

Configuration Management (CM) Support –

A plan is essential:

A configuration management plan is necessary for sound configuration management practice. Include the following in the plan:

Configuration Identification:

Identify the things to be managed and level of control at each level.

Identify all configuration items to be controlled: user requirements documents, requirements specifications and traceability, design artifacts, development documents, software version documents, interface control documents, drawings and parts lists, test plans and procedures, test scripts, test results, training materials; depending on the type of program, you may also have architecture products, data flows and network diagrams, simulation data, test harness/modeling and simulations, etc.

Identify the level of detail of each to be controlled: system-of-systems, system, configuration item, component, item, part number, network asset, etc.

Identify all baselines to be managed: user requirements, system requirements, design, development, test, sustainment, experimentation, etc.

Develop a schema or comply with organizational policy to provide unique identifiers for each item.

Determine the level of the configuration management hierarchy (stakeholders) for each identified "configuration item" to be approved (baselined).

Configuration Control:

Develop a closed-loop corrective action process to track all configuration item changes to closure and inclusion in appropriate baseline documentation.

Build or provide specifications to build work products from the software configuration management system or physical products from the hardware configuration management system.

Purchase or develop tools for version control of source code. This product should provide version control tracking to the line of code level. Assure implementation of an engineering release system to provide hardware version control.

Configuration Status Accounting: Publish periodic reports describing the current configuration of each configuration item. There should be a configuration version description document detailing each version of software undergoing integration, system, or acceptance test. There should be a set of engineering drawings detailing each developmental hardware item undergoing

integration and testing. Commercial hardware and software also needs to be under configuration control during integration and testing. Configuration status accounting applies to all fielded hardware, software, and other controlled assets during operations and maintenance for the life of the system.

Configuration Audits:

Perform periodic examinations of operational baselines for completeness (configuration verification audit). Prior to product delivery to the sponsor, ensure successful completion of a functional configuration audit to assure that the product meets its specified requirements. Also conduct a physical configuration audit to assure that the successfully tested product matches the documentation.

Accounting of requirements changes per month and changes processing time; also, the number of defects that are open and closed are metrics that may be used for configuration management.

Automate to manage complexity:

If the program is sufficiently complex, identify and install an automated tool to support the configuration management tasks. Consider the other stakeholders (engineers/programmers, users, contractors, interfacing systems, and sustainment organizations) in the selection of any automated configuration management tools.

Work your plan: Implement and conduct the configuration management activities according to the program's configuration management plan.

Quality Assurance (QA) Support –

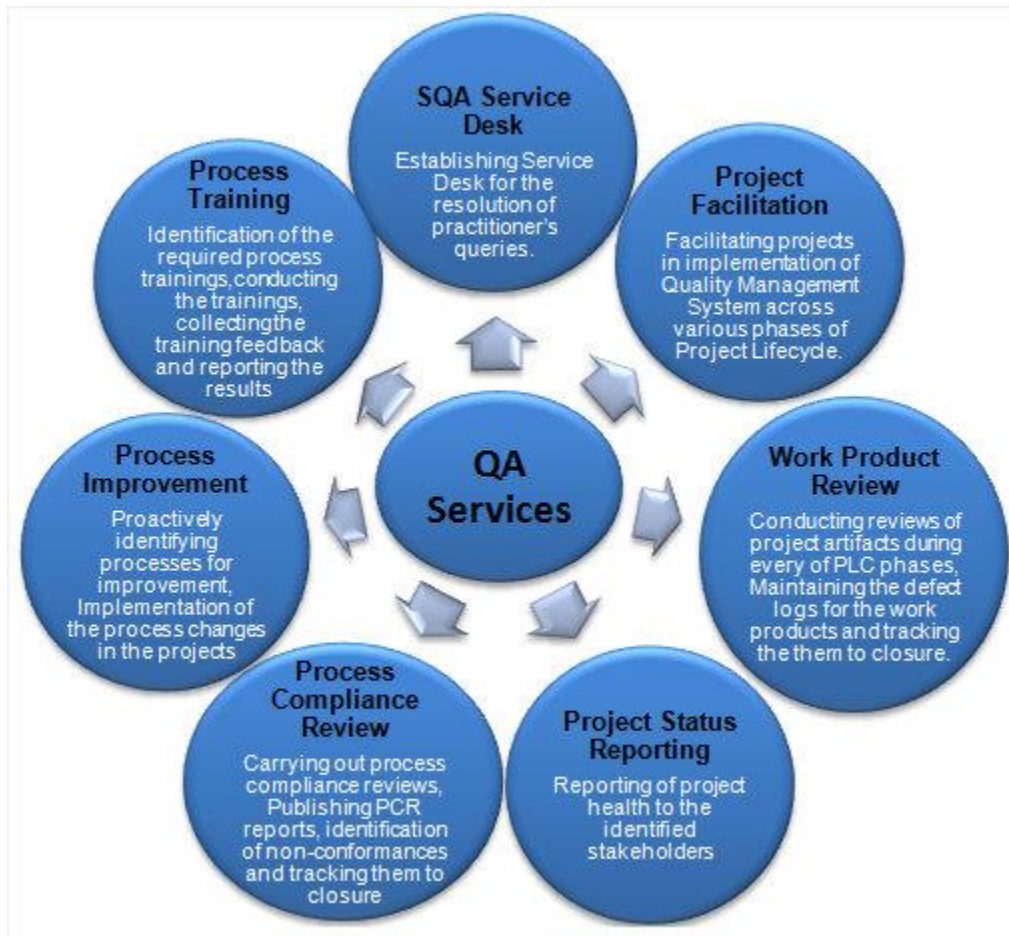
Quality Assurance is a planned and systematic means for assuring management that the defined standards, practices, procedures, and methods of the process are applied. (Source: Software Engineering Institute, Carnegie Mellon University)

The most effective way to manage quality cost is to prevent defects at the early stages of Project Life Cycle (PLC). As a universal known fact, it is always less expensive to detect & fix the defect at an earlier stage than fixing it once it has occurred in the production.

Lee Defense enables their customers to have a robust quality system by providing Quality Assurance Services to all types of projects (Development, Enhancement, Maintenance, Testing and Production Support). We help our clients to:

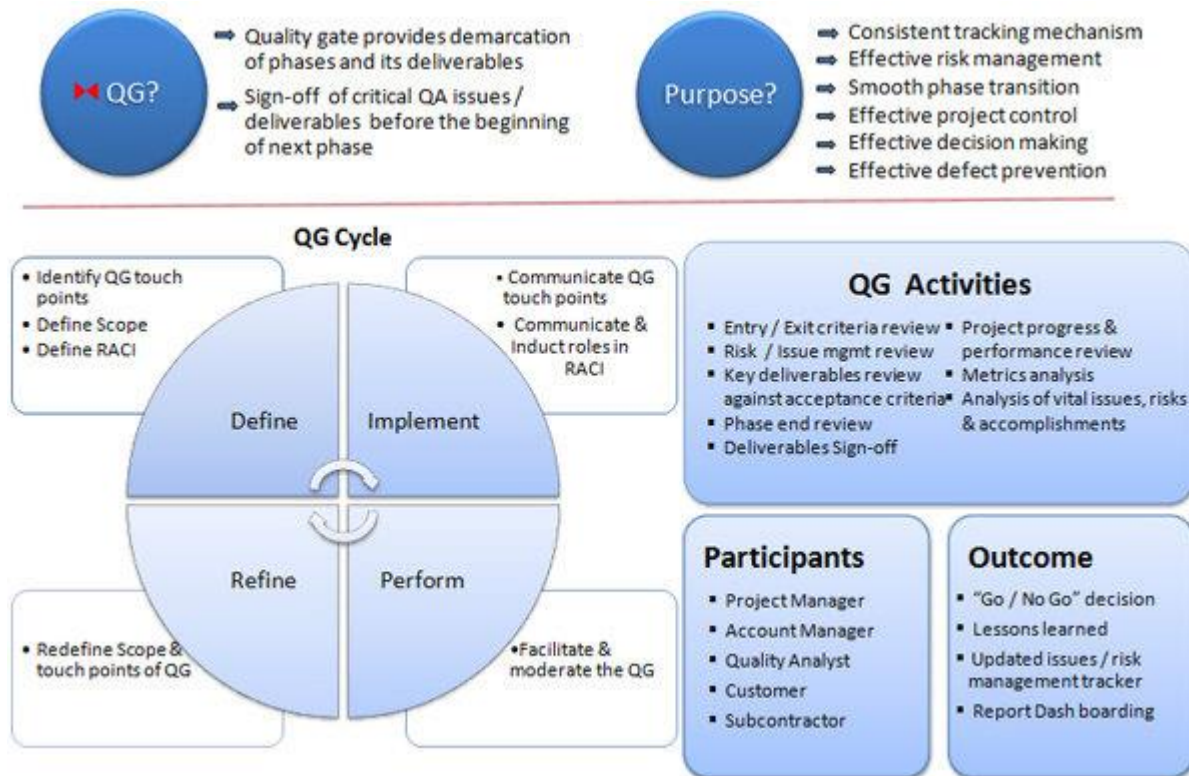
- Enable projects to detect defects early in the project lifecycle
- Establish a Defect Tracking System
- Establish Quality Gate (QG) framework
- Establish Defect Prevention Methods
- Enable Process Improvement initiatives

- Improve the Quality of Deliverables



Lee Defense Group's Subject Matter Experts have multiple years of process implementation & project facilitation experience.

Lee Defense Quality Gate Process:



Many of Lee Defense customers have benefited in:

- Improving the ability to manage and prevent the defects to production systems
- Reducing rework & cost
- Improving productivity
- Sustaining quality of deliverables throughout the project life cycle
- Early detection & fixation of defects in the software life cycle