# ConicIT Technical Brief
# Behavioral Performance Analysis for Dynamic Thresholds

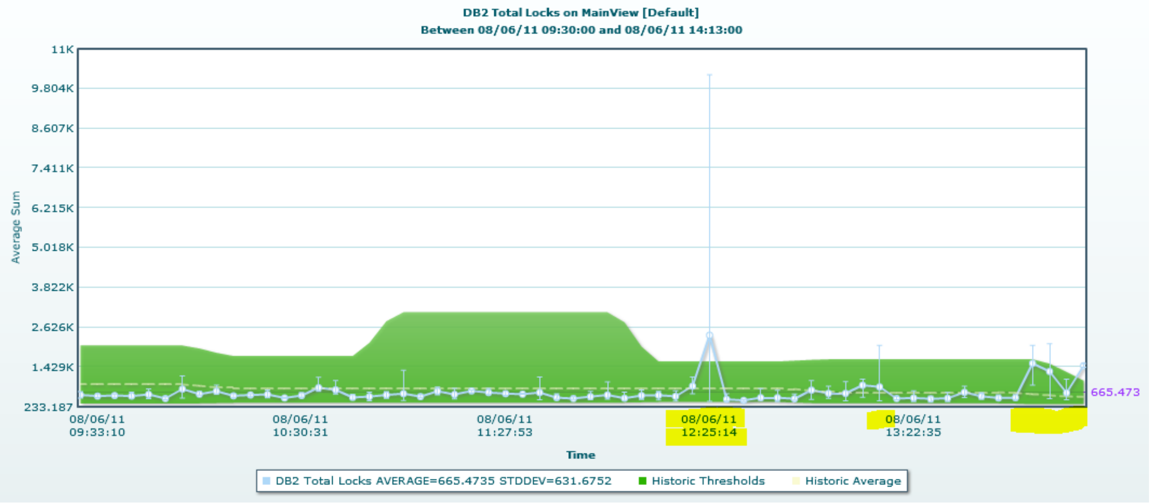
In performance monitoring thresholds are used to define the bounds of expected (or normal) performance of system variables like CPU, dispatching wait time, DB buffer usage or the current number of active DB locks.  Once thresholds are breached, the system is no longer acting normally – which is usually indicative of a problem that needs to be attended.

Thresholds are usually defined as fixed thresholds which are easy to define manually, but suffer from over generalization – not taking into account a variable's (values) behavior changes over time. Fixed threshold breaches are a lagging indicator - identifying an extreme failure (like 100% CPU utilization) after it has already happened – making them almost useless for real world performance alerts and management. Using fixed threshold systems leads to either setting the thresholds too low and receiving too many alerts (causing technical staff to ignore these alerts), or setting thresholds too high and missing real problems.

Dynamic thresholds solve the issues associated with fixed thresholds by using a behavior profile describing the dynamic nature of normal behavior (e.g. how a variable is different at different times of the day, days of the week, and special days like the first of the month). Dynamic thresholds provide the basis for accurate real time alerts by providing an unlimited set of context relevant thresholds. The reason that dynamic thresholds aren't in widespread use is that they can't be calculated manually. For example to define the dynamic threshold for variable like DB2-locking-rate or the CPU usage would require at least 8700 thresholds (one per hour) for a single system for single year. Just consider how difficult it would be to define the same for the hundreds and thousands of values of each system variable, manually.

While dynamic thresholds are much better than fixed thresholds, on their own they still aren't accurate enough for performance alerts. Not every breach of a dynamic threshold is a reason for an alert – that is where performance models come into play. These models understand semantics of system and application performance, and can differentiate between a threshold breach that should generate an alert and a threshold breach that should be ignored in the current context. The best performance models are layered models (e.g. generic performance models, virtualization performance models, mainframe performance models) where the models are combined to provide the most accurate alerts possible for a specific installation.

ConicIT automatically (without any user involvement) calculates dynamic thresholds defining a Dynamic Behavior Profile for each variable, and constantly updates it to adapt to changes in the system. These Dynamic Behavior Profiles define the dynamic thresholds needed for accurate, meaningful real time alert candidates. These alert candidates are then fed into the appropriate models which decide whether an alert should be issued, or if the anomaly can be ignored.

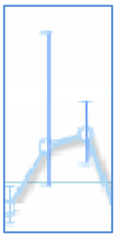


Graph components:

Current data (actual graph values)

Prediction area, expected values (max, average, min)
The "prediction green area" shows the expected (i.e. predicted) future values that ConicIT calculates for every parameter (variable).

These predicted values are calculated every day at midnight using the variable’s historical values.
Generally this is the most complicated task in ConicIT analysis system



Vertical bars that represent max and min values for an average point value

For example: in the graph shown, the green area represents the predicted normal behavior of the number of DB2 locks. Each point on the graph actually represents an average of the number of DB2 locks held during the time the point represents (the vertical line displays the min and max number of locks held). From the graph, it is clear that there is one unusual peak outside of the predicted behavior boundaries at around 12:25 a.m. and two less pronounced peaks (around 13:30 and 14:00). Using a fixed threshold with the threshold set too low would cause too many alerts to be issued (at 12:25, 13:30 and 14:00, which means 66% of the alerts are incorrect), while setting the threshold too high would cause the alert on the real problem to be missed.

ConicIT's dynamic threshold would automatically define different thresholds for 12:00-13:00 and 13:00-14:00 and alert candidates would be generated only when a real change in behavior has occurred which may be indicative of a real problem (i.e. only at 12:25). These alert candidates would be passed the performance models which would decide whether this change in behavior warrants issuing a real alert or not, given the current context.