Extend Selenium Grid Video Recording Capability

Hello, in the previous article, I described **how to extend Selenium Grid**. You can reach the tutorial in this link. Today, I am going to add **Video Recording capability** into our **Selenium Grid**. Let's get started Selenium Grid Video Recording Capability.

Create a Recorder for Selenium Grid Video Recording Capability

I am using **monte-repack library** to record videos. You can take a look at the source code in our repository. The implementation might differ from your need. That's why I don't want to get into its details. Maven dependency that I used is:

```xml
<dependency>
 <groupId>com.pojosontheweb</groupId>
 <artifactId>monte-repack</artifactId>
 <version>1.0.1</version>
</dependency>
```

Changing the Proxy Implementation

I want my proxy to start recording in case there are any custom desired capabilities about recording. So I add a String value for recording request. I will check its value and start the recording.

```java
private static String RECORD_VIDEO = "rec
```

The code below will get the value of RECORD_VIDEO capability from the TestSession object.

```java
record = (Boolean)session.getRequestedCap
```

Then recording will start.

```java
@Override
public void beforeSession(TestSession ses
    record = (Boolean)session.getRequeste
    if (record) {
        screenRecorder = new Recorder();
        screenRecorder.startScreenRecorde
        System.out.println("Video Record
    }
}
```

After the session ends, we need to stop recording by using the below code.

```java
@Override
public void afterSession(TestSession sess
    System.out.println("Selenium Extendi
    if(record)
        screenRecorder.stopScreenRecorde
}
```

How to Modify Your Desired Capabilities?

Add your desired capabilities in DesiredCapabilities object as below. Then run your test, and see your Grid records video like a charm.

We create the RemoteWebDriver with our newly added Desired capability in Before Annotation. By this way, every time a driver is created a new session will be created as well.

```java
@Before
public  void beforeClass(){
    DesiredCapabilities caps =  DesiredCa
    caps.setCapability("recordVideo",true
```

```
    try {
        driver = new RemoteWebDriver(
                new URL("http://localhost
                caps);
    } catch (MalformedURLException e) {
        e.printStackTrace();
    }
}
```

You can implement anything in the test.

```java
@Test
public void test(){

//Test Implementation
}
```

We should use quit() method of WebDriver in JUnit's After function. By doing so Selenium terminates the session and video recording is terminated.

```java
@After
public  void afterClass(){
   driver.quit();
}
```

Important Note:

Selenium Default Proxy doesn't know if a new test is started or not. It only knows if there's a new session or not. You should implement your new functions according to this fact.