Лекція 1.2
Модель Software Maintenance Maturity Model (SMMM)

Питання до розгляду

- Структура процесу супроводження програмного забезпечення.

1. Контекст супроводження програмного забезпечення.
2. Супроводження програмного забезпечення та унікальні для нього процеси.
3. Ключові процеси супроводження програмного забезпечення.

II. Мета та обґрунтування моделі SMMM.

III. Архітектура моделі SMMM.

IV. Області процесів та ключові процеси моделі.

V. Література

I. Структура процесу супроводження програмного забезпечення

Контекст супроводження програмного забезпечення (1)

- В організаційному контексті супроводження програмного забезпечення існує декілька типових інтерфейсів (рис. 1):

  • замовники та користувачі супроводження програмного забезпечення (позн. 1);

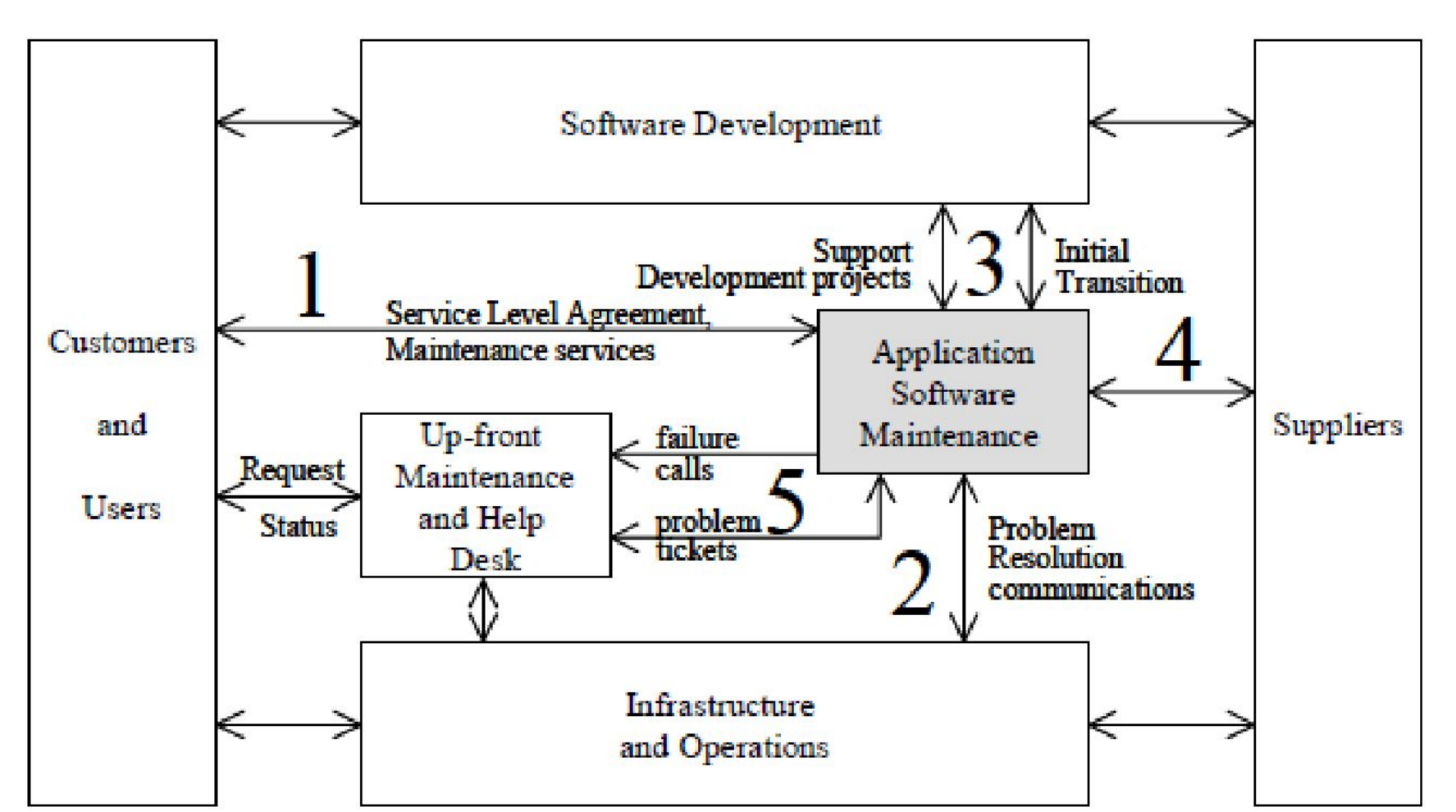
  • інфраструктура та операційний департамент (позн. 2);

  • розробники (позн. 3);

  • постачальники (позн. 4);

  • попереднє технічне обслуговування та helpdesk (позн. 5).

Рис.1 Контекстна діаграма супроводження програмного забезпечення



Контекст супроводження програмного забезпечення (2)

- ***Клієнтський інтерфейс (1).*** Діяльність включає в себе обговорення пріоритетності окремих запитів, планування, бюджетування, клієнтського обслуговування та підтримки роботи користувачів.
- ***Інфраструктура та операції (2).*** Включає в себе обробку всіх операцій, пов’язаних з робочими станціями, платформами та мережами. Також включаються заходи щодо резервного копіювання, відновлення даних та системного адміністрування.
- ***Інтерфейс розробників засобів супроводження (3).*** Знання програмного забезпечення спеціалістами зі супроводження має велике значення для розробників, які необхідно замінювати або включати в існуючу систему успадковане ПЗ. Приклади діяльності: проектування тимчасових або нових інтерфейсів; верифікація бізнес правил та допомога у розумінні даних існуючого програмного забезпечення; допомога у переміщенні даних.

Контекст супроводження програмного забезпечення (3)

- ***Постачальники (4).*** До складу постачальників входять:

а) ті, хто розроблює нове ПЗ чи конфігурує ERP (*Enterprise Resource Planning*) системи;

б) субпідрядники, які входять до складу персоналу зі супроводження та виконують вузько кваліфіковані завдання, допомагають підчас пікових навантажень;

в) компанії, що мають ліцензії на обслуговування визначеного програмного забезпечення, спеціалісти зі супроводження яких надають спеціалізовану підтримку цього програмного забезпечення;

г) аутсорс-компанії, які можуть замінити, частково або повністю, функції IT організації (*розробка, супроводження чи операційна та інфраструктурна робота*).

Контекст супроводження програмного забезпечення (4)

- ***Попереднє технічне обслуговування та helpdesk (5).*** Залежить від організаційної структури і може бути частиною функцій організації, що забезпечує підтримку програмного забезпечення, клієнтської організації або іншої незалежної організації, яка виконує підтримку ПЗ.

Процес супроводження програмного забезпечення та особливості використання (1)

SWEBOK визначає набір процесів, дій та практик, які є унікальними для спеціалістів зі супроводження, наприклад:

Перехід: контрольована або скоординована послідовність дій, завдяки якій система поступово передається від розробників до супроводжувачів;

SLA (**Service Level Agreement, угоди щодо рівня послуг)** та спеціалізовані (відповідно до доменної області) угоди зі супроводження, що укладаються спеціалістами зі супроводження;

Обробка запитів на модифікацію та звітів щодо проблем (**helpdesk**): процес обробки запитів, який використовується спеціалістами зі супроводження для визначення їх пріоритетності, документування та маршрутизації;

Прийняття/відхилення запитів на модифікацію: запити ретельно оброблюються у відповідності до розміру/трудоємності/складності та можуть бути відхилені та перенаправлені до розробників.

Процес супроводження програмного забезпечення та особливості використання (2)

У SWEBOK також зазначено, що ряд програмних продуктів та технологій адаптовано до специфічного середовища супроводження програмного забезпечення. Серед них:

Симуляція процесів: Ця технологія використовується для оптимізації супроводження при діяльності з покращення ПЗ.

Вимірювання процесу супроводження: Фахівці із супроводження часто використовують результати досліджень задоволеності користувачів, щоб зрозуміти, як працюють їхні клієнти. Деякі організації використовують комерційні додатки для розрахунку зовнішніх та внутрішніх показників супроводжуваності програмного забезпечення.

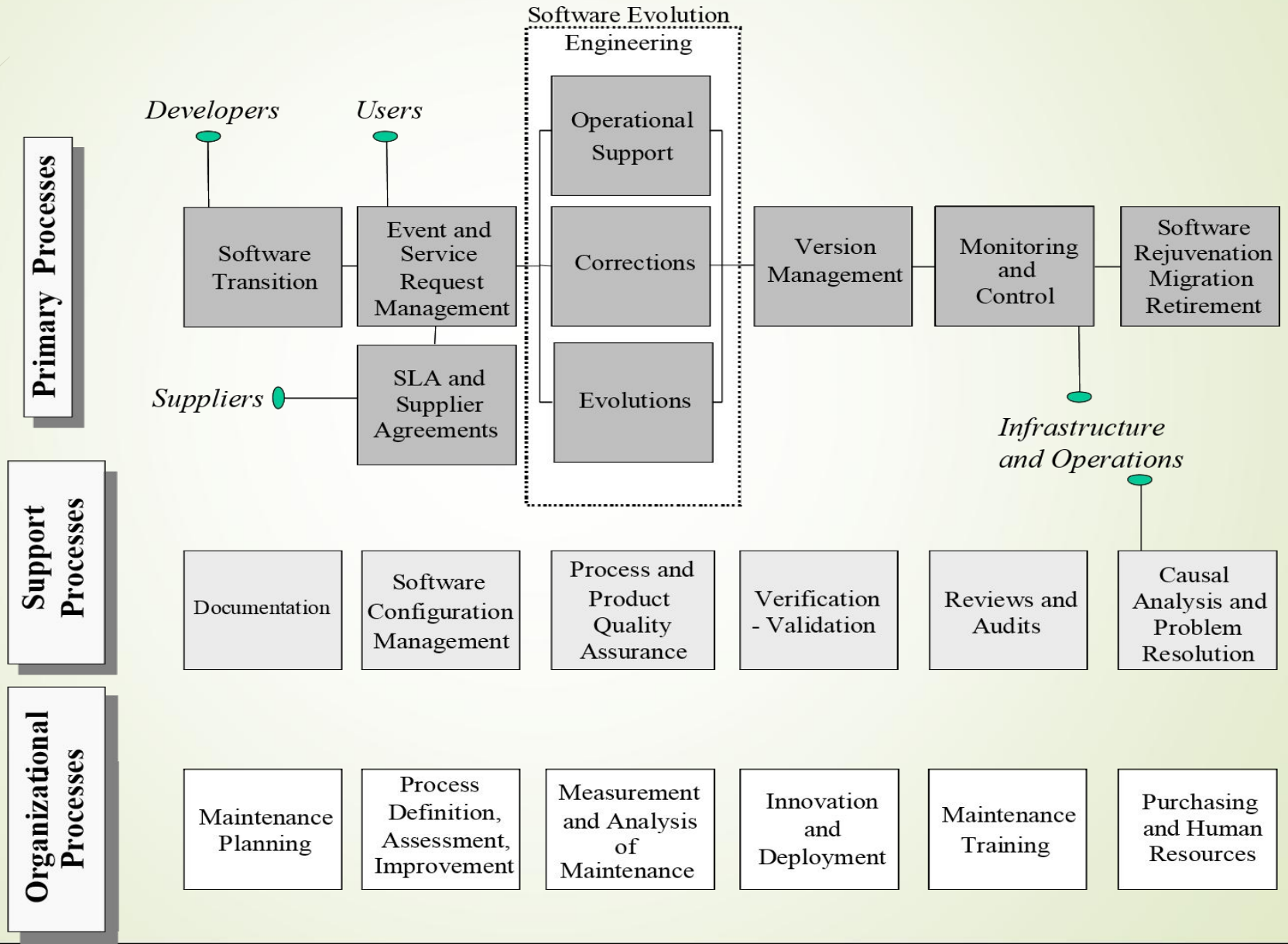
Процес супроводження програмного забезпечення та особливості використання (3)

- **Репозиторій запитів на супроводження:** Адекватна інформаційна система (часто спільна для helpdesk) повинна постачатися супроводжувачами для керування робочим навантаженням та відстеження великої кількості запитів користувачів. Такий репозиторій може стати основою для аналізу зусиль та важливою складовою для вимірювання.
- **Спеціалізоване навчання спеціалістів зі супроводження.**
- **Білінг послуг зі супроводження:** Найменування і ціни на послуги зі супроводження повинні бути описані та підтримуватися відповідними додатковими системами та процесами.
- **Виготовлення систем спостереження:** Організації, що надають послуги зі супроводження також повинні впроваджувати системи спостереження для щоденного дослідження операційного середовища на появу ознак погіршення роботи або відмови. Такі спостережні системи гарантують виявлення проблем на якомога ранніх стадіях (в ідеалі перш, ніж користувач дізнається про них).

Процеси супроводження програмного забезпечення (1)

- **Основні (ключові) процеси** (операційні процеси супроводження програмного забезпечення);
- **Технічна підтримка процесів** (підтримка основних процесів);
- **Організаційні процеси,** які представлені у Інформаційній системі (ІС) або у роботі департаментів організації (наприклад: навчання, фінанси, підбір персоналу, організація закупівель, тощо)

Ключові процеси супроводження програмного забезпечення (2)



Ключові процеси супроводження програмного забезпечення (3)

- ***Процес передачі.*** Цей процес забезпечує контрольованість програмного проекту, а також координованість та структурованість передачі програмного забезпечення спеціалістам зі супроводження. Вони будуть зосереджуватись на супроводжуваності нового програмного забезпечення, це означає, що процес передачі на супроводження відбувається протягом всіх етапів життєвого циклу розробки програмного забезпечення.
- Після того, як програмне забезпечення передане на супроводження, ***Процес управління подіями та запитами*** обробляє всі щоденні події, звіти щодо проблем, запити на модифікацію та запити технічної підтримки. Існують щоденні події, які потребують ефективного управління. Перший крок цього процесу оцінити, чи має бути запит виконаний, перенаправлений або відхилений (оцінка проводиться на основі **SLA,** характеру запиту та його розміру). **Угоди з постачальниками** пов’язані з організацією договірних аспектів (наприклад: ліцензії, можливість підписання договорів з третіми особами, тощо) та з SLA.

Ключові процеси супроводження програмного забезпечення (4)

- Прийняті запити документуються, пріоритезуються, розподіляються та обробляються у відповідності до категорії. Існують такі категорії: 1) ***Процес операційної підтримки*** (як правило, не вимагає змін програмного забезпечення); 2) ***Процес корекції програмного забезпечення***; чи 3) ***Процес еволюції програмного забезпечення***. Операційна підтримка складається з: а) відповідей на питання; б) надання інформації та порад; і в) допомога клієнтам краще зрозуміти роботу програмного забезпечення та документацію.
- ***Процес управління версіями*** передає об'єкти на розробку, а ***Процес моніторингу і контролю*** забезпечує відсутність погіршення роботи операційного середовища. Спеціалісти зі супроводження завжди слідкують за поведінкою операційної системи та середовища, в якому вона працює, для виявлення ознак деградації. Вони повинні швидко попередити інші групи підтримки (оператори, технічна підтримка, мережева підтримка, тощо), коли станеться щось незвичайне та оцінити, чи відбулось погіршення роботи, яке потребує розгляду.
- Останнім основним процесом є процедури **відновлення** для покращення супроводжуваності, процедури **переходу** для переміщення системи в інше середовище та процедури **утилізації**, коли система виводиться з експлуатації.

II. Мета та обґрунтування моделі SMMM

Основні джерела, покладені в основу розробки моделі SMMM

- ISO/IEC14764;
- IEEE1219;
- ISO/IEC12207;
- CMMi';
- SWEBOK;
- ISO/IEC 15504.

Мета моделі SMMM (1)

- Аудит можливостей постачальника або аутсорсера з надання послуг зі супроводження програмного забезпечення; або
- Покращення внутрішньої організації процесу супроводження програмного забезпечення.
- Модель була розроблена з точки зору споживача, як досвідченого учасника конкурентного, комерційного середовища. Кінцева мета покращення роботи програм в результаті застосування моделі **SMMM** - підвищення задоволеності користувачів, що є пріоритетніше за жорстку відповідність стандартам.

Мета моделі SMMM (2)

Більший рівень зрілості в контексті моделі **SMMM** означає **для клієнтських організацій**:

Досягнення цільових рівнів обслуговування та задоволення клієнтських пріоритетів;

Реалізація найкращих наявних способів супроводження програмного забезпечення;

Отримання прозорих послуг зі супроводження за конкурентну вартість;

Проведення найшвидшого обслуговування зі супроводження програмного забезпечення.

Мета моделі SMMM (3)

Для **організації, що проводить супроводження**, досягнення більшої зрілості може призвести до:

- Зниження витрат на супроводження та підтримку;
- Скорочення фаз життєвого циклу та інтервалів;
- Збільшення здатності досягнення рівнів обслуговування;
- Збільшення здатності збирати кількісні дані про властивості ПЗ на всіх стадіях процесу супроводження.

III. Архітектура моделі SMMM

Рівні зрілості супроводження моделі SMMM

Level	Level Name	Risk	Interpretation
0	Незавершений	Highest	No sense of process
1	Представлений	Very High	ad hoc maintenance process
2	Керований	High	basic request-based process
3	Встановлений	Medium	state-of-the-art maintenance process
4	Прогнозований	Low	generally difficult to achieve now
5	Оптимізуючий	Very Low	technologically challenging to attain

Level 0 – Incomplete Process

Level– Leve Name	Capability Level Definition	Process Generic Attributes
0- Incomplet Process	The process is not being executed by the organization, or there is no evidence that the process exists. Level 0 implies that the activity is not being performed by the organization	a) There is no evidence that the process exists; b) Upper management is not aware of the impact of not having this activity or process in the organization; c) The activity or process does not meet the goals stated by the model; d) There is no knowledge or understanding of the activity or process; e) Discussions concerning the activity or process take place, but no evidence can be found that the activity or process exists; f) Historical records show that the activity has been performed, but it is not being done at this time.

Level 1 – Performed Process

1- Performe Process	Improvised: Recognition that the practice is executed informally. Level 1 implies that something is being done or that the activity is close to the intention of the practice presented in the model. The execution of the practice depends on the knowledge and presence of key individuals. The practice is typically ad hoc and not documented. It is local and would not necessarily appear in another software maintenance group. There is no evidence that the attributes of the processes are systematically executed or that the activities are repeatable.	a) The organization is aware of the need to conduct this activity or process; b) An individual conducts the activity or process and the procedures are not documented (note: typically, staff must wait until this individual arrives on-site to learn more about the process; when this individual is not on-site, the activity or process cannot be executed fully); c) A few of the software maintainers execute this activity or process; d) We cannot recognize precisely the inputs and outputs of the activity or process; e) There is no measure of the activity or process; f) The deliverables (outputs) are not used, not easily usable, and not kept up to date, and their impact is minimal; g) Who performs the activity or the qualifications/training required cannot be identified.

Level 2 – Managed Process

2- Managed Process	Awareness of the practice, which is deployed or a similar practice is performed. Level 2 implies that the practices suggested by the model are deployed through some of the software maintenance groups. What characterizes this level is the local and intuitive aspects of the activities or processes, which makes it difficult to harmonize them across all the software maintenance organizations.	a) The process is documented and followed locally; b) Training or support is provided locally; c) The goals of the process and activities are known; d) Inputs to the process are defined; e) Deliverables supporting the goals of the activity or process are produced; f) Qualitative measures of some attributes are performed; g) Individuals' names and qualifications are often described.

Level 3 – Established Process

3- Established Process	The practice or process is understood and executed according to an organizationally deployed and documented procedure. Level 3 implies that the practice or process is defined and communicated, and that the employees have received proper training. We expect that the qualitative characteristics of the practice or process are predictable.	a) The practice or process suggested by the model is executed; b) The same practice is used across software maintenance groups; c) Basic measures have been defined and are collected, verified, and reported; d) Employees have the knowledge to execute the practice or process (i.e. implying that the roles and responsibilities of individuals are defined); e) The required resources have been assigned and managed to achieve the identified goals of the process; f) Techniques, templates, data repository, and infrastructures are available and used to support the process; g) The practice or process is always used by the employees; h) Key activities of the process are measured and controlled.

Level 4 – Predictable Process

4- Predictable Process	The practice is formally executed and quantitatively managed according to specified goals within established boundaries. There is an important distinction with respect to Level 4, in terms of the predictability of the results of a practice or process. The expression 'quantitatively managed' is used when a process or practice is controlled using a statistical control or similar technique well suited to controlling the execution of the process	a) Intermediate products of a process are formally reviewed; b) Conformance of the process has been assessed based on a documented procedure; c) Records of reviews and audits are kept and available; d) Open action items from reviews and audits are monitored until closure; e) Resources and infrastructures used by the process are planned, qualified, assigned, controlled, and managed; f) The process is independently reviewed or certified; g) Key activities of the process have historical data and an outcome that is measurable and controlled; h) Key activities have a numerical goal that is set and is attainable; i) Key activities have quantitative measures that are controlled in order to attain the goals; j) Deviations are analyzed to make decisions to adjust or correct the causes of the deviation.

Level 5 – Optimizing Process

5– Optimizing Process	The practice or process has quantified improvement goals and is continually improved. Level 5 implies continuous improvement. Quantitative improvement targets are established and reviewed to adapt to changes in the business objectives. These objectives are used as key criteria for improvements. Impacts of improvements are measured and assessed against the quantified improvement goals. Each key process of software maintenance has improvement targets.	a) Major improvements to process and practices can be reviewed; b) Innovations to technologies and processes are planned and have measurable targets; c) The organization is aware of and deploys the best practices of the industry; d) There are proactive activities for the identification activities of process weaknesses; e) A key objective of the organization is defect prevention; f) Advanced techniques and technologies are deployed and in use; g) Cost/benefit studies are carried out for all innovations and major improvements; h) Activities of reuse of human resources knowledge are performed; i) Causes of failure and defects (on overall activities/processes and technologies) are studied and eliminated.

IV. Області процесів та ключові процеси моделі

Process Domain	Key Process Area	Roadmap
Software Maintenance Process Management	Maintenance Process Focus	Responsibility and Communications
		Information gathering
		Findings
		Action plan
	Maintenance Process/Service Definition	Documentation and Standardization of processes/services
		Process/Service adaptation
		Communication processes /services
		Repository of processes/services
	Maintenance Training	Requirements, plans, and resources
		Personal training
		Initial training of newcomers
		Projects training on transition
		User training
	Maintenance Process Performance	Definition of maintenance measures
		Identification of baselines
		Quantitative management
		Prediction models
	Maintenance Innovation and Deployment	Research of innovations
		Analysis of improvement proposals
		Piloting selected improvement proposals
		Deployment of improvements

Software Maintenance Request (MR) Management		Benefit measurement of improvements
	Event and Service Request Management	Communications and contact structure
		Management of events and service requests
	Maintenance Planning	Maintenance Planning (1 to 3 yrs)
		Project transition planning
		Disaster Recovery planning
		Capacity planning
		Versions and upgrade planning
		Impact analysis
	Monitoring and Control of Service Requests and Events	Follow up on planned and approved activities
		Review and analyze progress
		Urgent changes and corrective measures
	SLAs and Supplier Agreements	Account Management of users
		Establish SLAs and contracts
		Execute services in SLAs and contracts
		Report, explain and bill services

Process Domain	Key Process Area	Roadmap
Software Evolution Engineering	Software Transition	Developer and owner involvement and communications
		Transition process surveillance and management
		Training and knowledge transfer surveillance
		Transition preparation
		Participation in system and acceptance tests
	Operational Support	Production software monitoring
		Support outside normal hours
		Business rules and functionality support
		Ad hoc requests/reports/services
	Software Evolution and Correction	Detailed design
		Construction (programming)
		Testing (unit, integration, regression)
		Documentation
	Software Verification and Validation	Reviews
		Acceptance tests
		Move to production

Support to Software Evolution Engineering	Software Configuration Management	Change Management Baseline configuration Reservation, follow-up, and control
	Process and Product Quality Assurance	Objective evaluation Identify and document non-conformances Communicate non-conformances Follow up on corrections/adjustments
	Measurement and Analysis of Maintenance	Define measurement program Collect and analyze measurement data Repository of maintenance measures Communicate measurement analysis
	Causal Analysis and Problem Resolution	Investigate defects and defaults Identify causes Analyze causes Propose solutions
	Software Rejuvenation, Migration, and Retirement	Redocumentation of software Restructuring of software Reverse engineering of software Reengineering of software Software migration Software retirement

V. Література

- Guide to the Software Engineering Body of Knowledge (SWEBOK). – California: IEEE Computer Society, 2001. – 219 p.

1. Alain April, Jane Hayes, Alain Abran and Reiner Dumke. Software Maintenance Maturity Model. The software maintenance process model.: Wiley, 2004.