

# Context Factor in Choosing a Debugging Strategy

Context factors refer to the specific elements and circumstances surrounding a particular situation or problem that can significantly influence the choice of a debugging strategy for solving it. These factors play a crucial role in shaping how developers approach and address software issues. They encompass a wide range of aspects, including:

## DEFECT CHARACTERISTICS

such as its complexity, frequency of occurrence, and criticality

- The bug is happening every time you run the program.
- Can't make the bug happen.
- The defect is intermittent.
- The bug is not evident, or you cannot find the problem.
- The bug is related to data.
- The bug is related to a function call(focused).
- Can localize a variable with an incorrect value(focused).
- Cannot clearly identify the code line associated with the failure / If there is a particular line of code that breaks the program.
- The defect is related to front end.
- There are many different error messages.
- The fault localization gets tough and you randomly poking the system to find the fault.
- The defect affected the performance.

Commented [1]: replicability

Commented [2]: code-independent

Commented [3]: Root (focused, frontend, Errorstack)

Commented [4]: STUCK

Commented [5]: Network

## CODEBASE CHARACTERISTICS

like its size, architecture, and maintainability:

- Software takes too much memory or takes too long to respond.
- Program freezes.
- The program already has test cases.
- Error associated with small segment of the code.
- Size of the code.
- Legacy code.
- Modularity of the code
- Architecture of the code
- UI logic
- Testability

## ORGANIZATION CHARACTERISTICS

such as its size, structure, and development methodologies:

- Cost consideration
- Time consideration
- Knowledge sharing culture
- Any development culture (e.g., Agile, Collaborative, ....)
- Availability of managers or experienced teammate.
- Project with tight deadlines.

## TOOL CHARACTERISTICS

how a developer is familiar with a specific tool, or how the use of a tool in the codebase affect your choice:

- Availability of underlying structure of software and supportive tools
- The availability of Performance Profiler
- The availability of logs
- The availability of debugger tools
- The availability of configuration management tools (Git, Subversion, Mercurial)
- The availability of testing

## INDIVIDUAL CHARACTERISTICS

preferences of the developers involved:

- Developer expertise
- Familiarity with the programming environment and language
- Habit to specific approach

## OTHERS