

Wrong way to fix tests

This does not seem to be a good way to fix tests:

```
log.DebugFormat("Writing 100000000 values took {0} ms", (DateTime.Now - startTime).TotalMi
//Assert.Less((DateTime.Now - StartTime).TotalMilliseconds, 30000);
```

Also when writing Performance tests - make sure to time the right thing. For example measuring log.Write may give very varying time durations, measure as little block as possible, so instead of:

```
...
DateTime startTime = DateTime.Now;
regularGridCoverage.Resize(10000, 10000, 1, 1);
log.DebugFormat("Writing 100000000 values took {0} ms", (DateTime.Now - startTime).
TotalMilliseconds);
Assert.Less((DateTime.Now - startTime).TotalMilliseconds, 25000);
...
```

Write:

```
...
var startTime = DateTime.Now;
regularGridCoverage.Resize(10000, 10000, 1, 1);
var duration = (DateTime.Now - startTime).TotalMilliseconds;

log.DebugFormat("Writing 100000000 values took {0} ms", duration);
Assert.Less(duration, 25000);
...
```


Or:

```
...

var stopwatch = new Stopwatch();
stopwatch.Start();
regularGridCoverage.Resize(10000, 10000, 1, 1);
stopwatch.Stop();

log.DebugFormat("Writing 100000000 values took {0} ms", stopwatch.Elapsed.TotalMilliseconds);
Assert.Less(stopwatch.Elapsed.TotalMilliseconds, 25000);
...
```

Also make sure that message is written showing how long it took to do something, otherwise it is hard to check how long it takes on build server.

 reviewed performance tests and decreased required time to run tests, also 30% of tests are wrong! Time is not measured or not a performance test at all. Keep quality of the tests **higher**.

... for example this test as a performance test is useless:

```
[Category("Performance")]
public void ReadRegularGridPerformance()
{
    //todo: get a bigger grid file :)
    var store = new NetCdfFunctionStore();
    store.FunctionBuilders.Add(new RegularGridCoverageBuilder());
    store.Open(".././.././data/NCEP-NAM-CONUS_80km_best.ncd.nc");
    var grid = (IRegularGridCoverage) store.Functions.First(f => f is IRegularGridCoverage);
    log.DebugFormat("Reading {0} slices of {1} values", grid.Time.Values.Count, grid.SizeX*grid.SizeY);
    DateTime start = DateTime.Now;
    foreach (DateTime dt in grid.Time.Values)
    {
        Assert.AreEqual(6045, grid.GetValues(new VariableValueFilter<DateTime>(grid.Time, dt)).Count);
    }
    log.DebugFormat("Took {0} ms", (DateTime.Now - start).TotalMilliseconds);
}
```