Composable Specifications for Structured Shared-Memory Communication
Benjamin Wood, Adrian Sampson, Luis Ceze, and Dan Grossman

Parallel programming is here to stay. Shared memory is fast and popular... ...but it's hard!

It's difficult because communication is implicit...

What if programmers specify where communication occurs?

@Group("Progress")
public class Renderer {
    volatile int curLine;
    final int lines;
    ConcurrentHashMap<Integer,Pixel[]> outputImage;

    // Render line (nthreads * n + tid) for every n.
    @Writer("Progress")
    void render(int tid, int nthreads) {
        for (curLine = tid; curLine < lines; curLine += nthreads)
            outputImage.put(curLine, expensiveCall(...));
    }

    // Return after rendering is finished.
    @Reader("Progress")
    ConcurrentHashMap<Integer,Pixel[]> getImage() {
        while (curLine < lines) /*spin*/;
        return outputImage;
    }
}

master thread
(running getImage)
read curLine
read curLine
read curLine
read curLine
...

worker thread
(running render)
write curLine
write curLine
write outputImage
write curLine
write curLine
write outputImage
...

worker thread
(running render)
write curLine
write curLine
write outputImage
write curLine
write curLine
write outputImage
...