Open Source Software

Lecturer: Taghreed Alreffaee

Lecture #7

Objective:-

- Community
- Community Driven Development
- Software Development
- Software Building
- Software Testing
- Software Testing Types
- Verification and validation

Community

- A community is a diverse group of people sharing their thoughts, ideas, works, and experiences, at a common place.

- a group of people who live in the same area (such as a city, town, or neighborhood) and have the same interests.

Community -Driven Development

- Community-driven development(CDD) is an initiative in the field of development that provides control of the development process, resources and decision making authority directly to groups in the community.

- Community-Driven Development (CDD) is an approach to local development that gives control over planning decisions and investment resources to community groups.

- Community-driven development (CDD) is a modality of project design and delivery which transfers decision-making power and, often financial and technical resources, directly to communities or groups of end-users

Community Driven Development(contd.)

- In the software industry, community driven development means an initiative under which a group of technologists work together with a common vision of producing open source software. Most of the OSS development communities who are successful today, organize themselves in a similar way as a professional organization or proprietary software company. In a well-structured OSS community; you can find each member has a job role within the team. Groups of people performing similar jobs, form sub teams within the community (multiple sub teams collaborate to develop open source software).

1 – *The developers' group:* is responsible for writing code for independent modules of the software.

2 – *The builders' group:* takes these modules from the developers and put them together to build a new version of the software.

3 – *testers' group:-* These software builds are internally tested by the *testers' group*. In case a bug or failure is found, they report back to the appropriate developer who debugs the code and includes a proper fix. This is an iterative process that repeats itself until the product satisfies all its requirements and becomes error free.

4 – *The release management group:* packs together the final version (final build) of the software with all the necessary documents once the objectives of the project are met, and then hands it over to the customers.

Software Development(The developers' group):-

- is a group of people who work together to develop a software, product, or service .

- a project development team can actually be any team focused on developing a particular project, whether it be constructing a building or manufacturing a new toy.
- A project development team often includes:-

1- project manager.

2- business analyst.

3- software architect.

4- software designer.

4- Developer…. etc.

Software Building (Software's Builders):-

- If you are an IT professional who is working with a software development team you will come to know this term "Build" various times.

- The software build is a general term in the software development world. The software build is an activity to translate the human-readable source code into an efficient executable program.

- It is the process of creating the application for a software release, by taking all the relevant source code files and compile them and then creating a build artifact (such as executable program)

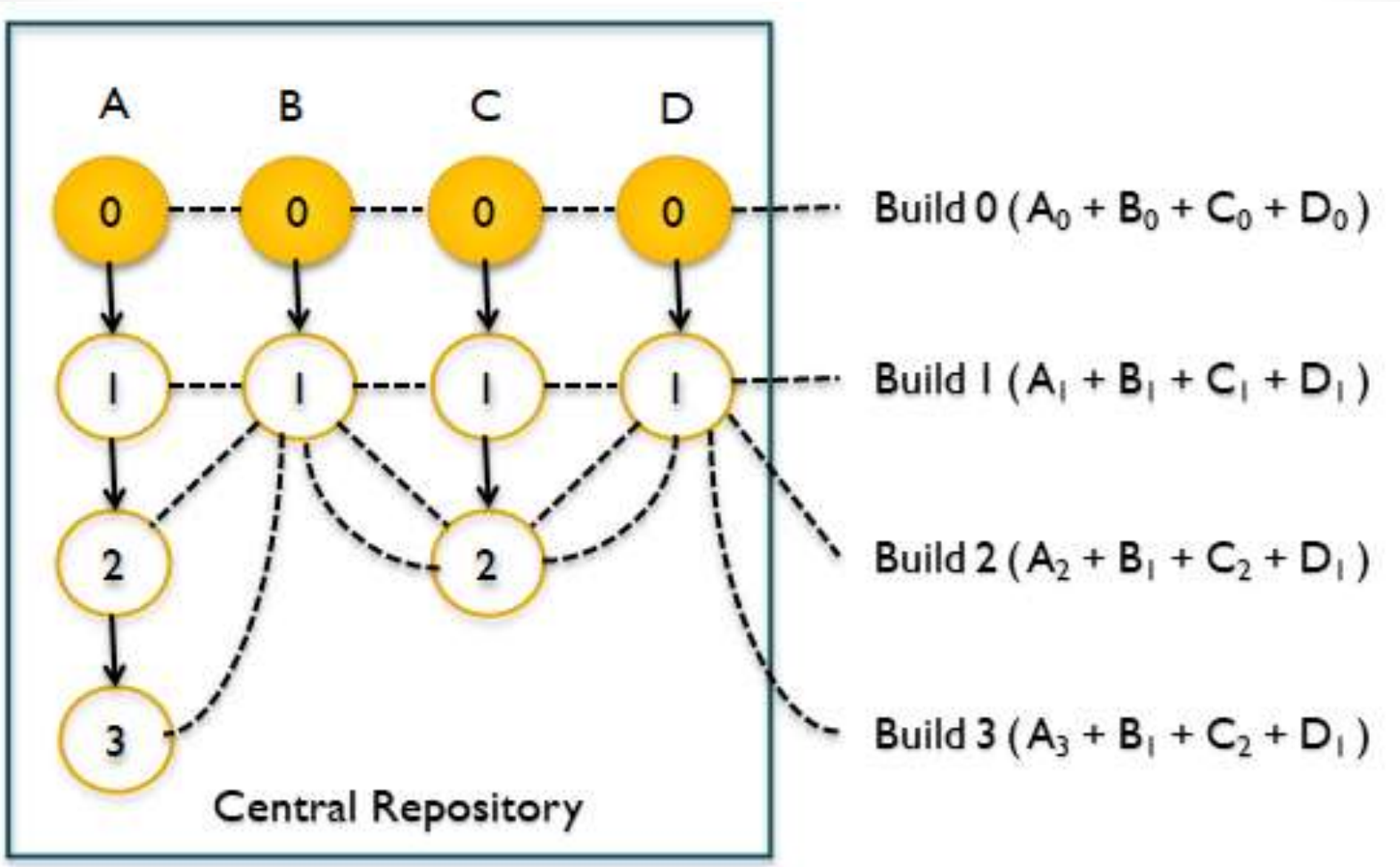
- The compilation process, in which source code files are turned into executable code, is one of the most important processes in the software development process.

- A build tool is typically used to manage the software development process. Builds are made when the code has been considered ready for validation or release.

- a build in software development is when your engineers convert the source code into a functional, standalone program with executable code that can be run on a desktop computer or mobile device.

- A software build is created when your project reaches a certain point in development. First, your designers, software architects, and business analysts outline their vision for product implementation. This vision is then turned into project specifications that your developers base the source code on. When the developers finish writing the source code, and the backbone of your program meets your project specifications, this usually means that the source code is deemed ready for implementation. It will be turned into a software build and thoroughly tested before the release.

- Development communities generally have a separate build team to carry out this task. It is important to mention that the build process recompiles only those source code files which were modified since the previous build was released. For the rest of the elements which remained uncharged, the older compiled modules are picked up.



"Software Builds"

- In the figure, the central repository contains FOUR source code files A, B, C, and D. The base version of these files is illustrated with the number ZERO, so A_0 represents the base version of source code file A for example. When compiled and linked they create 'Build 0'. On the next iteration, all the files were modified and their versions were upgraded to 1. Since all the files were modified, all are compiled and linked to create 'Build 1'. On the next iteration, only source code file A and C are modified creating second versions of them (A_2 and C_2). As a result when creating 'Build 2', only A_2 and C_2 are compiled, and then linked with B_1 and D_1. Similarly, 'Build 3' is created based on the newly compiled A_3 and precompiled B_1, C_2 and D_1.

Software Testing(Tester's Group):-

- is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free.

- examination of the behavior of a program by executing the program on sample data sets.

- It is a process used to identify the correctness , completeness and quality of developed computer software.

- It is the process of executing a program \application under positive and negative conditions by manual or automated means . It check for the:-
- Specification
- Functionality
- performance

H.W

- What is the difference between:-

- Defect
- Error
- Failure

Who does Testing?

- In the IT industry, large companies have a team with responsibilities to evaluate the developed software in context of the given requirements. Moreover, developers also conduct testing which is called Unit Testing. In most cases, the following professionals are involved in testing a system within their respective capacities:
- Software Tester
- Software Developer
- Project Lead/Manager
- End User
- Different companies have different designations for people who test the software on the basis of their experience and knowledge such as Software Tester, Software Quality Assurance Engineer, QA Analyst, etc. It is not possible to test the software at any time during its cycle.

When to Start Testing?

- An early start to testing reduces the cost and time to rework and produce error-free software that is delivered to the client. However in Software Development Life Cycle (SDLC), testing can be started from the Requirements Gathering phase and continued till the deployment of the software. It also depends on the development model that is being used. For example, in the Waterfall model, formal testing is conducted in the testing phase; but in the incremental model, testing is performed at the end of every increment/iteration and the whole application is tested at the end. Testing is done in different forms at every phase of SDLC:

- During the requirement gathering phase, the analysis and verification of requirements are also considered as testing.
- Reviewing the design in the design phase with the intent to improve the design is also considered as testing.
- Testing performed by a developer on completion of the code is also categorized as testing.

When to Stop Testing?

- It is difficult to determine when to stop testing, as testing is a never-ending process and no one can claim that a software is 100% tested. The following aspects are to be considered for stopping the testing process:

✓ Testing Deadlines

✓ Completion of test case execution

✓ Bug rate falls below a certain level and no high-priority bugs are identified

✓ Management decision

Software Testing types:-

There are many types of software testing:

1 – **Black Box Testing**: in this type of testing, the actual output of the code is compared with the expected result. Testers are supposed to feed the system with a set of inputs, and note down the corresponding outputs. In case of any mismatch between the obtained and the expected results they report it to the developer without analyzing the code.

2 – **White Box Testing**: this type of testing provides the tester with complete knowledge of the application being tested, including access to source code and design documents. This in-depth visibility makes it possible for white box testing to identify issues that are invisible to black box testing

H.W

- State other types of Software Testing.

Why Software Testing is Important?

- **Software Testing is Important** because if there are any bugs or errors in the software, it can be identified early and can be solved before delivery of the software product. Properly tested software product ensures reliability, security and high performance which further results in time saving, cost effectiveness and customer satisfaction.

- The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

- Testing is important because software bugs could be expensive or even dangerous. Software bugs can potentially cause monetary and human loss

- In May of 1996, a software bug caused the bank accounts of 823 customers of a major U.S. bank to be credited with 920 million US dollars.

- In 1985, Canada's Therac-25 radiation therapy machine malfunctioned due to software bug and delivered lethal radiation doses to patients, leaving 3 people dead and critically injuring 3 others.

Verification & Validation

Verification: "Are we building the product right".

- The software should conform to its specification
- Verification should check the program meets its specification as written in the requirements document for example.
- This may involve checking that it meets it functional and non-functional requirements.

Validation: "Are we building the right product".

- The software should do what the user really requires.
- Validation ensures that the product meets the customers expectations
- This goes beyond checking it meets its specification; as we have seen, system specifications don't always accurately reflect the real needs of users

Thanks For Listening