Data Analytic Tools for Inconsistency Detection in Large Data Sets

sdmay18-27
Faculty Adviser: Dr. Ying Cai
Client: Kingland Systems
Team: Christopher Konopka, Logan Heitz, TJ Rogers, Camden Voigt

Introduction

Problem

Kingland performs inconsistency detection on 500,000 + entries against a database of 100 million records. This process currently takes an average of a day (24 hours) to complete.

Need

Kingland needs a system than can perform these inconsistencies more quickly to save time and resources as well as being able to add more inconsistencies as needed.

Solution

Our solution improves the efficiency by reducing memory used, making smaller SQL queries, and using multi-threading.

Design Requirements

Functional Requirements

- ❏ Configurable
- ❏ Don't use SQL Inner-Join statements
- ❏ Solution must validate all fields are present in data
- ❏ Solution must compare current records to previous records as well as other current records
- ❏ Solution must detect all inconsistencies

Non-Functional Requirements

- ❏ Completes faster than current solution
- ❏ Solution must work with central Database of more than 100 million records

Engineering Constraints

- ❏ Compatible with MySQL
- ❏ Must handle XML Input

Operating Environment

- ❏ Amazon Web Service

Users & Uses

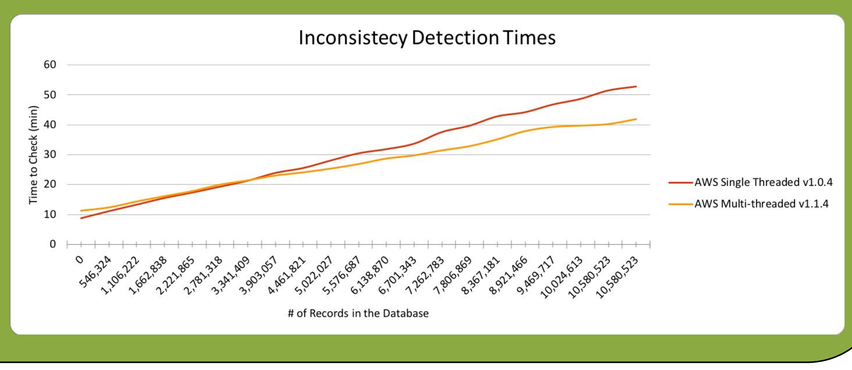
Users

- ❏ Kingland's Data Analysts.

Uses

- ❏ To detect inconsistencies in Kingland's daily reports

Performance Results



Standards

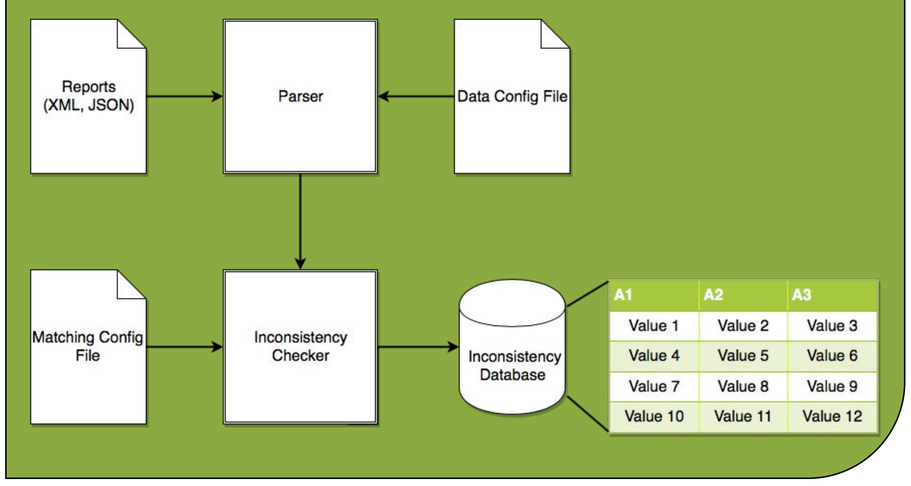
Testing Protocols

- ❏ IEEE 829
  - ❏ Software testing documentation
- ❏ SO/IEC/IEEE 29119-4
  - ❏ Test techniques

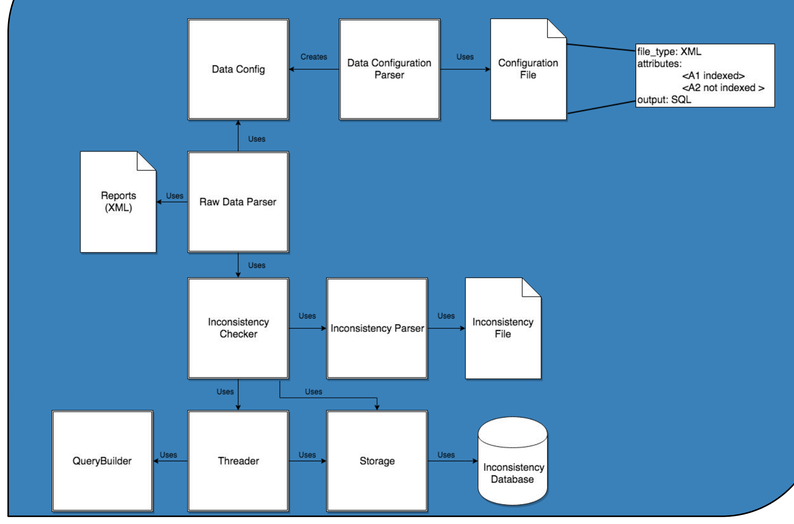
Ethics

- ❏ ISO 17799
  - ❏ Regarding information security

Concept Diagram



Block Diagram



Functional Modules

Data Configuration

- ❏ Allows users to configure how the project stores data
- ❏ Reads and XML Configuration file to get user options

Raw Data Parser

- ❏ Parses the input XML file
- ❏ Calls inconsistency checker to check each record read

Inconsistency Parser

- ❏ Parses the inconsistency file to see which inconsistencies the program should check

Inconsistency Checker

- ❏ Checks new records against database for inconsistencies

Threader

- ❏ Manages Threads and Thread Pools

Storage

- ❏ Provides an interface to save records to various storage mediums
- ❏ Provides inconsistency detection queries

Technical Details

Details of Functional Modules

- ❏ All modules implemented in Java
- ❏ Developed in Intellij
- ❏ Inconsistency Checker
  - ❏ Uses Apache Commons CLI to parse command line options
- ❏ Raw Data Parser
  - ❏ Uses Sax XML Parser
- ❏ Storage
  - ❏ Uses JDBC to connect to database
  - ❏ Uses Apache Commons DBCP for connection pooling
- ❏ Apache Log4j utilized for logging throughout project

Testing

Testing Environment

- ❏ Followed Test Driven Development principles
- ❏ Automated pipeline to run after every git push
- ❏ AWS Instance (db.m4.2xlarge)
  - ❏ vCPU: 8
  - ❏ Memory (GiB): 32
- ❏ Local Machine
  - ❏ Intel® Core™ i7-7700K CPU @ 4.20GHz
  - ❏ Memory (GiB): 32
  - ❏ Storage (GiB): 356

Testing Strategy

- ❏ JUnit Tests
- ❏ Integration Tests
- ❏ Mockito for independent Unit Tests
- ❏ Maven Plugins to manage Integration Testing Goals
- ❏ Performance Testing