Contribution ID: **89**

Type: **not specified**

Addressing scalability challenges in Monitoring and Control

Wednesday, 24 March 2010 16:35 (15 minutes)

This presentation identifies scalability challenges associated with Monitoring & Control and some possible solution approaches.
• Automated control and coordination of distributed, heterogenous elements. Guiding architectural principle: semi-autonomous control, where parent nodes specify objectives and child nodes work out how to achieve objective given local constraints. Provide facilities within the automated control scripts for dealing with heterogeneity and for situation handling.
• Automated fault detection and handling. Approach: Fault detection algorithms at each level of the system hierarchy, and associated handling actions to achieve the defined objectives if at all possible while preserving system integrity. Fault detection at multiple time horizons (level 1 / level 2 level 3 feedback control) for early detection of problem situations based on behaviour patterns.
• Automated system upgrade, with local constraints that necessitate staggering of upgrades and interoperation of different versions. Guiding architectural principle: version at parent node must be same or newer than version at child node.
• Operator monitoring and control of huge numbers of elements. Guiding architectural principles: Minimize required interventions. Enable status-at-a-glance summary views with drill-down. Create rules for prevention and handling of conflict among operators at different levels of the system.
• System management. Need to understand the behaviour of parts, their integration, and the system as a whole, in order to continuously improve and optimize the system. Approach: Automated analysis of logs and engineering data to develop models of system behaviour and detect potential problems and optimization opportunities. What-if analysis of resulting models to select optimizations, including cost reduction opportunities.
• Software maintenance and evolution costs, agility. Guiding architectural principles: Use common software across the system as much as possible. Leverage public domain and COTS components, and make it easy to integrate future COTS software. Use specification-driven approaches and plug-ins to increase agility.
• System rollout and evolution. Need to support phased construction and deployment. Guiding architectural principles: Specification-driven controllers that can be configured to accommodate control of any type of system element. Hierarchical modular approach to system control based on semi-autonomy, so that only controllers at the integration point need to be modified to accommodate new system elements.
• Engineering properties. Achievement of performance, reliability, availability and security goals. Approach: Toolsets for automated analysis of engineering properties, and calibration of engineering models based on acquired data about system behaviour.

Primary author: Dr NATARAJAN, Swaminathan (Tata Research Development &amp; Design Centre, Tata Consultancy Services)

Co-authors: Dr AHUJA, Amrit (TRDDC, Tata Consutlancy Services); Dr VIN, Harrick (TRDDC, Tata Consultancy Services)

Presenter: Dr NATARAJAN, Swaminathan (Tata Research Development &amp; Design Centre, Tata Consultancy Services)

Session Classification: Signal Transport, Signal Processing, Software, & Data Management

Track Classification: Signal Transport, Signal Processing, Software, and Data Management