

## 1 Een simple GUI voor het weergeven van XML-data

In deze opdracht willen we je laten oefenen met het ontwerpen en implementeren van een applicatie met een eenvoudige Grafisch User-Interface. Verder leer je omgaan met data in XML-formaat afkomstig van een externe bron, in dit geval <http://www.buienradar.nl/>. Buienradar stelt (gratis) weers- en verkeersinformatie beschikbaar om, bijvoorbeeld, op je eigen website te plaatsen of op een andere manier te gebruiken.

## 2 Leerdoelen

Na afloop van deze opdracht ben je in staat om:

- xml-data afkomstig van een externe bron in te lezen, te parseren en te doorzoeken op relevante informatie;
- een GUI te ontwerpen, waarmee informatie kan worden geselecteerd en weergegeven;
- door gebruik te maken van de juiste abstractie-middelen ervoor te zorgen dat de GUI zo min mogelijk met het model interfereert.

## 3 Probleemschets

### Weersinformatie

XML staat voor *eXtensible Markup Language*. Deze taal wordt gebruikt voor het vastleggen van data op een gestructureerde manier. De structuurinformatie die aan de data wordt toegevoegd maakt gebruik van zogenaamde *tags*, die up hun beurt weer voorzien kunnen zijn van attributen. XML lijkt enigszins op HTML. Groot verschil is echter, dat HTML vooral bedoeld is voor de opmaak van gegevens terwijl men met XML gegevens kan indelen. XML wordt veel gebruikt om gegevens via internet te versturen of uit te wisselen.

In deze opdracht ga je xml-data afkomstig van Buienradar ophalen en zichtbaar maken. Je kunt xml gewoon als platte tekst beschouwen en op een ad-hoc manier hierin op zoek gaan naar bepaalde data, bijvoorbeeld de weergegevens van het meetstation in Arnhem. Veel makkelijker is het echter om gebruik te maken van bestaande Java-bibliotheken om xml-data te parseren die dan als een `Document`-object beschikbaar komt. De `Document` klasse bevat verschillende methodes waar je op systematische wijze mee kunt zoeken naar specifieke informatie. Welke methodes voor jouw gebruik handig zijn zul je zelf moeten gaan uitzoeken. Voor het parseren kun je het beste eerst een `DocumentBuilderFactory` object maken waarmee je vervolgens een `DocumentBuilder` genereert. Deze laatste klasse bevat een methode genaamd `parse` waar je de url van de xml-data aan meegeeft (in dit geval <http://xml.buienradar.nl/>) en die je dan de bijbehorende `Document`-instantie oplevert.

### Weergave

Het is de bedoeling dat de gebruiker een weerstation kan kiezen en dat hij vervolgens de weerssituatie op die locatie te zien krijgt. Voor het kiezen dien je gebruik te maken van een `JComboBox` die alle keuzemogelijkheden in de vorm van een *drop-down-list* weergeeft. Verder dient je applicatie ook een 'ververs'-knop te bevatten om de meeste actuele informatie op te halen en moet het weerssymbool (een plaatje van een al dan niet door wolken bedekt zonnetje) worden getoond. Deze weerssymbolen zijn als url opgeslagen in het xml bestand. Om ze weer te geven moet je het plaatje op de aangegeven locatie nog wel even ophalen.

## **4 Producten**

Als producten moet je een JAVA applicatie, zoals gebruikelijk bestaande uit een verzameling klasse-definities, inleveren, die voldoen aan de kwaliteitscriteria zoals die gesteld zijn voor deze cursus. De minimale kwaliteitseis is dat je eigen producten alle van JavaDoc worden voorzien. Vergeet niet om de namen van alle groepsleden te vermelden.

## **5 Inleveren van je producten**

**Vóór zondag 26 april, 11:00 uur, via Blackboard.**