CYBAGE
Delivering Value. Scientifically.

Role Based Training – Developers

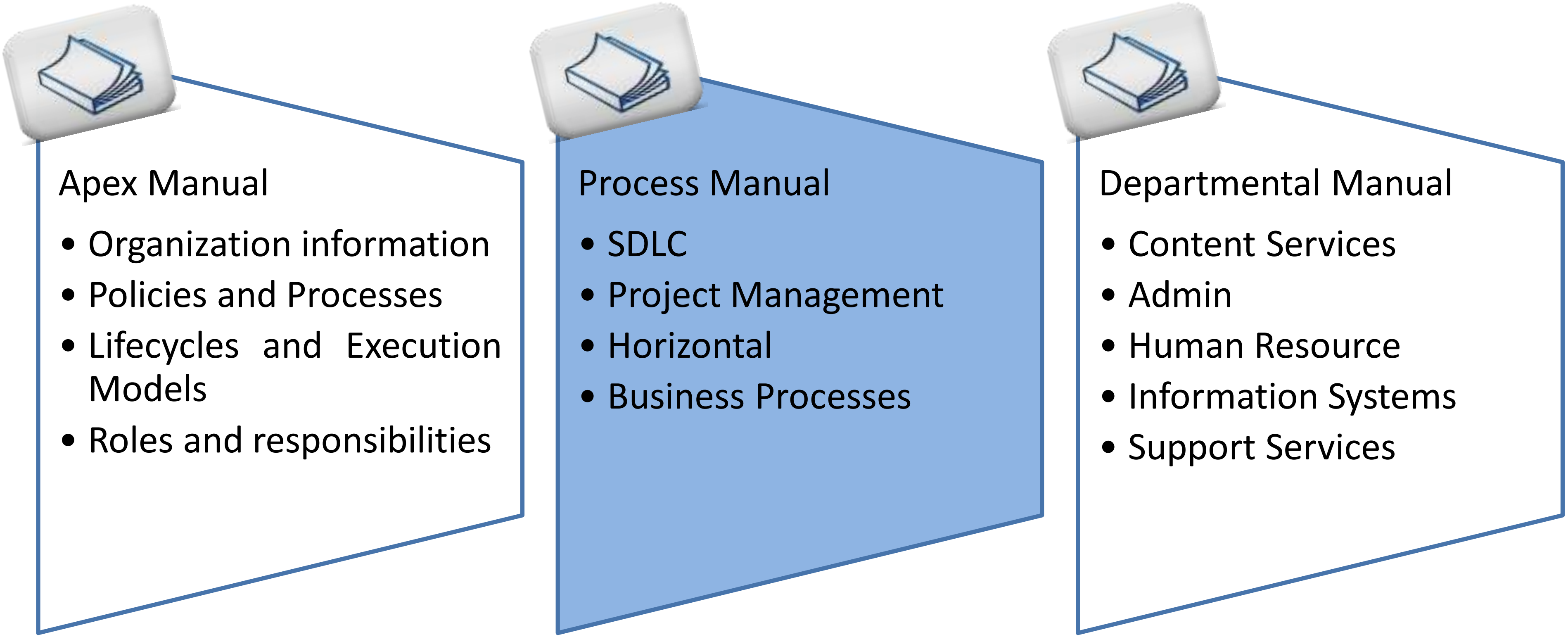
Authored and Presented by: Process Team

www.cybage.com

Agenda:

- Quick Glance at Q@Core!
- Responsibilities of Developer
- Involvement of Developer in various processes
- What is DevOps?
- Commonly used Development Metrics
- Walkthrough of Q@Core artifacts
- Q & A

Q@Core Framework



Apex Manual

- Organization information
- Policies and Processes
- Lifecycles and Execution Models
- Roles and responsibilities

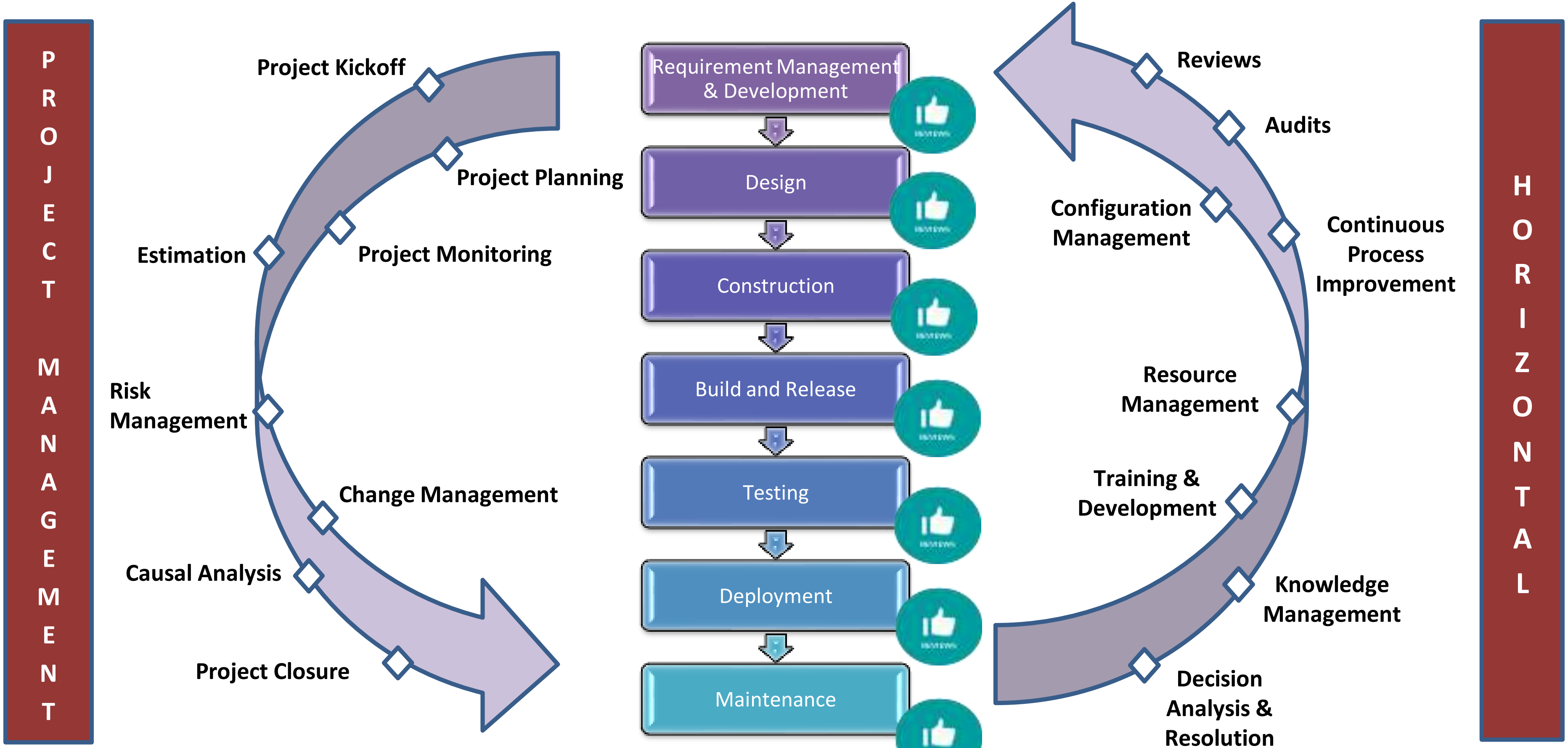
Process Manual

- SDLC
- Project Management
- Horizontal
- Business Processes

Departmental Manual

- Content Services
- Admin
- Human Resource
- Information Systems
- Support Services

Process Manual – At a glance



Developer – A Role!

- Understand and elicit the requirements
- Attend Client calls
- Prepare/Study design
- Set-up Development Environment
- Coding using appropriate Coding Standards
- Reviews
- Unit testing
- Highlight Issues/Risks to Project Manager
- Communicate with relevant stakeholders
- Participate in performing Root Cause Analysis
- Bug Fixing (QA defects)

Involvement of Developers in various processes : SDLC

SDLC Processes

Requirements Management

- Participate in Requirement Gathering and Understanding/ Refinement
- Participate in elicitation of requirement specifications
- Update requirement related queries responded by client
- Get requirements reviewed and agreed with client
- Create and Maintain Requirement Traceability Matrix

Use of tools is recommended for requirement management e.g. Jira, Confluence, TFS etc.

SDLC Processes

Design

- Participate in creating/understanding design
- Identify adequate design solutions/patterns
- Create technical design as per standards
- Get the design reviewed and agreed with client

Use of tools is recommended for designing e.g. Visual Studio, Eclipse, Visio, etc

SDLC Processes

Unit Testing

- Is a level of software testing where individual units/components of a software are tested
- Review and execute test cases
- Make use of Unit Testing Tools like nUnit, Jasmine, jUnit
- Ensure to update UTCs for changes in functionality
- Recommended to have module wise UTCs maintained

Note : UTC Document may not be required if Unit Testing is done using tools.

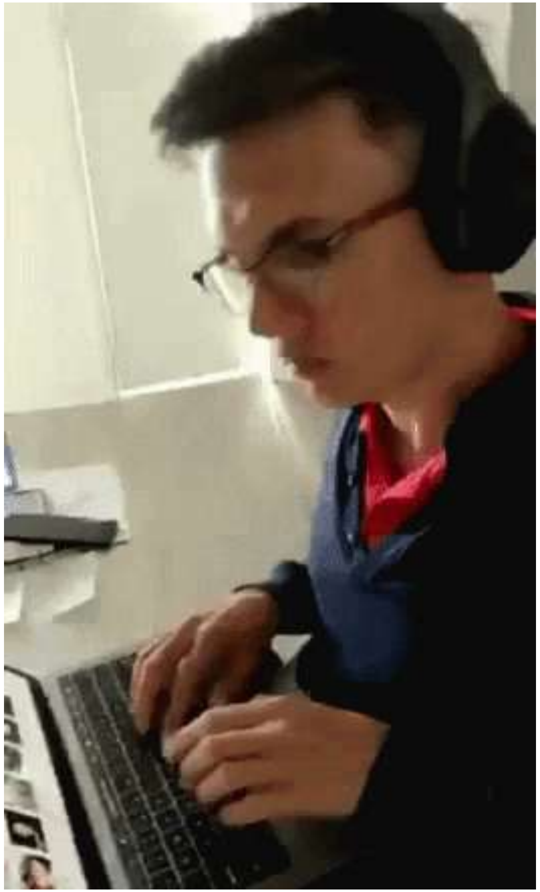
```
Assert.AreEqual(
    GetTimeOfDay(),
    "Morning" ))
```

SDLC Processes

Construction

- Study the design
- Write the Unit Test Cases/Scripts
- Set up the Development Environment
- Develop code using Coding Standards
- Execute the Unit Test Cases/Scripts
- Perform Reviews
- Fix the Code Review defects and QA Defects
- Perform Static Code Analysis using tools like SonarQube, FxCop

Use of Code Review Tools like Crucible, Collaborator is recommended



SDLC Processes

Use Code Quality Tools

Discuss with Project Manager/Architect/Technical Hierarchy, about the relevant code quality tool and configure the required rule set

Example: In SonarQube as per project requirement

1. Define Quality Profiles (rules)
2. Set Quality Gate
3. Create Dashboards

SDLC Processes

Build & Release

- Create build using build versioning guidelines
- Create Read Me and Release Notes
- Communicate build details to relevant stakeholders (QA/Client)
- Release build to intended environment

Continuous Integration process using tools can be adopted

Ex. TFS, Jenkins, Bamboo, Cruise Control, etc.

Types of Build:

- **QA Internal Build**: Unit Tested code/build from Developer to QA
- **Patch**: Quick fixes either to QA or to Client
- **Client Release**: Final release (QA Tested Build) to customer

SDLC Processes

Deployment

- Understand Deployment Environment
- Prepare ***Deployment Schedule and Roll Back Plan***
- Perform Deployment
    - ➢ Back-up existing system
    - ➢ Set-up Preparation, Product Installation
- Roll Back (in case of Issues)
- Provide post production/Deployment Support
- Participate in Root Cause Analysis for Production Issues

SDLC Processes

Maintenance Lifecycle Process

Every task/request forms the basis for planning and tracking.

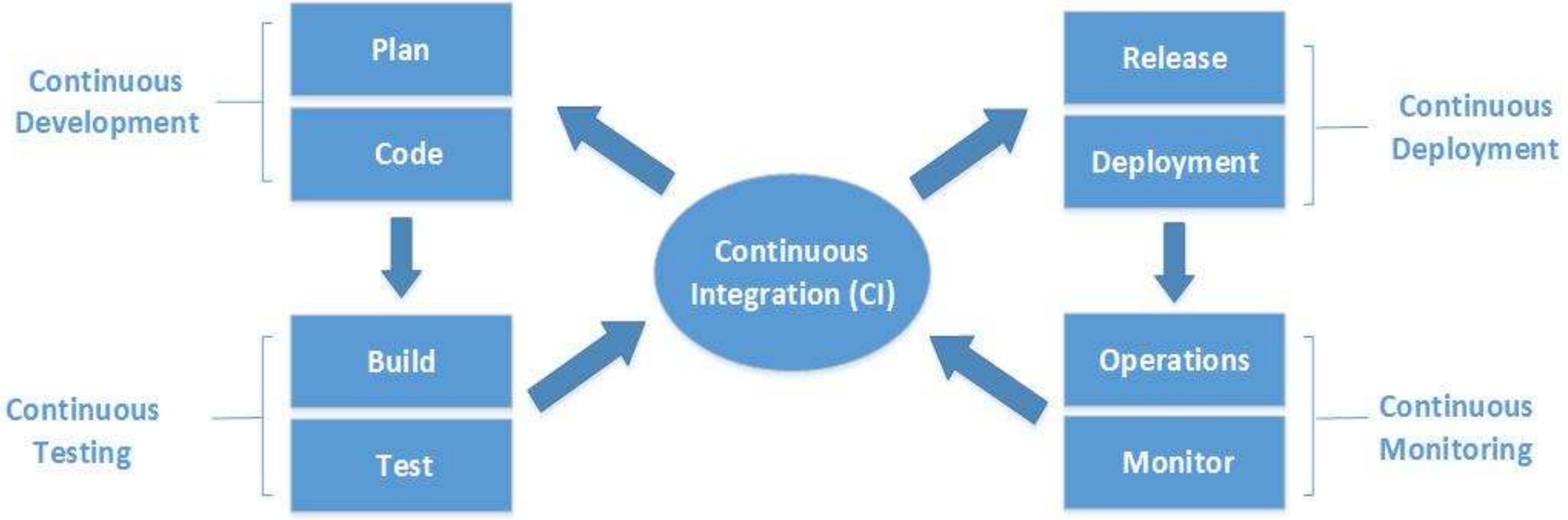
Bug Fixing / Enhancement

Change / Maintenance Request

- Estimate Maintenance request
- Perform Impact Analysis,
- Identify unit test scenarios
- Perform reviews
- Implement the request
- Track request to closure
- Create and Maintain Requirements Traceability Matrix

What is DevOps?

DevOps is a set of practices that automates the processes between software development and IT teams, in order that they can build, test, and release software faster and more reliably.



Involvement of Developers in various processes : Project Management

Project Management

Estimation

- Share estimates of assigned modules/requirements
  - Size Estimation
    (Ex. FP, Story point)
  - Detailed Estimation
    (Ex. Task Based Estimation, Delphi-Expert Judgment)
- Identify and raise the need for re-estimation

Note: Size estimation is mandatory, recommended technique is Function point analysis.

Project Management

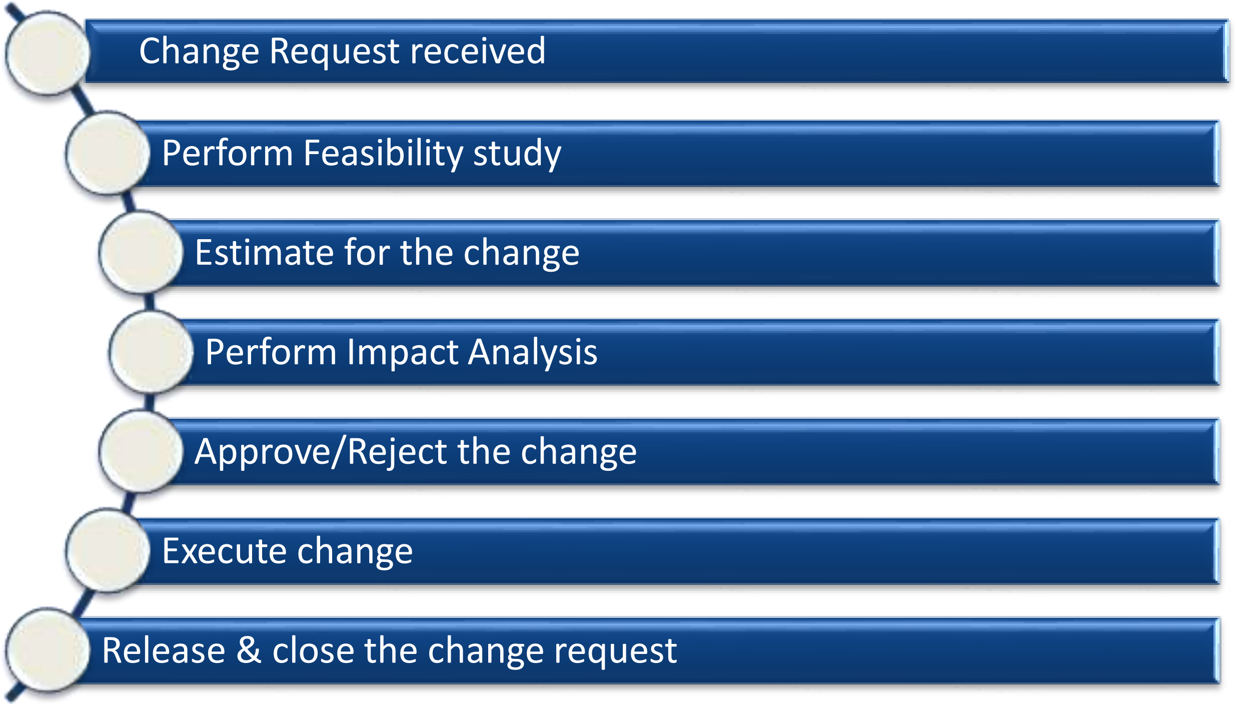
Project Planning

Share estimates of assigned modules/requirements

- **Pre-planning:** Understand & estimate for requirements/ User Stories with Product Owner- Prioritize User Stories for the Sprint.
- **Planning:** Constructing Sprint Backlog, Task Breakdown- self assignment of tasks.
- Decide Sprint Goal and Release Plan.

Project Management

Change Management



- Change Request received
- Perform Feasibility study
- Estimate for the change
- Perform Impact Analysis
- Approve/Reject the change
- Execute change
- Release & close the change request

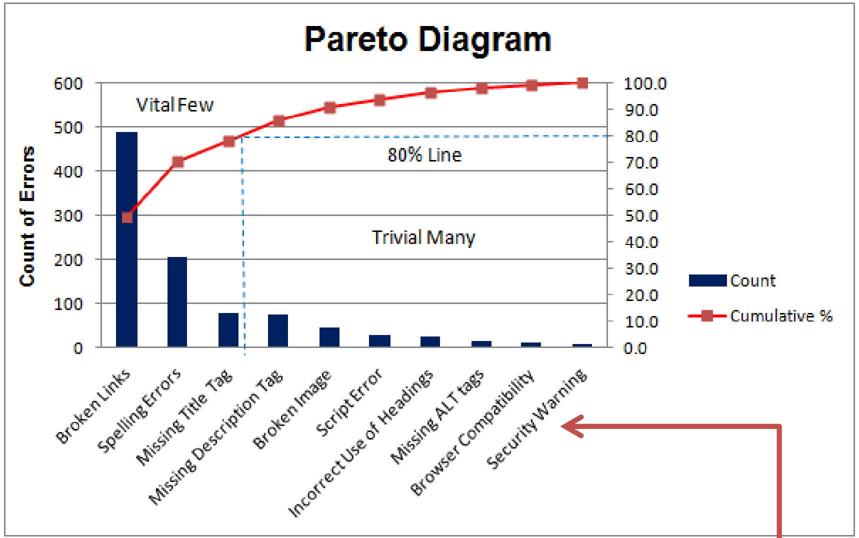
Project Management

Causal Analysis & Resolution (Defect Prevention and Root Cause Analysis)

- Participate in collecting and analyzing the data for causal analys
- Provide attributes to defects
  - Defect type, Defect Introduced at, Defect Category, etc.
- Participate in CAR meetings to conduct root cause analysis
- Execute the action plan to prevent future occurrence of sim problems
- Identify and document the impacted change

Techniques for CAR - Pareto Analysis

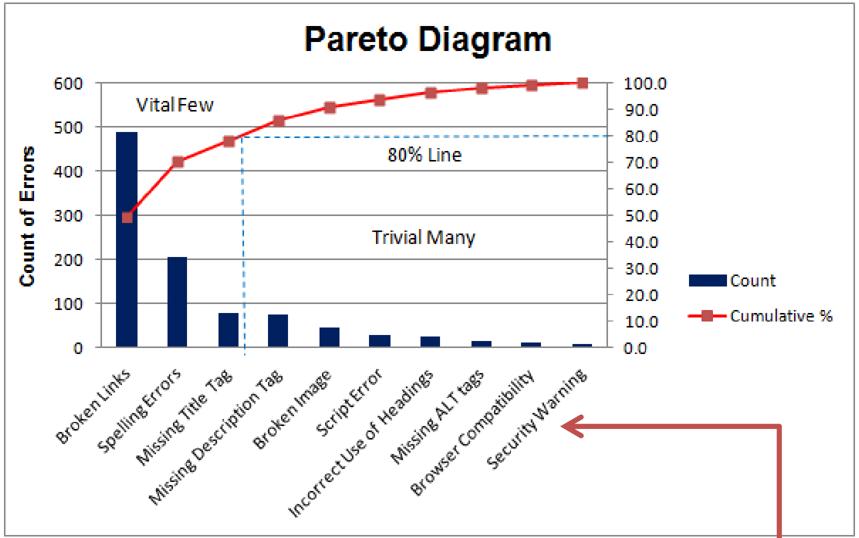
- A tool to determine the "vital few" out of the "trivial many" possible causes to a given problem
- 80-20 rule – "20% of the causes account for 80% of the problems"



Possible Causes

Techniques for CAR - Pareto Analysis

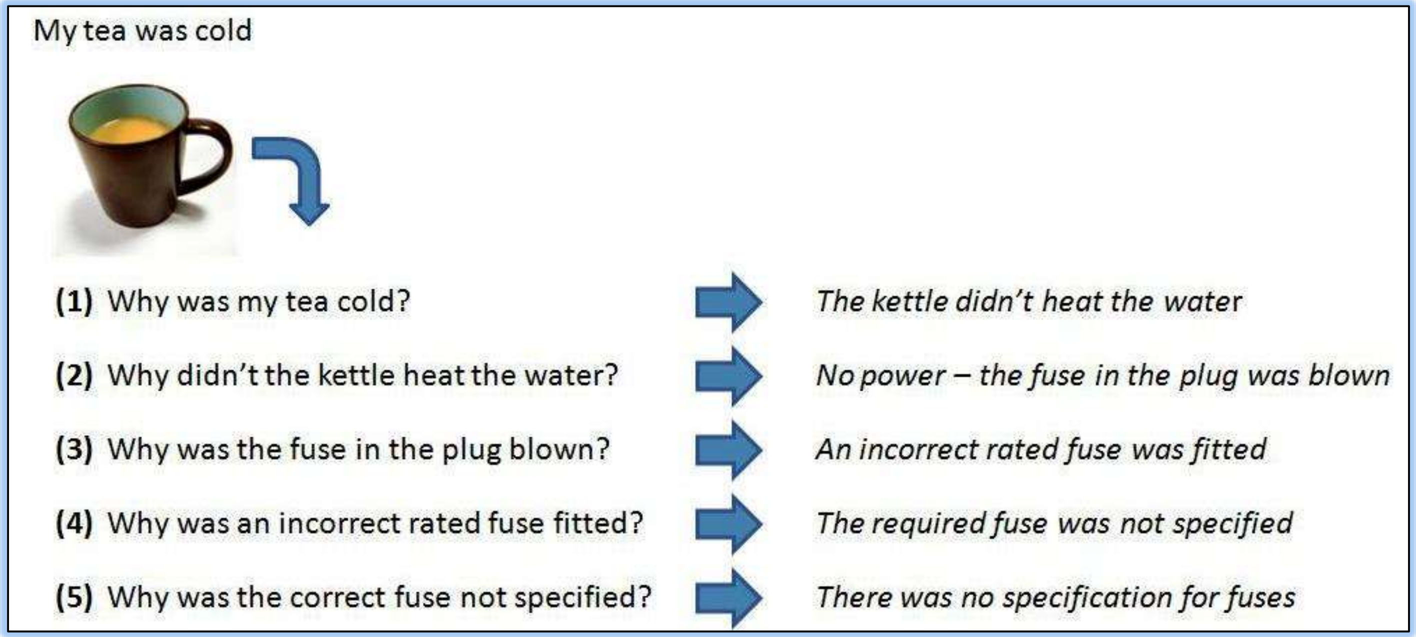
- Answers the following questions:
  - What are the largest issues facing our team or business?
  - Which 20% of sources are causing 80% of the problems (80/20 Rule)?
  - Where should we focus our efforts to achieve the greatest improvements?



Possible Causes

Techniques for CAR - 5-Why Technique

An iterative interrogative technique used to determine the root cause of a problem or a defect.

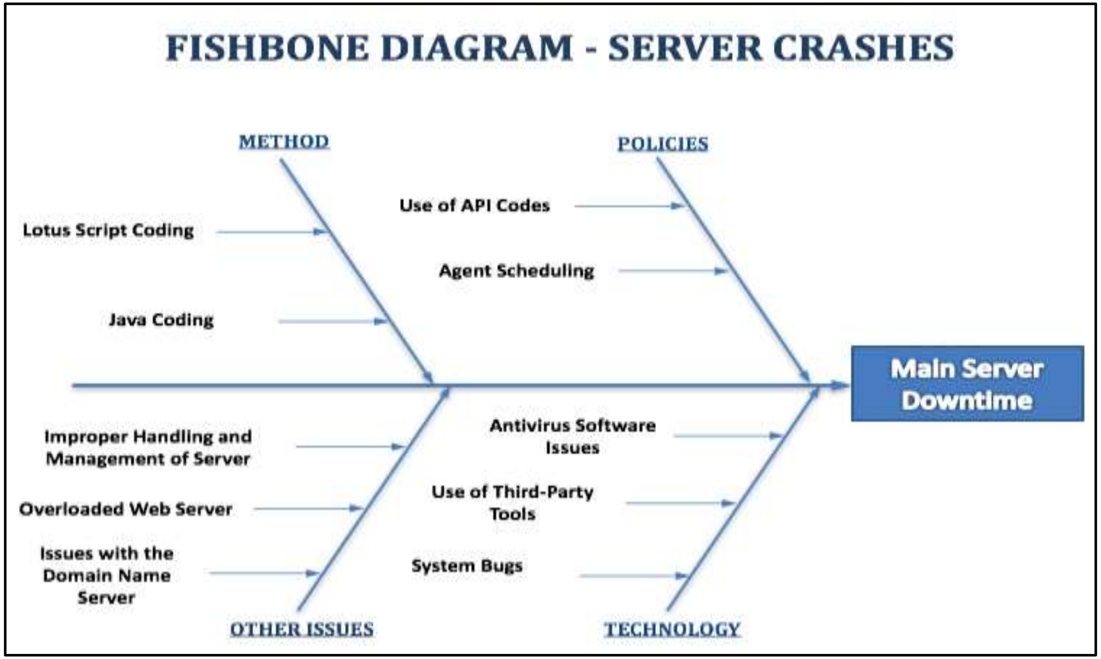


My tea was cold

(1) Why was my tea cold?	*The kettle didn't heat the water*
(2) Why didn't the kettle heat the water?	*No power – the fuse in the plug was blown*
(3) Why was the fuse in the plug blown?	*An incorrect rated fuse was fitted*
(4) Why was an incorrect rated fuse fitted?	*The required fuse was not specified*
(5) Why was the correct fuse not specified?	*There was no specification for fuses*

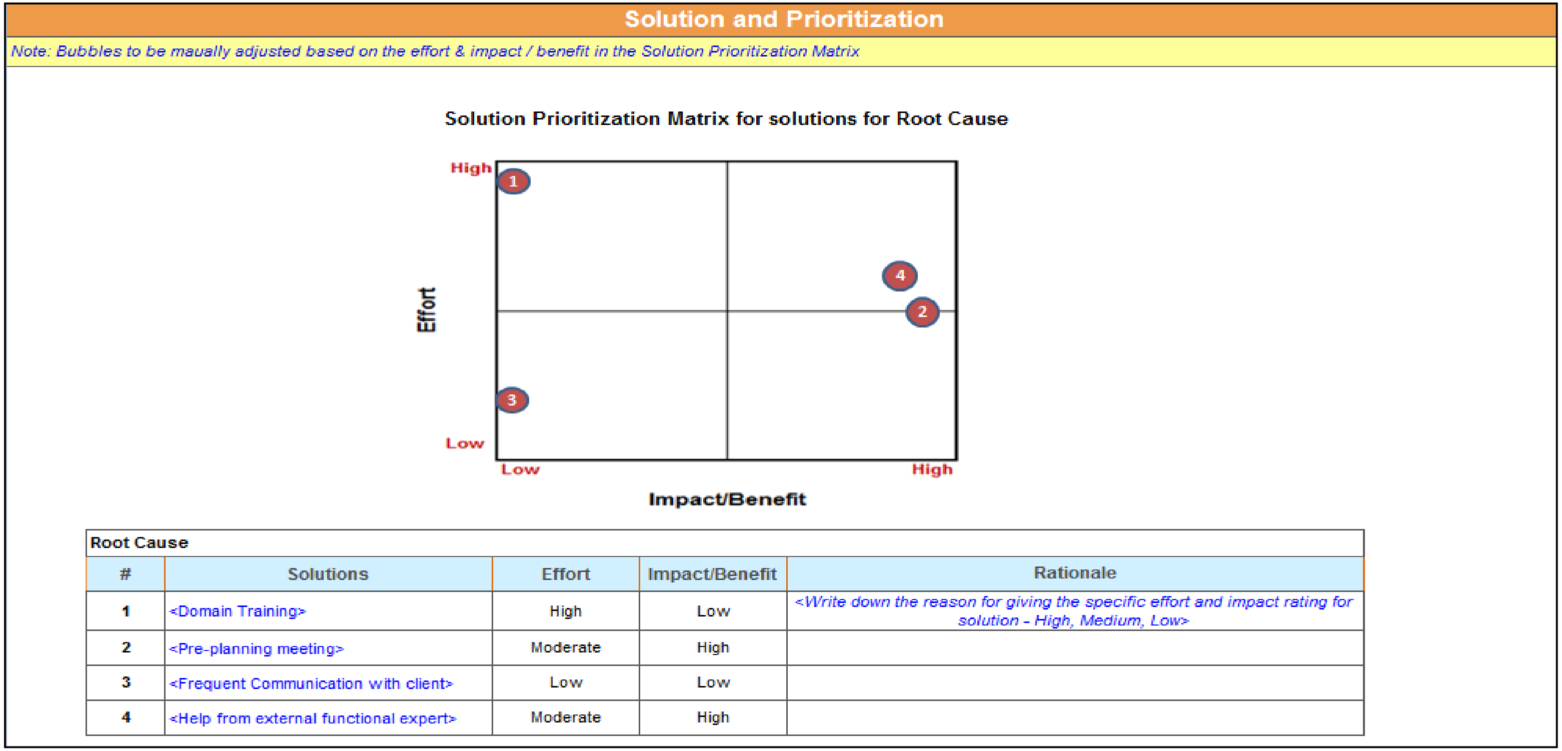
Techniques for CAR - Fishbone analysis

- Known as Fishbone/Ishikawa/Cause-and-Effect diagram.
- A tool used to identify all the possible root causes of a problem or defect.



Fishbone analysis - Solution and Prioritization Matrix

- Used to arrive at a consensus to a set of proposed solutions.
- Used to prioritize the solutions proposed to be implemented for the identified root causes of the given problem.



Solution and Prioritization

Note: Bubbles to be maually adjusted based on the effort & impact / benefit in the Solution Prioritization Matrix

Solution Prioritization Matrix for solutions for Root Cause

Root Cause

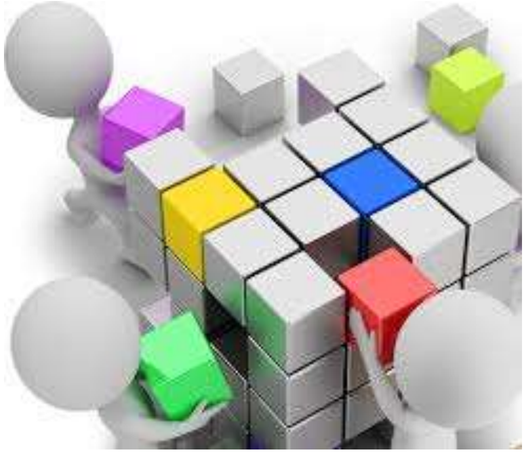
#	Solutions	Effort	Impact/Benefit	Rationale
1	<Domain Training>	High	Low	*<Write down the reason for giving the specific effort and impact rating for solution - High, Medium, Low>*
2	<Pre-planning meeting>	Moderate	High	
3	<Frequent Communication with client>	Low	Low	
4	<Help from external functional expert>	Moderate	High	

Involvement of Developers in various processes : Horizontal Processes

Horizontal Processes

Software Configuration Management

- Know and follow the folder structure in repository
- Understand and use the naming conventions
- Ensure to mention check-in/ check-out comments
- Ensure to baseline the required artifacts (Label/Tag)
- Ensure the right artifacts are available for the entire team



Tools for Software Configuration Management : TFS, CVS, SVN, Perforce

Horizontal Processes

Resource Management

- Participate in the ramp-up activity for the new joiners
- Participate in the handover and takeover activity (if triggered)

Horizontal Processes

Decision Analysis & Resolution(DAR)

- Participate in decision making by providing:
  - Possible Alternative solutions
  - Providing rational (Pros/Cons)
  - POC/Analysis of risk involved in selected solution
- Few techniques for conducting DAR:
  - Decision Matrix
  - Force-Field Analysis
  - Brainstorming

Commonly used Development Metrics

Metrics	Formula
Code coverage%	(No. of lines of code covered by automated unit tests/Total Lines of code) * 100
Unit Testing %	[Unit Test Case Execution Efforts (Hrs.)/Total Coding Efforts (Hrs.)] * 100
Code Review %	[Coding Review Efforts(Hrs.)/Total Coding Efforts (Hrs.)] * 100
% Quality Kick-back	(No. of Rejected work items/No. of Delivered work items) * 100
% Velocity Achieved	(# Story Points Completed/# Story Points Committed) * 100

Other Responsibilities

- Discuss doubts & queries with client, PM or other stakeholders
- Regularly attend client calls
- Ensure clarity by providing reports on time
- Participate in team meetings
- Coordinate with QA team
- Ensure to log your timesheet correctly
- Participate in audits as auditee

NOTE : Always follow the Process; it's a need, NOT A Fancy!

Walkthrough of Templates

Requirement Phase	Design Phase	Coding Phase	Build and Release Phase	Deployment Phase	Maintenance Phase
RUD	AD	UTCD	RN	RPL	MRL
SRS	SSLD		RM	DS	MRD
RTM	UID		BNMT		
QL					

CRL	RR	RFC

Any questions?

Quick Test !

1. Understanding of requirements is done using?

2. Definition of functional & non-functional requirements is done using?

3. What is RTM?

4. List some design artifacts/tools available in Q@Core?

5. What is the purpose of writing UTCs?

6. What are various types of builds defined in Q@Core?

Quick Test !

7. What are different estimation techniques?
8. What is developers' involvement in CAR process?
9. What are the techniques under DAR?
10. What is developers' involvement in configuration management?
11. What is change management process?

CYBAGE
Delivering Value. Scientifically.

Thank You!

www.cybage.com