Fault-Tolerant Job-Flow Management for Grid Environments

Team: Gargi Dasgupta[3], Liana Fong[1], S. Masoud Sadjadi[2], Onyeka Ezenwoye[2], Balaji Viswanathan[3], Selim Kalayci[2], David Villegas Castillo[2], Norman Bobroff[1]

1: IBM T. J. Watson Research Center, Yorktown Heights, NY 10598,

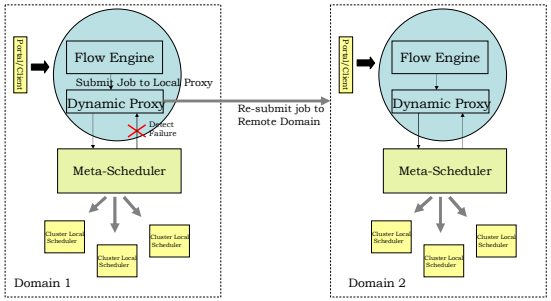
2: Florida International University, Miami, FL 33199,

3: IBM India Research Labs, New Delhi

1. Overview

- **Motivation**
  - Provide flexible job-flow orchestration and submission for scientists
  - Integrate service-based flow orchestration with job scheduling
  - Provide fault-tolerance handling that is transparent to the job-flow
- **Approach**
  - Isolate job-flow orchestration from fault handling using layered design
    - Top Layer: Job-flow (Service) orchestration; using BPEL for job-flows
    - Second layer: Job trapping & fault handling; using JSDL for jobs
  - Introduce fault-handling proxy to implement job recovery policies
    - Use the Dynamic Proxy from TRAP/BPEL framework
    - Fault handling policies can be job specific depending on
      - Type of job
      - Type of failure
      - Level of fault-tolerance specified by user

2. Architecture & Operation



- **Job-flow modeling and flow engine layer**
  - Recovery policies specified for job invocations that need "monitoring"
  - Flow adaptation replaces monitored invocations with invocation to proxy
- **Dynamic proxy and fault handling layer**
  - Dynamic Proxy monitors the status/progress of job invocation
  - Job failures trapped at proxy transparent to upper job-flow layer
  - Proxy implements recovery policy (e.g., route job to another domain, or retry)

3. Job-flow Management Framework