Resilient X10

**X10** is an APGAS programming language that is designed to provide a simple and clean programming model for developing scale-out applications.

As supercomputers grow larger, the Mean Time Between Failure reduces, and the need for writing fault tolerance applications becomes more critical.

**Resilient X10** [1] allows X10 programs to survive process failures. By introducing the *Happens Before Invariance Principle*, it guarantees the correct repair of the global program structure after a failure.

```scala
try{
  /*Task A*/
  at (p) {
    /*Task B*/
    finish {at (q) async { /*Task C*/ } }
  }
}
catch(dpe:DeadPlaceException){ /*recovery steps*/}
D;
```

By applying the HBI principle, Resilient X10 will ensure that statement D executes after Task C finishes, despite the loss of the synchronization construct (finish) at place p.

**Resilient X10 over MPI ULFM**

Although MPI is the preferred transport layer for scale-out computing, Resilient X10 was initially supported only over sockets.

ULFM (User-Level Failure Mitigation) is the most recent proposed specification for fault tolerant MPI [2]. An implementation of ULFM is available based on OpenMPI 1.7.

We integrated X10 with ULFM to allow Resilient X10 applications to benefit from the scalability and performance of MPI.

**Conclusion:** Using a fault tolerant MPI implementation (ULFM), resilient X10 applications can achieve better performance with the optimized MPI communication routines and the support for high speed network protocols provided by MPI (e.g. Infiniband verbs).

References: