Extract Method:

‘Duplicated Code’ code smell was detected in several methods in Network class.

Accounting code was duplicated in printDocument method and logging code was duplicated in requestWorkstationPrintsDocument and requestBroadcast methods.

These code smells were fixed by extracting methods inside the same class, and JUnit tests were re-executed.

Move Behavior Close to Data:

‘Feature Envy’ code smell was detected in the methods extracted above. Node and Packet classes were mainly being used as data classes, and methods belonging to other classes were accessing this data.

Node logging code was moved to Node class.

The extracted printDocument() code in Network class had a lot of complicated logic for checking the type of document and for retrieving author and title.

We moved these checks to Packet class (Move method) and replaced them by getMessageAuthor() and getMessageTitle() methods. Now that excessive use of Packet data attributes was reduced, we moved the printDocument function in a new PrinterNode class (so that HTML and XML related printing logic can be eventually added to this class).

Eliminate Navigation Code:

‘Duplicated Code’ smell was detected in Network class in requestWorkstationPrintsDocument and requestBroadcast methods for locating the current node for printing.

We extracted this code into a separate method and changed it to a recursive call.

Transform Self Type Checks:

‘Switch case’ code smell was detected at multiple places, as node type was being checked using switch statements.

We created individual sub classes and used polymorphism to invoke these newly created methods.

‘Feature Envy’ code smell was detected. The printDocument method in Network is moved to the Node and Packet class respectively.