

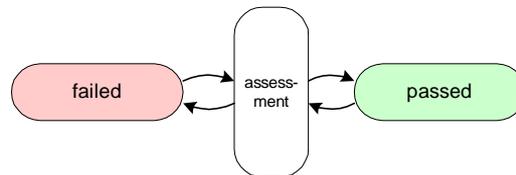
**Question 1**

*Discuss the relative advantages and disadvantages of CMMI and ISO 9001.*

The more safety-critical software is developed all over the world, the more industry and government feel an urgent need for a standardized and well-accepted methodology in order to predict, prove and ensure software quality. Two major approaches handle the topic from different points of view, their foundations were laid in the past ten years. The very generally written ISO 9000 standard aims at the whole manufacturing industry as well as at the service sector. On the other hand, CMM and CMMI were explicitly designed with software and its special features and attributes in mind.

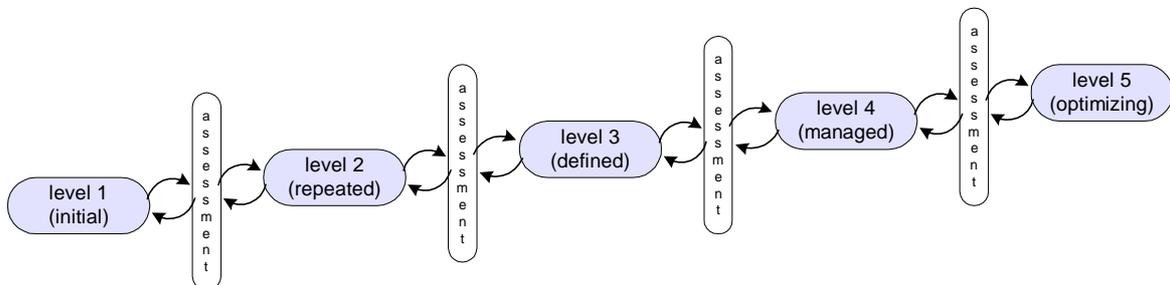
Due to its generality, ISO 9000 has been refined by ISO 9001 which discusses quality assurance for the complete development plus maintenance cycle and by ISO 9000-3 that sets guidelines for software organizations. However, these documents remain fairly uncertain, they cover just a few sheets of paper (about 50). CMMI goes far deeper into detail, its specification is approximately ten times larger in size. Purely concentrating on software, CMMI seems to fit best to the needs of modern software development while ISO 9000/9001 can be applied to the entire company. I discovered to my surprise that ISO 9000/9001 is widely known to the upper management level, where usually only a small fraction of trained developers can be found since economics dominates that area, while solely a few web sites even mention CMMI. Hence, the people actually authorized to decide which quality enhancing methodology to introduce tend to choose ISO 9000/9001. Quite similar is the political aspect: the Geneva (Switzerland) based ISO committee gains more attention in Europe opposed to the Carnegie Mellon (Pittsburgh, USA) rooted CMMI naturally obtains a widespread use in North America. One should not forget about this political fact that often influences management's decision to a crucial degree.

The basic intention of ISO 9001 is to define minimal requirements for a quality system. In contrary, CMMI focuses at a continuous process improvement. That difference arises many contrasting properties such as the rating of an examined company or department (level of approval). There are just two possible outcomes of an ISO assessment: the company may either pass or fail. The figure below visualizes the relationship:



**Figure 1:** ISO 9001 assessment

CMMI's basic principle to differentiate among several levels of maturity leads to a far more complex structure:



**Figure 2:** CMMI assessment

I omitted some transitions: indeed, one can “jump” forward or even backwards from one level to another without visiting all intermediate levels. However, these cases should be very rare and seldom.

One of the initial forces that led to ISO 9001 was to enhance the quality seen from the customer's point of view. The standard's role was defined by a need to properly assess a potential supplier before contracting him. The main goal of CMMI, maturing the internal processes, in the end nicely maps to ISO's goal. The big advantage of CMMI, achieving maturity, effects in longing for future improvements of used processes. Thus, CMMI turns out to be a more on-going approach while ISO 9001 just verifies that certain quality ensuring techniques are in place. ISO 9001 defines structures of the administrative management level as well as it directs the creation, responsibility and storage of documents.

As ISO 9001 leaves many details of the assessment unspecified, there is much space left for interpretation. In my opinion, one cannot always accurately compare different companies just based on their assessment result since it might heavily depend on the assessor. Even educated and well-trained auditors, required for ISO 9001 assessment, are prone to failures. The CMMI specification does repeat that mistake and provides an in-depth coverage of almost all software quality related aspects. Though, CMMI is allowed for self-assessments but its very objective structure seems to be quite invariant to the actual assessor. One common example is the introduction of statistical measurements (metrics) on higher levels or the request for certain techniques.

In the beginning, I mentioned a shortage of truly accepted ways to compare software quality. CMMI's variety of levels give a more suitable hint which company fits best to one's demands. Not in all cases, a high level is desirable: small projects sometimes do not require a large overhead and may successfully be carried out by a young, devoted team of not-too-experienced programmers (at a lower cost). For huge projects with strong emphasis on quality, level 3 should be the lower limit to ensure a satisfying realization. There is no decent feature in ISO 9001 that distinguishes competing offers before signing a software team, an alarming drawback.

It is possible to map some parts of ISO 9001 to CMMI: the lowest three levels of CMMI cover more or less the quality demands of ISO 9001. Hence, a company that resides on a fairly high CMMI level should easily "survive" a ISO 9001 assessment what leads to a well-accepted recognition and reputation. The same may be valid for the opposite direction, ISO 9001 certified software teams are well-suited for a CMMI assessment. In addition, it is possible to be certified ISO 9001 and CMMI compliant at the same time. However, the overhead to run two parallel assessments concurrently is enormous and should not be neglected.

## Question 2

Prepare an assessment report based on the interview in the exercise session, covering the process areas *Requirements Management*, *Project Planning* ~~and *Project Monitoring and Control*~~. Assume that everything said in the interview is correct (unless contradictory) and no additional confirmation is necessary.

Two experienced ☺ assessors, namely Sebastian Schenk and Jakob Magiera, started an assessment on May 16<sup>th</sup>, 2003 at 3:40pm. They inspected the process of a client/server project: about fifteen developers take advantage of a J2EE Standard Environment, the projects should be finished within six months.

### Requirements Management

#### *Specific Goal 1*

SP1.1 The studied project uses a database as a basic tool for its Requirements Management. That database holds all requirements with no further distinction between customer and product requirements. Nevertheless, a priority based system ensure to set up an order of importance. The whole database consists of data gathered jointly by both the customer and the software team (or its managers) which leads to a clearer understanding. Each record includes the requirements, their priority and the source of that requirement. Even though the database has been set up in cooperation, there is a necessity to verify if the requirements and their priorities represent valid and reasonable data.

SP1.2 The contract does not further specify who is in responsibility to track the requirements. During the interview, it revealed that anything related to SP1.2 is vague and imprecise.

SP1.3 Most changes to the specification arise when special meetings between the customer and the assessed company take place. These changes can be tracked via a simple text based versioning system. I detected frequent updates to the requirements despite the system is not explicitly a true and powerful versioning system, it just adds the new or changed requirements and marks their dependencies. A history of changes is available, too. However, the people use the system efficiently, and “they must do it”.

SP1.4 The functional design fulfils the demand for a bidirectional traceability in a bottom-up manner. There are still some problems when tracing some requirements from the functional design to the customer, though.

SP1.5 When detecting inconsistencies between requirements and the resulting product, the workshops held provide an opportunity to take a closer look at the system. In addition, changes to the requirements are followed by a consistency check. The mentioned database stores problem reports then. The person present at the assessment admitted a missing validation: the team does not ensure/validate to produce a complete solution, some requirements may be “forgotten”.

Conclusion: The main practices of specific goal 1 (managing requirements) are put into work. Some minor details have to be corrected or improved.

*Generic Goal 2*

GP2.1 Different levels of CMMI expect the accomplishment of different generic goals. Since each level includes all generic goals of all lower levels, at least generic goal 2 has to be completed in order to achieve an higher level than 1: the institutionalisation of a strategy of a managed process in the whole organization. Even though there are techniques that belong to a managed process, a thorough documentation – needed in order to preserve continuity – could not be found.

GP2.2 Indeed, the plan seems to be mature: early phases, such as the initial assembling of requirements are reasonable. Late changes to any requirements are well-defined although the tools used do not meet modern demands.

GP2.3 As mentioned, the utilized tools do not prepare a optimal ground for an advanced software development process. Nevertheless, they are known to the staff and thus no further educational efforts are required. On the other hand, the assessed company did not give any information on the monetary aspect and the availability of staff, in particular skilled persons able to handle special tasks or being responsible for creating and presenting results of the requirements management process.

GP2.6 Visual Source Safe, used in the process, provides all necessary services of a proper configuration management tool as satisfies the substantial requirements of the generic practice 2.6.

GP2.7 That configuration management tool is under administration of some persons. Exact names were not available in the interview but the assessed underlined the existence of an identification process. In consequence, an evaluation of their involvement could be made.

GP2.8 The realization of generic goal 2.8 turned out to be as fuzzy as generic goal 2.7. I was unable to become aware of any process supervision and controlling with the exception of the mentioned problem reports.

GP2.9 An objective evaluation of be aligned to the defined requirements demands the invocation of some metrics, external persons or assessment methods. The only verification by non-core members is done at the workshops at an very low degree.

GP2.10 I did not reveal any information throughout the interview concerning a review by the upper management. Even the regular status reports – which remain a passive action – do not state anything about an involvement of the upper management.

Unlike the specific goal 2, there are some obvious shortages when it comes to accomplishing generic goal 2. Nevertheless, some generic practices are just unknown to the assessor, therefore a second, more directed interview may disclose further details. The company may (!) be considered to reside on level 2 whereas a quite conservative assessor possible will rate the organization as being on an “initial” level.

*Synopsis*

CMMI is an on-going process focussing on continuous enhancement of software quality issues. A company typically likes to achieve an upper level but usually is regarded at a low one on its initial assessment.

The interviewed organization clearly showed some deficiencies concerning the specific goal 2 and the generic goals 2. Although there only a few lacks related to specific goal 2, it is evident that the company failed to reach generic goal 2.

The report reveals the found insufficiencies in order to help the company to perform better in their next assessment. According to the slides provided in the lecture, that assessment will take place within about 18 months, so there is some time left to improve the internal quality of the requirements management process.

Commitment

Ability

Directing  
Implementation

Verification

## Project Planning

### *Specific Goal 1*

SP1.1 Preparing estimations substantially drives the future growth of a company. A Work Breakdown Structure splits up the whole project into smaller pieces easier to handle that later serve as a profound basis for all estimations. Such a plan does not exist or at least there is no detailed proposal.

SP1.2 Three basic attributes describe the required efforts: a manpower of roughly 15 developers work in three teams for a time span of about 6 months at a total price of approximately 150,000 €. Unfortunately, it cannot be ensured that these numbers (except for the time span, see SP1.3) are actually tracked. A further important attribute, the code size (LOC) undergoes no estimation. The numbers given above are an outcome of subjective estimations based on personal experience and feelings instead of established methods like metrics.

SP1.3 The definition of several milestones gives a fine-grained plan. Nevertheless, it is not clear how the phases Requirements Analysis, Functional Design, Technical Design, Implementation and Test map to these milestones. The used process model is quite similar to the Waterfall model.

SP1.4 Just the time needed for the whole project is estimated, no approximations on each milestone are available.

Altogether, specific goal 1 is not arrived due to a missing Work Breakdown Structure. That practice plays a very important role and cannot be omitted under any circumstances.

### *Specific Goal 2*

SP2.1 As mentioned in SP1.2, the project costs up to 150,000 € and lasts for 6 months. One can be sure that the company carefully tracks these values, however, in the interview these procedures did not come to light.

SP2.2 An identified problem is integrating new developers since it takes some time to train them. The second problem that the team discovered is being possibly unable to cope with changed requirements and still keep on time. No further problems were mentioned (like reducing the final price if the team does not implement all features).

SP2.3 Data management is closely connected to SP1.3 and GP2.6 of Requirements Management. This time, Visual SourceSafe and the Concurrent Versioning System are involved: the latter to store source code while SourceSafe manages all documents.

SP2.4 In addition to the data management tools, Java and a suitable IDE are available (exact name unknown in the interview). It is impossible to infer from the interview whether all of the planned 15 developers will actually work on that project. I like to point out the uncertainty of the payment: maybe there is one final payment or several payments during the project's course.

SP2.5 The project leader likes to underline that the developers are trained quite well. If it becomes a necessity to engage new developers then an additional education may be required. Yet the interview did not reveal a detailed plan of the knowledge urgently needed, furthermore no jobs are assigned to special persons.

SP2.7 A plan has been created but it is neither extraordinarily transparent nor can an appropriate tracking and verification be ensured.

All in all, specific goal 2 has been achieved with some minor flaws in the tracking process. That area should be optimized in order to perform better in the next CMMI assessment.

### *Specific Goal 3*

SP3.1 Virtually no plans are reviewed with the exception of the main (technical) architecture and some not further specified so-called "special" plans. It is unclear to what degree the upper management reviewed the plans, in the interview the project leader just stated that they did it and finally signed the plans.

SP3.2 It can be concluded that there are not obvious discrepancies between the plan and the actual available resources. One should not forget that the estimated numbers were neither proved nor verified (SP1.2) by standardized and accepted methods.

SP3.3 Both the team and its leader are responsible for successfully carrying out the project with an emphasis on the leader. The role of the upper management is not accurately defined.

Like specific goal 2, there are some minor flaws in the realization of specific goal 3. The use of not properly reasoned estimates (SP3.2) causes some serious headache. Nevertheless, specific goal 3 has been achieved.

*Generic Goal 2*

GP2.1	The plan covers major areas of the company (team, team leader, upper management), the procedure used is not optimal but well-understood by all participants. I got the impression that the process could be easily repeated.	Commi- ment
GP2.2	I could not hardly detect a written description used to create the Project Planning plan. In contrary, all employees have a thorough knowledge and comprehension of the single step taken.	Ability
GP2.3 GP2.4 GP2.5	As mentioned before, there are nearly no information available on the concrete process of Project Planning in that company. So it is impossible to determine whether appropriate resources are available to the company, the responsibility is adequately distributed among the team and all participating persons are satisfactory trained.	Directing Implemen- tation
GP2.6 GP2.7 GP2.8	Anything said in GP2.3 to GP 2.5 applies to the Implementation Direction as well. It was impossible to reveal how the configuration is managed and traced, how involved persons are identified and how the process is tracked and controlled.	Verifi- cation
GP2.9 GP2.10	The verification of the Project Planning process is neither assessed objectively nor reviewed by the upper management.	
	One for sure agrees that the company in no way fulfil the requirements of generic goal 2 of Project Planning. The biggest deficit is a broad lack of information available in the company. There are no written guidelines and in addition the team leader did not give enough – better to say: did not give <i>any</i> – details on the procedure.	

*Synopsis*

Project Planning is one of the basic components of CMMI and thus mandatory for any company achieved at least level two. Nevertheless, there is some planning although one cannot determine what kind of. I rate the company as “initial” concerning Project Planning.

**Summary**

There are big problems in performing well in areas slightly more advanced than Requirements Management. According to the CMMI specification, the company resides at most on level 1, it is “initial”. In order to get a better rating in the next assessment, severe issues on the field of Project Planning have to be resolved. However, I dare to doubt that the company will climb to level 2 or even higher within a short period without undertaking rigorous and fundamental changes in their quality process.

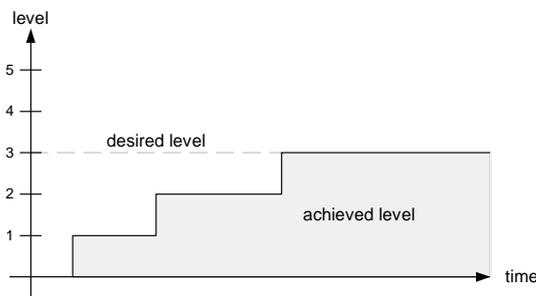
### Question 3

*How does the distribution of levels achieved in assessments vary over time ? Explain !*

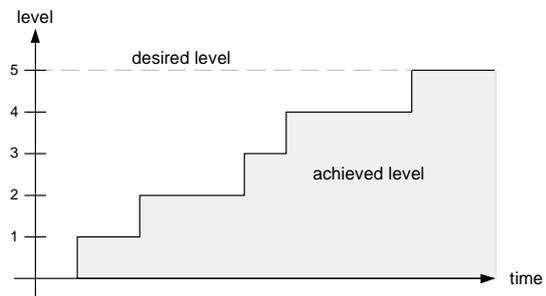
I discussed in question 1 that CMMI is an on-going process needed to be repeated from time to time. Once an initial assessment took place, the company or department is responsible to maintain or enhance the current level of quality. An idealized CMMI aided way would steadily increase the CMMI level until it reaches the top level.

The reality of software development turns out to be immanent to such idealized visions. Typically, a company is initially assessed to reside on one of the lower levels, usually level 1 or 2. All participating persons have to invest lots of efforts in order to get up one level. It is not always desirable to achieve a higher level since it the financial resources and, more important, the time spent require to pay off. So if the written software involves not too safety-critical (hence heavily quality depended) code then it may be worthless to be on a high CMMI level.

The few companies that recognize CMMI as something like “advertisement” or “trust policy” because their customers show an elevated interest in quality intend to reach level 5. Unfortunately, it takes some time to get there, some surveys discovered a time span of about 18 to 24 months for an increase of one level.



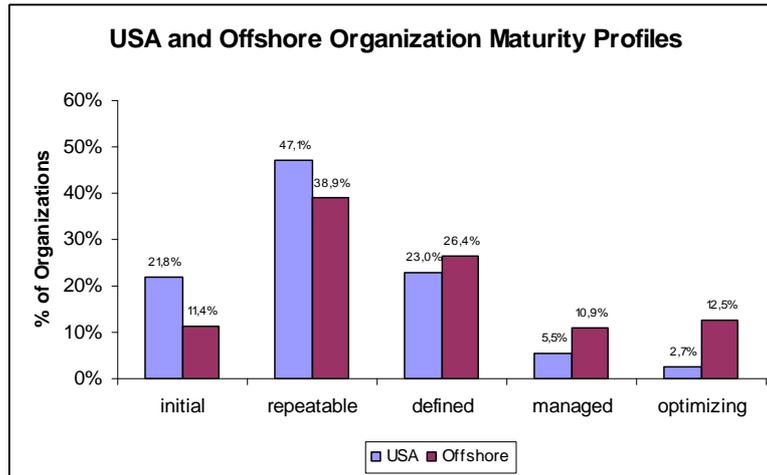
**Figure 3:** Reaching CMMI level 3



**Figure 4:** Reaching CMMI level 5

Figure 3 and Figure 4 show some imaginary maturity improvements. Even though they are not based on actual data sets, they give some insights in the typical property of CMMI: enhancing the process takes unequally distributed time spans. Furthermore, higher levels require a longer period of preparation since the requirements become more and more complicated to fulfill. In fact, education and experience cannot be “added immediately” to a software team.

Indeed, these figures omit the possibility of being assessed at a lower level than before. This represents a vital danger for companies undergoing high fluctuations of responsible manager and qualified personnel. Especially the soft skills of these persons, such as education and experience, play an important role in software quality and hence identify the CMMI level to a high degree.



**Figure 5:** Distribution of Maturity Levels  
(from SEI, “Process Maturity Profile 2002”, 2003)

Figure 5 clearly shows an astonishing distribution: more than one out of two companies were assessed as “initial” or “repeatable”. These numbers are due to a quite simple reason: CMM and CMMI has been defined just a few years ago and therefore is too new to be early adapted by the software industry. Most company seem to be on the way to their desired level but did not achieve it yet (see Figure 3 and Figure 4).

In the not-too-close future, I expect a slight shift of the median from level 2 to level 2-3. The more CMMI matures and gains reputation, the more the industry will recognize a need for “repeatable” or “defined” processes. However, the fraction of “managed” or “optimizing” may remain approximately the same as these who have a deep interest in extraordinary quality already adapted CMMI and are part of the figure given about.