

Pellet Reasoner Code



The Reasoner Manager



- To get an instance of Protégé Reasoner for an OWL model: `getInstance()` .
- To obtain an instance of a reasoner: `createProtegeReasoner(OWLModel owlModel, Class reasonerJavaClass)` should be used. “`reasonerJavaCode`” is the Java class of the reasoner for which you would like to obtain an instance.
- Reasoner implementations:
 - **ProtegePelletJenaReasoner**: provides a direct connection to the Pellet reasoner accessed through the Jena API.
 - **ProtegePelletOWLAPIReasoner**: provides a direct connection to the Pellet reasoner accessed through the OWL-API.

Getting a Pellet reasoner instance (Jena)



- The following code shows how to get a Pellet reasoner instance that is accessed through Jena:

```
public ProtegeReasoner createPelletJenaReasoner(OWLModel owlModel) {  
    // Get the reasoner manager instance  
        ReasonerManager reasonerManager = ReasonerManager.getInstance();  
  
    //Get an instance of the Protege Pellet reasoner  
        ProtegeReasoner reasoner = reasonerManager.createProtegeReasoner(owlModel,  
        ProtegePelletJenaReasoner.class); return reasoner;  
}
```

Getting a Pellet reasoner instance (OWL-API)



- The following code shows how to get a Pellet reasoner instance that is accessed through OWL-API:

```
public ProtegeReasoner createPelletOWLAPIReasoner(OWLModel owlModel) {  
    // Get the reasoner manager instance  
        ReasonerManager reasonerManager = ReasonerManager.getInstance();  
    //Get an instance of the Protege Pellet reasoner  
        ProtegeReasoner reasoner = reasonerManager.createProtegeReasoner(owlModel,  
        ProtegePelletOWLAPIReasoner.class);  
    return reasoner;  
}
```

Querying the reasoner for inferred information



- Once a reasoner instance has been obtained, the reasoner can be queried for information about the ontology.

```
// Get the VegetarianPizza OWLNamedClass from the OWLModel
```

```
    OWLNamedClass vegetarianPizza = owlModel.getOWLNamedClass("VegetarianPizza");  
    if(vegetarianPizza != null) {
```

```
// Get the number of asserted subclasses of VegetarianPizza
```

```
    Collection assertedSubclasses = vegetarianPizza.getNamedSubclasses();
```

```
    System.out.println("Number of asserted VegetarianPizzas: " + assertedSubclasses.size());
```

```
// Now get the inferred subclasses of VegetarianPizza
```

```
    Collection inferredSubclasses = reasoner.getSubclasses(vegetarianPizza);
```

```
    System.out.println("Number of inferred VegetarianPizzas: " + inferredSubclasses.size());
```

```
    System.out.println("VegetarianPizzas:");
```

```
    for(Iterator it = inferredSubclasses.iterator(); it.hasNext();) {
```

```
        OWLNamedClass curClass = (OWLNamedClass) it.next();
```

```
        System.out.println("\t" + curClass.getName()); }
```

Pellet Example



- Example
- <http://protege.stanford.edu/plugins/owl/download/ProtegeReasonerExample.java>

Debugging: Pellet Test with Swoop



```
package org.mindswap.swoop.reasoner;  
  
import java.io.PrintWriter;  
import java.io.Writer;  
import java.net.URI;  
import java.net.URISyntaxException;  
  
import org.semanticweb.owl.impl.model.OWLConnectionImpl;  
import org.semanticweb.owl.io.owl_rdf.OWLRDFErrorHandler;  
import org.semanticweb.owl.io.owl_rdf.OWLRDFParser;  
import org.semanticweb.owl.model.OWLException;  
import org.semanticweb.owl.model.OWLOntology;  
import org.semanticweb.owl.util.OWLConnection;  
import org.semanticweb.owl.util.OWLManager;  
import org.xml.sax.SAXException;
```

Debugging: Pellet Test with Swoop



```
public class PelletTest {  
    public static void main(String[] args) throws Exception {  
        PelletTest pt = new PelletTest();  
        OWLOntology ont = pt.loadOntology(new URI("http://ontology.owl"));  
        System.out.println("done parsing ontology");  
        pt.classify(ont);  
    }  
    private void classify(OWLOntology ont) {  
        final OWLOntology onto = ont;  
        new Thread() {  
            public void run() {  
                PelletReasoner reasoner = new PelletReasoner();  
                try {  
                    System.out.println("starting classification");  
                    reasoner.setOntology(onto);  
                } catch (OWLException e) {  
                    e.printStackTrace();  
                }  
            }  
        }.start();  
    }  
}
```


Debugging: Pellet Test with Swoop



```
private OWLOntology loadOntology(URI uri) throws Exception {
    OWLRDFParser parser = new OWLRDFParser();
    parser.setOWLRDFErrorHandler(new OWLRDFErrorHandler() {
        public void owlFullConstruct(int code, String message)
            throws SAXException {

        }
        public void error(String message) throws SAXException {
            throw new SAXException(message.toString());
        }
        public void warning(String message) throws SAXException {
            System.out.println("RDFParser: " + message.toString());
        }
        public void owlFullConstruct(int code, String message, Object obj) throws SAXException
    {

    }
    });
    OWLConnection connection = new OWLConnectionImpl();
    parser.setConnection(connection);
    return parser.parseOntology(uri);}
}
```