Quick Start Guide to DXC'10 Industrial Track

This guide is intended to assist participants in DXC10. We spell out what information is provided to participants, what is expected from them, and the overall process. For a schedule of when information/documents will be provided, please see the DXC10 announcement document.

1. Development Phase: Build models and develop diagnostic strategies
   a. Provided: System description information in the form of xml catalogs, schematics, additional documentation, sample data, and an Oracle that maps diagnoses to recommended recovery actions. Refer to the Industrial Track Descriptions document for details.
   b. Expected: Diagnostic Algorithm (DA) developers understand system components, failure modes, parameters, and build models and diagnostic strategies to diagnose failures in the system and recommend recovery actions (with the assistance of the Oracle) that mitigate the failure(s). Refer to the Oracle section in the DXC10 Framework README for more details.
2. Preliminary Integration Phase: Interface preliminary DA with DXC framework
   a. Provided: DXC10 Framework API, this will be detailed in a document to be released later, however refer to DXC09 API documentation for background information.
   b. Expected: DA reads sensor data and commands and sends diagnostic output and recommended recovery actions using API calls. Refer to the sample DA provided with the DXC10 Framework.
3. Testing Phase: Verify DA can connect with DXC10 Framework using sample scenarios
   a. Provided: DXC scenario loader and sample scenarios.
   b. Expected: Developers run their DA on sample input .scn files using the scenario loader on their own machine to make sure the DA runs without errors and creates an output .scn file (one per scenario). Refer to DXC10 Framework README for details on how to run the scenario loader.
4. Self Evaluation Phase: Verify DA scenario output can be evaluated
   a. Provided: Evaluator software.
   b. Expected: DA runs evaluator software on output .scn files to check for conformance to system catalog naming conventions. Developers can update their models and algorithms based on the evaluator output and repeat steps 1-4 to correct errors and improve performance. Refer to DXC10 Framework README for details on how to use the evaluator.
5. Submission Phase:
   a. Provided: Website for algorithm submissions
   b. Expected: DA developers submit a single compressed archive, which contains a Windows or Linux executable with no external dependencies and all supporting files, including models, configuration files, etc. Refer to DXC10 website for more information.
6. Optional Paper Submission (strongly encouraged)
   a. Provided: Opportunity to get paper published in DX10 workshop proceedings.
   b. Expected: Paper describing algorithm and approach taken to DXC10 competition.