

ЖИЗНЕН ЦИКЪЛ НА ИС

Същност, стандарти и особености на ЖЦ

Концепцията „жизнения цикъл“ (ЖЦ) на информационните системи датира от началото на 70-те години на миналия век и е свързана с необходимостта от регламентиране на дейностите по създаване и използване на ИС. По своята същност жизненият цикъл на ИС представлява съвкупността от дейности, определящи живота на системата и свързани с разработването и експлоатацията ѝ. Той обхваща времето от момента на вземане на решение за нейната разработка до преустановяване на функционирането ѝ.

Основният нормативен акт, регламентиращ ЖЦ, е международният стандарт ISO/IEC 12207. Той определя структурата на ЖЦ, включените в него процеси, задачи и стъпки.

Съгласно ISO/IEC 12207 ЖЦ на информационните системи включва 3 групи процеси:

1. Основни – проучване, разработка (покупка), внедряване, експлоатация и съпровождане.
2. Спомагателни – документиране, управление на конфигурацията, проверка и оценка (управление на качеството).
3. Организационни – управление на проекта, създаване на инфраструктурата на проекта, оценка и подобряване на ЖЦ, обучение.

Основните процеси са задължителни. Това са базовите процеси, без които е невъзможно създаването и функционирането на ИС. Спомагателните процеси съпътстват основните и осигуряват качествено им реализиране. Организационните осигуряват условията за изпълнение на основните и за качествено функциониране на системата.

По-късно е разработен, а през 2002 г. е публикуван стандартът ISO/IEC 15288 System life cycle processes (SLCP). Стандартът има широка приложимост, но основното му предназначение е да поддържа процеса на създаване и

експлоатация на компютъризирани системи. За разлика от ISO/IES 12207 в SLCР са предвидени следните видове процеси и дейности:

- Договорни процеси – придобиване, доставка;
- Процеси на предприятието – управление на инвестициите, управление на ресурсите, управление на качеството, управление на ЖЦ на системите и др.
- Проектни процеси – планиране, оценка, контрол на проекта, управление на конфигурацията, управление на риска, проектиране и др.
- Технически – определяне и анализ на изискванията, разработка на архитектурата, внедряване, интеграция, верификация, експлоатация, съпровождане и др.
- Специални – определяне и реализация на връзки и зависимости, изхождайки от спецификата на задачите и целите.

В стандарта ISO/IEC 15288 се предвиждат 6 стадия на създаване на информационната система.

Таблица 2.1
Етапи на създаването на ИС според ISO/IEC 15288¹

| Етапи | Описание |
|-----------------------------|---|
| 1. Формиране на концепцията | Анализ на изискванията, избор на концепция и проектни решения |
| 2. Разработка | Проектиране |
| 3. Реализация | Програмиране |
| 4. Експлоатация | Въвеждане в експлоатация и използване на системата |
| 5. Поддържане | Осигуряване функционирането на системата |
| 6. Снемане от експлоатация | Прекратяване на използването, демонтаж, архивиране |

В таблица 2.1 е приведен списък на етапите и основните резултати от тях.

Както се вижда в ISO/IEC 15288² има промени и доуточняване на основните процеси. Специално са отделени етапите „Поддържане“ и „Снемане от

¹ http://www.intuit.ru/studies/professional_skill_improvements/1901/courses

² http://www.intuit.ru/studies/professional_skill_improvements/1901/courses (12.07.2013 г.)

експлоатация“, което показва тяхното значение. Поддържането е процес, при който се осигурява качествено и дълготрайно функциониране на системата. В етап 6 пък се обръща внимание на архивирането като операция, осигуряваща запазване на резултатите от работата на системата за последващо използване.

Предписанията на стандартите се използват в конкретните методики за проектиране на ИС най-често с допълнения и доуточняване. Проектирането на ИС е сложен, трудоемък и продължителен процес. Съвременните технологии на проектиране предполагат поетапна разработка с непрекъснато детайлизиране на първоначално дефинираната обща идея на системата. Изхождайки от това, в някои методическите разработки се предлага известно модифициране на основните процеси: проучване, концептуално проектиране, логическо проектиране, реализация (програмиране), внедряване, съпровождане. С това същността и логиката на процеса на проектиране не се променя, единствено се доуточняват и детайлизират някои крупни процеси. Всеки процес от своя страна се разбива на подпроцеси и отделни дейности (видове работа).

Модели на ЖЦ

Моделът на ЖЦ определя съдържанието и последователността на изпълнение на процесите, задачите и стъпките по разработката на системата, техните връзки и условията, при които се реализират. Моделът на ЖЦ зависи от спецификата на разработка на ИС и от условията, при които тя се създава и функционира – обем на работата, срокове, комуникация с възложителите и др.

Във връзка с използването на понятието „модел на ЖЦ“ следва да се уточнят други два термина – етап на проектирането и дейност.

Етапът на проектиране е фаза (phase), която има начало, край и определен резултат. Етапите следват един след друг, характеризират се с оформяне на определена документация (отчетност) и доста често след приключване на междинна фаза – със съответно финансиране (в зависимост от условията на договора). Етапите могат да бъде разбити на подетапи за по-доброто управление на проекта и разпределение на отговорностите.

Дейността (activity) е определен тип работа, изпълнявана в процеса на разработка. Различните дейности изискват различна квалификация и се

изпълняват от различни специалисти – например управлението на проекта се изпълнява от мениджъри, тестването – от т.н. тестери. В рамките на един етап могат да се изпълняват няколко вида дейности – напр. разработка на общата концепция, контрол на качеството и управление. Освен това един вид дейност може да се изпълнява на различни етапи. Например на етап „проектиране“ може да се започне разработката на тестове, а тяхното изпълнение се осъществява на етап „програмиране“.

Към моделите на жизнения цикъл се предявяват следните основни изисквания:

- резултатност – да осигуряват условия за завършване на разработката в срок и с очакваното качество;
- гъвкавост - да осигуряват възможност за отразяване на променящите се на условия на функциониране на обекта (предприятието) в хода на разработката, така че създадената система да е адекватна на изискванията към момента на внедряването ѝ;
- да осигуряват дисциплина на проектирането.

Дългогодишният опит по изграждане на ИС води до създаването на различни модели на ЖЦ, които могат да бъдат причислени към две групи:

- последователни,
- итеративни.

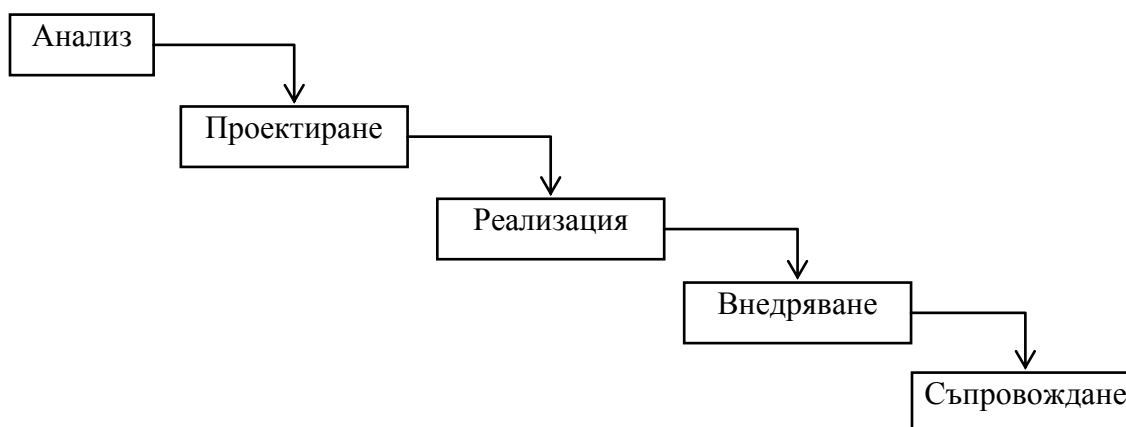
Последователните модели предполагат разделяне на цялостната дейност по изграждане на ИС на последователно реализиращи се завършени процеси, без застъпване или итеративно изпълнение. При итеративните разработката се осъществява с няколкократно повторение на последователността от процеси при постепенно доуточняване и доразвитие на идеите.

Като основни представители на тези групи модели ще разгледаме каскадния и спираловидния модел на ЖЦ.

1.Каскаден модел на ЖЦ

Каскадният модел (фиг. 2.1) е създаден в началото на 70-те години от У. Ройс. Характеризира с разделяне на всички работи по изграждането на системата на етапи, при това преходът от един етап към друг се осъществява само при

окончателно завършване на работите по предходния етап. В края на всеки етап се разработва необходимата документация. Наборът от документи е регламентиран и задължителен. Така оформените (документирани) резултатите от предходния етап се използват като спецификация (изисквания) за следващия. Възложителите и потребителите се запознават със системата при нейното внедряване и предаване в експлоатация. Моделът е подходящ за разработки с добре дефинирани изисквания, за типови системи с ясна визия за изискванията, за проекти, изискващи качествено управление.



Фиг. 1 Каскаден модел на ЖЦ

Положителните страни на каскадния модел са:

- Възможности за добро планиране и управление на проекта, тъй като дейностите са точно дефинирани, могат да се разпределят ресурсите за изпълнение на проекта и поетапно да се контролира процесът;
- Добро ниво на документираност – нито един етап не може да бъде завършен без да се оформи регламентираната за него документация;
- Добри възможности за аутсорсинг.

Слаби страни:

- Отделяне на разработчика от потребителя – потребителят (възложителят) участва при дефиниране на изискванията към системата на етапа проучване и при приемане на системата в експлоатация, т.е. в началната и завършващата фаза на проектирането. По този начин възложителят не може да влияе върху хода на проектирането, не може своевременно да внася корективи. Това води до увеличаване риска от неразбирателство между изпълнител и потребител (възложител), което може да предизвика провал на проекта;

- Събраните на етап „проучване“ изисквания са “замразени” във вид на „Техническо задание“ и последващи спецификации и не се променят в хода на разработката. На практика обаче, в изискванията трябва непрекъснато да се внасят промени не само заради динамиката на обекта, а и защото те предварително трудно се формулират, налага се да се доуточняват и доразвиват в хода на изпълнението на проекта;

- Окончателната оценка на системата се дава при нейното внедряване, а това означава късно откриване на грешки, необходимост от връщане към началото на работата практически в края на разработката и е съпроводено със сериозни допълнителни разходи, неспазване на сроковете и проблеми между изпълнител и възложител;

- Ако има допуснати грешки в началните етапи на проектирането, те многократно се мултиплицират при изпълнението на следващите етапи;

- Значителна продължителност на разработката, не се предполага застъпване, паралелно изпълнение на дейности и етапи, което би довело до съкращаване на времетраенето на процеса;

- Твърда схема на работа, която не осигурява гъвкавост при промяна на условията – например при проблеми с финансирането или преразпределение на бюджета;

Независимо от тези проблеми каскадният модел се предпочита и досега поради неговата простота и натрупан опит.

За да се осигури необходимата гъвкавост на разработката каскадният модел претърпява развитие. Създаден е вариант на модела, при който е внесена определена итеративност, като се допуска връщане към предходен етап и промяна, допълване и доуточняване на изискванията.

2. Спираловиден модел на ЖЦ

Предложен е от Боем през 1988 г. за преодоляване недостатъците на каскадния модел и за по-добро управление на риска. Спираловидният модел е алтернатива на каскадния, цели да се ускори разработката и текущо, в хода на проектирането, да се дефинират изискванията (фиг. 2.2). При него акцент се поставя на началните етапи на ЖЦ – анализ, концептуално и логическо

проектиране и на постоянния контакт с потребителите. Идеята е предлаганите решения за системата да бъдат обсъждани с потребителите през кратки периоди от време (няколко дни) и своевременно да бъдат внасяни изменения. Същевременно се разширяват и доуточняват изискванията. Използва се проектиране по метода „отгоре-надолу“, като в началото се разработват общосистемните решения и функционалността, а след това всяка функция се разработва на ниво задачи и модули. Системата се изгражда итеративно чрез прототипиране, като на всяка итерация от спиралата се създава фрагмент или версия на системата, обсъжда се с потребителя, конкретизират се детайлите, доуточняват се изискванията към следващата версия и ако е завършена - се интегрира с вече разработените елементи на системата. Така разработката се задълбочава, разширява и конкретизира.

За разлика от каскадния при спираловидния модел няма предварително определен набор от версии и регламентиране на тяхната последователност. Съдържанието на версиите също не е регламентирано, зависи от конкретните условия. В рамките на една версия се реализират различни действия с определена последователност – определяне на изискванията, анализ, реализация, интеграция, представяне на потребителя. По същество изпълнението на една версия е завършен етап от разработката. Например най-напред може да се определи целесъобразността на разработката, след това да се разработят системните решения, по-нататък да се представи декомпозицията и основните функции и т.н. Възможен е и друг начин на работа – всяка версия да представя отделен компонент на системата. След завършването на една версия се планира следващата.

Идеите на спираловидния модел лежат в основата на различни методологии за бърза разработка, напр. RAD – Rapid Application Development³ (технология за бърза разработка на приложения), група методологии AGIL (Agile Software Development)⁴, известни като гъвкави и др.

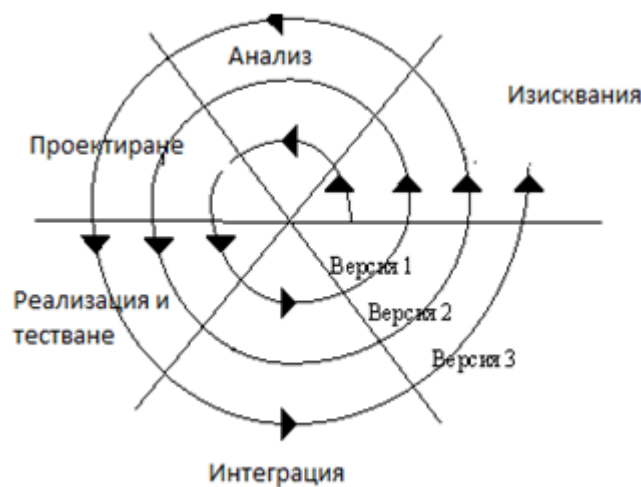
³ J. Martin. Rapid Application Development. New York. Macmillan, 1991

⁴ <http://www.planit.net.au/resource/industry-stats-project-methodologies-2012/> (15.07.2013 г.)

Спираловидният модел е подходящ за неголеми разработки с кратки срокове за изпълнение. Успешно се прилага при динамични системи и такива с неясни изисквания в началото на работата.

Положителни страни:

- Съкращаване на сроковете, бърза разработка;
- Непрекъсната връзка с крайния потребител, възможност за текущо внасяне на промени;
- Намаляване риска от провал на проекта. Позволява фокусиране върху най-рисковите елементи на проекта.



Фиг. 2 Спираловиден модел на жизнения цикъл

Слаби страни:

- Процесът не подлежи на добро планиране и управление – работа от предходна итерация може да се остави за следващата и да не се завърши, няма точни критерии кога да се премине към следваща итерация;
- В повечето случаи документацията се разработва при завършване на проекта, когато екипът вече има други ангажименти, поради което най-често е непълна или некачествена, а понякога изцяло отсъства.
- Възникват проблеми при отсъствието на добри средства за прототипиране.

Разгледаните модели на жизнения цикъл са базови. На тяхна основа са разработени множество разнообразни модели, които фокусират вниманието върху определени характеристики на процеса.

Изборът на модел на жизнения цикъл зависи от характера на разработваната система, обема на работа, характера на изискванията, нивото на риск при разработката и др. За реализацията на един проект е възможно съчетаването на модели на ЖЦ.

Изводи:

- изборът на модел на ЖЦ определя успеха на проекта;
- няма универсални модели – всичко зависи от характера и мащаба на проекта;
- моделите могат да се комбинират;
- правилното прилагане на моделите на ЖЦ изисква висока квалификация, добра организация и по-високо равнище на зрялост на разработчика;
- всички модели имат както преимущества, така и недостатъци, но винаги това е свързано с особеностите на конкретния проект.

Съдържание на основните процеси на жизнения цикъл

Проучване. Етапът може да се осъществява самостоятелно или пък да е интегриран със следващите етапи. Има за цел да осигури възможно най-пълна информация за обекта, неговите особености и изисквания към системата. Проучването включва:

- изследване на мисията, целите и стратегията на развитие на предприятието;
- проучване на нормативната уредба и правилата за дейността;
- изследване на организационната и функционална структура;
- анализ на бизнес-процесите и техните особености;
- анализ на данните;
- събиране и анализ на конкретни изследвания към процесите и данните.

По-подробно въпросите на проучването да разгледани в отделна тема.

Разработка или покупка. Изпълняват се всички дейности по изграждане на новата система. В случаите на разработка се извършва проектирането на системата като се преминава през два етапа: концептуално проектиране и детайлно проектиране.

Концептуалното проектиране има за задача да формира по-общите решения за системата, изхождайки от особеностите на предметната област и включва:

- определяне обхвата и ограниченията;
- разработка на общосистемни решения (хардуер, мрежа, избор на програмно осигуряване и средства за разработка, обща идеология на системата) и вътрешни стандарти;
- изграждане на инфраструктурата;
- изграждане на архитектурата на системата;
- разработка на функционалността;
- разработка на концептуалния модел на БД и др.

При детайлното проектиране се детайлизират всички решения, взети на предходния етап и се разработват елементите на системата по функции - постоянна информация и номенклатури, файлове, логически и физически модели на БД, вход и изход, модели и алгоритми на процесите, система на защита, потребителски интерфейс, интеграционни компоненти и др. Целта е да се разработят всички решения, позволяващи последващо програмиране. Една част от тестовете за системата се подготвят на този етап.

При покупка се изпълняват следните процеси:

- избор на готова система (приложение);
- закупуване
- подготовка и обучение на специалистите;
- подготовка за внедряване;
- разработка на решения за интеграция със съществуващи системи.

Реализация. Всички решения от предходния етап се довеждат до програмна реализация. Разработват се спецификациите на програмите и

програмния код. Системата се подлага на разнообразни тестове с цел откриване и отстраняване на грешки, непълноти, неточности и други проблеми при реализацията ѝ.

Внедряване. Това е толкова отговорен етап, колкото и предходните. Внедряването има за цел да осигури въвеждането на системата в експлоатация. Изпълнява се в обекта и заема ограничен период от време. При въвеждането на системата в експлоатация е съществена ролята на бизнес организацията. Налага се активно ангажиране на специалистите от предприятието в дейностите по внедряване.

Изпълнението на етапа се предхожда от подготовка за внедряване – закупуване на оборудването, изграждане на мрежата, обучение на специалистите. Важна и достатъчно сложна задача на подготовката за внедряване е реорганизацията на предприятието и на бизнес процесите.

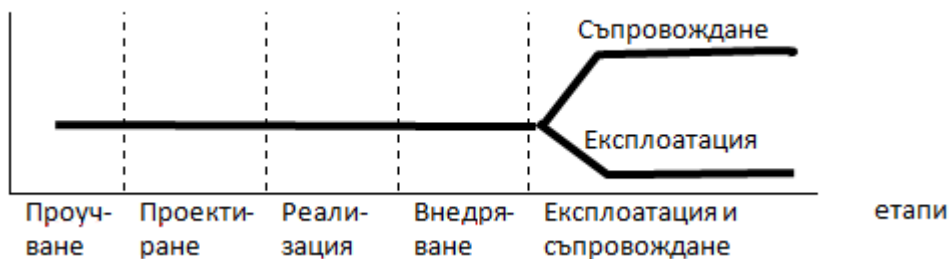
Внедряването включва следните дейности:

- инсталиране на системата;
- въвеждане на постоянната информация;
- паралелна работа със старата система;
- откриване и отстраняване на грешки и проблеми.

Внедряването е един от особено тежките етапи, тъй като се изпълнява на място, резултатите от новата система се сравняват със старата и обикновено възникват много непредвидени проблеми. Тяхното своевременно отстраняване е задача, изпълнявана на този етап.

Експлоатация и съпровождане. Това е дейност извън жизнения цикъл на проекта, но в рамките на ЖЦ на системата. Вложените инвестиции и усилия за разработка и внедряване на БИС изискват изградената система да бъде използвана в продължение на достатъчно дълъг период от време. Но в резултат на множество фактори: непълни изисквания; настъпили изменения на средата (на самото предприятие и на външната среда); непълноти и неточности при реализацията на системата и други; даже при внедряване на системата, нейните характеристики могат да се различават от потребителските изисквания. Тези различия се задълбочават в хода на експлоатацията. Има случаи, когато при

рядко изпълнявани процеси се откриват сериозни проблеми едва след достатъчно продължителна работа на системата. Затова в хода на нейната експлоатация паралелно се извършва „съпровождането“ ѝ (фиг. 2.3).



Фиг. 2.3 Място на съпровождането

Съпровождането е процес на постоянно наблюдение върху работата на системата с цел разкриване на проблеми и своевременното им отстраняване. Заедно с това динамиката на бизнес процесите, променящите се условия на функциониране на обекта и измененията във външната среда налагат периодична актуализация на системата. Например, промените в законодателната уредба често налагат промяна на правила и алгоритми на работа. Без съпровождане не би било възможно поддържането на БИС в адекватно състояние. Целта е да се осигури използването на съществуващата система без да се налага разработването или покупката на нова.

По такъв начин в хода на експлоатацията на системата се осъществява:

- корекция;
- усъвършенстване и обновяване на системата.

В резултат не са редки случаите, когато за съпровождане се налага да се заделят не малко средства (до 60-70% от общата себестойност на системата по данни на НР).

Съпровождането може да се осъществява по следните начини:

- от екип на разработчика при сключване на съответен договор;
- от разработчика в рамките на гаранционното обслужване;
- от ИТ отдела на предприятието;
- от друга организация.