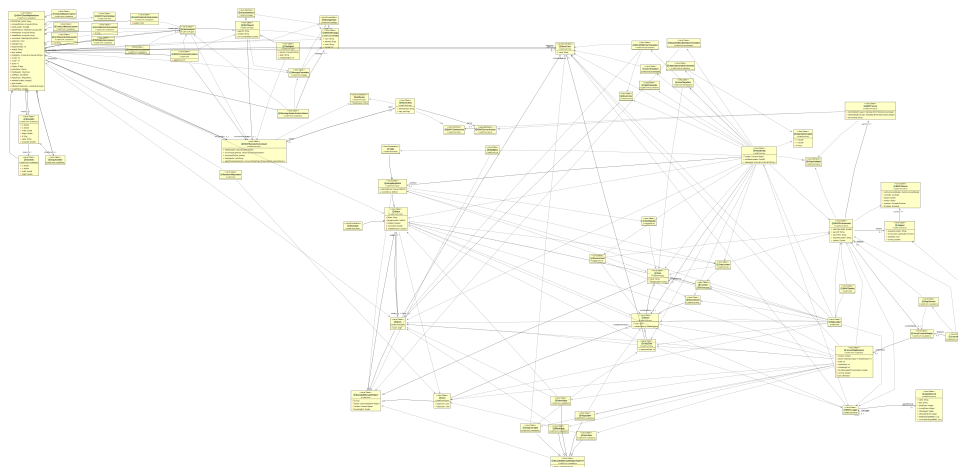


BW4T - Refactoring

May 19, 2014

1 Hoe het nu is



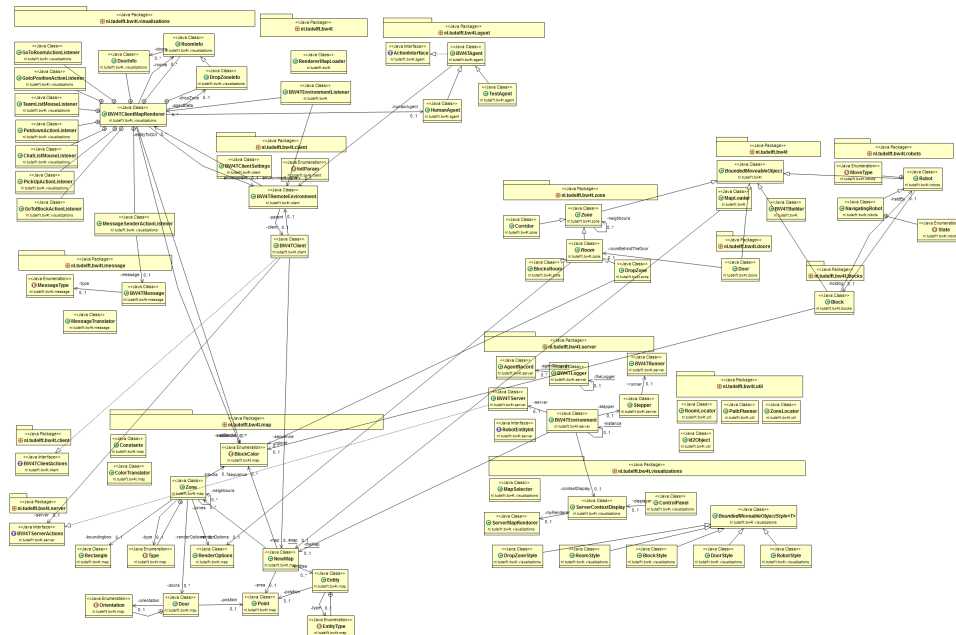
Zoals in het UML diagram te zien is zijn veel klassen met elkaar verbonden. De documentatie is erg matig en er zijn geen testen. Deze factoren zorgen ervoor dat het heel lastig is om de code te begrijpen en te debuggen. Het is lastig uit te vinden wat een stukje code doet, wat de functie van een methode is, wat het verschil is tussen bepaalde methodes en klassen en wat de connectie is tussen stukken code.

In het huidige systeem zit Repast, maar veel functies die ook in Repast zitten worden apart geïmplementeerd. Hier zorgt Repast ervoor dat de totale size van de systeem erg groot is (500 mb voor Repast alleen).

In het huidige systeem zit een client en server klasse, maar de client en server worden niet consistent gebruikt. De client en server zitten niet los van elkaar, en er zijn klassen die met elkaar communiceren buiten de client en server om terwijl ze dit niet zouden moeten doen bij een client-server architecture.

Het opstarten van het systeem is nu ook niet heel gebruiksvriendelijk, de client en server moeten apart opgestart worden. Zie ook bijlage voor voorbeelden van 'slechte' code en comments.

2 Wat hebben we tot nu toe bereikt



Repast is nu een plugin en de bijbehorende jar files zijn uit het systeem, wat veel minder ruimte kost (nu nog maar 8mb). De plugin is de nieuwste versie van Repast.

We zijn bezig geweest met het splitsen van de server, client en een core. In core zitten de klassen die zowel door de server en de client gebruikt wordt. Elk van deze onderdelen is nu een apart Maven project. Met deze structuur hopen wij meer overzicht te krijgen in de code en dat het voor andere mensen ook (snel) duidelijk is hoe het systeem in elkaar zit. Verder is het testen van het systeem met deze structuur een stuk makkelijker.

3 Repast

Repast is zeker een goed platform voor agent-based modeling. Echter in dit project niet heel nuttig, dit omdat GOAL totaal onafhankelijk werkt van Repast. Het is niet mogelijk om te zorgen dat GOAL en Repast goed synchroon lopen zonder GOAL daarop aan te passen. Wij zijn van mening dat Repast alleen gebruikt kan worden als het systeem helemaal opnieuw wordt opgebouwd en er aanpassingen aan GOAL gemaakt worden.

Repast erin laten zitten heeft voor ons op dit moment geen toegevoegde waarde, alles wat Repast biedt moet in dit systeem net anders gebruikt worden. De klassen en methodes moeten dan toch op een eigen manier worden geïmplementeerd. Dit geldt ook voor eventuele toekomstig functies van Repast, deze kunnen heel nuttig zijn voor systemen zoals BW4T maar alleen als deze hierop gebouwd zijn.

4 Wat kunnen wij aankomende weken nog bereiken

Wat wij de aankomende weken nog kunnen doen is zorgen dat het opstarten van het systeem makkelijker is. Dus dat de server en client niet apart opgestart hoeven te worden, maar dat er een launcher is die alles voor je doet. Ook zullen we de client-server architecture verder uitwerken. Hierbij willen we proberen om het systeem te versimpelen door overbodige connecties tussen klassen te verwijderen of om te zetten naar client-server communicatie. Ook zullen we de documentatie van het systeem proberen te verbeteren.

Een ander punt waar we aan gaan werken is het schrijven van testen. Dit zullen voornamelijke functionaliteitstesten zijn.

5 Wat zou er kunnen gebeuren

Als we meer tijd en ervaring zouden hebben zouden we het hele systeem omgooien. Zodanig dat de server-client architecture nog beter wordt geïmplementeerd, de documentatie duidelijk is, en alles in het systeem goed getest is. Bij een langerlopend project is het te overwegen om alsnog Repast te gaan gebruiken. Dit zou dus betekenen dat er ook goed naar de link tussen GOAL en Repast gekeken moet worden.

Het is dan ook heel belangrijk dat het overzicht bewaart blijft, dat er kritisch naar de code wordt gekeken zodat het systeem optimaal werkt.

Bijlagen Deze bijlage bevat een aantal voorbeelden van 'slechte' code en comments. Het zijn situaties die we willen verbeteren en voorkomen.

Lege methodes

```
public void execute(RunState toExecuteOn) {  
    // required AbstractRunner stub. We will control the  
    // schedule directly.  
}
```

Twee methodes die op elkaar lijken maar niet duidelijk het verschil wordt aangegeven.

```
/**  
 * Free an agent on the server  
 *  
 * @param agent  
 *         , the agent to free  
 * @throws RemoteException  
 *         , if an exception occurs during the execution of a remote  
 *         object call  
 * @throws RelationException  
 *         , if an attempt to manipulate the agents-entities-relation  
 *         has failed.  
 */  
public void freeAgent(String agent) throws RemoteException,  
    RelationException {  
    server.freeAgent(agent);  
}  
  
/**  
 * Unregister an agent on the server  
 *  
 * @param agent  
 *         , the agent to unregister  
 * @throws AgentException  
 *         , if an attempt to register or unregister an agent has  
 *         failed.  
 * @throws RemoteException  
 *         , if an exception occurs during the execution of a remote  
 *         object call  
 */  
public void unregisterAgent(String agent) throws AgentException,  
    RemoteException {  
    server.unregisterAgent(agent);  
}
```

Hacks

```
BW4TRunner runner; // HACK should be private.  
    .  
    .  
    .
```

Uitgecommenteerde code, zeker bij een eindversie is het netter om dit eruit te halen

```
public void init(Map<String, Parameter> parameters)
    throws ManagementException {
    // System.out.println("re-initializing repast simulator");
    setState(EnvironmentState.INITIALIZING);
    takeDownSimulation();

    Parameter map = parameters.get("map");
    if (map != null) {
        mapName = ((Identifier) map).getValue();
    }
    try {
        launchRepast();
    } catch (IOException e) {
        throw new ManagementException("launch of Repast failed", e);
    } catch (ScenarioLoadException e) {
        throw new ManagementException("launch of Repast failed", e);
    } catch (JAXBException e) {
        throw new ManagementException("launch of Repast failed", e);
    }

    setState(EnvironmentState.RUNNING);
    // System.out.println("re-initialised");
}
```

Er wordt niet gechecked of er inderdaad aan de voorwaarde wordt voldaan dat alle entities en agents ontbonden zijn

```
/**
 * kill the client interface. kill at this moment does not kill the server,
 * it just disconnects the client. Make sure all entities and agents have
 * been unbound before doing this.
 *
 * @throws RemoteException
 * @throws NotBoundException
 * @throws MalformedURLException
 */
public void kill() throws RemoteException, MalformedURLException,
    NotBoundException {
    server.unregisterClient(this);
    Naming.unbind(bindAddress);
    UnicastRemoteObject.unexportObject(this, true);
}
```

Niet getest, op de gok iets proberen

```

/**
 * This class implements the repast {@link Runner}. This handles the calls to
 * the repast stepping - when do bots move, how often, and pausing etc. This is
 * modified copy of TestRunner_2 (see #2009 and #2236). I did not give this much
 * thought, I just plugged it in and it did what I hoped for - being able to
 * control Repast. Otherwise, scheduling/running is not done here at all, but
 * from the {@link BW4TEnvironment} directly by calling {@link #step()}.
 *
 * @author W.Pasman 11mar13
 */
public class BW4TRunner extends AbstractRunner {

```

Zou gedaan moeten worden, maar is dus niet gedaan

```

public void runInitialize() {
    Parameters params = new Parameters() {
        /**
         * This should be changed. Temporary fix in order to be able to run BW4T.
         */
        @Override
        public void setValue(String paramName, Object val) {
            // TODO Auto-generated method stub
        }

        @Override
        public boolean isReadOnly(String paramName) {
            // TODO Auto-generated method stub
            return false;
        }

        @Override
        public String getValueAsString(String paramName) {
            // TODO Auto-generated method stub
            return "asd";
        }
    }
}

```