The History of Parallelism Support in Java

Douglas C. Schmidt
d.schmidt@vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science
Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA
Learning Objectives in this Lesson

- Learn the history of Java parallelism from 2010 to 2022
Learning Objectives in this Part of the Lesson

- Learn the history of Java parallelism from 2010 to 2022
- Understand the evolution of Java from concurrency to parallelism
A Brief History of Parallelism in Java
A Brief History of Parallelism in Java

- Foundational parallelism support

See [en.wikipedia.org/wiki/Java_version_history#Java_SE_7](en.wikipedia.org/wiki/Java_version_history#Java_SE_7)
A Brief History of Parallelism in Java

- Foundational parallelism support
- Focus on fine-grained object-oriented data parallelism

See en.wikipedia.org/wiki/Data_parallelism
A Brief History of Parallelism in Java

- Foundational parallelism support
- Focus on fine-grained object-oriented data parallelism
  - e.g., runs the same task on different elements of data by using the “split-apply-combine” model

See [www.jstatsoft.org/article/view/v040i01/v40i01.pdf](http://www.jstatsoft.org/article/view/v040i01/v40i01.pdf)
A Brief History of Parallelism in Java

- Foundational parallelism support
- Focus on fine-grained object-oriented data parallelism
- e.g., runs the same task on different elements of data by using the “split-apply-combine” model

Use a common fork-join pool to search input strings to locate phrases that match famous quotes by the Bard

List<List<SearchResults>>
listOfListOfSearchResults = ForkJoinPool
  .commonPool()
  .invoke(new
    SearchWithForkJoinTask
    (inputList,
     mPhrasesToFind, ...));

See github.com/douglascraigschmidt/LiveLessons/tree/master/SearchForkJoin
A Brief History of Parallelism in Java

- Foundational parallelism support
- Focus on fine-grained object-oriented data parallelism
- Powerful & scalable, but tedious to program directly
A Brief History of Parallelism in Java

• Advanced parallelism support

See en.wikipedia.org/wiki/Java_version_history#Java_SE_8
A Brief History of Parallelism in Java

- Advanced parallelism support
- Initial focus on fine-grained functional programming frameworks for data parallelism

See [en.wikipedia.org/wiki/Data_parallelism](en.wikipedia.org/wiki/Data_parallelism)
A Brief History of Parallelism in Java

- Advanced parallelism support
- Initial focus on fine-grained functional programming frameworks for data parallelism & asynchrony

See gist.github.com/staltz/868e7e9bc2a7b8c1f754
A Brief History of Parallelism in Java

- Advanced parallelism support
- Initial focus on fine-grained functional programming frameworks for data parallelism & asynchrony

```java
List<Image> images = urls
    .parallelStream()
    .filter(not(this::urlCached))
    .map(this::downloadImage)
    .map(this::applyFilters)
    .reduce(Stream::concat)
    .orElse(Stream.empty())
    .collect(toList());
```

Synchronously download images that aren’t already cached from a list of URLs & process/store the images in parallel

A Brief History of Parallelism in Java

• Advanced parallelism support
• Initial focus on fine-grained functional programming frameworks for data parallelism & asynchrony

CompletableFuture

<Stream<Image>>

resultsFuture = urls

.stream()

.map(this::checkUrlCachedAsync)

.map(this::downloadImageAsync)

.flatMap(this::applyFiltersAsync)

.collect(toFuture())

.thenApply(stream ->

    log(stream.flatMap(
        Optional::stream),
    urls.size()))

.join();

Combines streams & completable futures to asynchronously download images that aren’t already cached from a list of URLs & process/store the images in parallel

A Brief History of Parallelism in Java

• Advanced parallelism support
  • Initial focus on fine-grained functional programming frameworks for data parallelism & asynchrony
  • Later focus on pub/sub reactive streams frameworks

  e.g., Java reactive streams made available in Java 9 have enabled the RxJava & Project Reactor frameworks

See en.wikipedia.org/wiki/Java_version_history#Java_SE_9
A Brief History of Parallelism in Java

- Advanced parallelism support
  - Initial focus on fine-grained functional programming frameworks for data parallelism & asynchrony
  - Later focus on pub/sub reactive streams frameworks

List<Image> filteredImages = Flux
  .fromIterable(urls)
  .parallel()
  .runOn(Schedulers
    .boundedElastic())
  .filter(url -> !urlCached(url))
  .map(this::blockingDownload)
  .flatMap(this::applyFilters)
  .sequential()
  .collectList()
  .block();

Applies Project Reactor reactive streams to asynchronously download images that aren’t already cached from a list of URLs & process/store the images in parallel.
A Brief History of Parallelism in Java

• Java’s advanced parallelism frameworks are designed to strike a balance between productivity & performance
A Brief History of Parallelism in Java

• Java’s advanced parallelism frameworks are designed to strike a balance between productivity & performance

• However, these frameworks can be overly prescriptive
The Evolution of Java from Concurrency to Parallelism
The Evolution of Java from Concurrency to Parallelism

- Brian Goetz has an excellent talk about the evolution of Java from concurrent to parallel computing.

See [www.youtube.com/watch?v=NsDE7E8sIdQ](http://www.youtube.com/watch?v=NsDE7E8sIdQ)
The Evolution of Java from Concurrency to Parallelism

- Brian Goetz has an excellent talk about the evolution of Java from concurrent to parallel computing.

His talk explains how modern Java combines functional programming with fine-grained data parallelism to leverage many-core processors.

See [www.infoq.com/presentations/parallel-java-se-8](http://www.infoq.com/presentations/parallel-java-se-8)
The Evolution of Java from Concurrency to Parallelism

- Rob Pike also has a good talk that explains the differences between concurrency & parallelism

His talk explains how concurrency is about *dealing* with lots of things at once, whereas parallelism is about *doing* lots of things at once

See [www.youtube.com/watch?v=cN_DpYBzKso](http://www.youtube.com/watch?v=cN_DpYBzKso)
End of the History of Parallelism Support in Java