ParaFormance™ : An Advanced Refactoring Tool for Parallelising C++ Programs – Part 2

Chris Brown, Vladimir Janjic, Kevin Hammond

University of St Andrews, Scotland

@chrismarkbrown

@rephrase_eu

Outline

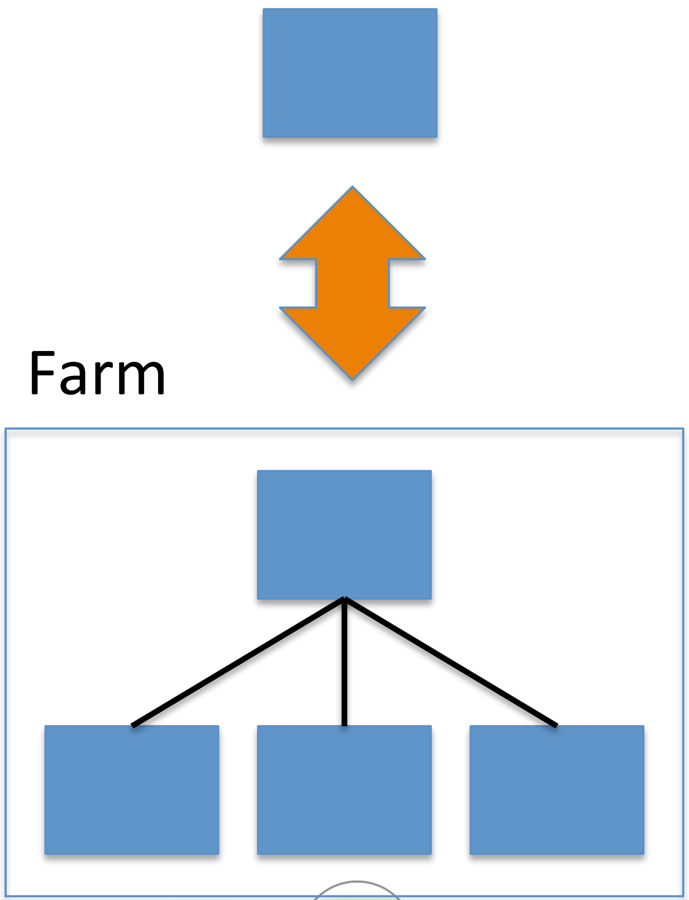
1. Introduction and overview to Insertion
2. Live Demonstration of Insertion
3. Follow along interactively
4. Image Convolution and Ant Colony examples

What is Insertion?

1. Introduces parallelism into an application semi-automatically
2. Refactors a sequential portion of code into a parallel version
3. Introduces all parallel 'business logic'

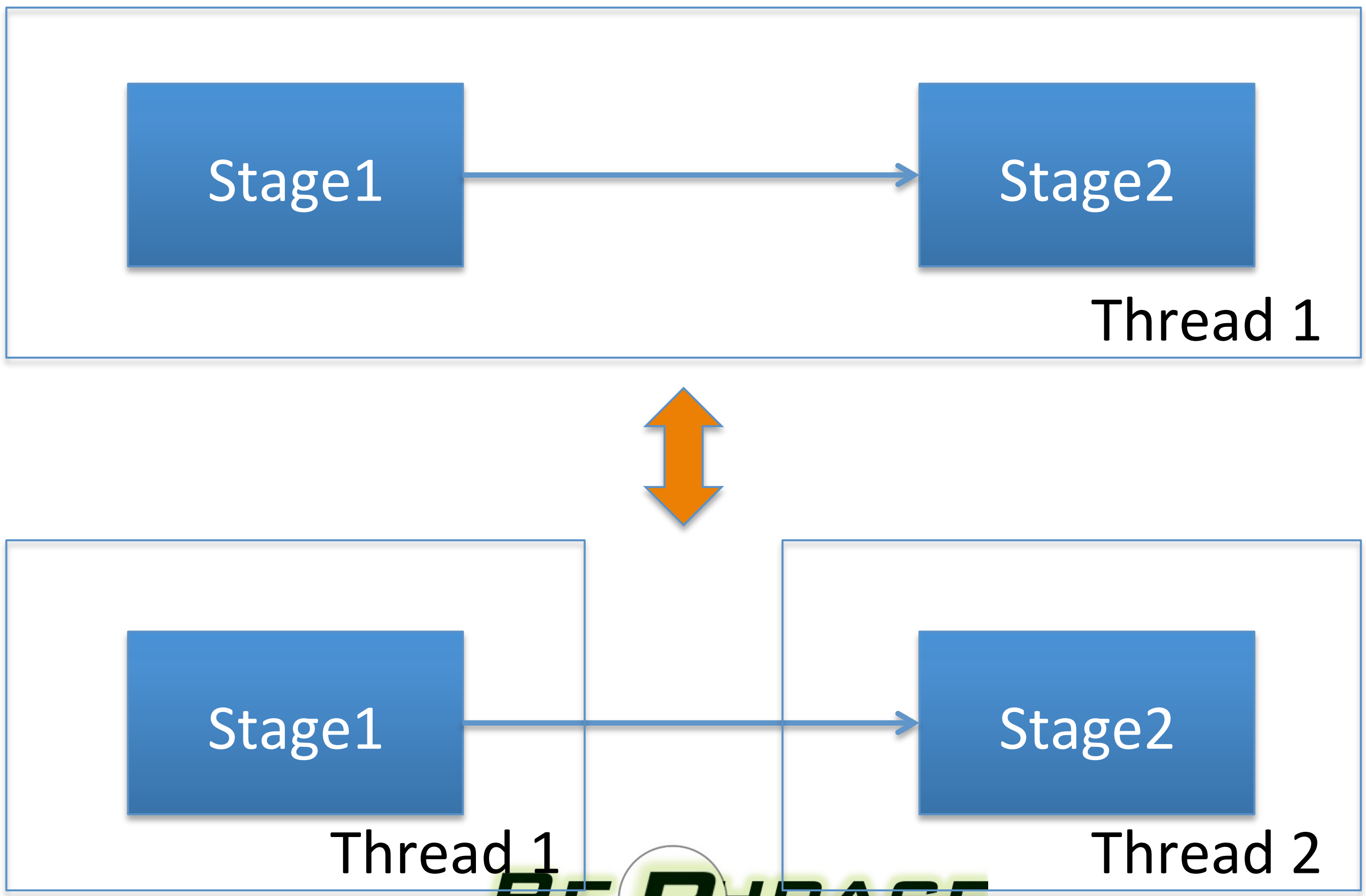
Inserting a Farm

S	$\equiv$	$Farm(S)$	*farm intro/elim*



Inserting a Pipeline

$S_1 \circ S_2 \quad \equiv \quad Pipe(S_1, S_2)$ *pipe seq*

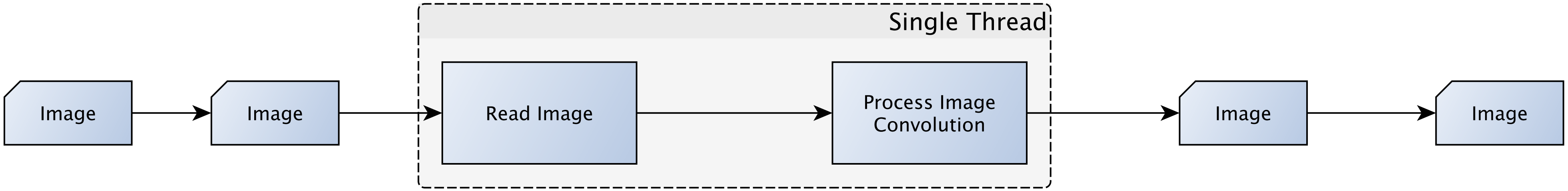


Why?

- Huge saving in effort over manual
- Difficult to get right!
- We have seen 40 hours of manual programming effort reduced into 5 hours work (a 8x productivity improvement)
- The programmer doesn't have to learn lots of boilerplate
- The programmer doesn't have to keep track of library/standards changing

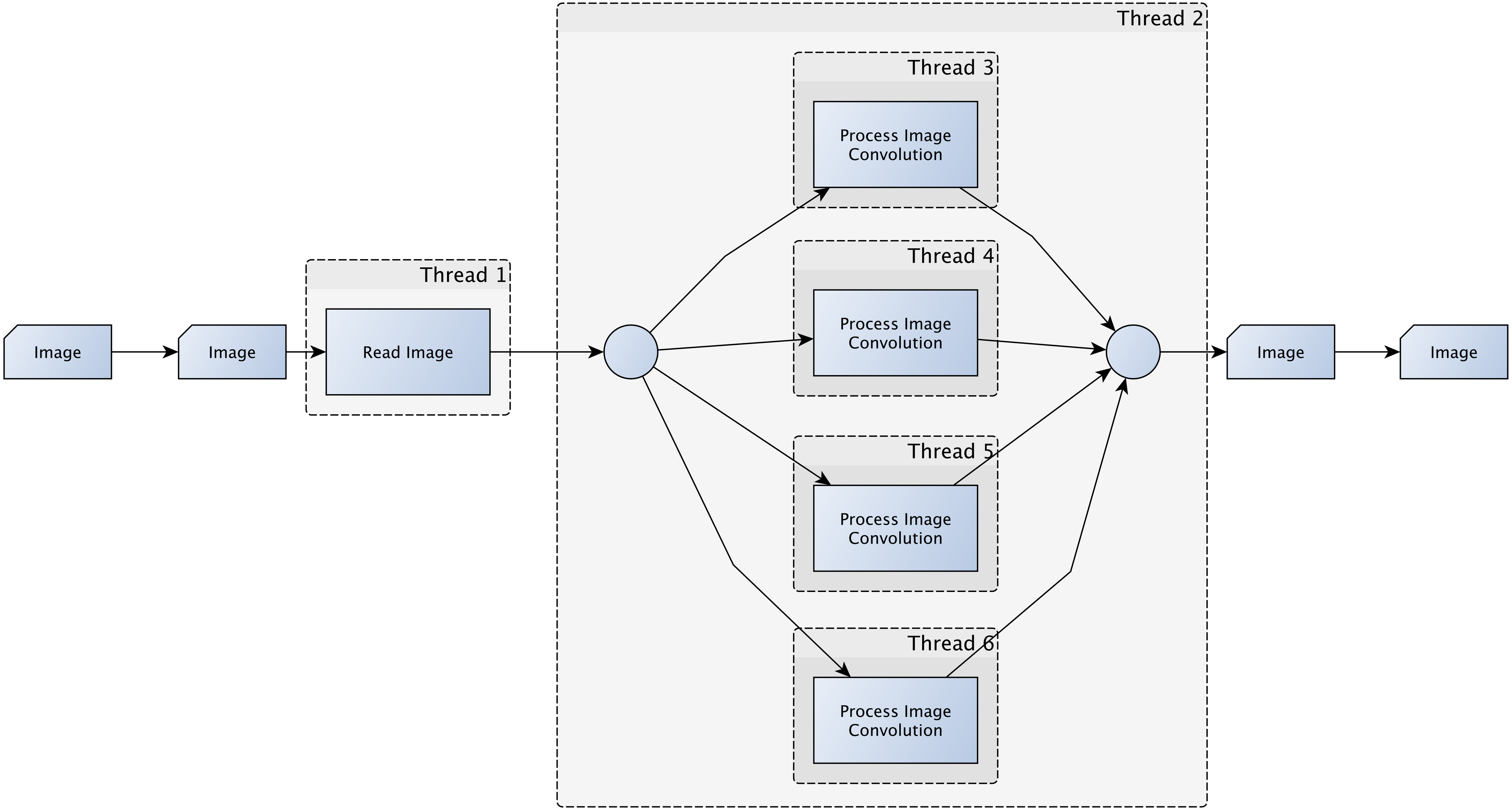
Image Convolution

University of St Andrews



REPHRASE

Image Convolution, Refactored



University of St Andrews

THANK YOU!

`http://rephrase-ict.eu`

@rephrase_eu