Chapter-8
Tool support for Testing

classmate
Date
Page

- The tool are grouped by the testing activities.
- Some tools perform a very specific and limited function but many of the commercial tools provide support for many different functions.
- Tool support is very useful for repetitive tasks. The computer doesn't get bored and will be able to exactly repeat what was done before without any mistakes.

- Since the tool will be fast, it can perform its tasks much more efficiently and reliabely.

Different types of test tools according to test process activities:

1) Tool support for management of testing and tests.
   - Test management tools.
   - Requirements management tools.
   - Incident management tools.
   - Configuration management tools

2) Tool support for static testing
   - Review process support tools
   - static analysis tools
   - Modelling tools.

3) Tool support for test specification
   - Test design tools
   - Test data preparation tools.

classmate
Date
Page

4) Tools support for ~~[illegible]~~ test execution and logging
- Test execution tools
- Unit Test framework tools
- Test comparators
- coverage measurement tools.
- security tools.

5) Tools support for performance and monitoring
- Dynamic analysis tools.
- Performance testing, load testing and stress testing tools.
- Monitoring tools.

(1) ✱ Static Analysis Tools
- These tools are generally used by developers as part of the development and component testing process.
- Here, the code is not executed but the tool itself is executed and the source code we are interested in is the input data to the ~~data~~ tools.
- These tools are an extension of compiler technology.

Features/ characteristics
- To calculate metrics such as cyclomatric complexity or nesting levels.
- To enforce coding standards.
- To analyze structures and dependencies
- To help in code understanding
- To identify the defects in the code

classmate
Date
Page

(2) Test Management Tools:

Feature / characteristics
- To manage the tests like
  - keeping track of same kind of data for given set of tests.
  - know which test need to run in common environment
  - know number of test planned, written, run, passed or failed

- To schedule the tests that need to be executed.

- To manage test activities like.
  - time spent in test design, test execution.
  - keep track on whether we are on schedule and on budget.

- To track the test results and defects

- To provide interface to other tools such as
  - test execution tools
  - incident management tools
  - requirement management tools etc.

- To log the test results
- To prepare the progress report based on metrics such as
  - total no: of test runs.
  - total " " test passed
  - " " " incidents raised.
  - " " " ~~test objects~~ defects fixed

③ Configuration Management Tools.

- These tools are not strictly testing tools but these tools are used for good configuration management for controlled testing.
- Features:
  - To store information about version and builds of the software and testware.
  - To keep track of which versions belong with which configurations (e.g. operating systems, libraries, browsers)
  - To perform build and release management
  - To perform baselining
  - To perform access control (checking in and out).

4) Incident Management Tools.

- It is aka bug-tracking tool, defect-tracking tool or a bug-management tool.
- Here the information about the failure (not the defect) that was generated at the time of testing and the info. about the defect that caused the failure are recorded.

Features:

- To store the information about the attributes of incidents (eg severity)
- To store attachments (eg screenshot)
- To prioritize incidents.
- To store the status of failures (eg: duplicate, ready of confirmation test etc).
- To report the metric about incidents (eg: total no. of occurance etc.)

⑤ Requirement Management Tools:-

- The better the quality of the requirements the easier it will be to write tests from them.

Features:

- To store the requirement statements.
- To store the information about requirement attributes.
- To check consistency of requirements.
- To identify undefined, missing or 'to be defined later' requirements.
- To prioritize requirements for testing purposes.
- To trace the requirements to test and tests to requirements
- To trace through all the levels of requirements.

⑥ Test Data preparation tools:-

- It allows data to be selected from an existing database
- It also allows data to be created, generated, manipulated and edited for use in tests.

Features/characteristics

- To select data and records from file and databases.
- To enable records to be stored or arranged in different order.
- To construct a large number of similar records from a template

⑦ Test Execution Tools : (aka Test Running Tool)

- These tools need a scripting language in order to run the tool.
- These tools can repeat test actions (in loops) for different data values (i.e. test inputs)
- There are 5 ways of scripting.
  - Linear scripts (which could be created manually or captured by recording ~~amount~~ a manual test)
  - Structured scripts (using selection and iteration programming structures)
  - shared scripts (where a scripts can be called by other scripts).
  - Data-Driven scripts (where test data is in a file or spread sheet to be read by a control script).
  - Keyword-Driven scripts: (where all of the information about the test is stored in a file or spreadsheet).

They are actually best used for regression testing.

Features / characteristics of test execution tools are:

- To capture (record) test inputs while tests are executed manually.
- To store an expected result in the form of a screen or object to compare to, the next time the test is run.
- To execute tests from stored scripts and optionally data files accessed by the script.
- To do the dynamic comparision (while the test is running).

- To initiate post execution comparision.
- To log results of tests run.
- To mask or filter the subsets of actual and expected results.
- To measure the timings for test.
- To send the summary results to a test management tool.

Imp Benefits of using testing tools:

① Reduction of repeatative work

- Repetitive work is boring, if it is done manually.
- People tend to make mistakes when doing the same task over and over (eg: Regression Testing).

② Greater consistency and repeatability:

- People have tendency to do the same task in a slightly different way even when they think they are repeating something exactly.
- A tool will exactly reproduce what it did before, so each time it is run, the result is consistent.

③ Objective assessment:

- If a person calculates a value from the software or incident reports, by mistakes they may ommit something or they may use own ~~jdg~~ judgement. This may lead them to interprete that data incorrectly.
- So, using a tool, the assessment is more consistently calculated. (eg : of assessment may be, calculating the cyclomatic complexity, or nesting levels of behaviour) etc.

(4) Ease to access to information about tests or testing

- Tools helps to present the information visually using graphs, charts etc.
- Information presented visually is much easier for the human mind to understand & interpret.

(5) Automate activities that cannot be executed manually.

Imp Risks or Disadvantages of using the testing tools.

1. Unrealistic expectations from the tool.
   - The tools are just software and we all known that there are many problems associated with any kind of software.
   - It is very important to have clear and realistic objectives for what the tool can do.
2. People often make mistakes by understimating the (time, cost and effort for the initial introduction of a tool is high)
3. People frequently miscalculate the (time and effort needed to achieve significant and continuing benefits from the tool is high)
4. Mostly people underestimate the (effort required to maintain the test assests generated by the tool is high)
5. People depend on tool a lot (over reliance on the tool).
6. Neglating version control of test assets within the tool.
7. Risk of tool vendor going out of business.
8. Poor response from vendor for support & upgrades.
9. Some tools are unable to support new platform.

Factors for the software testing tool selection.

- Assessment of the organization's maturity (eg readiness for change), strengths and weakness.
- Identification of the areas within the organization where tool support will help to improve testing processes
- Evaluation of tools againsts clear requirement & objective criteria.
- Evaluation of the vendor (training, support & other commercial, aspects) or open-source network of support.
- identifying and planning internal implementation (including coaching and monitoring for those, new to the use of tool).
- A proof-of-concept by using a test tool during the evaluation phase to establish whether it performs effectively with the software under test and within the current infrastructure or to identify changes needed to that infrastructure to effectively use the tool.

- Evaluation of training needs considering the current test teams, test automation skills.
- Estimation of a cost-benefit ratio based on a concrete business case.
- ways of using, managing, storing and maintaining the tool and the test assets.
- Assess whether the benefits will be achieved at reasonable cost.