

Autonomous Computing – Rule-based improvements

By Tim Dellas. Ascora GmbH

Some questions for you

- Would you want your monitoring system to not just notify you if a measured KPI reaches a critical value, but instead directly take action on it?
- If data needs to be looked at and complex rules must be considered to react to undesired changes, is there an easy way to set this up?

Motivation

Autonomous Computing is a research field, where feedback loops and AI are concerned with process control and keeping or improving the status quo. AI has come a long way, but still the decisions met by machine learning cannot be understood or justified logically. When control of expensive machinery is considered, and when a standstill of eg a conveyor belt can lead to massive losses and contractual penalties, responsibility of control of such machines currently must be with human operators. This is the main reason ZDMP decided to implement autonomous computing in the form of an easily configurable GUI controlled rule-based machine.

The ZDMP Autonomous Computing component allows you to define triggers and actions on top of it. Simple triggers allow limits on numeral values that can be defined with comparative operators (ie checking if a value is above, equal or below a certain value), and an action can be triggered when the condition is fulfilled. Complex triggers allow functions such as evaluating strings with simple String operators (beginsWith, endsWith, contains, substrings), as well as using regular expressions on them, or even combining multiple conditions with boolean operators, and involving timing constraints. Moreover, variables can be defined to be forwarded to an action.

Requirements and initial assumptions

The autonomous computing is based on the Monitoring component, as here all the information and values converge. On top of this, an “if-this-then-that”-like approach was created, to be easily able to create rules that would then automatically trigger actions, which, in turn, could be used to send information and alerts to the operators or even directly be targeted at APIs within the system.

Triggers, KPIs and the Message Bus

Supervisors and Managers usually “walk” within factories, checking if the schedules are on time, if the warning lights show green, the shifts and workers are at their designated workplaces, the transports arrive and leave as planned, and workers talk to them when there are problems, quality issues or personnel issues of any kind. Much of this can be translated into real-time data using physical or virtual sensors (from ZDMPs Data Acquisition component) or by using standard interfaces such as Prometheus. This data will then be sent via the ZDMP Message Bus whereas the ZDMP Monitoring and Alerting component can be used to define KPIs to watch for.

Key performance indicators (KPIs) are values in data that an operator might wish to look for. Is the temperature of the machine right? Are the cycles per minute in the right area? By how much is the failure tolerance of the test used, indicating a slightly off configuration. KPIs might be hidden in data items or structured as a text. They might have to be extracted using a query where a query topic is defined and the KPI can be extracted from incoming data items.

Extracted numeral KPI values can be compared to standards or possible deviations from a standard can be defined.

Actions, Variables and Receivers

A KPI comparison triggers a notification/alarm, when the measured value is beyond the defined threshold. Within ZDMP the Monitoring and Alerting component allows the alerting of a manager or a supervisor as a standard action. The value of the KPI, the complete data item, the topic and the sender of the KPI are variables that can be forwarded with the message to a receiver (manager, supervisor, admin, etc.) or to a receiver group (eg “line-10-workers”, “sector-A-managers”, etc).

The autonomous computing component can be used with the actions to call URLs or REST interfaces to directly take measures when the KPIs are beyond the defined thresholds. For example, when the flywheel of a machine spins too fast, the machine speed can be reduced with an API before involving operators.

Documentation

Current documentation / tutorials of the Autonomous Computing and the related Monitoring and Alerting can be found [here](#).

What will ZDMP achieve

ZDMP will enable the user to track KPIs via the Monitoring and Alerting. Using the Autonomous Computing, the user can define conditions / triggers when the KPIs and set thresholds are contravened and configure actions to automatically handle the situation based on rules or even scripts before or additionally to notifying operators.

ZDMP Links

• Architecture Component(s)	Autonomous Computing
• Work Package	WP5 – ZDMP Platform Building
• Tasks	T5.5 – Distributed & Autonomous Computing