Information Management I - Group Project Report

Group 18 - Car Insurance Broker

1. **Introduction**

As part of this project, we were tasked with creating some XML documents describing some classes from our class diagram. We also had to create interesting XQuery queries based of our XML documents based of the use case diagrams which we made in the previous project.

1. **Did we make any design changes to our UML diagrams to suit the XML format?**

No, we found that the design of our UML diagrams fit well into our XML design and as a result, no changes were made to the diagrams when writing our DTD and XML files.

1. **Who did what**

After having presented our UML diagrams, we met up the next week and divided the work between the members of the group that were constantly present in our weekly meetings. We then met up the next week with the DTDs and XML files done, that is when we divided the work for the different queries amongst ourselves.

Samuel Petit : I personally worked on the Vehicle DTD and XML file. I also wrote both the queries that use built in xquery functions.

Everlyn Poh : I worked on the Company XML and DTD files and wrote 3 of the interlinked XQueries.

Conor Ryan: I worked on the Claim and QuoteSystem XML and DTD files and wrote the user-defined xquery.

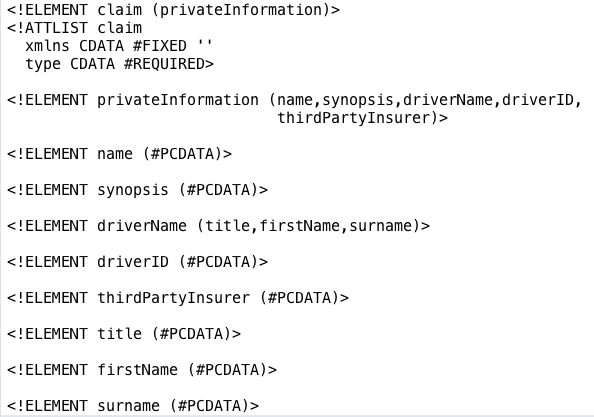
Kamil Przepiórowski : I worked on the Driver XML and DTD files, and wrote the “penalty points info” xquery.

Sean Roche : I worked on the Policy XML and DTD files and wrote the xquery which uses the LET function. I also assessed and documented the strengths of XML and Xquery design.

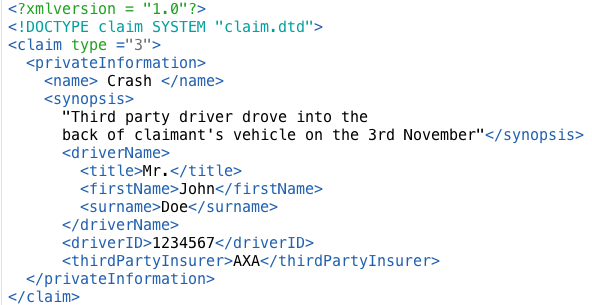
Con Óg Ó Laoghaire: I wrote Customer Support.xml and analysed and wrote the weaknesses of our XML approach

1. **XML and DTD files**
2. **Claims**

* The DTD file:

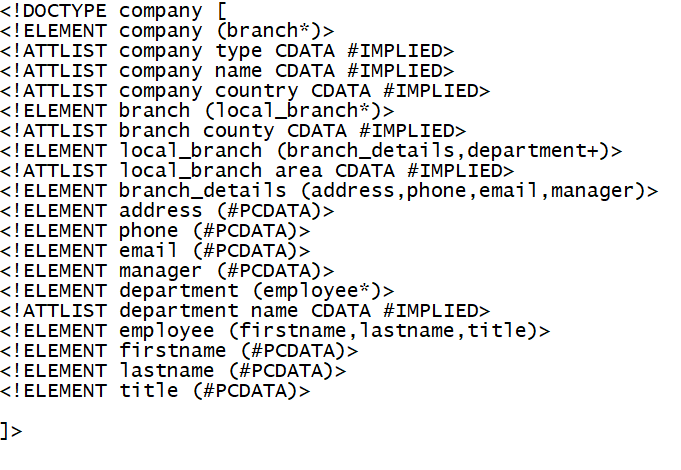


* The XML file:



1. **Company**

* The DTD file :

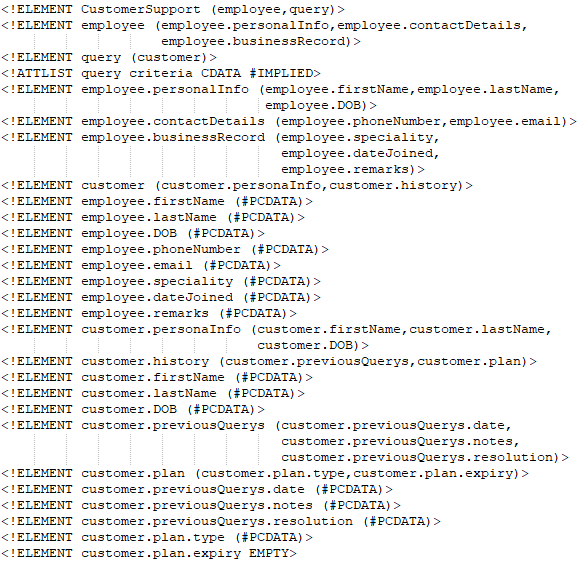


* The XML file



1. **Customer Support**

* The DTD file:

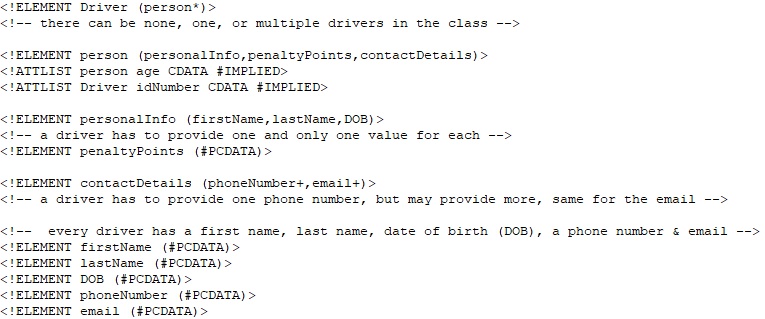


* The XML file:

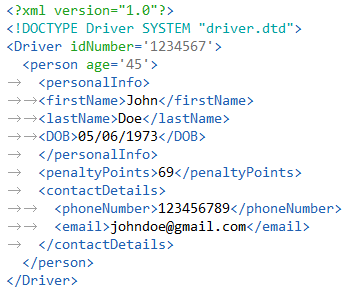


1. **Driver**

* The DTD file :

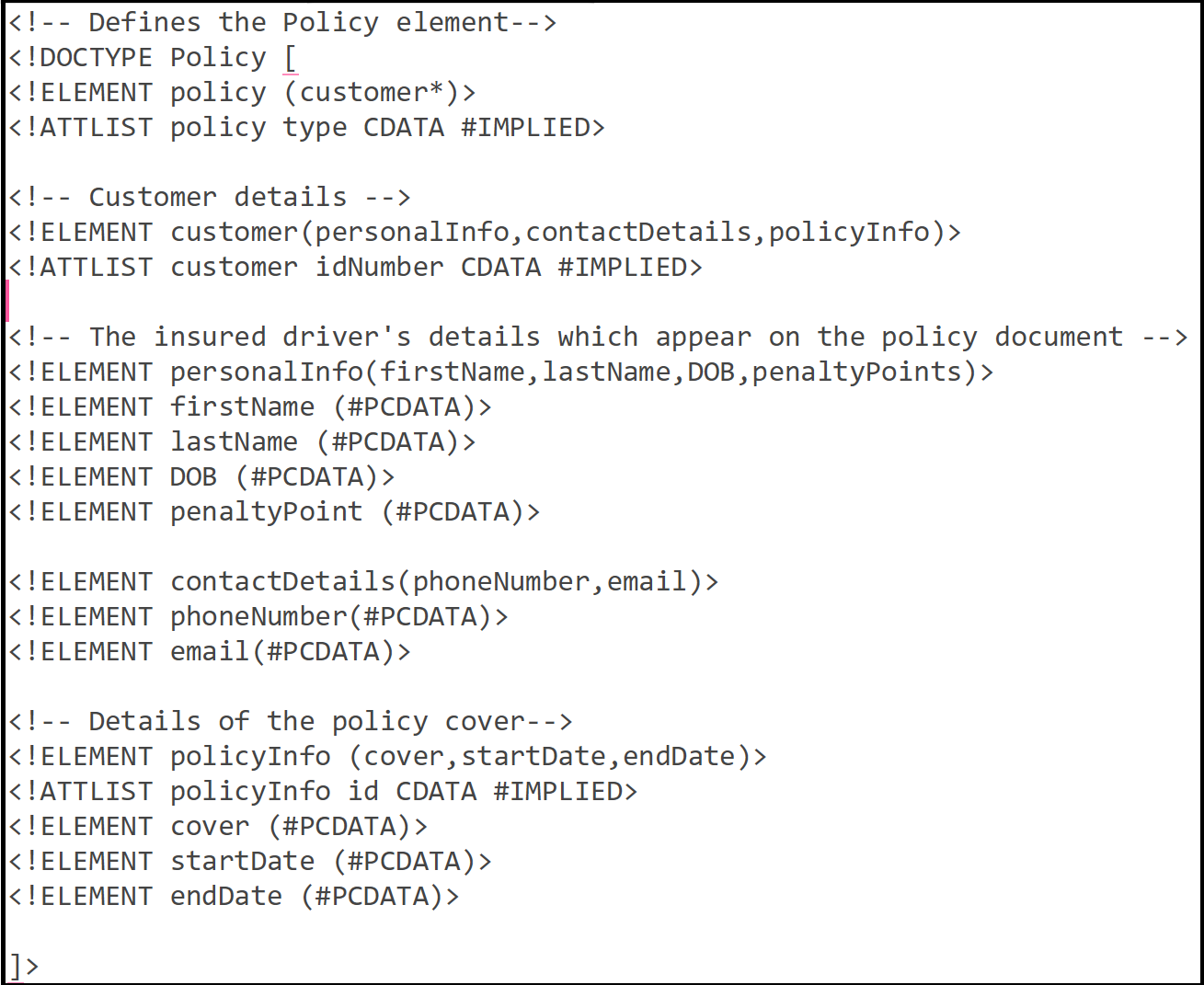


* The XML file :



1. **Policy**

* The DTD file:

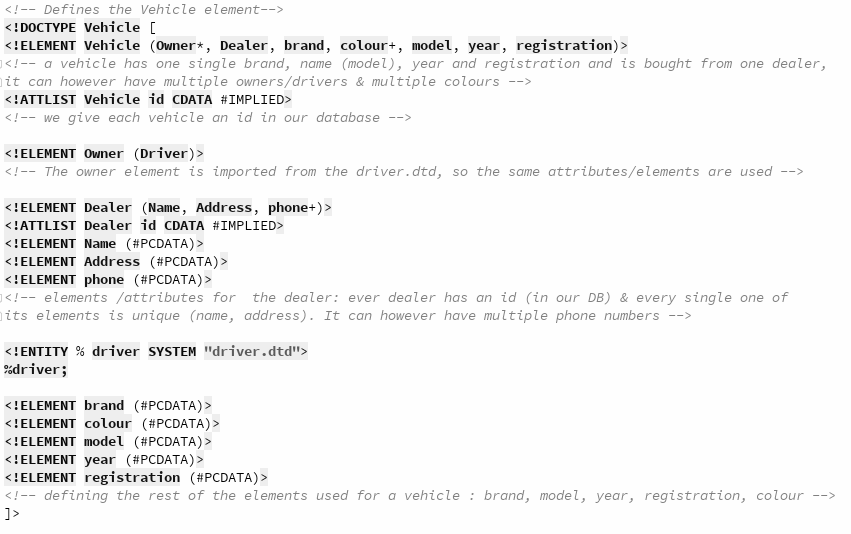


* The XML file:

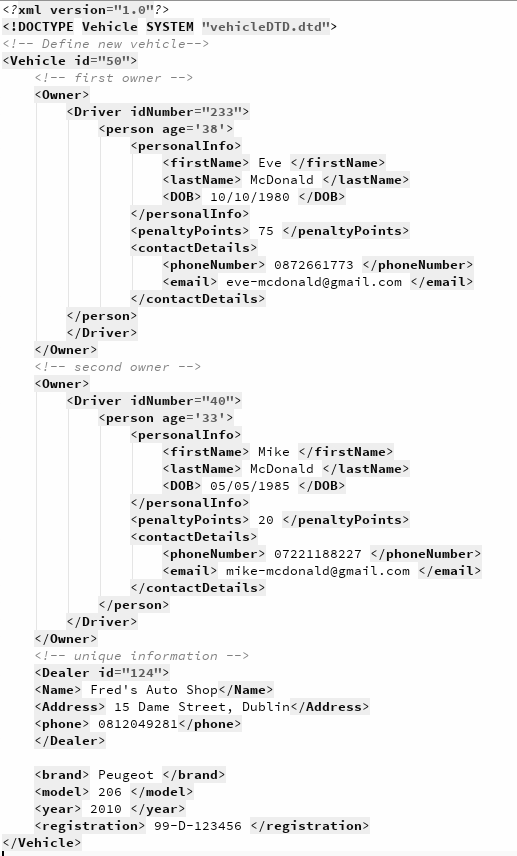


1. **Vehicle**

* The DTD file :



* The Vehicle XML file :

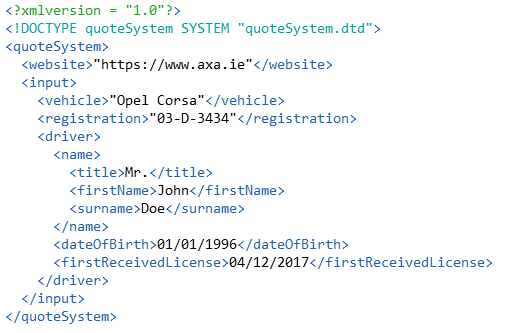


1. **Quote System**

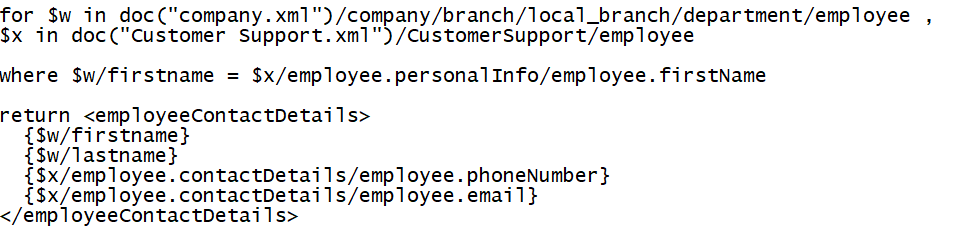
* The DTD file:



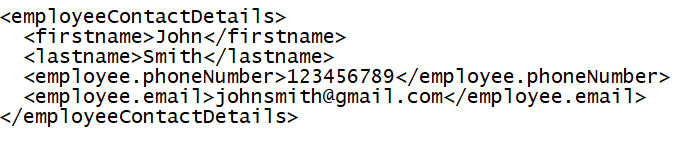
* The XML file:



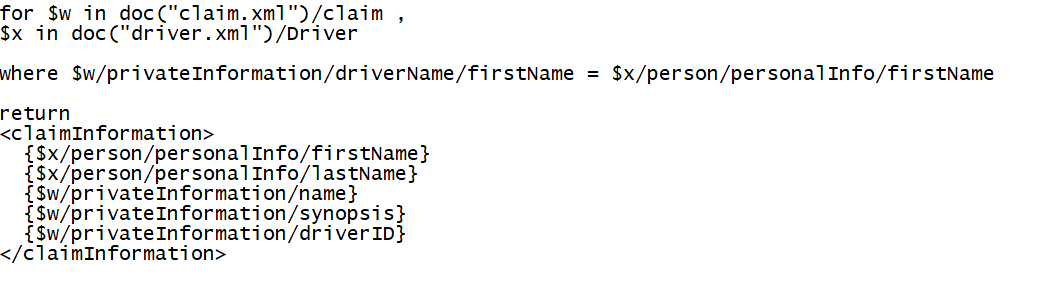
1. **XQueries**
2. **Query 1 (Interlinked)**



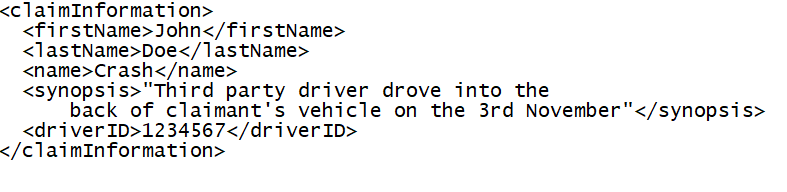
* Identification of UML Case that it supports : Company class and Customer Support Class.
* Purpose of Query : To retrieve contact details of an employee from another interlinked XML file by searching their name.
* Example of output when query is executed :



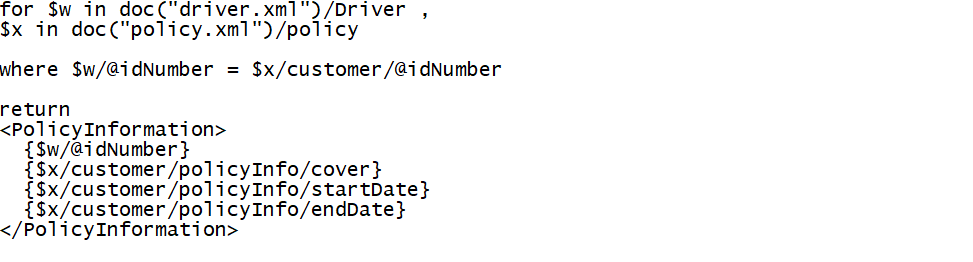
1. **Query 2 (Interlinked)**



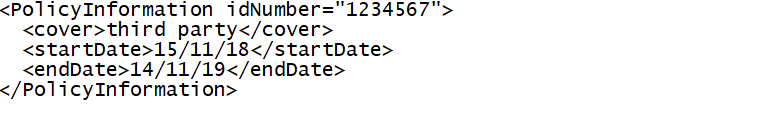
* Identification of UML Case that it supports : Claim class and Driver Class.
* Purpose of Query : To retrieve claim information of a driver from another interlinked XML file by searching the driver’s name.
* Example of output when query is executed :



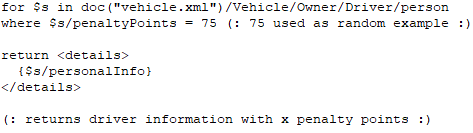
1. **Query 3 (Interlinked)**



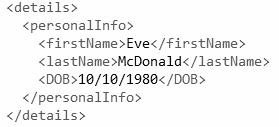
* Identification of UML Case that it supports : Policy class and Driver Class.
* Purpose of Query : To retrieve policy information of a driver from another interlinked XML file by searching the driver’s customer ID number.
* Example of output when query is executed :



1. **FOR query**

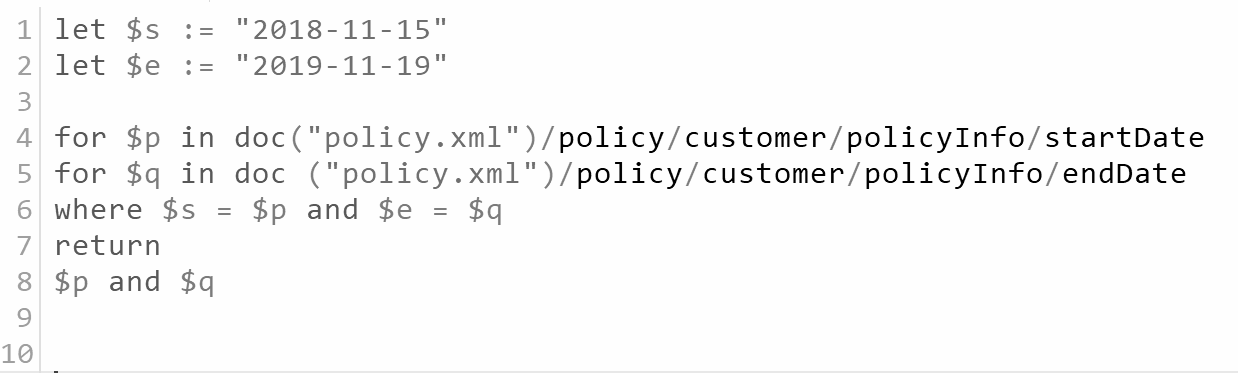


* Identification of UML Case that it supports : Registering / Getting a quote
* Purpose of Query : To retrieve the information of a Driver with x amount of penalty points.
* Example of output when query is executed : If there exists a driver in the database, with x penalty points, the output will be:

****

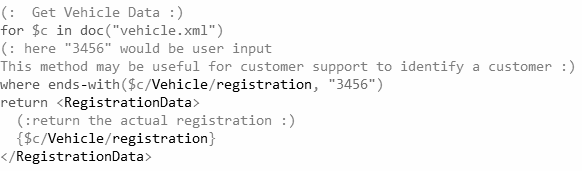
* Otherwise, the query will not return anything as the driver doesn’t exist.

1. **LET query**

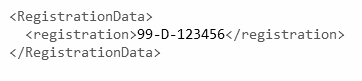
****

* This query may be useful when a claim is being made to check its validity. It could be implemented to ensure that the policy is active and the vehicle is covered for the the date of the claim or incident.
* Therefore, this query supports our UML Use case #1 which was “Making a Claim”.
* This query has two potential outputs; true or false.

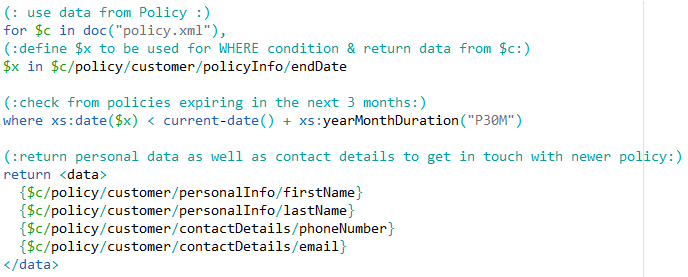
1. **Built in function query #1**



* This query may be useful for customer support when dealing with a customer. The employee may ask for the customer to provide certain information to verify that he is indeed who he claims to be before processing the customers query.
* As a result of the above explanation, this query illustrates part of our UML Use case #1 which was “Making a Claim”. But it also is relevant for our 3rd UML Use case : “Customer Service”.
* Here is a sample output when this specific query is executed:



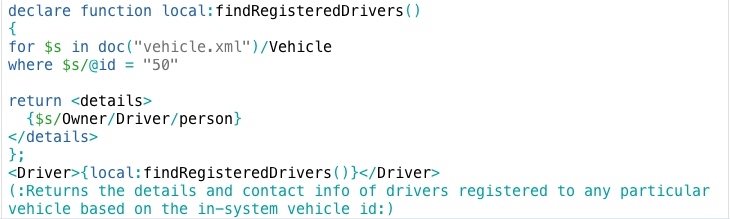
1. **Built in function query #2**



* This query would be used for renewing customer policies before they expire. It returns data and contact details from every customer with policies expiring in the next 3 months from when the query is executed. A computer system could make use of this query, run it once a day or so and get in touch with those customers the query returns.
* As a result of the above explanation, this query is useful for our 2nd UML Use case : Insurance Renewal System.
* Here is a sample output of this query :



1. **User defined function query**



* This query could be used for finding the drivers on any vehicle that is involved in a claim, or if a driver’s policy may be changing.
* Example of output:



1. **Strengths and Weaknesses of the XML design and the XQueries design.**

**Strengths**

* Relatively straightforward to write, it is much like HTML however it has more advanced linking abilities and does not use acronyms which increases it’s understandability.
* It is easily extendable.
* It is essentially a human language as opposed to a computer language and therefore can be easily understood by anyone.
* Allows for quick prototyping.
* XQueries allow for quick lookup of data in a database in a relatively simple manner

**Weaknesses**

* Information portrayed in a very raw manner ie. impossible to write or test without knowledge of XML and XPath
* Layout style dependant on who is writing it
* Can be overwhelming to someone who isn’t familiar with it