

# Accelerating Financial Evaluations using Haskell and the GPU

Alex Olivier Appetiti

## Bit of Background

Prudential is an investments/insurance company.



A common task for them is to evaluate the estimated value of a financial product they want to sell for many different parameters.

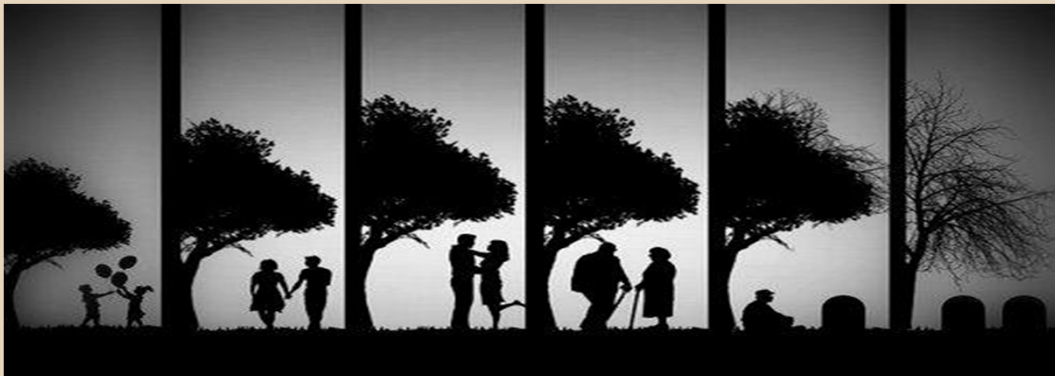
## Bit of Background

Currently, they do these calculations on a large cluster of computers using a program called Moses (written in C++)



## Bit of Background

... and it takes them a long time



## Bit of Background

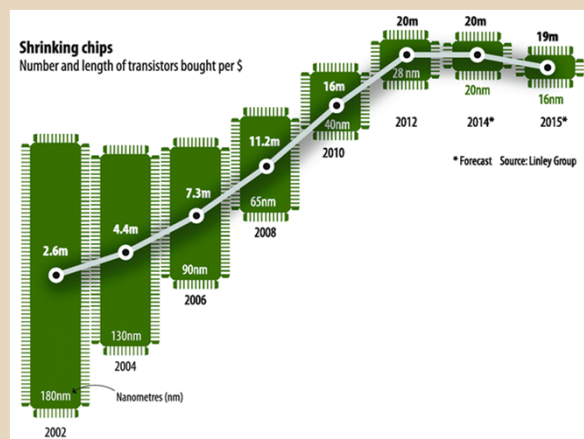
They are interested in seeing how to make these simulations run faster.

- More simulations
- More precise simulations



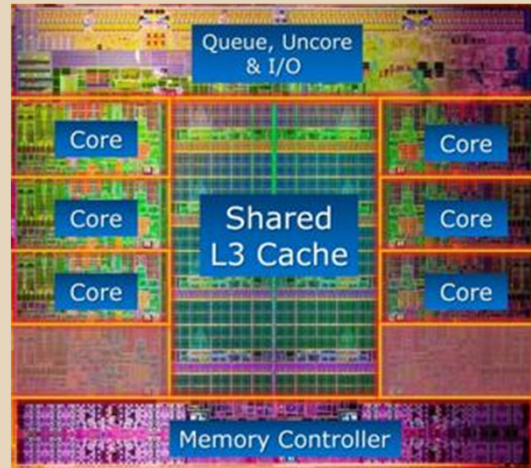
## Can Hardware Solve the Problem?

Can Moore's Law save us??



## Enter Multicore... or better yet...

Do many things at once, instead of one thing at a time very quickly!



## Use a GPU!

The core's of a GPU have become more powerful with time... and they have many many cores!



## And Why Haskell?

Haskell is a functional programming language  
Maps well to mathematical problems, often  
more elegant solutions!

For example...

## Quicksort..

```
// To sort array a[] of size n: qsort(a,0,n-1)
void qsort(int a[], int lo, int hi)
{
    int h, l, p, t;

    if (lo < hi) {
        l = lo;
        h = hi;
        p = a[hi];

        do {
            while ((l < h) && (a[l] <= p))
                l = l+1;
            while ((h > l) && (a[h] >= p))
                h = h-1;
            if (l < h) {
                t = a[l];
                a[l] = a[h];
                a[h] = t;
            }
        } while (l < h);

        a[hi] = a[l];
        a[l] = p;

        qsort( a, lo, l-1 );
        qsort( a, l+1, hi );
    }
}
```

Vs.

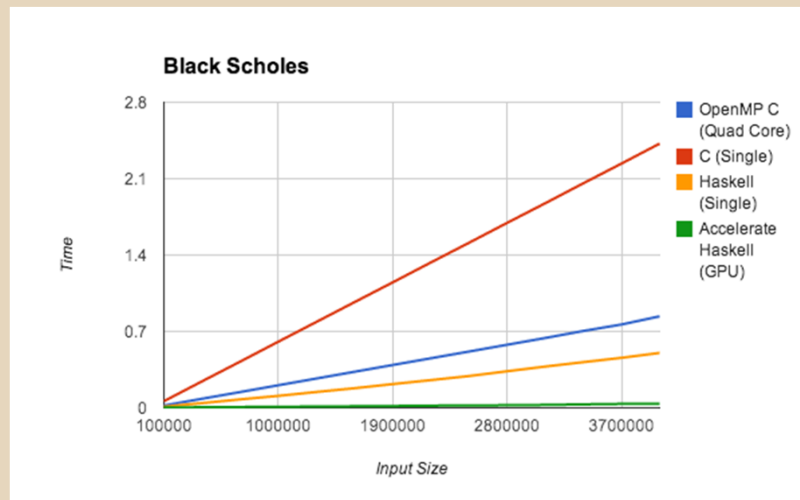
## Quicksort...

```
quicksort :: Ord a => [a] -> [a]
quicksort [] = []
quicksort (p:xs) = (quicksort lesser) ++ [p] ++ (quicksort greater)
  where
    lesser = filter (< p) xs
    greater = filter (>= p) xs
```

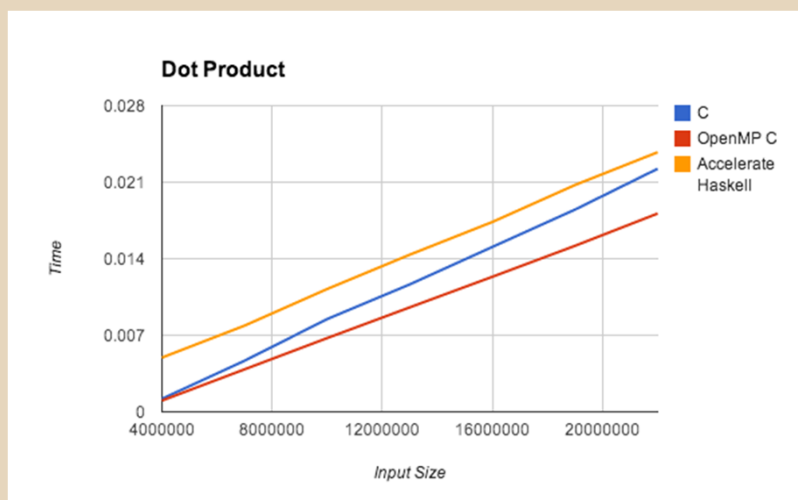
## Other Advantages...

- No pointer manipulations.
- Functions as first class citizens.
- Automatic memory management.
- Strong Static Typing.
- Accelerate!

# It's Pretty, but is it Fast?



# Cont..



## Next Step: Cash Flow Valuation!

Any questions?