
What is Doc. Composer Template Language?

DCTL ([Doc. Composer](#) Template Language) is an XML-based language that enables the transformation of design specification into document content. DCTL comes with a well-defined structure and syntactic rules for writers to define what and how project data should be extracted from a Visual Paradigm project, and how these data should be presented in a document.

The following shows a basic template:

Name

Doc. Composer provides you with a dynamic and efficient document editing experience by letting you produce document through simple drag-and-drop. You just need to select a piece of model data, like a use case or a sequence diagram, then drag out a build-in template and drop it onto the Doc. Composer to create content.

All the build-in templates are opened for editing. To make the document more adoptable to your company's needs, you can edit a template or to design your own templates and re-use it document-by-document. For example, design a class specification by printing out the description of classes that contains "Controller" in their names.

This section is divided into two main parts. The first part is going to talk about the steps for creating a template in Doc. Composer. The second part comes with a detailed specification of template constructs you can use in constructing a template.

As the template shows, it tries to create a table, and add table rows for showing the names of use cases within a diagram. This example is a fairly simple one yet it outlines two main thing that every element template tries to achieve:

- Data retrieval - To query the use cases from a diagram, and then get the name of each use case.
- Layout of content - Table construction.

That's what you can do with a DCTL - You compose an element template with DCTL, drag the template into your document in Doc. Composer, let Doc. Composer interpret your template and output content accordingly.

In the coming sections you will see how to retrieve project data as well as to layout the content with the use of DCTL.