Reading Report: arKItect 2.2

Bojan Arnaudovski, University of Antwerpen

December 19, 2013

ArKItect is a domain- specific meta-modeling tool for representing and designing complex hierarchical systems. It was developed by Samuel Boutin, Joe Matta and Konstantin Smolin from the French software company Knowledge Inside in 2007. The first version of arKItect was release in 2007 and Renault and Cheuvreux were its first costumers. Since 2008, the number of costumers grew from year to year which made arKItect a crucial software tool for every system engineer/architect. Apart from the meta-modelling creation, arKItect also provides a graphical representation of the defined model and offers the possibility of generating documents for that model representation (Word, Excel, XML, HTML). Two user types are supported in arKItect: Developer and Designer. The key strengths of the Developer are: the hierarchical and graphical system representation, multi-view visualization, variants management and additional functionalities along Python scripting. In the Developer, the end user can only modify the project data and variants while on the other with the Designer, the user can access the meta-modeling, can define rules, filters, types and attributes for the formal language. The Designer has the same key strengths as the Developer and also has the advantage of defining your own modeling language and multiple views on the system. In this report I am going to focus on the Designer user type which will be later used for modeling the RPG.

The Designer Environment of the Designer user types consists of the Rules and Filters panel, the Internal Block Diagram/Relation Block Diagram/Tabular view and the palette with the object representation. The Rules and Filters are the main construction tools in the Designer.

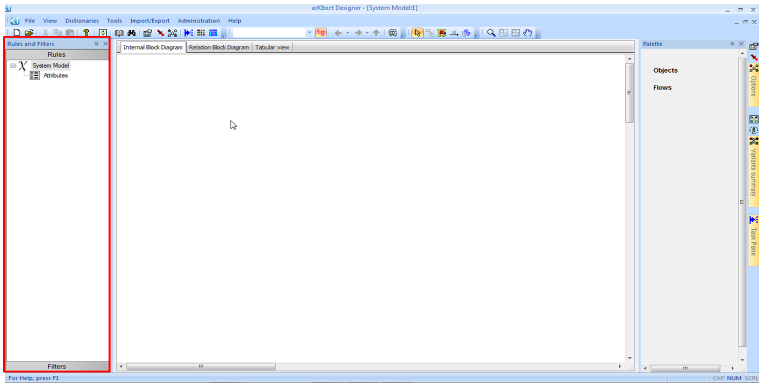


Figure 1: arKItect designer environment.

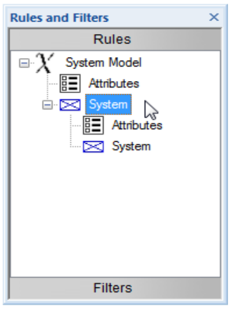


Figure 2: Rules and Filters and Palette Panel.

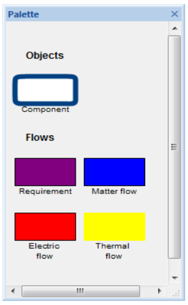
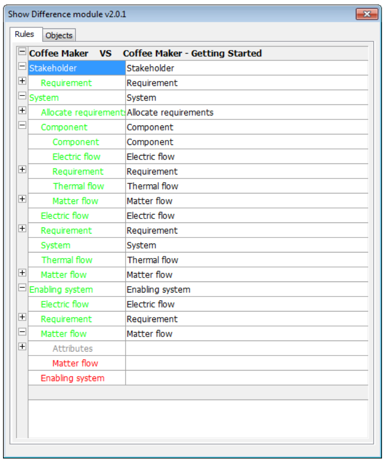


Figure 3: Palette Panel.

To begin with the meta-modeling first we begin by defining the Rules for our language together with the corresponding object (visual) representation. Aside from the objects which are the basic project entry in arKItect (having one or more attributes) there are another special sub-category of objects called, flows. The flows represent the relationship between the objects, usually defined as parent-child relation, input, output, input/output and bidirectional flow. There is also another special object in arKItect called the root. It is the only object that doesn't have any parents i.e it is not contained in another object.

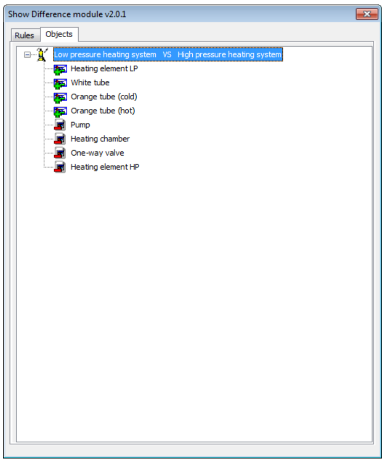
After defining the Rules and type attributes for our modeling language we continue by adding one or more filters to our formalism. The Filters allow us to create projections which describe different aspects of our model i.e describing which types are visible in the given context and which are not. The Internal Block Diagram is the main view of the objects in our project. It is related to the defined projection and to the object view in the treeview (tree-form view of our objects). The Relation Block Diagram represents the current objects and all the siblings liked with it while the Tabular view of an object, represents all of the objects children along with the attributes.

arKItect offers two tools for collaboration between users. The first tool is used for sharing a project between multiple users. The administrator grants access to users so they can both begin working on the project. If all the users don't work at the same time on this project, the modifications made by the last user will be loaded immediately the next time the project is opened. All the data with the projects are not stored locally. The other tool for collaboration is used for sending and receiving notifications and alerts. There are two ways



Coffee Maker VS	Coffee Maker - Getting Started
Stakeholder	Stakeholder
Requirement	Requirement
System	System
Allocate requirements	Allocate requirements
Component	Component
Component	Component
Electric flow	Electric flow
Requirement	Requirement
Thermal flow	Thermal flow
Matter flow	Matter flow
Electric flow	Electric flow
Requirement	Requirement
System	System
Thermal flow	Thermal flow
Matter flow	Matter flow
Enabling system	Enabling system
Electric flow	Electric flow
Requirement	Requirement
Matter flow	Matter flow
Attributes	
Matter flow	
Enabling system	

Figure 4: Show Difference in projects 1a.



Low pressure heating system VS High pressure heating system
- Heating element LP
- White tube
- Orange tube (cold)
- Orange tube (hot)
- Pump
- Heating chamber
- One-way valve
- Heating element HP

Figure 5: Show Difference in projects 1b.

of sending email notifications in arKItect. The first one is from within the tool and the second one is done via Python script which we can modify and also send the URL of the model representation. These features are similar with the collaboration options in AtoMPM except that AtoMPM also supports online chat which is not the case with arKItect.

There are three features available in arKItect that help us manage the versions of our projects. The History feature displays all the past modifications along with which types were modified and in what projection. The second feature allows us to compared one project with another where the results are color-coded. The last feature allows us to merge two different projects.

One of the most important special advanced features of arKItect is that it permits us to import and export data in the form of XML, Word, Excel and HTML. When exporting to Excel, new spreadsheet is created for every child together with all the instance objects of that type. For exporting HTML data, arKItect uses Scalable Vector Graphic (SVG) technology for rendering the diagrams in the web pages.

As I mentioned above, Python scripts can be executed in arKItect, defined as Program attributes of the objects. Worth mentioning is that arKItect saves all the data (ports, objects, labels) on a server so every time we open arKItect we have to going to the server so we can access our data. When the saved data becomes too voluminous it begins the slow down the performance of arKItect which can be seen when modifying the Internal Block Diagram of the current projection. When this is the case, arKItect premits us to change the performance settings by clearing the previous port positions and speed up the charging of diagrams.

Essential to mention is that we can add image for any arKItect object. ArKItect proposes four ways of attaching the image to the object, as a foreground or background image (graphically attached to the object and displayed in the diagram), as attached file (the object with the image is uploaded on the server where users need to downloaded it), as a link (similar as the attached file except that the object/image is not uploaded to a server but in a repository) and as an RTF attributes.

1 References:

1. ArKItect web page, http://www.k-inside.com/web/
2. ArKItect documentation, https://support.k-inside.com/display/ARKI22/arKItect+2.2.x+documentation+home