Personal Agents – Servlets:

Each personal agent in the Rule Responder system is implemented via a Java servlet. However, there is a major difference between the SymposiumPlanner and WellnessRules instantiations.

- **SymposiumPlanner Servlets:**
  SymposiumPlanner servlets are much less complex, yet more unorganized than their WellnessRules counterparts. They have the following format:

  MainClass

  MessageGenerator *MessageParser* *QueryBuilder*

  MainClass: This Class handles the incoming query by converting it to POSL, executing POSL on it, and converting the answer(s) to RuleML before sending them back to the OA. It does so via the three supporting classes:

  MessageGenerator: Generates messages for HTTP transfer from the provided RuleML. There is no particular format in the main class itself. It calls upon OO jDREW after first retrieving its knowledge base.
  MessageParser: Parses the RuleML query.
  QueryBuilder: Helps convert RuleML to an object oriented format.

  ** Please see the documentation within the code for detailed information**

- **WellnessRules Servlets:**
  Like SymposiumPlanner, WellnessRules servlets consist of a main class and many supporting classes. However, the main class and supporting classes are now capable of both OO jDREW and Euler executions.

  MainClass

  MessageGenerator *MessageParser* *QueryBuilder*

  Now introduces new supporting classes:

  GeneralHandler *N3Handler* *POSLHandler*

  GeneralHandler*:* This class handles general method calls for PA's such as retrieving the time and date, and determining a PA's responsible profiles and user's requested profile.

  N3Handler*:* This class handles N3 method calls for PA's such as generating N3 profiles parsing a RuleML query, and answering an N3 query with Euler EYE.

  POSLHandler*:* This class handles POSL method calls for PA's such as generating POSL profiles parsing a RuleML query, and answering a POSL query with OO jDREW.

  ** Please see the documentation within the code for detailed information**