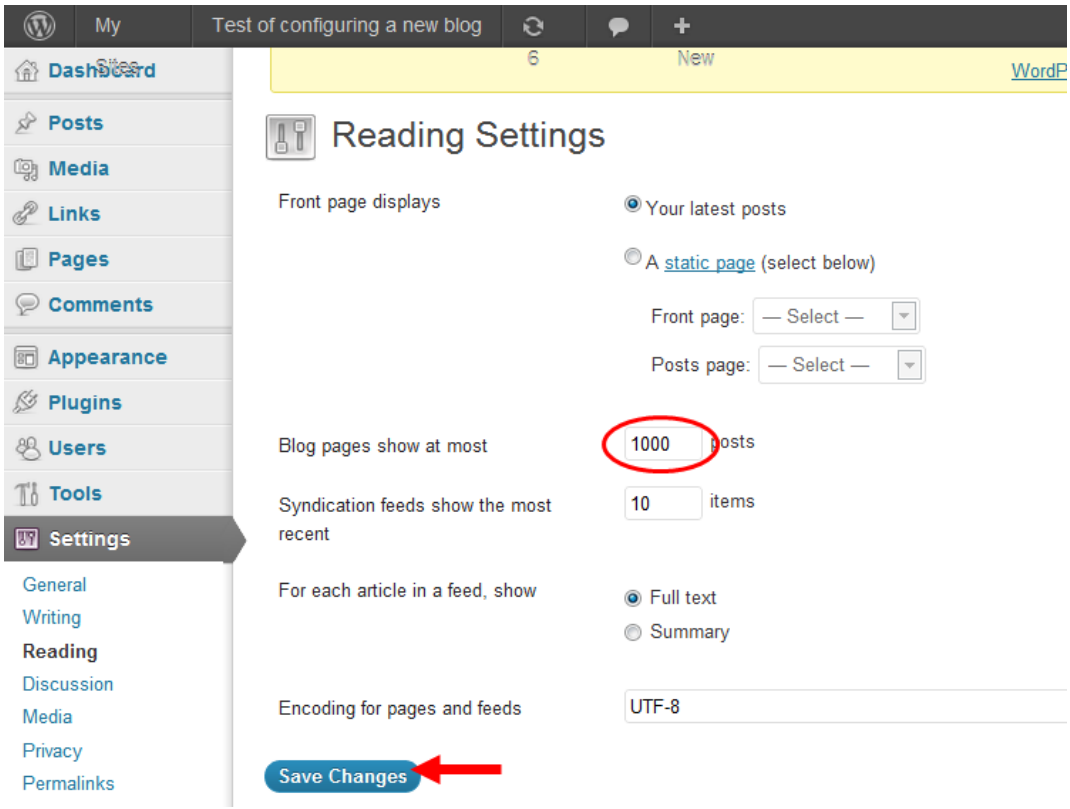
Step 6)

Configure **Settings/Writing** (the purpose is to disable emoticons and enable XML-RPC publishing, which will be needed at the step of populating the blog automatically through BiNoM, see below)



Step 7)

Configure **Settings/Reading** (the purpose is to limit the maximum number of blog pages to show up to 1000)

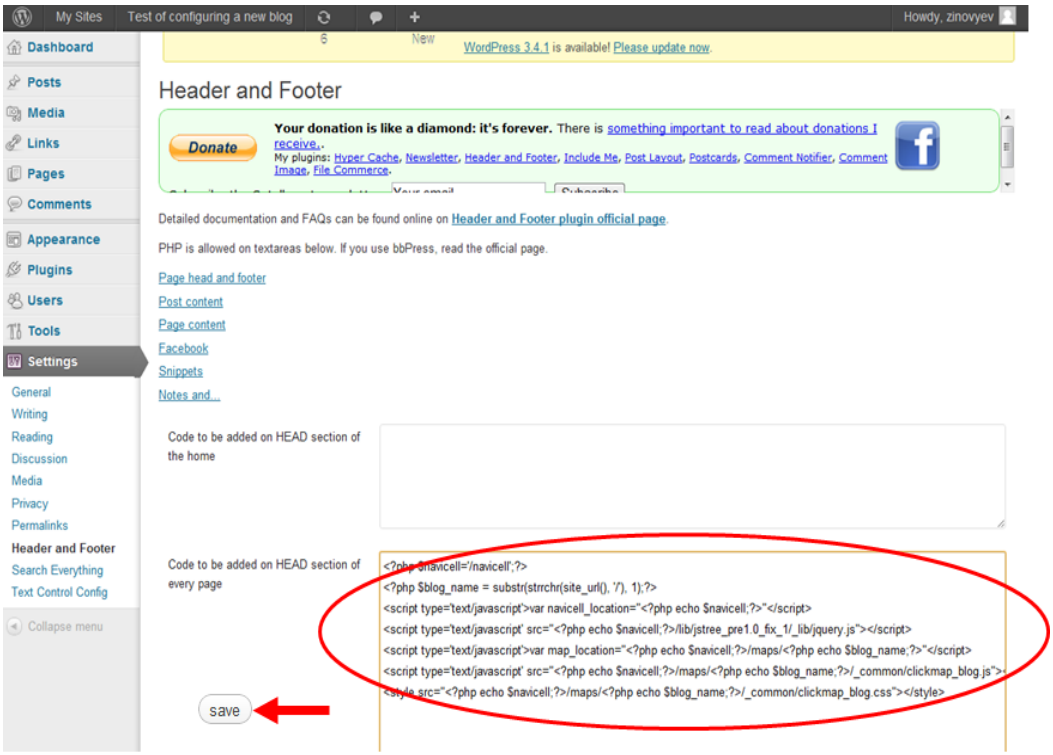


Step 8)

Configure **Settings/Header and Footer** (the purpose is to insert some JavaScript instructions in the beginning of each page automatically). The text to be inserted can be copy-pasted from below:

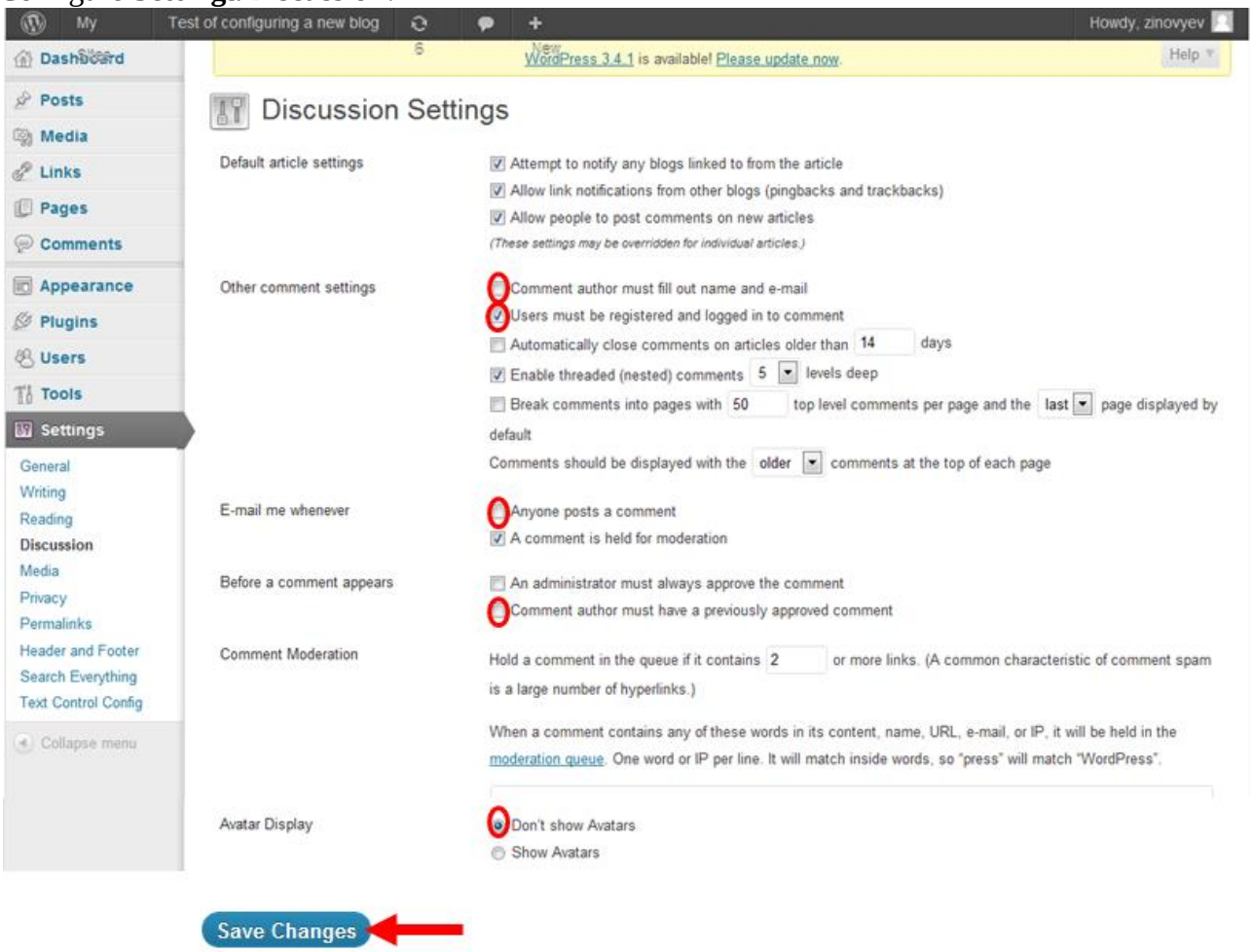
```
<?php $navicell='/navicell';?>
<?php $blog_name = substr(strrchr(site_url(), '/'), 1);?>
<script type='text/javascript'>var navicell_location="<?php echo $navicell;?>"</script>
<script type='text/javascript' src="<?php echo $navicell;?>/lib/jstree_pre1.0_fix_1/_lib/jquery.js"></script>
<script type='text/javascript'>var map_location="<?php echo $navicell;?>/maps/<?php echo $blog_name;?>"</script>
<script type='text/javascript' src="<?php echo $navicell;?>/maps/<?php echo $blog_name;?>/_common/clickmap_blog.js"></script>
<style src="<?php echo $navicell;?>/maps/<?php echo $blog_name;?>/_common/clickmap_blog.css"></style>
```

NOTE: *The map pages generated will learn the location of the blog from a parameter "WordPress Server URL" in BiNoM interface (see below). The blog should also know the location of the map pages location. This location is specified in the JavaScript code above and highlighted in bold ("**/navicell**" in our case). If the maps are installed not in "/navicell" then this parameter should be modified accordingly, otherwise it won't be possible to shuttle between the blog and the maps.*



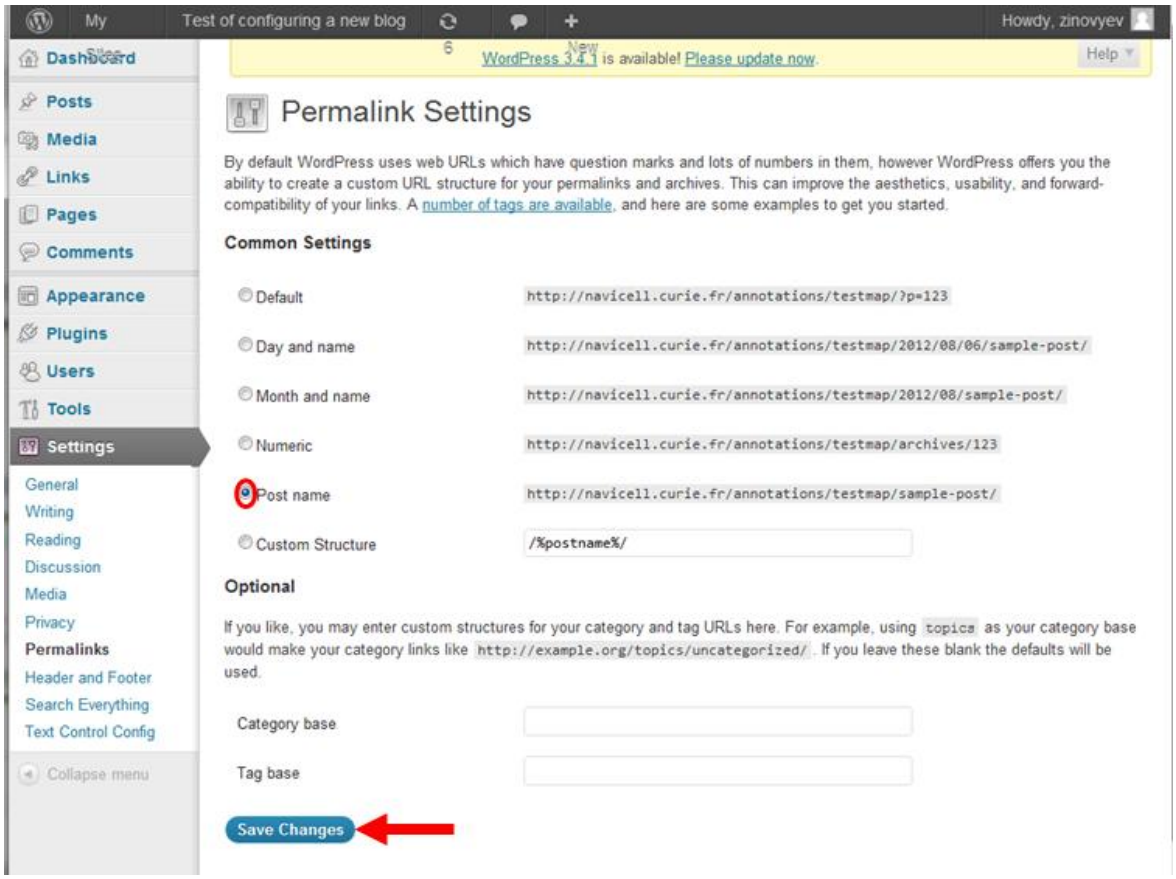
Step 9)

Configure **Settings/Discussion**:



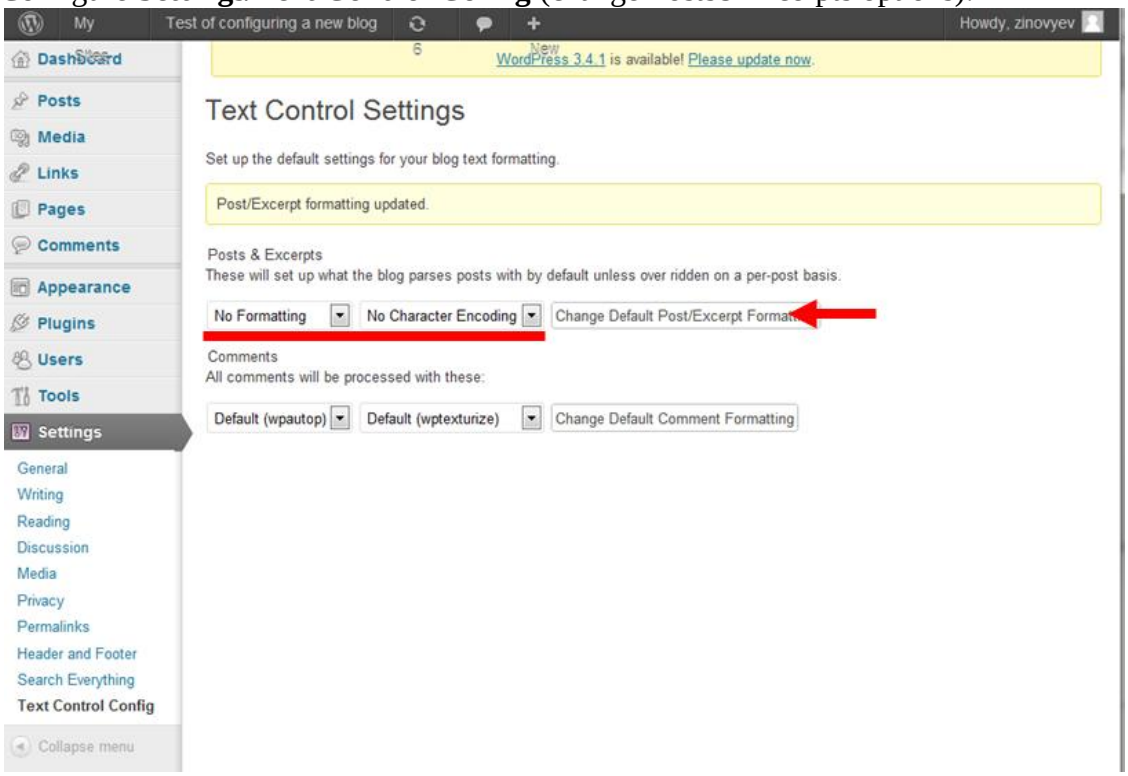
Step 10)

Configure **Settings/Permalink** (use the Post name option):



Step 11)

Configure **Settings/Text Control Config** (change Posts&Excerpts options):



Step 12)

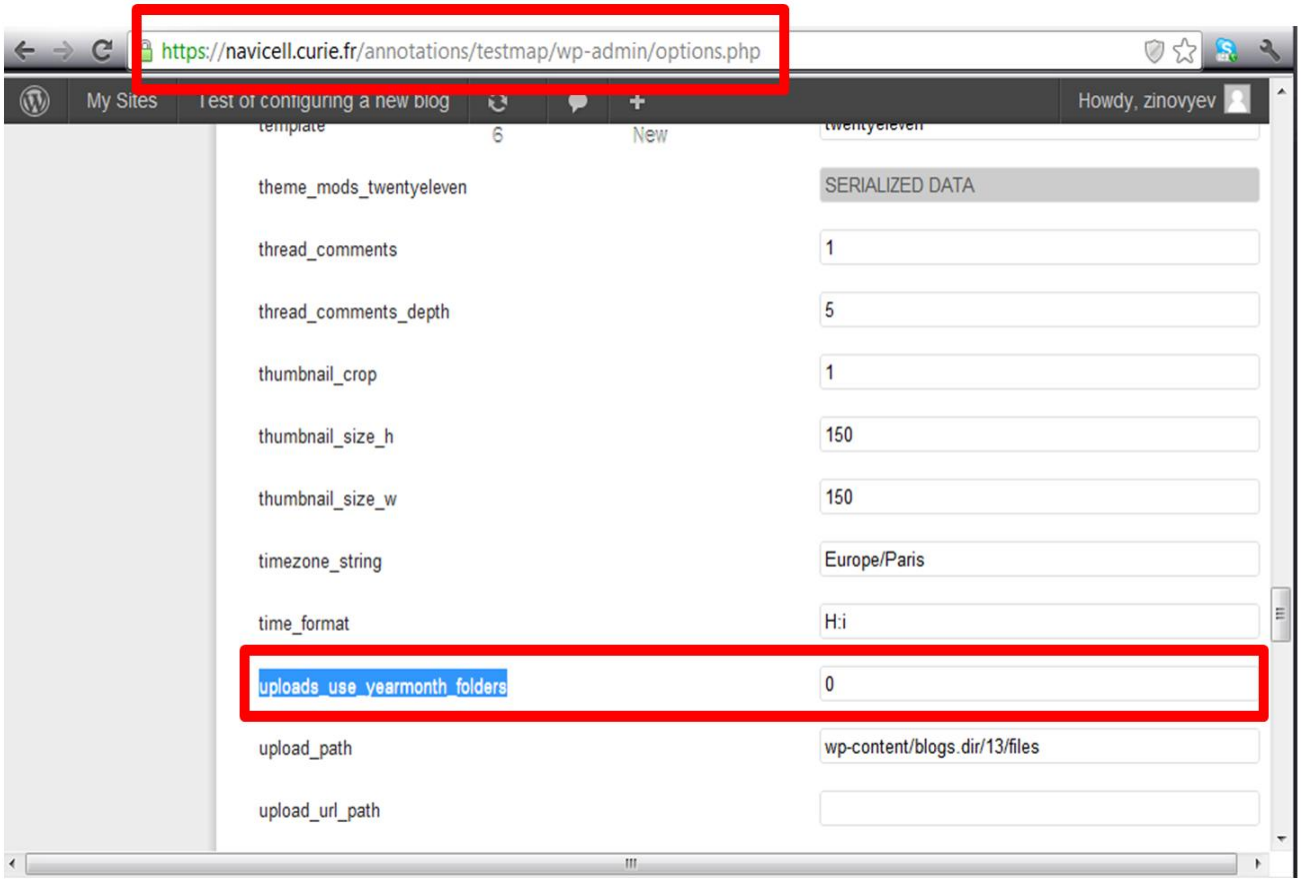
Change the default way where the files are uploaded. For this, one has to access to options through the following URL (../options.php):

https://[NAVICELL_URL]/[NAME_OF_THE_BLOG]/wp-admin/options.php .

In our case, it is

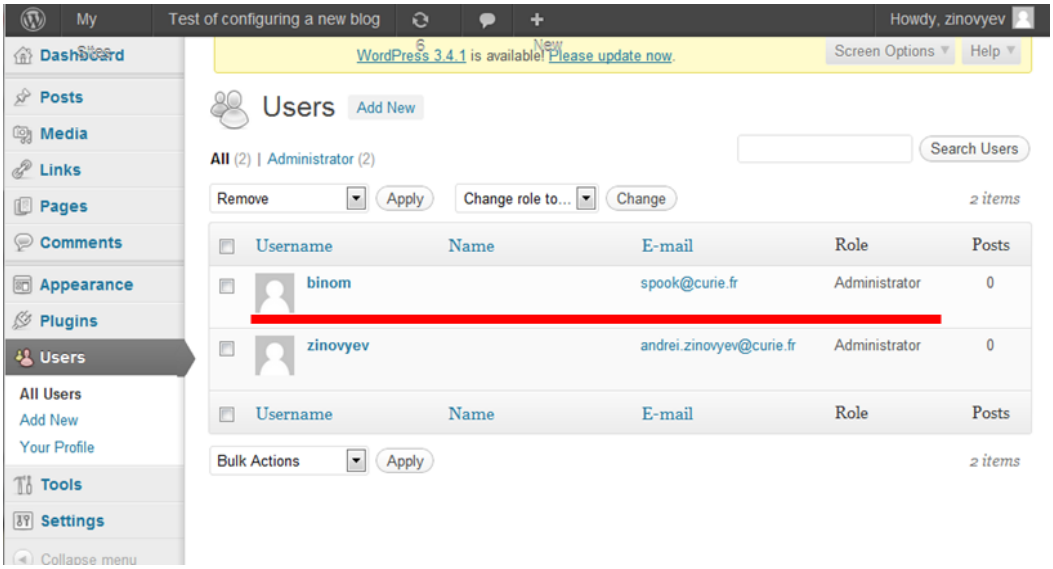
https://navicell.curie.fr/annotations/testmap/wp-admin/options.php

Change uploads_use_yearmonth_folders option to "0" value:



Step 13)

Create a new blog user with **Administrator** rights. In the example below this is "binom" user. This information will be needed for automatically filling the blog by BiNoM plugin, see the next section.



4.5 Generating Javascript code and populating the blog

4.5.1 Pre-defined folder structure for generating the map files

In order to generate a set of html files necessary to launch NaviCell locally or on the web-server, create a folder with the following sub-directories:

lib
map_icons
maps

The content of the *lib* and *map_icons* directories should contain files that can be downloaded at http://navicell.curie.fr/pages/NaviCell_lib_map_icons.zip. This should be done only once for all maps created.

The *maps* directory may contain several maps, one per a sub-directory. Each map directory with source files (xml*s* and png*s* and the *config* file) should be named [MAP_BASE]_*src* .

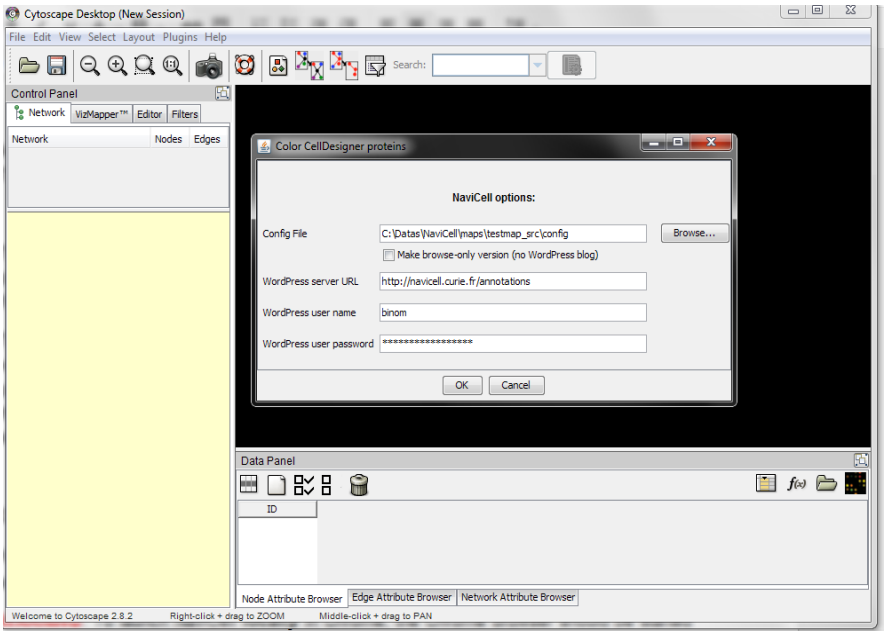
BOX 13: Example of the map directory structure

```
maps/

------- testmap_src/

------------------- config
------------------- test_master-0.png
------------------- test_master-1.png
------------------- test_master-2.png
------------------- test_master-3.png
------------------- test_master.xml
------------------- test_MITOCHONDRIA-0.png
------------------- test_MITOCHONDRIA-1.png
…
------- apoptosismap_src/

------------------- config
…
```

4.5.2 Launching BiNoM Cytoscape plugin to create the files

The latest version of Cytoscape (http://cytoscape.org) and BiNoM Cytoscape plugin (http://binom.curie.fr) should be installed first.

After starting Cytoscape, click “Plugins/BiNoM 2.0/BiNoM I/O/Produce NaviCell map files...”. The following dialog will appear (fill it accordingly):



The default option “Make browse-only version (no WordPress blog)”, when selected, will produce maps without any WordPress blog installed, only as collection of html and image files. Evidently, this version will not allow any commenting on the map.

Then click “Ok”. Generating posts for WordPress can take a while (an hour or so for a map with few hundred entities but it depends much on the WordPress server reactivity time). The progress of file generation can be seen in the command line as well as at the Dashboard of the WordPress site which is being created. By contrast, generation of the light browse-only version should be very fast. The generated maps can be put on a web-server or used locally.

4.5.3 Launching NaviCell

NaviCell is launched by opening *index.html* file in the *master* subfolder of the created map in a browser.

NOTE: To launch NaviCell **locally** in Chrome, the Chrome browser should be started with “--allow-file-access-from-files” option.