logi.DIAG
Test Driven Automation and Condition Monitoring

20 Jahre Erfahrung in der Automatisierung
**Automation** is the use of control systems (such as numerical control, programmable logic control, and other industrial control systems), in concert with other applications of information technology (such as computer-aided technologies [CAD, CAM, CAx]), to control industrial machinery and processes, reducing the need for human intervention.[1] In the scope of industrialization, automation is a step beyond mechanization. Whereas mechanization provided human operators with machinery to assist them with the physical requirements of work, automation greatly reduces the need for human sensory and mental requirements as well. Processes and systems can also be automated.

http://en.wikipedia.org/wiki/Automated
Motivation - Automation

- Increasing Requirements
- Ever-increasing Complexity
- Larger and more complex software
Two Approaches for one Solution

- **Test-Driven Automation**
  - To assist during design & development of automation solutions

- **Condition Monitoring**
  - To identify upcoming system failure at an early stage
logi.DIAG: Test Driven Automation

- Reusing concepts from business IT for automation systems development
  - Test Driven Development Process
  - Testable Application Architecture
  - Testing Practices like Unit-Tests
- Integration with process supporting tools like
  - Lifecycle-Management Tools
    (Requirement-, Test-, Issue-, Configurationmanagement)
  - Tools for Automated Testing
logi.DIAG: Condition Monitoring

- Identify a failure before it occurs, e.g.,
  - Monitor the sound of a motor’s bearing and warn if the motor “sounds like becoming defective”

- Current approaches are very simple
  - Monitoring of single (or a few) sensor values
  - Compute some characteristic data/value (e.g. using an FFT)
  - See if characteristic data/value fits into an acceptance region
  - Issue an error

- Benefits
  - Maintenance intervals can be scheduled according to machine condition, not just every \( n \) months
  - Spare parts can be ordered in advance and do not have to be put on stock
  - Downtime can be minimized (minimum loss of production)
logi.DIAG: Condition Monitoring

- Connection of Automation and Condition Monitoring
  - Integration of diagnosis into automation application architecture
  - Utilize diagnosis interfaces used for unit-testing to supply condition monitoring applications with process related data

- Development of Data-Mining Algorithms for Real-Time Data Processing

- Add statistical methods to the simple algorithms currently used in condition monitoring systems
  - Also allows to “collect data” over a rather long period of time without extensive need of resources
  - Can extend Condition Monitoring from single components (e.g. monitoring a drive’s bearings) to larger parts of automation solutions (e.g., machine or even complete production line/plant)
Condition monitoring:
Storing statistical data should be efficient timewise and spacewise

☐ We represent data by a set of representatives of a fixed size

☐ Instead of computing summary statistics with the original (large) data set which is constantly growing we compute the statistic using the “empirical statistics” with the representation data set

☐ Data are collected continuously, and when a given number of new data is available, the representation data set is updated

☐ Method is “universal” we do not restrict the set of available statistical methods by “condensing” the data
logi.DIAG: Efficient data monitoring

Original data set (growing) -> Summary statistic

Representation data set -> Summary statistic
Algorithm (Chambers et al)

- For the set of representatives compute the interpolating distribution function (representatives are used as quantile points)
- For the data set to be used for adjustment compute the (discontinuous) empirical distribution function
- Compute the mixture distribution of these two distributions. Weights in the mixture are accumulated number of points and size of the new data set
- Compute quantiles of mixture distribution and use these as the new representation data set
- This algorithm does NOT allow generalization to multivariate distributions easily.
logi.DIAG: Efficient data monitoring

- Multidimensional algorithm
  - Compute principal component directions for the data
  - Compute projections of the data sets on the components
  - Perform one-dimensional Chambers algorithm for each projection
  - Combine updated projected representatives to new set of representatives
Showcase application: coffee machine
- We compute the sound spectrum of the noise
- We want give a warning if coffee will run out soon
- Spectrum has characteristic behavior for fineness of grinding
- Spectrum is noticeably different when coffee will run out

Implementation
- Compute average energy in selected spectral segments in real time
- Do multidimensional plot (lattice plot) with marked regions for different states of the machine
Showcase Application: Coffee Machine
logi.DIAG: Issues with R Implementation

- Algorithms are run on automation systems
  - Slow CPU (even 16-bit CPUs with < 10MHz)
  - Few memory (beginning with 2KB RAM+Flash up to ~ 1MB)
  - No math processor
  - Real-time operating systems

- Reimplementation required
  - Porting from R to C
  - Using as few R packages (and functions) as possible
  - Tuning for memory and operations (floating point?)
Thank you for your interest

Thomas Baier (mailto:thomas.baier@logicals.com)
Erich Neuwirth (mailto:erich.neuwirth@univie.ac.at)

www.logidiag.at