

Additional constant variables

Odysseus allows to implement additional constant variables. For this, the developer has to implement the interface `IReplacementProvider` from the bundle `de.uniol.inf.is.odysseus.script.parser`. The implementation has to be provided as a service, then. The interface is as follows:

```
public interface IReplacementProvider {
    public Collection<String> getReplacementKeys();
    public String getReplacementValue( String replacementKey );
}
```

The method `getReplacementKeys()` has to return a collection of keys which the provider wants to offer. This will be called each time before a `OdysseusScript-File` is executed (so the list of available keys can change during run-time, if needed). The second method `getReplacementValue()` finally has to return the value for the given key. It is expected that since the provider provides the key, a value non-null is returned. **If the second method return null, OdysseusScript uses an empty string ("").** The method is called each time the key is used inside one `OdysseusScript` file.

Example

The developer wants to add the current date as a constant variable. The class could be implemented as follows:

```
// Example how to add constant variables
// Caution: This class has to be provided as a service!
public class DateReplacementProvider implements IReplacementProvider {

    // We provide one key here to get the currentDate in OdysseusScript
    @Override
    public Collection<String> getReplacementKeys() {
        List<String> keys = new ArrayList<String>();
        keys.add("currentDate");
        return keys;
    }

    // Each time, "currentDate" is used in one OdysseusScript-File, this
    // method is called
    @Override
    public String getReplacementValue( String replacementKey ) {
        // obsolete since we provide only one key
        if( replacementKey.equalsIgnoreCase("currentDate") ) {
            return new Date().toString();
        }

        return "";
    }
}
```