**SOFTWARE ARCHITECTURE ASSIGNMENT**

**SUBMITTED BY - PADMAPRIYA I**

**BITS ID - 2019AT15017**

**SUBMITTED TO - PROF. K. B. RAMESH**

**How would you measure the specific value of architecture in a project? How would you distinguish the value added by architecture from the value added by other activities such as quality assurance or configuration management?**

* Measuring the specific value of architecture in a project can be a complex task.
* One way to approach this is to quantify the benefits that the architecture provides, such as increased efficiency, better maintainability, reduced risk, improved user experience, etc.
* This can be done through metrics such as the number of bugs found, the speed of fixing problems, the speed of adding new features, and the overall user satisfaction.
* To distinguish the value added by architecture from the value added by other activities such as quality assurance and configuration management, it is important to understand the unique role that each of these activities plays in the development process.
* Quality assurance focuses on ensuring that the software meets the specified requirements and standards, while configuration management is concerned with tracking changes to the software over time and ensuring that the correct version is used at all times.
* Architecture, on the other hand, is focused on the overall design and structure of the software, including decisions about technology, scalability, security, and other high-level concerns.
* By analyzing the impact of each of these activities on the overall success of the project, it is possible to determine the specific value that each contributes.

**In a distributed system each computer have its own clock. It is difficult to perfectly synchronize those clocks. How will this complicate making performance measures of distributed systems? How would you go about testing that the performance of a particular system activity is adequate?**

* In a distributed system, the use of multiple clocks can lead to clock drift, which can cause inconsistencies in timing between the different components of the system.
* This can make it difficult to accurately measure the performance of the system, as different components may be operating at slightly different speeds.
* To test the performance of a particular system activity, one approach is to use a benchmarking tool that can measure the time taken for a specific task to be completed on each component of the system.
* This provides a baseline for evaluating performance, which can then be compared against system requirements to determine if the performance is adequate.
* Another approach is to use a distributed performance testing tool, which allows multiple components to be tested simultaneously and the results to be aggregated to determine overall system performance.
* This can provide a more comprehensive view of the performance of the system and help identify any bottlenecks or areas for improvement.
* In order to account for the effect of clock drift on performance measurements, it is important to use a consistent method of measuring time, such as using a common time source, such as NTP (Network Time Protocol), to synchronize the clocks of all components in the system.
* This helps to minimize the impact of clock drift on performance measurements, allowing for more accurate and consistent results.

**In embedded systems faults often occur “in the field” and it is difficult to capture and replicate the state of the system that led to its failure. What architecture mechanisms might use to solve this problem. Describe this scenario with an example.**

* One mechanism that can be used to capture and diagnose faults in embedded systems is a "fault tolerant architecture".
* This type of architecture is designed to detect, isolate, and recover from faults in the system without losing critical system functionality.

**Example:**

* Let's consider an industrial control system for a chemical plant. This system monitors and controls various processes, such as temperature and pressure, to ensure the safe and efficient operation of the plant.
* If a fault occurs in this system, it can be difficult to identify the root cause and reproduce the issue.
* To address this challenge, the system can be designed with redundant hardware and software components that can detect and isolate faults.
* For instance, the system can be designed with multiple sensors for each process variable, such as temperature and pressure, and the system can use voting mechanisms to determine the correct value in case of disagreement between sensors.
* Additionally, the system can have multiple processors running in lockstep to detect and recover from faults in the software.
* Moreover, the system can also be equipped with a real-time data logging mechanism that records the state of the system at the time of the fault.
* This data can be used to diagnose the fault and identify the sequence of events that led up to it.
* By analyzing the recorded data, engineers can identify the root cause of the fault and make changes to the system to prevent similar faults from occurring in the future.
* In summary, a fault-tolerant architecture can help address the challenge of diagnosing faults in embedded systems by providing redundancy, fault detection, isolation, and recovery mechanisms, and by logging data to help identify the root cause of the fault.

**Go the website of your favorite open source system. On the site, look for the architectural documentation for that system. What is there? What is missing? How would this affect your ability to contribute code to this project? Describe.**

* I have visited the website of VLC media player, and I was not able to find a comprehensive architectural document for the system.
* The website provides user documentation, technical documentation, and developer documentation, but there does not appear to be a single document that provides an overview of the system's architecture.
* Without a comprehensive architectural document, it may be more challenging for new developers to understand the overall design and structure of the system.
* This could make it more difficult for them to contribute code to the project, as they may not have a clear understanding of how their changes will affect other parts of the system.
* On the other hand, VLC media player is a well-established open source project with a large and active developer community.
* This means that there are likely many resources available to help new developers get up to speed on the project.
* For example, there may be forums or mailing lists where developers can ask questions and get help from more experienced members of the community.
* Overall, while the lack of a comprehensive architectural document may make it more challenging for new developers to understand the system, the active developer community and other available resources may help to mitigate this issue.