Figure 44 illustrates the key elements of a defect management process.

Defect Prevention

As many quality experts have pointed out, the best approach to defects is to eliminate them altogether.

Figure 45 illustrates a defect prevention process with three major steps that are described below.

- Identify Critical Risks
- Estimate Expected Impact
- Minimize Expected Impact

Identify Critical Risks

- A key requirement is missing.
- Critical application software does not function properly.
- Vendor-supplied software does not function properly.
- Software does not support major business functions – necessitates process reengineering.
- Performance is unacceptably poor.
- Hardware that malfunctions.
- Hardware and software do not integrate properly.
- Hardware that is new to installation site.
- Users are unable or unwilling to embrace new system.
- User’s ability to actively participate in project, etc.

**Estimate Expected Impact**

**Minimize Expected Impact**

Appropriate techniques to reduce expected impact are a function of the particular risk. Techniques to prevent defects include:
- Quality Assurance
- Training and Education (Work Force)
- Training and Education (Customers)
- Methodology and Standards
- Defensive Design
- Defensive Code

**Deliverable Baseline**

You baseline a deliverable, or work product when it reaches a predefined milestone in its development. This milestone involves transferring the product from one stage of development to the next. Deliverable baseline involves the following activities:

- Identify key deliverables
- Define standards for each deliverable

**Defect Discovery**

The steps involved in defect discovery are illustrated in Figure 47.

![Figure 47. Defect Discovery](image)

**Find Defect**
Techniques to find defects can be divided into three categories:
- Static techniques
- Dynamic techniques
- Operational techniques
Record Defect
You should record defects for these four major purposes:

- To correct the defect
- To report status of the application
- To gather statistics used to develop defect expectations in future applications
- To improve the software development process

Report Defects

Acknowledge Defect
Strategies to address this problem include:

- Instrument the code to trap the state of the environment when anomalous conditions occur.
- Write code to check the validity of the system.
- Analyze reported defects to discover the cause of a defect.

Defect Resolution

The steps involved in defect resolution are illustrated in Figure 48

![Figure 48. Defect Resolution](image)

Prioritize Fix
The purpose of this step is to answer the following questions and initiate any immediate action that might be required:

- Is this a previously reported defect, or is it new?
- What priority should be given to fixing this defect?
- What steps should be taken to minimize the impact of the defect prior to a fix? For example, should other users be notified of the problem? Is there a workaround for the defect?

A suggested prioritization method is a three-level method, as follows:

- Critical – Would have a serious impact on the organization’s business operation.
- Major – Would cause an output of the software to be incorrect or stop.
Minor – Something is wrong, but it does not directly affect the user of the system, such as a documentation error or cosmetic GUI (graphical user interface) error.

Schedule Fix

Fix Defect

Report Resolution

References

Guide – CSTE Common Body Of Knowledge, V6.1