Java™

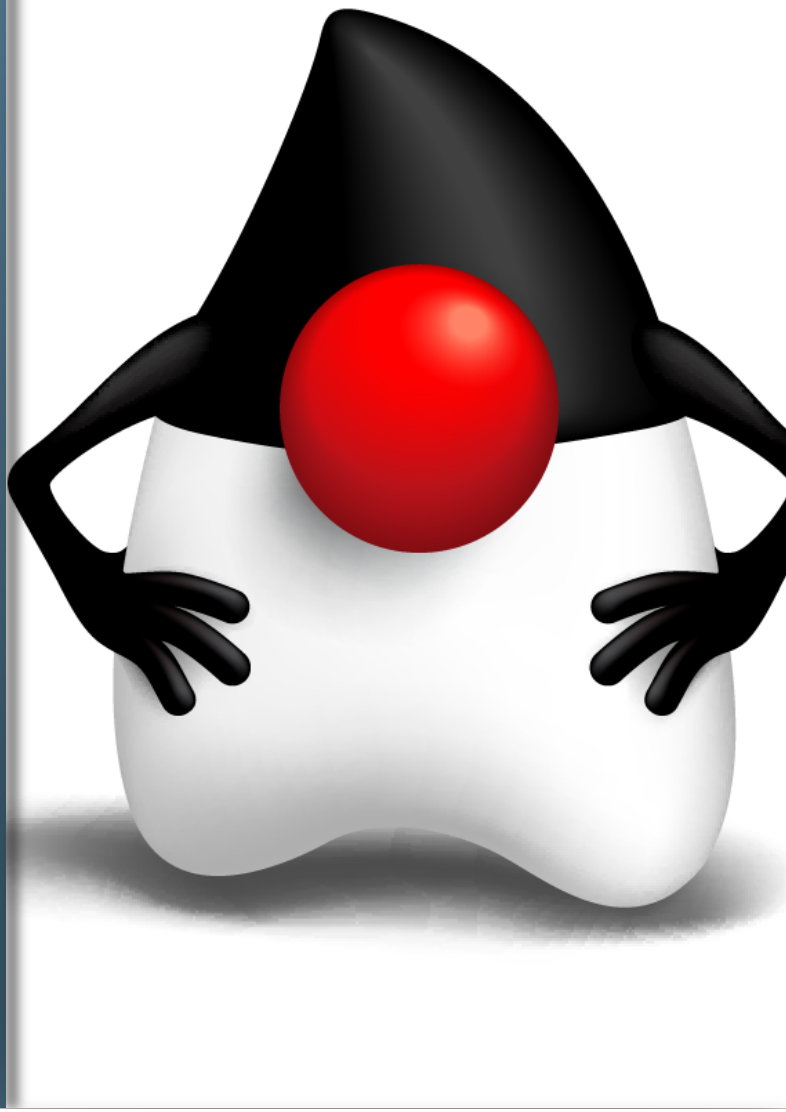
The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Parallel Streams Workshop

Stuart W. Marks
Oracle JDK Core Libraries
Virtual JUG 13 March 2014

@stuartmarks



Streams

- What is a stream?
  - A stream is a ~~collection~~ of ~~objects~~.
  - A stream is a ~~sequence~~ of ~~objects~~.
  - A stream is an ***abstraction*** that represents zero or more ***values***.
- Values, not objects: avoid mutation
- Not a collection: values might not be stored anywhere.
- Not a sequence: order might not matter.
- “A stream is an iterator on steroids.”

 Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Java™ ORACLE®

Pipelines

- A pipeline consists of:
  - a stream ***source***
  - zero or more ***intermediate*** operations
  - a ***terminal*** operation
  - a pipeline can be executed in parallel

- What has this got to do with Lambda?
  - Lambdas tell these operations what to do.

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

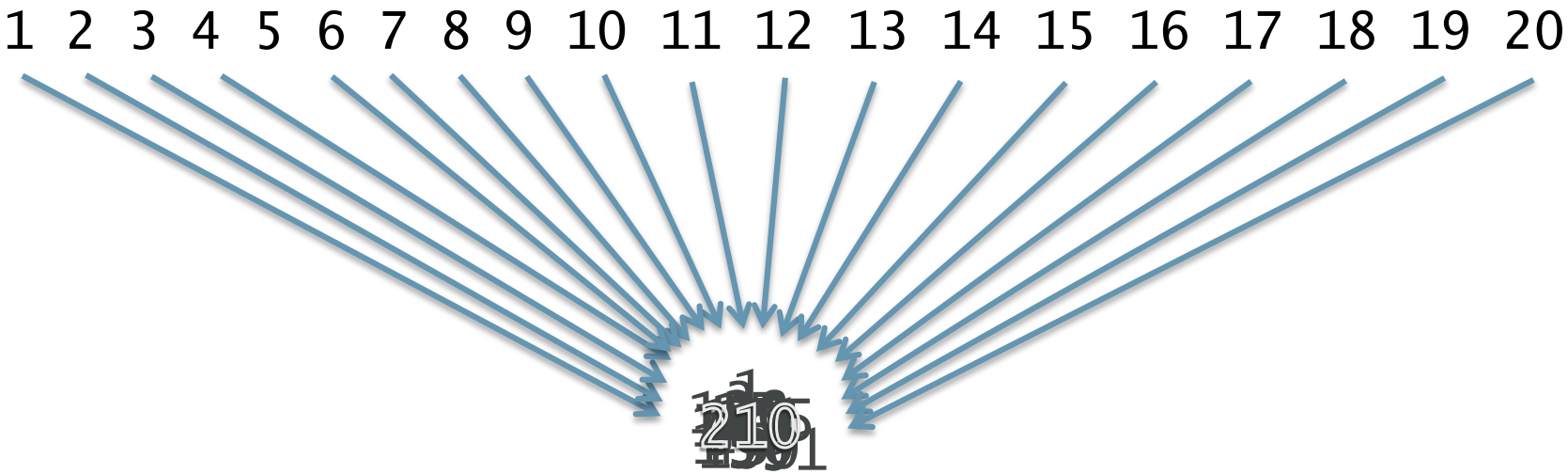
Java™ ORACLE®

Code and Demos!

 | Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Java™ ORACLE®

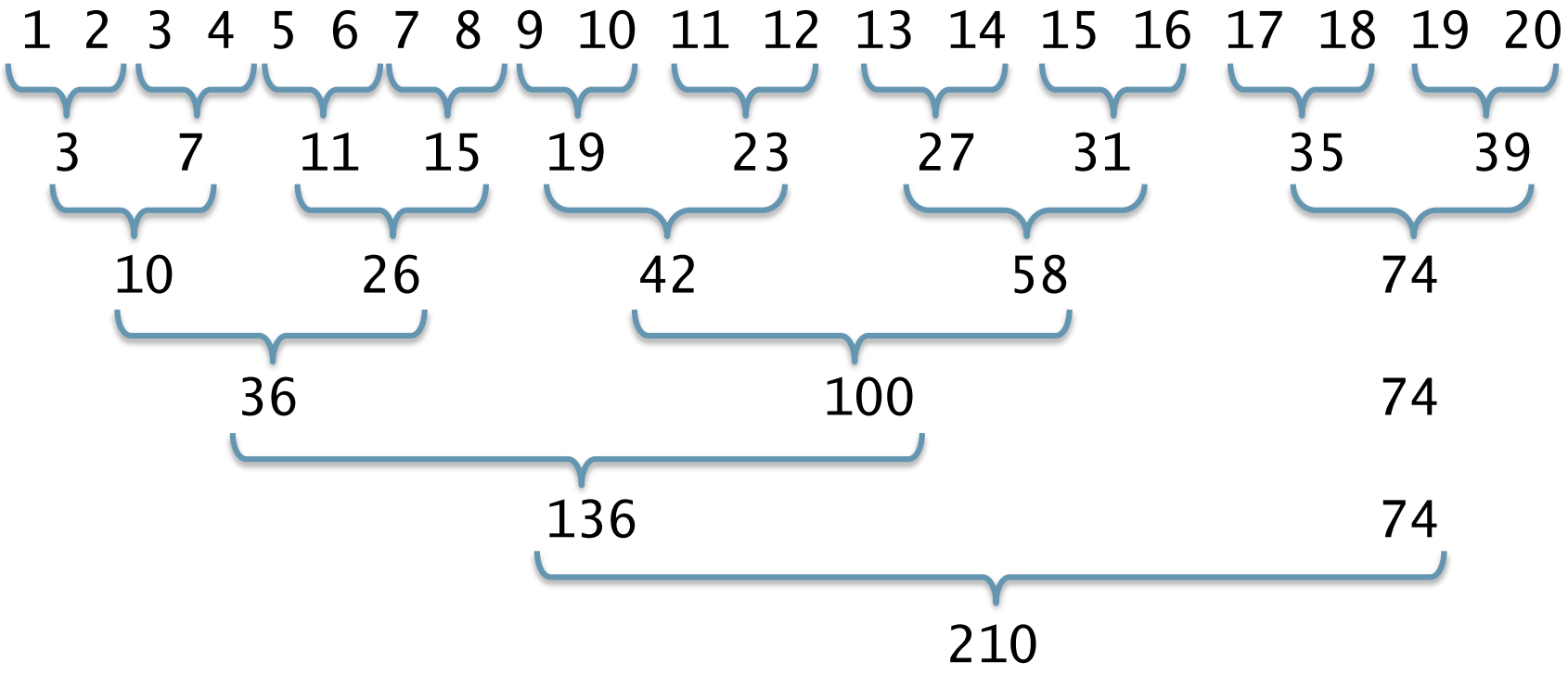
Summation by Accumulation



 | Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Java™ ORACLE®

Summation by Reduction



 | Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Java™ ORACLE®

Moar Code and Demos!

 | Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Java™ ORACLE®

Summary

- Writing parallel programs is easy! Or is it?
- You still have to be careful:
  - avoid mutation and interference
  - avoid contention
  - think reduction, not accumulation
- Benchmark, don't guess:
  - JMH is easy to use and avoids common pitfalls benchmarking Java
  - "multi-core" machines can be very slippery

 Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Java™ ORACLE®

Links

- Java 8:
  - http://openjdk.java.net/projects/jdk8/ (OpenJDK sources)
  - https://jdk8.java.net/ (snapshot builds, currently b132 is RC3)
- FAQ: http://www.lambdafaq.org/
- JMH: http://openjdk.java.net/projects/code-tools/jmh/

- Contact me on Twitter: **@stuartmarks**
- Thanks to @sjmaple and @virtualJUG !!

 Copyright © 2014, Oracle and/or its affiliates. All rights reserved. |

Java™ ORACLE®

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

 Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Java™