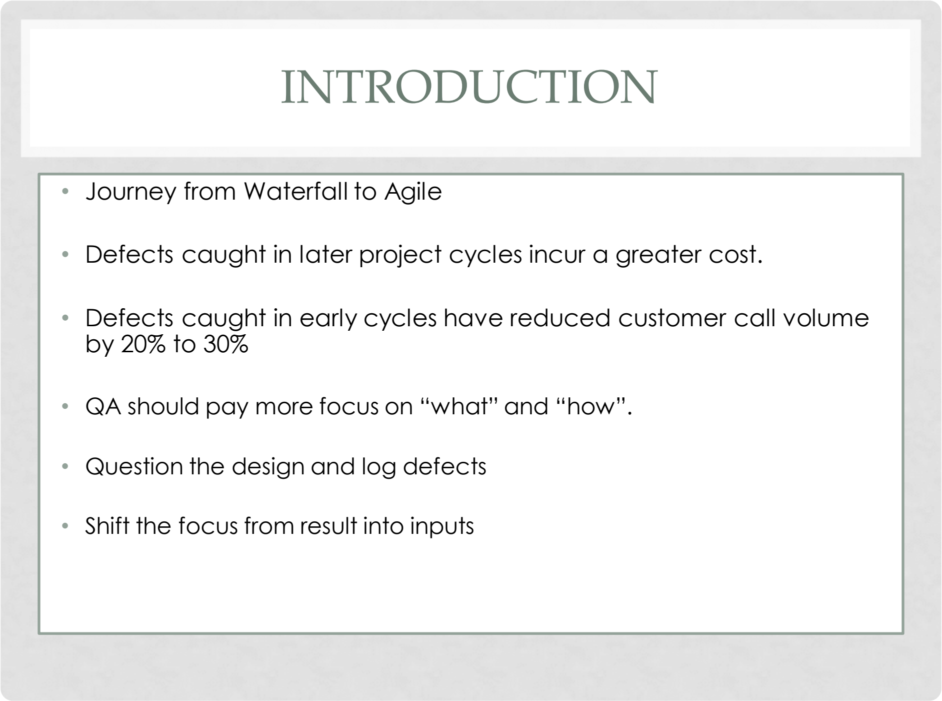
QUALITY BEGINS WITH DESIGN

AUTHOR: JEYASEKAR MARIMUTHU (MCAFEE)
DATE: 15-OCT-2013

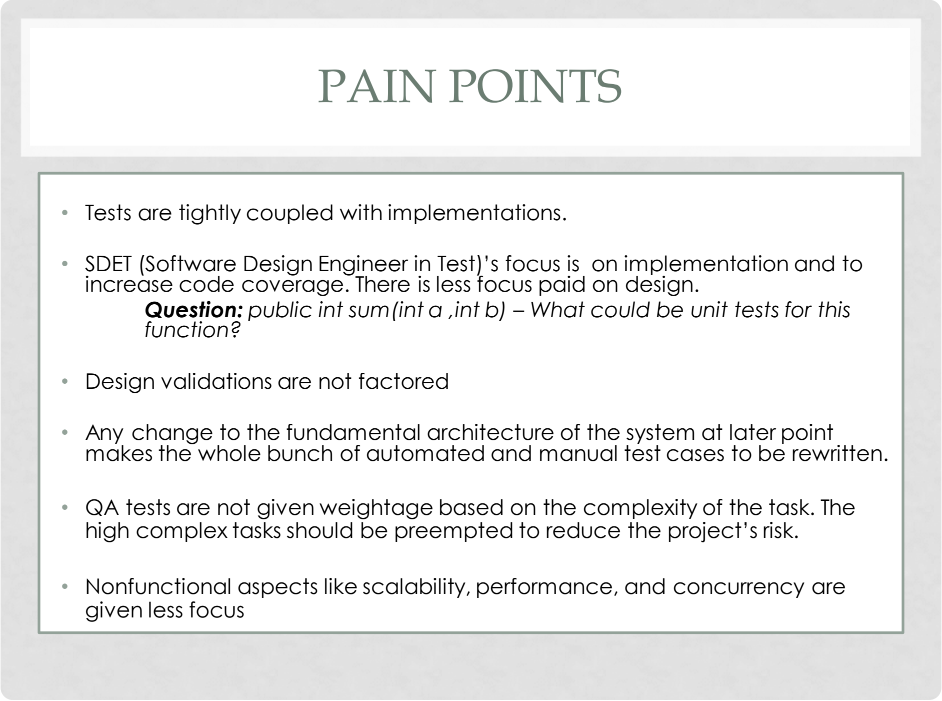
ABSTRACT

- Involving QA in the design phase
- Significance of having a Quality architect
- A new perspective to look at user stories by means of complexity, inter dependency, number of functions, Distributed coverage across layers
- Detection of issues in advance
- Optimized test plans and ownership assignment
- Well understood designs produce effective automation
- Non functional insights (Fault tolerance, extensibility, reliability, Maintainability and availability)



INTRODUCTION

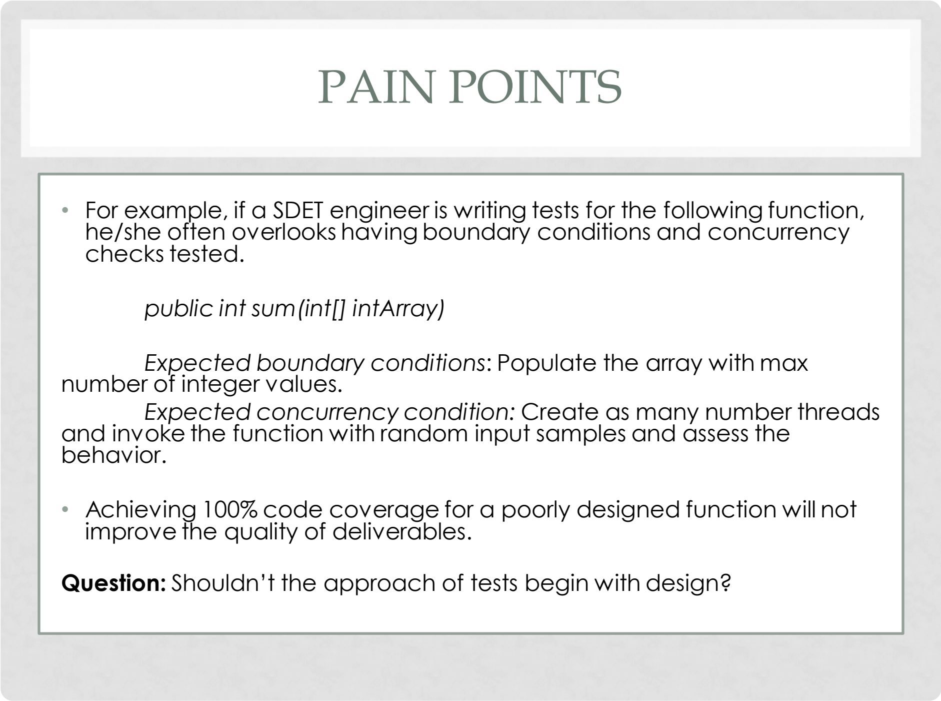
- Journey from Waterfall to Agile
- Defects caught in later project cycles incur a greater cost.
- Defects caught in early cycles have reduced customer call volume by 20% to 30%
- QA should pay more focus on “what” and “how”.
- Question the design and log defects
- Shift the focus from result into inputs



PAIN POINTS

- Tests are tightly coupled with implementations.
- SDET (Software Design Engineer in Test)’s focus is on implementation and to increase code coverage. There is less focus paid on design.

  Question: *public int sum(int a ,int b) – What could be unit tests for this function?*
- Design validations are not factored
- Any change to the fundamental architecture of the system at later point makes the whole bunch of automated and manual test cases to be rewritten.
- QA tests are not given weightage based on the complexity of the task. The high complex tasks should be preempted to reduce the project’s risk.
- Nonfunctional aspects like scalability, performance, and concurrency are given less focus



PAIN POINTS

- For example, if a SDET engineer is writing tests for the following function, he/she often overlooks having boundary conditions and concurrency checks tested.

public int sum(int[] intArray)

Expected boundary conditions: Populate the array with max number of integer values.

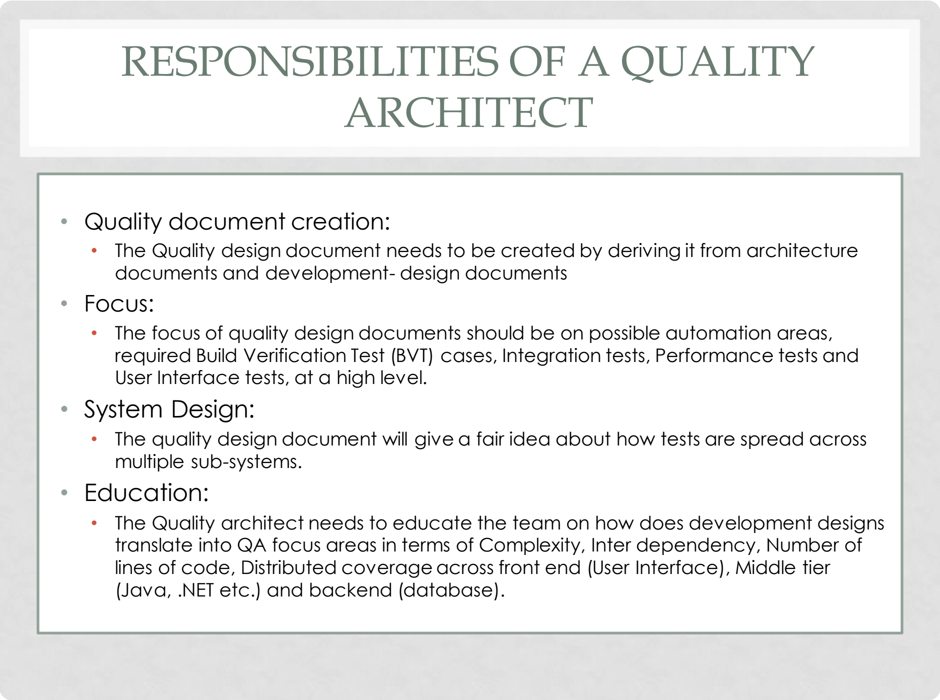
Expected concurrency condition: Create as many number threads and invoke the function with random input samples and assess the behavior.

- Achieving 100% code coverage for a poorly designed function will not improve the quality of deliverables.

Question: Shouldn't the approach of tests begin with design?

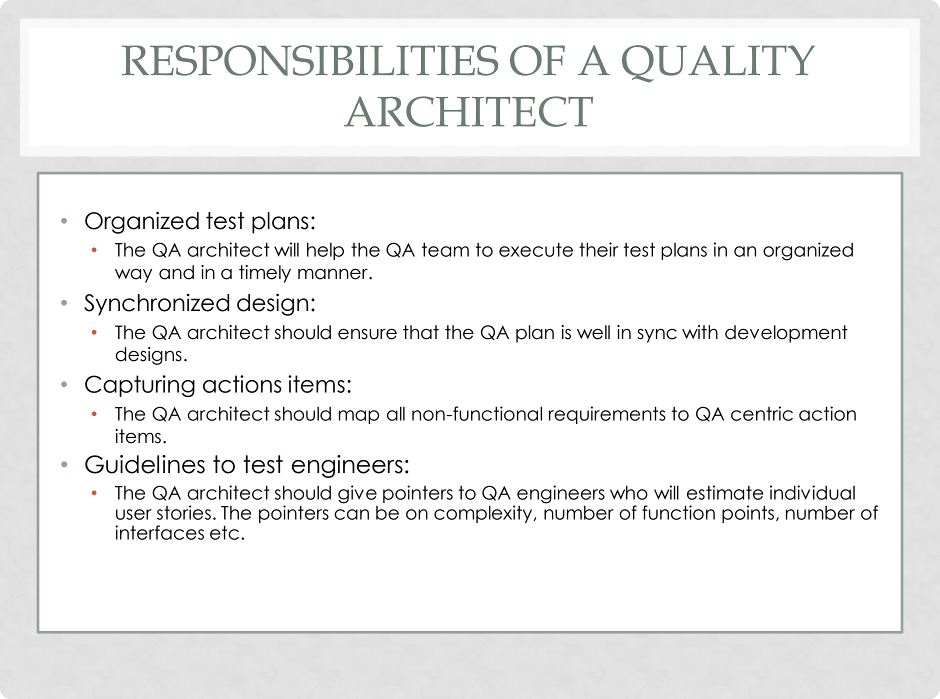
NEED FOR QUALITY ARCHITECT

- The design phase allows architects to come up with detailed system diagrams that capture the following.
  - Logical breakdown of functionalities against functional and nonfunctional requirements, focusing on Separation of concerns
  - Separation of concerns - An established way of reducing complexity.
  - Interface designs for internal and external facing APIs.
  - Defining nonfunctional system attributes such as Fault tolerance, specific set of tradeoffs, extensibility, reliability, maintainability and availability.
- In order to QA the designs, the QA architect should prepare test plans for each of the item mentioned above.
- The designated Quality architect will work closely with product architects and other design owners and will create a quality design document for the System Under Development.



RESPONSIBILITIES OF A QUALITY ARCHITECT

- Quality document creation:
  - The Quality design document needs to be created by deriving it from architecture documents and development- design documents
- Focus:
  - The focus of quality design documents should be on possible automation areas, required Build Verification Test (BVT) cases, Integration tests, Performance tests and User Interface tests, at a high level.
- System Design:
  - The quality design document will give a fair idea about how tests are spread across multiple sub-systems.
- Education:
  - The Quality architect needs to educate the team on how does development designs translate into QA focus areas in terms of Complexity, Inter dependency, Number of lines of code, Distributed coverage across front end (User Interface), Middle tier (Java, .NET etc.) and backend (database).



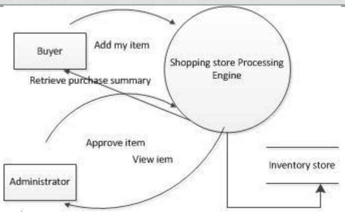
RESPONSIBILITIES OF A QUALITY ARCHITECT

- Organized test plans:
  - The QA architect will help the QA team to execute their test plans in an organized way and in a timely manner.
- Synchronized design:
  - The QA architect should ensure that the QA plan is well in sync with development designs.
- Capturing actions items:
  - The QA architect should map all non-functional requirements to QA centric action items.
- Guidelines to test engineers:
  - The QA architect should give pointers to QA engineers who will estimate individual user stories. The pointers can be on complexity, number of function points, number of interfaces etc.

SYSTEM DESIGN VERSUS QUALITY DESIGN

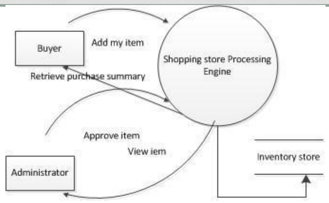
- Call out dependencies
- Define Complexities
- Test optimization
- Technical knowledge
- Cost Saving
- Well defined boundaries
- Collaboration
- Automation plan

DEV. DESIGN - ILLUSTRATED



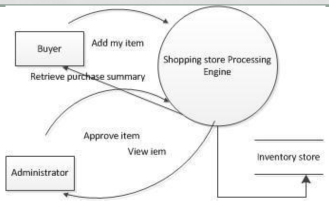
- Development Design:
  - Buyer object sends the shopping information to shopping cart.
  - The shopping store business component identifies the buyer by his session id and adds buyer's shopping data to inventory store.
  - Admin object requests for shopping info added by buyers.
  - The shopping store business component validates the admin's id and retrieves purchase requests from inventory store.
  - The shopping store Business component constructs purchase object and sends it back to admin.
  - Admin object views the info in the front end (view JSP) and clicks on approve
  - The shopping store Business component commits the order.

QA DESIGN - ILLUSTRATED

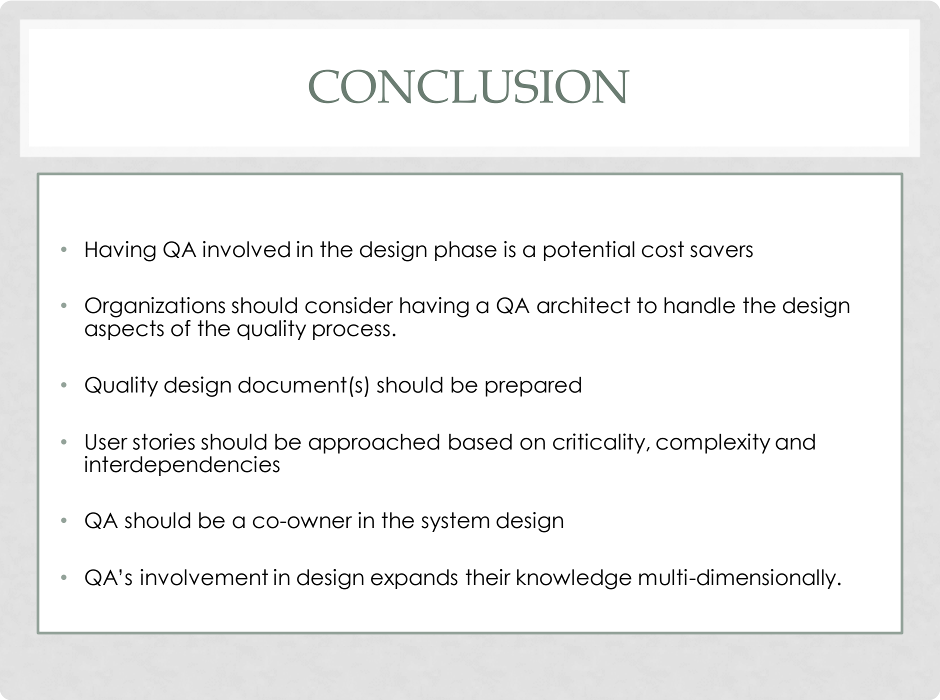


- QA Design:
  - Make sure unit tests are written for buyer object in order to check whether the object contains valid values like session id and shopping parameters.
  - Write a test API to check whether the business component commits the data into data store successfully.
  - Check the maximum number of requests that the business component can handle (nonfunctional Performance test).
  - Check the correctness of admin id and its corresponding request object.
  - Validate the validation logic by exercising with a number of unit test values.
  - Write a test API to check whether data submission was successful.
  - Check relevant tables in the inventory store and make sure the commit operation happened
  - End.

QA DESIGN - ILLUSTRATED

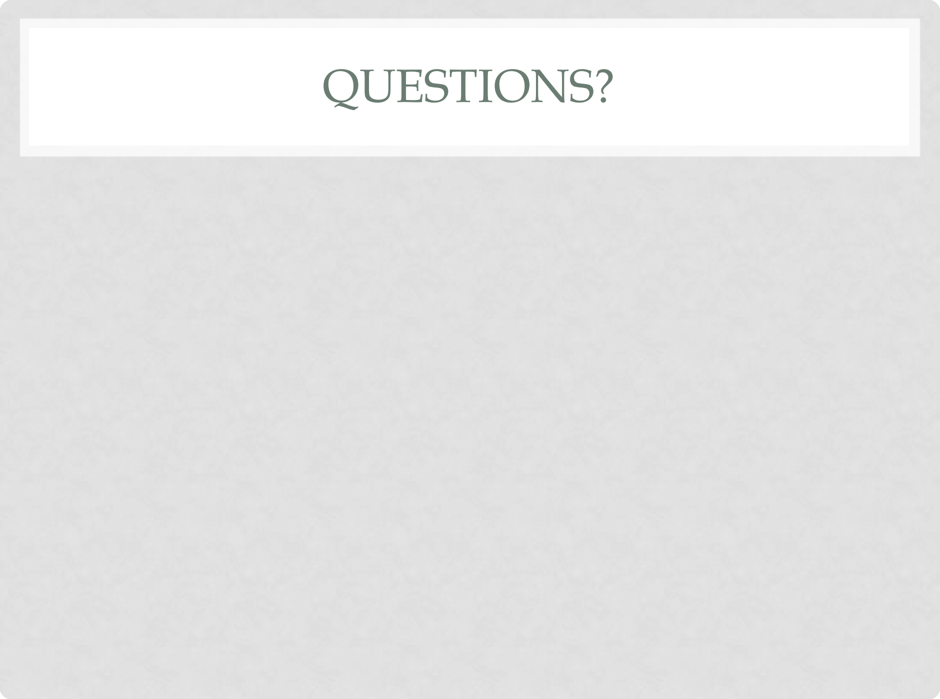


- Deriving non functional requirements from QA Design:
  - Interoperability tests covered between Buyer object, Business component and admin object
  - System availability, durability and maintainability test cases.
  - Load test, performance test coverage.
  - System test boundaries: API test, serviceability of the component in standalone and distributed environments.



CONCLUSION

- Having QA involved in the design phase is a potential cost savers
- Organizations should consider having a QA architect to handle the design aspects of the quality process.
- Quality design document(s) should be prepared
- User stories should be approached based on criticality, complexity and interdependencies
- QA should be a co-owner in the system design
- QA's involvement in design expands their knowledge multi-dimensionally.



QUESTIONS?