task of solving problems that may arise when implementing SQA tools. These arrangements are targeted especially at team leaders and unit managers, who are expected to follow up and support the implementation efforts made by their units.

Throughout this stage, internal quality audits are carried out to verify the success in implementation as well as to identify units and SQA issues that require additional attention. The internal quality audit findings will enable determination of whether the organization has reached a satisfactory level of implementation.

23.3.4 Undergoing the certification audits

The certification audits are carried out in two stages:

(1) *Review of the quality manual and SQA procedures developed by the organization.* The review ascertains completeness and accuracy. In cases of non-compliance with standards, the organization is obligated to complete the corrections prior to advancing to the second stage of certification.

(2) *Verification audits of compliance with the requirements defined by the organization in its quality manual and SQA procedures.* The main questions to be answered are:

- Have the staff been adequately instructed on SQA topics and do they display a satisfactory level of knowledge?
- Have the relevant procedures – project plans, design reviews, progress reports, etc. – been properly and fully implemented by the development teams?
- Have documentation requirements been fully observed?

The main sources of information for certification audits are (a) interviews with members of the audited unit, and (b) review of documents such as project plans, design documents, test plans and procedures, and design review records. In order to assure reliable results and avoid biased conclusions, audits are based on a random selection of projects and/or teams.

23.3.5 Procedures for retaining ISO certification

Periodic re-certification audits, usually carried out once or twice a year, are performed to verify continued compliance with ISO 9000-3 requirements. During these audits, the organization has to demonstrate continuing development of its SQA management system, which is expressed in quality and productivity performance improvements, regular updates of procedures to reflect technological changes, and process improvements.

23.4 Capability Maturity Models – CMM and CMMI assessment methodology

Carnegie Mellon University's Software Engineering Institute (SEI) took the initial steps toward development of what is termed a *capability maturity model* (CMM) in 1986, when it released the first brief description of the maturity process framework. The initial version of the CMM was released in 1992, mainly for receipt of feedback from the software community. The first version for public use was released in 1993 (Paulk *et al.*, 1993, 1995; Felschow, 1999).

23.4.1 The principles of CMM

CMM assessment is based on the following concepts and principles:

- Application of more elaborate management methods based on quantitative approaches increases the organization's capability to control the quality and improve the productivity of the software development process.
- The vehicle for enhancement of software development is composed of the five-level capability maturity model. The model enables an organization to evaluate its achievements and determine the efforts needed to reach the next capability level by locating the process areas requiring improvement.
- Process areas are generic; they define the "what", not the "how". This approach enables the model to be applied to a wide range of implementation organizations because:
  - It allows use of any life cycle model
  - It allows use of any design methodology, software development tool and programming language
  - It does not specify any particular documentation standard.

The CMM and its key process areas (KPAs) are presented in Figure 23.2.

23.4.2 The evolution of CMM

After 1993, the SEI expanded the original Software Development and Maintenance Capability Maturity Model (SW-CMM) through diversification. Its main structure was retailored to fit a variety of specialized capability maturity models. The following variants have been developed:

- **System Engineering CMM (SE-CMM)** focuses on system engineering practices related to product-oriented customer requirements. It deals with product development: analysis of requirements, design of product systems, management and coordination of the product systems and their integration. In addition, it deals with the production of the developed product: planning p uction lin nd their operation.

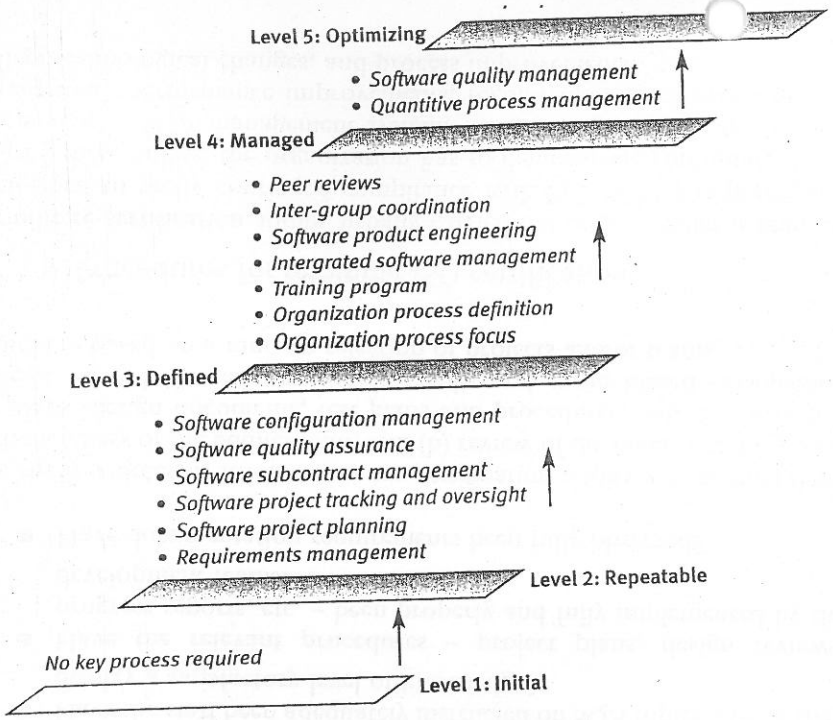

Figure 23.2: The CMM model levels and key process areas (KPAs)

Source: After Paulk *et al.* (1995)

- **Trusted CMM (T-CMM)** was developed to serve sensitive and classified software systems that require enhanced software quality assurance.
- **System Security Engineering CMM (SSE-CMM)** focuses on security aspects of software engineering and deals with secured product development processes, including security of development team members.
- **People CMM (P-CMM)** deals with human resource development in software organizations: improvement of professional capacities, motivation, organizational structure, etc.
- **Software Acquisition CMM (SA-CMM)** focuses on special aspects of software acquisition by treating issues – contract tracking, acquisition risk management, quantitative acquisition management, contract performance management, etc. – that touch on software purchased from external organizations.
- **Integrated Product Development CMM (IPD-CMM)** serves as a framework for integration of development efforts related to every aspect of the product throughout the product life cycle as invested by each department.

For an expanded discussion of the diversity of CMM applications, see Johnson and Brodman (2000).

Capability Maturity Model Integration (CMMI)

In the late 1990s a new developmental direction was taken – development of integrated CMM models. Development of specialized CMM models involved development of different sets of key processes for model variants for different departments that exhibited joint processes. In practice, this created a situation where departments that applied different CMM variants in the same organization faced difficulties in cooperation and coordination. The CMMI approach solved these problems at the same time as the moduals better conformed to the emerging ISO/IEC 15504 standard (see Royce, 2002).

At the beginning of 2002, SEI could offer the 1.1 version of three CMMI models, with each model presenting different integrated components:

- CMMI-SE/SW integrates the system engineering and selected aspects of software.
- CMMI-SE/SW/IPPD/SS integrates system engineering, software and integrated product/process aspects.
- CMMI-SE/SW/IPPD integrates system engineering, software, integrated product/process and supplier sourcing aspects.

23.4.3 The CMMI structure and processes areas

The CMMI model, like the original CMM models, is composed of five levels. The CMMI capability levels are the same as those of the original, apart from a minor change related to capability level 4, namely:

- Capability maturity level 1: Initial
- Capability maturity level 2: Managed
- Capability maturity level 3: Defined
- Capability maturity level 4: Quantitatively managed
- Capability maturity level 5: Optimizing.

A substantial change has nonetheless evolved with respect to the processes included in the models. The 18 key process areas of CMM (frequently referred to as *KPAs*) were replaced by 25 process areas (PAs). The PAs are classified by the capability maturity level that the organization is required to successfully perform. For each process area, objectives, specific practices and procedures are defined.

Appendix 23A presents the revised process areas and their descriptions.