Global View Resilience Questions

I. X-stack front-end

- What is your model for incorporating resilience mechanisms / libraries into your system?

II. X-stack back-end

- What is your model for incorporating resilience mechanisms / libraries into your system?

III. High Level Representations

- How will you capture invariants and correctness/consistency checks in program and export them to the runtime/implementation?  (we care for resilience)

- For global view data, what are the consistency assumptions?  what is the model?

- Can/will you produce detailed "effects" reader/writer maps for data structures (and at fine segmentation) by computation phase? procedure? (dematerialize structures)

- What are the error detection/failure modes of the program?

IV. Runtime Systems

- What are the error detection/failure modes of the runtime?

- What are the expectations for communication primitives (RDMA hardware vs active-message style - programmable)?

- How will you integrate access to NVRAM? (as part of memory, storage, intelligent memory, access control?)

V. Low-level Representations

- What are the error detection/failure modes of the program implementation?

- What are the interoperation requirements (layout, distribution formats) for libraries, routines?

Bonus: Applications (no category for this!)

- what payoff is needed to compensate for resilience effort?  (performance, correctness, ?)

- is there something new an application could do if resilience increased 10-fold?  (faster convergence, larger runs, ?)

- what is acceptable performance/storage overhead for resilience?