

Real Life Experience Using CMMI L2 Processes and XP Practices

Orhan KALAYCI
Nitelik Danışmanlık Ltd.

Sarit DİNMEZ
Emel SAYGIN
Serden FERHATOĞLU

Gülfer AKGÜN
Şenol BOLAT

Hasan ÖZKESER

BİMAR Bilgi İşlem Hizmetleri A.Ş.

Orhan.kalayci@nitelik.net

sdinmez@izm.bimar.com.tr

eondul@izm.bimar.com.tr

sferhatoglu@izm.bimar.com.tr

gakgun@izm.bimar.com.tr

sbolat@izm.bimar.com.tr

hasan.ozkeser@izm.bimar.com.tr

Abstract

In December 2003, BİMAR has initiated an SPI project to reach CMMI L2 in 2004 using XP practices. Nitelik Danışmanlık Ltd. has planned and executed the project. This paper aims to describe the pilot project where three CMMI L2 processes were implemented. These processes were Requirements Management, Project Planning and Configuration Management. The three CMMI processes were implemented, together with three XP practices of, Customer as a Member of Team” (the Card System), Pair Programming and First-Test-Then-Coding. The pilot project concluded very successfully. The main lesson to be drawn from the pilot was that the CMMI processes and the XP practices worked well in great harmony.

1 Introduction

In 2002 BİMAR has initiated an Organizational Change Program aimed at enhancing customer satisfaction. One main part of the Change Program was the improvement of the software development processes. BİMAR launched its SPI program in 2003. CMMI was chosen as the maturity model. XP was chosen as the software lifecycle model to be used together with CMMI. [4] In this sense, it is the first experience in Turkey in using CMMI processes and XP practices together.

The objective of this paper is to share our experience in piloting CMMI L2 processes together with XP practices. The processes piloted in the study were Requirements Management, Project Planning, and Configuration Management. The XP practices piloted were Planning the Game (cards), Pair Programming, First-test-then-coding and Customer as a Team Member.

For each process piloted we have given the following information: the purpose of the process, the metaphor used in designing the process, the main activities identified for the process, and the strategic decisions about the process.

For the XP practices, the following topics will be covered: Observation made about using the cards In XP for planning the game, experiences about pair programming, and the implementation of first-test-then-coding. [1]

2 CMMI Processes Piloted

2.1 Requirements Management

2.1.1 Purpose [5]

- Taking the customers’ needs accurately and precisely.
- Making sure that both BİMAR and the customer are committed to the project.
- Tracking these commitments so that project completed is on time and budget is met with zero defects.

2.1.2 Metaphor

While designing the metaphor, we imagined a restaurant where the food is prepared in the kitchen by the cooks under the watchful eyes of the patrons.

The role we designed for the waiter ensures that the waiter greets the customers at the door and promptly gets their orders. However, newly arriving patrons are not immediately shown to the kitchen until the previous group of customers are being served in the kitchen. Once taken to the kitchen, the customer can do much more than be simply involved with cooking: He/She can also contribute to decisions on the time allocated to cooking and the ingredients that are used.

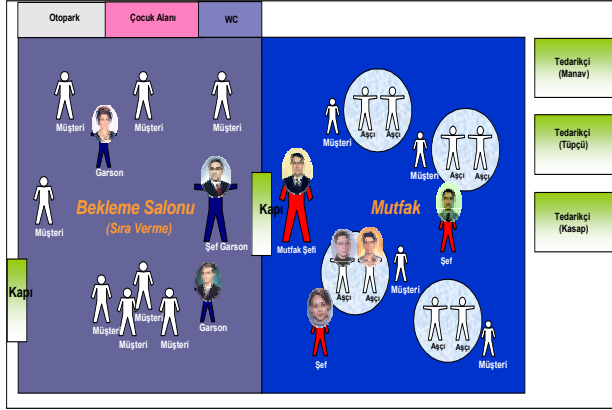


Figure 1 – Metaphor

After investigating the metaphor in detail, we identified six major steps to define the requirements management process. These steps were as follows:

- Take the order at the front door
- Show a proactive approach
- Forward the order to the kitchen
- Track the order
- Serve the customer
- Close the order (check the bill)

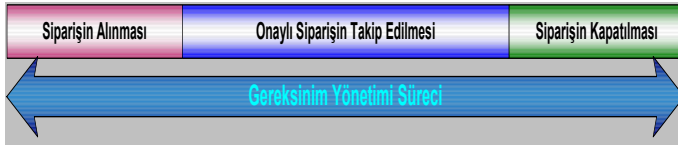


Figure 2 – Main Steps for Requirements Management

As we combined the ‘Forward the order’ step with the ‘Take the order’ step, the three main activities were determined as follows: (See Figure 2.)

- Taking the order
- Keeping track of the accepted order
- Closing the order.

We concluded that showing a proactive approach and a sophisticated approach was necessary for each step as they directly impacted service quality and customer satisfaction. Therefore, the following two steps are implied within each of the above three steps:

- Proactive Approach
- Sophisticated Approach

2.1.3 Main Activities

Taking the Order;

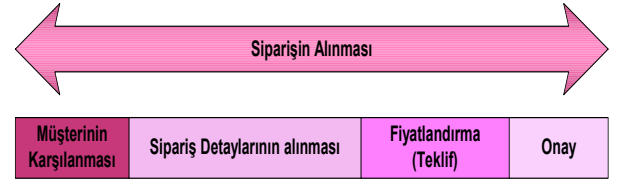


Figure 3 – Taking the order

- In this step, we first check if the customer has the authority to place an order. If so, we put the customer in touch with the right person on the Bimar side..
- After this appointment, the customer needs are gathered by BIMAR’s Estimation Team which is composed of qualified software personnel who help the customer to fill the Customer Cards and the verification (Test Scenario) cards
- The estimated time magnitudes that are given by the Software Estimation Team are used in order to form a price proposal and eventually let a final offer to be formulated.
- When the proposal of the software Project is approved, the Project Office is asked to supply an appointment date (Project Start Date).

Keeping Track of the Accepted Order;



Figure 4 – Keeping track of the accepted order

- We divide this step into two time intervals;
In the first level, the states between the approval date and the appointment date are tracked and managed. The second level takes place between the appointment date and the date of termination. Appointment date is the date when the Project Office launches the Project and the termination date is the date when the Project Office declares an end to the project.
- During the life cycle of the Software Project, scope and cost issues are handled by the Customer Relations Representative

whereas time and quality issues are handled by the Project Leader.

Closing The Order;



Figure 5 – Closing the order

This step starts as soon as notification comes from the Project Office indicating that the work is completed and the Project can now be closed in the following way:

- The assessment of the Project by the customer is taken by a Project inquiry questionnaire.
- As the technical closure is made, the Project resources become available to other projects.
- The Financial track includes the invoicing of the products and the control of the payment.

2.2 Project Planning

2.2.1 Purpose

The purpose of project planning process can be described by the following recursive steps: [2]

- Do the most valuable (the most important, urgent, etc...) task
- Coordinate the related parties
- If any change occurs, go to the first step

2.2.2 The Metaphor

In the design of the process, the metaphor of a Japanese Restaurant is used. In this restaurant, customers first wait at the entrance to be seated in the dining room. In the dining room, the food is cooked in the middle of the table arrangements. The two cooks prepare the food in accordance with the customers' orders.

The waiters take the customer orders at the entrance hall and forward the orders to the chef inside. The chef puts the orders in orders and shows the customers to their seats in the dining room. The food is prepared in the presence of the customer and the process continues

until the customer says "OK" The table then proceeds to enjoy the meal.

In this example, the chef is actually the project manager and the cooks are a pair of programmers. The chef brings the right customer and the right cooks together, plans what to do and in what order, executes the plan and solves any problems that may arise.

2.2.3 Main Activities

Project activities were performed according to the processes involved. The main steps were project planning, project tracking and project closure.

At the beginning of the project, iteration based planning was tested. This was the first meeting with cards. Customer cards were grouped to represent working program parts. One part would be completed in one iteration (4-6 weeks). Then, the technical cards were filled.

The customer cards were divided two day lasted technical cards and then the project plan was done. We observed that use of cards made project planning much easier. Likewise, it was became easier to keep track of the project tasks that are divided into smaller pieces. The average duration for the completion of technical cards was two days. At the end of this average period, the status of the card would become apparent, the problems that were negatively impacting the schedule would come to surface and hence such problems could be solved before it was too late.

As the two programmers worked as a pair, the actual task durations turned out to be shorter than planned. The resulting programs were of high quality, and the customers were very satisfied with the results.

The work environment is very important for the productivity of the programmers. Because of this the environment should be planned at the beginning of the project. The pair programmers should work together at one conformable table without being distracted by a ringing telephone, interrupts from other people or any other stimulus.

During the project, if the pair programmers and the customer met every day and worked together for several hours, project turned out to be successful. As a result of such communication, changes in the requirements could be easily accepted and acted on quickly.

We contend that the most powerful way of sharing information is face-to-face discussion.. During the project, regular status meetings were held to make the information distribution homogeneous and to address problems as a team As a result of the good team work, the project was successfully completed.

2.2.4 Strategic Decisions

1. In order to make the project portfolio system work successfully, project time schedules should be fixed and not deviate during the execution of projects.
2. To improve customer satisfaction and enhance the image of Bimar in the eyes of the customer, it was decided to allocate less resources to bug fixing. Paradoxical as it may seem, this decision need not result in sacrificing from quality. On the contrary, customer satisfaction should increase. The reason for this is that, even when a project is closes on paper, continued bug fixing means that it remains effectively open and the project's resources could not be allocated elsewhere. If less resources are allocated to this effort, more resources will be made available to other projects.
3. Invoicing should be made as soon as the working parts of the program are completed.
4. This process will be used only for the software development projects in Bimar.
5. According to the project's size and urgency, different approaches will be adopted for the project. For example, for an urgent and middle sized project, the approach will be 'start immediately and document later'.
6. The project will be planned iteratively.
7. The project will be planned by using the card system. [2]
8. Customer cards will be created as manageable, independent, small parts. A pair programmer will be able to complete the card in at most 2-3 weeks. [2]
9. The team will work in only one project at a time and to be able to concentrate on that given project, they will not be permitted to work for another project.
10. XP suggests the specialization in the business area and does not suggest technical specialization. [3]

2.2.5 What the pilot project means for Bimar

One of our colleagues in BIMAR acts in the project as the customer. The new process was tested with this "fake customer". Before applying the process to the real customer, this test provided the chance to make the necessary improvements in the process.

The opportunity of comparing the designed process with its actual implementation helped the process design teams to see the differences between the two.

During the project, preliminary quality assurance, measurement and analysis reports were prepared and process improvement was exercised. These exercises created the ideas to make the process more convenient. For example, to make the process more practical, standard content templates (communication plan and the like) were created.

During process design, good team work and taking every idea into consideration were seen to be the critical success factors. The process was designed by the team containing people from different departments. After the design period, the spreading period started. The project team members who took part in the process design worked actively to execute the process during the project. As a result, process spreading activities were seen to be effective.

All the process design and spreading activities show that Bimar took a huge step toward becoming a real "learning organization".At this age, the organizations who act quickly against the changes will be successful. We are proud of working in such a company that learns to live in a continuously changing environment.

2.3 Configuration Management

The Configuration Management System provides and maintains the consistency and integrity by using the identification, control, status assessment and audit activities for work products that form the essential values of an organization.

The Configuration Management System is designed for the purpose of protecting the data that is the most valuable asset of an organization. The existence and usability of the Configuration Management System in

an organization ensures speedy and easy access to the most reliable data within the limits of authorization.

2.3.1 Purpose

The purpose in the establishment of the Configuration Management System is to identify the work products that will be kept in this system, to follow up the modifications on work products and to provide and maintain their consistency and integrity by carrying out controls on them.

For this purpose, the special Goals and practices of CMMI level 2 are as follows; [5]

Special Goal 1. Establish Baselines

Specific Practice 1.1 Identify Configuration Items

Specific Practice 1.2 Establish a Configuration Management System

Specific Practice 1.3 Create or Release Baselines

Specific Goal 2. Track and Control Changes

Specific Practice 2.1 Track Change Requests

Specific Practice 2.2 Control Configuration Items

Specific Goal 3. Establish Integrity

Specific Practice 3.1 Establish Configuration Management Records

Specific Practice 3.2 Perform Configuration Audits

After the Configuration Management System was established in the company, a Configuration Management Plan was prepared for the internal pilot project with which the functionality of the system was tested. The work products to be included in the system, authorization on the work products and relations between them are defined in the plan. Conforming with the prepared plan during the project period, the consistency and integrity of the Configuration Management System for the project was ensured by following up the modification on work products and the version management.

During the performance period of the internal pilot project, the joint operation of the configuration management process and other processes were observed, the resulting suggestions were assessed at the status meetings and the improvements were made on the process.

2.3.2 Metaphor

In the process design and internal pilot project studies the metaphore of a bank was used.

We believe this is a fitting metaphor as there is a good match between a bank and when doing configuration management in terms of purpose, working method and customer satisfaction criteria.

The use of a metaphor during the process design and the internal pilot project ensured that idea sharing among the team members was carried out effectively.

2.3.3 Main Activities

In the design studies of the Configuration Management process, it was determined that there was a need for the roles of a General Configuration Manager and a Project Configuration Responsible, and appointments were made for these roles. While applying the configuration management process in the internal pilot project, three main steps that were established in the process design studies were taken.

Three main steps for the configuration management system were as follows;

1. Project-Starting : The top management has the authority to start a project in the Configuration Management System. A Project Configuration Responsible is appointed for the project. The Project Configuration Responsible, together with the Project Manager or the Project Customer Relations Representative, prepares a Project Configuration Management Plan. The prepared configuration management plan is sent with the order form to the General Configuration Manager by the person who orders the start of the project.

The General Configuration Manager executes the Project-Starting step.

2. Project-Operating : In the Project Configuration Management, a demand form is completed to perform a procedure on the work products. For the related order, the Project-Operating step is executed by the Project Configuration Responsible. The following procedures are included in the execution:

- To check in the work product
- To check out work product
- To delete the work product
- To create a baseline
- To withdraw the baseline
- To update the data on work product

3. Project-Ending: In the Project Configuration Management, the Project manager or in case he /she is not appointed, the Customer Relations Representative

has the authority to make the order for the ending of the project.

With the order of project ending, the Project Configuration Responsible executes the Project-Ending step by making sure that all the baselines taken places in the highest security level in the BIMAR storage system.

Certainly, during the application period of the Configuration Management System, some difficulties were experienced. To eliminate them, some improvements were introduced.

The purpose of the internal pilot project was to identify and eliminate the deficiencies for all the processes as well as to bring in improvements by evaluating suggestions and different approaches.

During the performance period of the internal pilot project, it has been determined at the beginning that the persons included in the project did not know the Project Configuration Management and its significance. Moreover, its application was seen as a big work load. Additionally, other processes did not trigger the application of the Project Configuration Management. In order to clarify this point, an informative guide was prepared to emphasize the significance of the Project Configuration Management and to make its application easier.

Again, in the studies carried out during the application period of the pilot project, the metaphor of a theatre was suggested. All the processes came together to work together with this regard, and an instructive scenario about the functionality of processes and their relations was prepared in BIMAR. The persons included in the process transformed the scenario into a play that was staged in the presence of all BIMAR employees. Thanks to this play, all the processes designed, including the Project Configuration Management process, became more understandable for the audience which included the process project teams.

Another issue was that when any work product was required to be taken off the Configuration Management system just for the purpose of reading/reviewing regardless of whether it was necessary to complete a form for such a procedure. As such questions were asked very frequently, the application process of the system was improved by giving read access authorization to all project members thereby eliminating this problem.

Based on the suggestions, it was decided to improve the Configuration Management System Order form that must be completed for each order in the Configuration Management System.

2.3.4 Strategic Decisions

The following strategic decisions were taken during the process design and internal pilot project studies and were approved by the top management;

- A tool must be used for the sound operation and functionality of the Configuration Management System. However, to determine the expectations and selection criteria related to the tool to be used, firstly a system was designed to be operated manually. For this reason a file server was allocated to store the work products and an MS Excel file was allocated to record the movements and relations of the work products within the system. The file server had confidential and private areas related to the project.

Confidential area: Allocated to the baselines and ended projects. Only the General Configuration Manager has authorization to access this area.

Private area: Allocated to the ongoing projects. Only the Project Configuration Responsible is authorized to access this area.

- As no tool was used for the Configuration Management System yet and as its manual follow up was difficult, the work products of the project and source codes were not included in the Configuration Management System.
- To be able to follow up the work products of the confidential area in the Configuration Management System, there must be a Company General Configuration Management Plan.
- For each project to be taken into the Configuration Management System there must be a Project Configuration Management Plan.
- There must be a General Configuration Manager and a Project Configuration Responsible role in the company.

General Configuration Manager: Responsible for the confidential area in the Configuration Management System. Carries out and follows up the procedures of the project starting, project ending and creates baseline steps in the confidential area. Responsible for

following up the company General Configuration Management Plan.

Project Configuration Responsible: Fully authorized person for the special area for the relevant project in the Configuration Management System. Responsible for the operation of all the process steps in and following up the project configuration management plan.

3 XP Practices Piloted

4.1 Cards

Cards are used to enable the customers to write their requirements in detail. They are also used to determine the project duration and cost. Customer cards should be written by business specialists that have the sufficient authority and business knowledge. These business specialists are to be assigned by their managers and they will work as a team member. They should work with the team preferably 5 days a week, and not less than 2 days.

Each card written by the customer should include a maximum of 2 weeks of work. The cards that are longer than 2 weeks should be divided into cards involving shorter time periods. [1]

The validation card is written by the customer, after the customer card, in order to define how to check the output. The Validation card includes the acceptance criteria for the relevant customer card.

The cards should be prioritized by the customer according to urgency and expected value,. The cards chosen are implemented through iterations and delivered to the customer. During the project, the customer may realize that some of the cards are not important for him and may cancel them.

The advantages of a doing a project using cards are as follows;

- The schedule and scope of the plan are easily tracked.
- Customer pays only for the finished parts (cards)
- Prioritization of the requirements is made much easier.

In our pilot project our new way of working was first presented to the customer. The customer was positive about the card system. He remarked that a project we did for them in the past would have been more successful if we had used this system.

A business specialist was assigned for the project by the customer. The business specialist (customer), working together with the Software Estimation Team, wrote the requirements on the cards. After the project

started, he worked together with he project manager to prioritize the cards and to plan the following iteration. The customer could track the status of the project easily as the periodic status reports were written in terms of cards. Customer satisfaction was increased as he worked with the team and could see the results as soon as the work was done. He could also make changes in his requirements and add new ones while working with the team.

At the end of each iteration, the programs for the chosen cards were delivered to the customer.

The pilot project was finished in 3 iterations. Based on the customer satisfaction survey that was carried out, we found that the customer was satisfied with our new way of working.

Card examples:

Figure 6 – Customer Card

Figure 7 – Validation Card

Figure 8 – Technical Card

3.1 Pair Programming

The pair programmer concept means that two programmers sit at the same table and look at the same screen and write code by rotation. As one of them writes the code, the other traces the code at the same time. [1]

At first sight, this may seem to be a waste of resources to allocate two programmers for the same project, in practice it is very good idea. In our experience, we observed that in the pilot project that the team of two programmers wrote code more quickly than the case of one programmer. As the second pair of eyes traces the code, the problems are discovered and solved early on. In addition to this, continuous control of the code reduces bugs; and as a result, the number of defects is reduced to a small number. This both saves time and increases quality.

We can express mathematically why pair programmers perform better than two individually working programmers when coding and then debugging.

In the beginning, the different knowledge levels of the two programmers may be an issue. However, with the progression of time, this issue becomes less and less important. When implementing XP practices, we observed that the programmer that is behind in knowledge could catch up with the other. One must note, however, that if the two programmers are of equal level of knowledge, then the project moves faster than otherwise. There is an important side advantage of this method: New programmers can quickly be put into action by pairing them with experienced programmers. This results in an organization where programming knowledge is shared effectively and where knowledge level becomes homogeneous over time.

level, projects go faster. However there is an important advantage of this work style that the new programmers in the area learns very quickly by working together with the expert programmer and start to produce the code in the

area. As a result the knowledge is shared and become homogeneous in the organization.

Of course, psychological factors should be taken into account when making the pair. If the pair programmers get along well, the project goes well too and the work is pleasurable.

3.2 First Test then Coding

In the project, program code was written in RPG on an IBM AS/400 system.

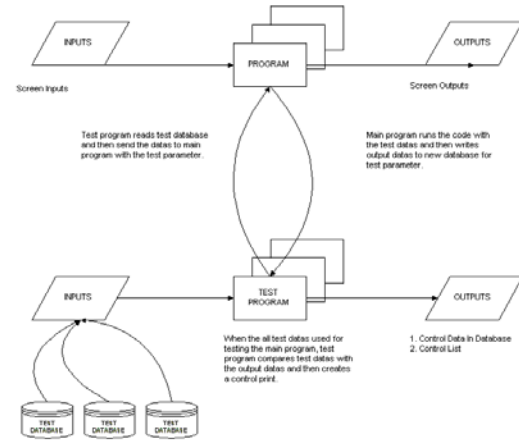


Figure 9: Test program calls the main module, provides it with inputs and checks if outputs fit the expected outputs

As distinct from classical approaches and in line with XP practices, the first action in coding was to write the test code. [1]

For this purpose, inputs and outputs were defined first.(Figure 9)

When writing the test code in RPG on an AS/400 system, the most difficult task was how to get the inputs. The reason for this is that, in RPG language, screens run integrated with the code, and without conditions in the source code, a program will not run in test mode. We tried to minimize such kind of changes in the main program.

The goal of the test method is to control the program by the another program. While writing the test code, at the beginning of every test, some test data were defined as constant. For example, to match the port number and the port names, both are defined as constant data, and then port number is obtained from the database

dynamically and verified. This method provides the test code of a long shelf life. Such test code can be used without any change in the future. Besides, for every test, the physical file is created on AS400 and the test data is entered into this file. Likewise, the data at the end of the test is saved into another physical file. At the end of the test, the control list is built by comparing the outputs of the main program with the test data. The outputs denote the data that the user wishes to obtain. This can cause some difficulties in file management, because the number of the files increases when testing complex modules.

Every piece of information entered by the user or that is available on the lists should be controlled by the test program. In RPG like programs, writing the test code of the program (whose purpose is to obtain lists) is more difficult than writing the test code for the screen controls. This difficulty increases as the lists get bigger and more complex.

In spite of these difficulties, this method provides quality software and, as a result, high customer satisfaction.

The other advantages of writing the test code are the following;

- 1- While writing the test code, it is possible to look at the program from different angles and the defects in the program can be discovered.
- 2- All the program conditions are tested with different test data.
- 3- Programs can be tested at any time by pressing a single key.
- 4- The test is made by the program and this ensures the objectivity of testing.
- 5- It is possible to observe the impact of the most recent changes on the program.

4 Conclusion

We found that XP practices supported CMMI processes very well: Requirements Management, and Project Planning processes became very simple when cards were used in hardcopy as suggested by XP. However, using hardcopy cards brought some disadvantages related with Configuration Management.

The final word can be “While CMMI creates an organizational discipline; XP eases the daily life

by providing pragmatic, end-result-oriented practices. CMMI and XP can be used together very well and their synergy is very strong.”

	Cards	Pair Programming	First Test then Coding
Requirements Management	++	N/A	+++
Project Planning	+++	++	++
Configuration Management	--	N/A	N/A

Table 1 – Relationship between some of CMMI process with some XP practices

5 References

1. Kent Beck, “Extreme Programming Explained: Embrace Change,” Addison-Wesley ,October 5, 1999
2. Kent Beck, Martin Fowler, “Planning Extreme Programming” Addison-Wesley, October 13, 2000
3. Ron Jeffries, Ann Anderson, Chet Hendrickson, Extreme Programming Installed, Addison-Wesley, October 13, 2000
4. Orhan Kalaycı, CMM vs Extreme Programming, Impact of Software Process on Quality, Bilkent University, June 2003
5. SEI, CMMI-SE/SW/IPPD/SS Version 1.1, Carnegie Mellon University, March 1, 2002

6 Acknowledgements

This paper would not possible without commitment of and support from the top management team whose members were İsmail Berkan, Hakan Erdem, Köksal Ataman, Umut Altınay, Azim Aydınlı, Osman Çiftçi, Sarit Dinmez.

Many thanks should go to the process members:
Requirements Management: Emel Saygın, Zerrin Sadi, Berkin Enbatan, Hakan Erdem, Kerim Önder, Necmi Şentuna

Project Management: Serden Ferhatoğlu, Aydan Kantar, Şenay Çalışkan, Habil Şenocak, Gökhan Dağhan

Configuration Management: Gülfer Akgün, Gökmen Eren, Hasan Özkeser, Hülya Küçükyörü, Özge Yılmazkurt,