The Data Deluge
The Data Deluge

1. What is the deluge?

2. What is parallel computing?

3. How to harness parallel power?
The Data Deluge

The DNA Deluge

A Parallel Computing Approach

Brendan Lawlor
The Data Deluge

1. What is the deluge?
2. What is parallel computing?
3. How to harness parallel power?

The DNA Deluge
A Parallel Computing Approach

Brendan Lavelle
The Data Deluge
The Data Deluge

Shrinking chips
Number and length of transistors bought per $
The cost of data is decreasing.
But the decrease in cost of processing is leveling off - at least on a single processor.
The Data Deluge

Multi Core Processors

Cloud/Clusters
The Data Deluge

Amdahl's Law:

\[ S(n) = \frac{T(1)}{T(n)} = \frac{T(1)}{T(1) \left( B + \frac{1}{n} (1 - B) \right)} = \frac{1}{B + \frac{1}{n} (1 - B)} \]

or... you can't make something faster than its slowest (= serialized) part.

Multithreaded programming
The Data Deluge

Multithreaded programming

Theory

Actual
The Data Deluge
The Data Deluge

We are going to need tools that are fit for purpose.

Functional Programming
Actor Architecture
Reactive Streams
The Smith-Waterman Algorithm

In bioinformatics, a sequence alignment is a way of arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences.
The Data Deluge

Central Dogma of Genetics

DNA → RNA → PROTEIN

DNA (transcription) → RNA (translation) → protein
The Data Deluge

The Smith-Waterman Algorithm

![Smith-Waterman Algorithm Diagram](image.png)
Two Kinds of Parallel

Data-Dependent

The SW algorithm is an example of this kind: some cells in the matrix require the results of others. It uses intricate loops and intimate knowledge of the hardware to drive as much data as possible through a CPU in a single clock cycle. It achieves SIMD using Intrinsics.

"Embarrassingly Parallel"

Running many query sequences against many database sequences is an example of this kind. Each run of the S-W algorithm is independent of those that go before and after.
Data-Dependent

The SW algorithm is an example of this kind: some cells in the matrix require the results of others. It uses intricate loops and intimate knowledge of the hardware to drive as much data as possible through a CPU in a single clock-cycle. It achieves SIMD using Intrinsics.
"Embarrassingly Parallel"

Running many query sequences against many database sequences is an example of this kind. Each run of the S-W algorithm is independent of those that go before and after.
The Data Deluge
Create a chassis to:
Spread the power over multiple queries and multiple database sequences
Plug in multiple engines to go faster (*scalability*)
Supply enough fuel to all engines to prevent stalling
Avoid flooding the engine with too much fuel
The Data Deluge

(Old) New Tools

Functional Programming

"The essence of functional programming is to concentrate on mechanisms of immutable values rather than on opaque modifications of stateful values."

- Charles Hughes, inventor of code programming language

In other words, with FP we can write code in an immutability-oriented manner and in the complex and extensive field of the parallel infrastructure,

- Engineering Actor to open, parallelize, and implement to balance out complex tasks, efficiently.

Actor Architecture

Reactive Streams
The Data Deluge

Functional Programming

"The essence of functional programming is to concentrate on **transformations** of **immutable values** rather than stepwise modifications of mutable values."

- Martin Odersky, inventor of Scala programming language

In other words, with FP we can write code in an intrinsically parallel manner and let the compiler and run-time schedule on the parallel infrastructure.

- Paraphrasing Robert Harper, Professor of Computer Science at Carnegie Mellon University
Actor Architecture
The Data Deluge

Reactive Streams

Publisher  demand  data  Subscriber
The Data Deluge
The Data Deluge

Proof of Concept: Architecture

Master

Query Sequences

High-Score Alignments

Worker 1

Database Sequences

Worker 2

Database Sequences

Worker 3

Database Sequences

PraceDays 2015
The Data Deluge

Fan-Out of Actors creates parallel processes on immutable data.

Reactive Stream of Databases

Reactive Stream of Database Sequences

Fan-In of Actors filters results by score.
Proof of Concept: Results

Results on a single Node: Linux box with an i7-4770 3.4GHz processor (quad-core)

200 S-W invocations (or about 1.8 billion "cells") per second

Using a single-core baseline, this indicates ~ 15% overhead (I/O & message passing).

~ 1000 lines of (readable!) Scala code.

What about multiple Linux boxes?
Multiple cheap computers, with the same cheap software, multiplies the performance.
The Data Deluge

What's next?

Optimization of Proof of Concept

Test on larger scale

Publish results

Consider a fully stream-centric solution.

(And So What!?)

Just one of many examples:
Clinical Microbiology

PraceDays 2015
The Data Deluge

The DNA Deluge
A Parallel Computing Approach
Brendan Lawlor