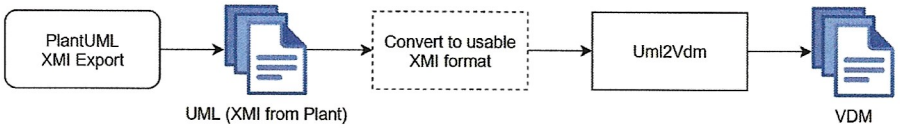

Figure 6: PlantUML to VDM Architecture Overview

#	XML	PlantUML
1	`<packagedElement xsi:type="uml:Class" name="Class1" visibility="public">`	`class Class1 {}`
2	`<ownedAttribute name="var_1" visibility="" type="//Class_1/Type">`	`var_1 : Type`
3	`<ownedOperation name="Operation" visibility=""> <ownedParameter type="Type" direction="return"/>`	`Operation() : Type`
4	`<ownedAttribute name="var_2" visibility="" type="Binding \%CClass1">`	`Class2 -right- "0..*" Class1 : -var_2`
5	`<packagedElement xsi:type="uml:Class" name="Class3" visibility=""> <generalization general="//Class1"/>`	`Class1 <|-- Class3`

Table 1: XML2PlantUML table

PlantUML Transformation Considerations

The table below highlights parts of XML that would be generated by transforming a VDM++ model into a UML following the XMI standard and using the functionality of VDM2UML.

The corresponding PlantUML syntax is then presented in the field next to the XML. This is done to showcase how an optimal translator would convert sections of the UML file into a working PlantUML diagram. The XML define the VDM++ core elements of classes, instance variables, operations and instance variables used by a class that does not define it.

The entries in the XML column of are somewhat simplified, as irrelevant information has been removed to improve legibility.

1. A class definition. All remaining elements are defined within the scope of a class, on both the XML and PlantUML side.

2. An Attribute definition. The attribute corresponds to a VDM instance variable.