



SUM 4 Java Threads Analyzer – product categories:

1. SUM4JTA as A Dynamic Code Analyzer (deadlocks, race-conditions, profiling)

[] **Multithreading contentions analysis.** There are two major Java multithreading contentions - **deadlocks** and race **conditions**:

a. deadlocks - once an application gets in a deadlock, it stays in a deadlock. Analysis here is well understood and is also provided by operating systems, JVM's and alike. [SUM4Java provides an additional level of automation in dead locks awareness, notification and automatic analysis.](#)

b. race conditions - silent and intermittent conditions of unpredictable results - [SUM4JTA rules in analysis of these conditions.](#)

Existing technologies suffer here in the following fronts:

- (i) High overhead - SUM4JTA overhead is **in the order of 10 times lower** than the other leading products
- (ii) High degree of False Positive Results - SUM4JTA provides **0% false positive** results comparing to the 15% and higher result of other leading products.
- (iii) Necessity of manual preparation and integrations with IDE's - Interactive Development Environments - SUM4JTA does not require IDEs and **does not need applications source code.**

[] **Profiling** – SUM4JTA provides *basic Profiling functionality* – analysis of bottlenecks – timing, invocation count

2. SUM4JTA as a Program Understanding Tool - we call it Black Box Opener - [SUM4JTA rules here](#). We have not seen the level of automation, the level of details, and the level of uniqueness of the graphical user Interface elsewhere.

3. SUM4JTA Enterprise Edition (SUMEE) as Data Centers Tech Support Knowledge Station - we call it a "mother ship station" collecting all experienced multithreading issues within an organization or in a data center - [SUM4JTA rules here](#)