

## Leertaak 2

Leertaak 2 maak je als practicumduo. Doel van deze leertaak is je te oriënteren op de uiteindelijke probleemstelling van dit thema, te weten het verwerken van een grote workload aan gegevens (op systemen met afgeknepen resources). Je zult in leertaak 3 hiervoor een systeem moeten bouwen met een eigen opslagsysteem en verdeeld over meerdere (virtuele) systemen.

In leertaak 2 maak je een proef- en demonstratieversie van het systeem op basis van een **relationeel** DBMS en dat op je **eigen laptop**. Deze versie onderwerp je aan een zogenaamde stresstesten om vast te stellen welke belasting je proefversie wel en niet kan verwerken. Bovendien kun je deze versie bij de eerste klantcontacten in leertaak 3 gebruiken.

Bij wijze van start ontvang je een Postgres-database/MySQL-database volgens het logisch databasemodel uit de casustekst UNWDMI waarin een aantal gegevens opgeslagen zijn. Je maakt een databaseapplicatie in de programmeertaal Java 1.7.0+ met gebruikmaking van Postgres/MySQL (eigen keuze) volgens de beschrijving in de casustekst UNWDMI. Je wordt aangeraden gebruik te maken van de ontwikkelomgeving Eclipse.

Je implementatie **moet** het behoud van referentiële integriteit afdwingen. Dat betekent concreet dat bij een *insert* van een *record* de waarden van *foreign keys* moeten kloppen. MySQL kent verschillende database engines of storage engines. De storage engine MyISAM ondersteunt geen foreign key relaties, maar InnoDB biedt dit wel.

Je applicatie bevat de volgende functies:

- ontvangst van nieuwe weergegevens in XML-formaat uit meerdere bronnen (je maakt gebruik van één server-socket);
- correctie van de weergegevens zoals beschreven in de casustekst;
- opslag van de weergegevens naar de database;
- één voorbeeld naar eigen inzicht van een selectiequery op de weergegevens zoals beschreven is in de casustekst.

De verwerking van XML-bestanden in Java maakt gebruik van hetzij het *Document Object Model (DOM)*, hetzij de *Simple API for XML (SAX)*. Het is bovendien ook toegestaan zelf een XML-Parser te schrijven (maar dat kost je wel extra inspanning, die zich later waarschijnlijk wel uitbetaalt). Informatie over het benaderen van relationele databases vanuit een Java-programma met behulp van JDBC is op Internet te vinden.

Als je databaseapplicatie gereed en functioneel getest is, voer je stresstesten uit. Daartoe ontvang je een weersimulator (Java-programma) dat maximaal weerstations simuleert, die elke seconde één XML-bericht met weergegevens genereren. Dit programma doet overigens geen 100.0000 weerstations na, maar een representatieve set van 8000 weerstations. Je voert stapsgewijs de frequentie van de gegevensgeneratie

op totdat je systeem de gegevens niet meer kan verwerken en je geeft daarvoor een verklaring.

Het resultaat van deze leertaak presenteert je schriftelijk aan de docent Operating Systemen. In deze rapportage moet de onderstaande informatie staan:

- een verklaring van alle programmaonderdelen die **de gevraagde functies vormgeven** (inclusief code snippets) en
  - verklaring hoe en in hoeverre je de ontvangen gegevens buffert
  - verklaring in hoeverre je gebruikt maakt van synchronisatie (e.g. *synchronized*)
- de resultaten van de stresstest inclusief je verklaring voor de maximale snelheid van de gegevenswerking. Je geeft hierbij aan:
  - een overzicht van de door jouw gebruikte systemen en infrastructuur (zoals besturingssysteem, CPU, geheugen, netwerkoverzicht etc.);
  - wat de gehaalde verwerkingssnelheid (aantal **verwerkte berichten per seconde**) is over een tijdspanne van minimaal 10 minuten; Dit toon je aan met metingen die je binnen de applicatie doet en of op de database.
  - welke **resource** de bottleneck vormt en door welk **proces/processen** dit wordt veroorzaakt;
  - een onderbouwing van je uitleg met behulp van gegevens van tools zoals taskmanager, procesexplorer, perfmon, top (of door jouw gebruikte, vergelijkbare tools). Voeg screenshots van deze gegevens toe.

Dit rapport zal naast inhoud ook van commentaar worden voorzien op het gebied van taalgebruik en structuur.

De applicatie kun je bewaren voor demonstratiedoeleinden en verdere ontwikkeling bij leertaak 3.

Je voert deze leertaak uit in themaweken 3, 4 en 5. Het schriftelijke eindrapport lever je uiterlijk aan het einde van de vijfde themawEEK in, via het daarvoor bestemde inleverpunt op BlackBoard. **Alleen PDF!** De source-code zet je in de file-exchange van je practicumduo.