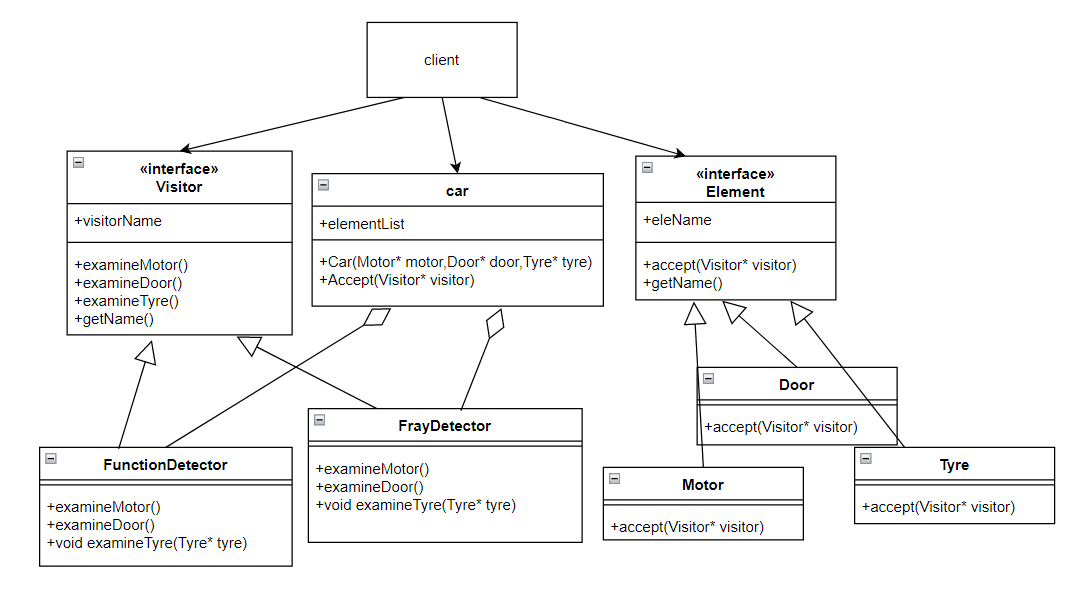
## 访问者模式

### 实现功能

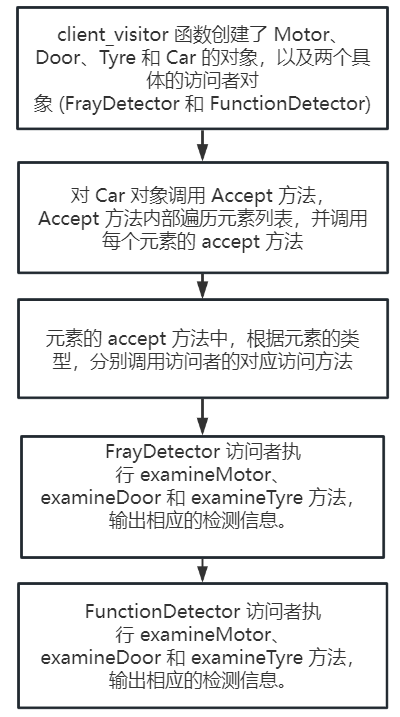
针对整体汽车和不同汽车的零件，需要进⾏质量评估和磨损评估。对于不同的评估，可能会有不同的测试者，以及不同的测试⽅法

### 类图



本图中，Element为接⼝，其具体实现为Motor、Door和Type，即组成汽车的零件。car是具体的汽车，由Motor、Door和Type组成。Visitor类是⼀个接⼝，其具体实现类是FunctionDetector和FrayDetector，两种测试者分别测试功能和损耗。通过car类的Accept函数，实现访问者的检验。

**流程图**



client\_visitor 函数创建了 Motor、Door、Tyre 和 Car 的对象，以及两个具体的访问者对象 (FrayDetector 和 FunctionDetector)。对 Car 对象调用 Accept 方法，Accept 方法内部遍历元素列表，并调用每个元素的 accept 方法。元素的 accept 方法中，根据元素的类型，分别调用访问者的对应访问方法。访问者执行操作：FrayDetector 访问者执行 examineMotor、examineDoor 和 examineTyre 方法，输出相应的检测信息。FunctionDetector 访问者执行 examineMotor、examineDoor 和 examineTyre 方法，输出相应的检测信息。

### 代价分析

#### 增加新访问者的代价：

如果需要增加新的访问者（例如，添加一个新的检测器类），需要修改所有的元素类和 Car 类，使其适应新的访问者。这可能会导致一定的代码修改工作。

#### 元素类层次结构的修改代价：

如果需要添加新的元素类（例如，添加一个新的车身元素），需要修改所有的访问者类，使其适应新的元素。这也可能导致一定的代码修改工作。

#### 元素类与访问者类之间的紧耦合：

元素类需要知道所有可能的访问者类型，并在 accept 方法中进行相应的调用。这导致了元素类与访问者类之间的紧耦合。如果访问者类的修改频率较高，这种紧耦合可能导致代码的脆弱性。