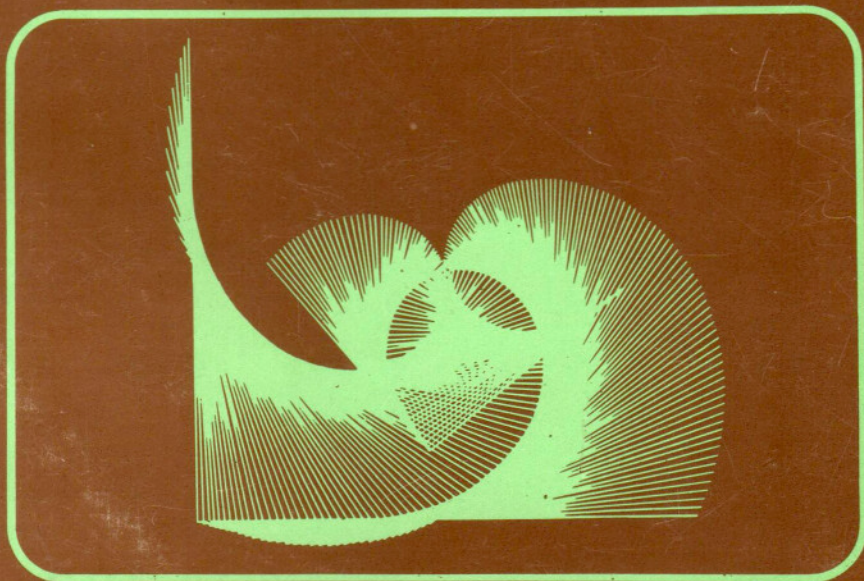


АВТОМАТИЗИРАНИ
СИСТЕМИ
ЗА ПРАВЛЕНИЕ



1984



СЪДЪРЖАНИЕ

ст. н. с. к. т. н. инж. Станислав Димитров, ст. н. с. к. т. н. инж. Бурия Димитрова, инж. Йордан Миндов, инж. Йордан Га- нов – Програмен пакет СМ – ЕЛИМА	35
ст. н. с. к. т. н. инж. Станислав Димитров, инж. Йордан Ганов, инж. Николай Атанасов, инж. Мариана Торо- манова – Микропроцесорен мрежов адаптер за реализация на процедурата LAPB, II ниво на препоръката X.25 на МККТТ	51
ст. н. с. к. т. н. инж. Станислав Димитров, ст. н. с. к. т. н. инж. Бурия Димитрова, к. т. н. инж. Цветана Гоче- ва, инж. Мария Момчева, инж. Йордан Миндов – Трето (пакетно) ниво от мре- жовия метод за достъп по препоръката X.25 на МККТТ	60
ст. н. с. к. т. н. инж. Станислав Димитров, ст. н. с. к. т. н. инж. Бурия Димитрова, инж. Мария Момчева, инж. Димитринка Стайкова – Транспортен протокол	71
ст. н. с. к. т. н. инж. Станислав Димитров, ст. н. с. к. т. н. инж. Бурия Димитрова, инж. Чавдар Стойнов, к. т. н. инж. Цветана Гочева, Светла Кова- чева – Сесионен протокол	89
к. т. н. инж. Константин Гълъбов, ст. н. с. д-р т. н. инж. Живко Железов – Алгоритъм за определяне на обобщен критерий за ефективност на много- машинни мрежи	116
ИЗ ЧУЖДЕСТРАННИЯ ОПИТ	
ст. н. с. к. т. н. инж. Бурия Димитро- ва – Япония гледа към 90-те години	122
ОБЗОРИ, СЪОБЩЕНИЯ	
инж. Михаил Кринков – Основни нап- равления в развитието на приложното програмно осигуряване за мини- и микрокомпютри.	125
Авторите в този брой	132
МЕТОДИЧЕСКИ ПРОБЛЕМИ И НАУЧНИ ИЗСЛЕДВАНИЯ	
ст. н. с. к. м. н. Аврам Ескенази – Автоматизиране на една фаза от соф- туерното производство	5
инж. Петър Крайчев, Петър Анастасов, инж. Петър Сребров – Администратор на база данни и унищожаването на данните	10
к. т. н. инж. Георги Павлов, Драгомир Паргов – Алгоритъм за нелинейна оптимизация с използване на стати- стически метод за търсене	16
инж. Тодорка Московска – Моделиране на технология на аналитичните разра- ботки с използване на ЕИМ	22
МАТЕМАТИЧЕСКО И ПРОГРАМНО ОСИГУРЯВАНЕ	
Програмно осигуряване на протоко- лите за информационен обмен в много- машинната мрежа на ЕССИ	34

МЕТОДИЧЕСКИ ПРОБЛЕМИ И НАУЧНИ ИЗСЛЕДВАНИЯ

Аврам ЕСКЕНАЗИ

АВТОМАТИЗИРАНЕ НА ЕДНА ФАЗА ОТ СОФТУЕРНОТО ПРОИЗВОДСТВО

Създаването на програмни продукти е дейност с кратка история. Осъзнаването на факта, че това е производствена дейност със своя организация, методология и специфика, стана не твърде отдавна, а опитите за систематични изследвания са още по-отскоро. Очевидно все още не е намерен единен подход, но общо взето авторите гледат анализите си, разбивайки производствения процес или на хронологичен принцип (като говорят обикновено за „фази“), или на функционален (отделяйки „функции“) /1/, /2/. За нашите цели ще се придържаме близо до изложените в /3/ виждания, които считаме за ясни и удобно приложими. Според тях всеки програмен продукт преминава през 6 фази на развитие:

- изследване;
- анализ за осъществимост;
- конструиране (проектиране);
- програмиране;
- оценка;
- използване.

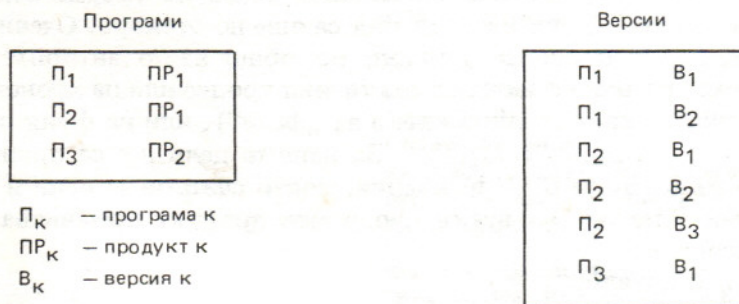
През тези фази с различна интензивност се изпълняват няколко функции — планиране, разработване, съпровождане, поддържане и др. Използването започва от момента на разпространяването и обикновено продължава от 2 до 6 години. Това е времето, през което продуктът се намира в експлоатация и се използва ефективно. През тази фаза се извършват най-интензивно функцията поддържане (обучение на персонала, внедряване, настройка) и функцията съпровождане (отстраняване на грешки, разширяване, усъвършенстване).

Като решаващ фактор за повишаване ефективността на софтуерното производство трябва да се разглежда неговото автоматизиране. Фазите на изследване и на анализ за осъществимост съдържат твърде много неподдаващи се на формализация елементи и поради това трудно могат да се автоматизират. Немалко са разработените средства за автоматизация на фазата проектиране, а — както добре се знае — най-мощните средства за автоматизация се прилагат по време на фазата програмиране.

Нашето внимание е съсредоточено върху фазата използване, където засега има твърде малко резултати. Тази фаза по някои данни

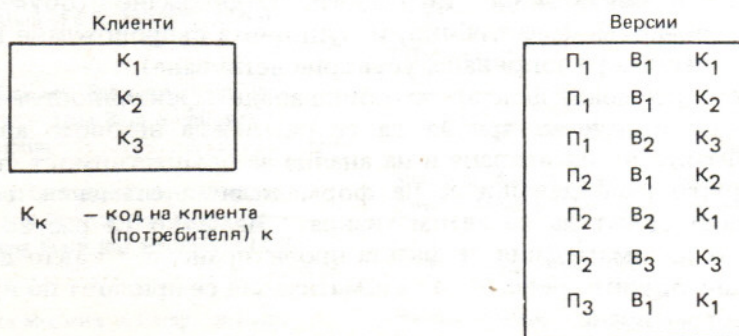
поглъща до 70 % от разходите по продукта. Ще опишем един модел, довел и до практическа реализация.

Използването (от гледна точка на фирмата-производител) все повече се усложнява. Обикновено дадена фирма се занимава с няколко продукта, които са във фаза на използване. Тези продукти могат да са нейно производство или тя само да ги разпространява, поддържа и съпровожда. Всеки от продуктите, обикновено независимо от другите, търпи някакво развитие. По този начин за много продукти във фаза на използване, т. е. на редовна експлоатация, има по няколко версии и/или модификации. (За простота нататък ще казваме само версии). В почти всички случаи един продукт се състои от няколко програми. Обикновено при съпровождане се изменят част от тези програми (фиг. 1).



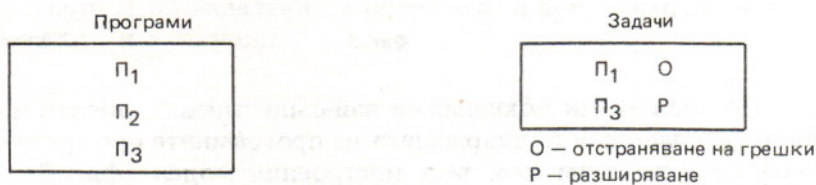
Фиг. 1

От друга страна, фирмата работи с известен брой клиенти (потребители). Един клиент може да не притежава всички продукти на фирмата. Ако притежава даден продукт, то потребителят използва обикновено само една версия или поне може да се счита, че фирмата отговаря за съпровождането и поддържането само на една версия (фиг. 2).



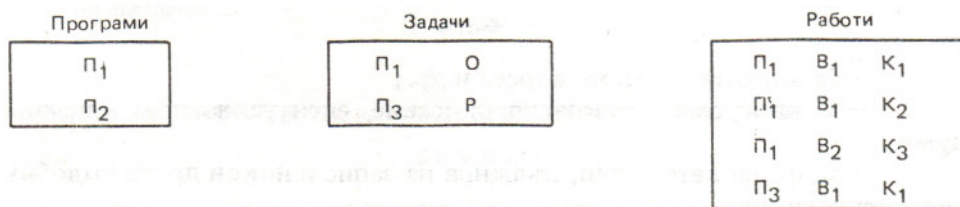
Фиг. 2

Самото съпровождане се състои в отстраняване на грешки, разширяване възможностите на продукта (което означава изменение на част от неговите програми или добавяне на нови програми), усъвършенстване (което обикновено означава изменение на програми с цел подобряване на някои от характеристиките им). Следователно всеки такъв акт може да бъде отделен и наречен задача. Задачата е свързана само с една програма (фиг. 3).



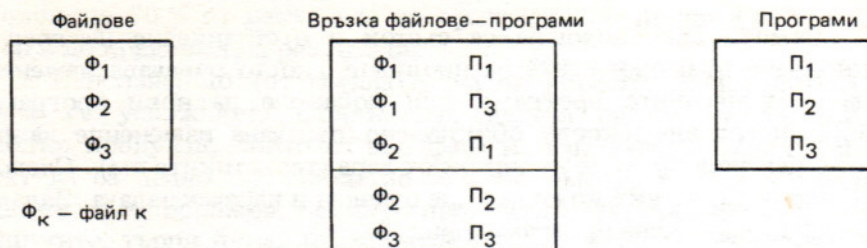
Фиг. 3

Поради наличието обаче на различни версии на продукта (а това значи и на съставлящите го програми) подходящо е на всяка задача да се съпоставят толкова работи, колкото версии в момента са в експлоатация. Очевидно в много случаи тези работи ще имат много общо, а могат и почти да съвпадат. Ако се вгледаме малко по-внимателно обаче, ще забележим, че всъщност крайната цел в случая е клиентът да получи новата програма и при това положение е по-подходящо под „работа“ да се разбира изпълнение на задачата за даден клиент, тъй като дори няколко клиенти да притежават една и съща версия, винаги има дейности (копиране на дистрибутивния носител, генериране върху инсталацията на клиента и др.), които са специфични за отделния клиент (фиг. 4).



Фиг. 4

В повечето продукти като самостоятелни обекти биха могли да се разглеждат и файловете (набори от данни). Обикновено един файл се използва в няколко програми. Изменението на дадена програма може да означава между другото и изменяне характеристиките на файл, участващ в тази програма. Следователно ще бъде необходимо този файл да се измени и във всички други програми, в които участва. (Тук очевидно става въпрос за промяна не само на декларацията на файла, а и за всички съпътстващи я изменения в програмата). Тази взаимовръзка е отразена на фиг. 5.



Фиг. 5

По този начин обхванахме най-съществените обекти и процеси от съпровождането и поддържането на програмните продукти в дадена фирма. Ако добавим към така построения модел (фиг. 6) и някои допълнителни данни:



Фиг. 6

- за клиента — името, адреса и др.;
- за програмите — кратко описание, евентуално чрез ключови думи;
- за файловете — тип, дължина на запис и някои други подобни характеристики;
- за задачите — срок на изпълнение, действителна дата на завършване, кодове на отговарящите изпълнители, необходими ресурси за изпълнението (човекочасове, машинно време и др.), състояние и др. подобни;
- за версиите — кратко описание и код на клиента;
- за работите — дата на действително завършване, достигане до информационна система, чрез която може значително да се подобрят и улеснят съпровождането и поддържането през фазата на използване.

На базата на така построения модел и с използване на СУБД от релационен тип INFO е реализирана и внедрена интерактивна инфор-

мационна система за нуждите на софтуерната фирма ORBIS (ФРГ). Системата функционира върху миникомпютър PRIME—550. Потребителят получава всички услуги по поддържане на информационната база и по търсене на информация в диалогов режим посредством меню. Въвеждането и изменението на данните става чрез екранни форми, към които са вградени твърде мощни средства за контрол. При няколкостотин програми и значително повече от тях версии, както и сравним брой файлове времето за отговор на произволно запитване в съответствие с менюто и по зададени от потребителя аргументи за търсене се движи между 1 и 5 секунди.

ЛИТЕРАТУРА

1. Hamilton, M., S. Zeldin. Functional Life Cycle Automation USE. IT. In The Journal of Systems and Software, 3, 1983, pp. 25–62.
2. Sneed, H. Softing – Software Engineering System. In The Journal of Systems and Software, 3, 1983, pp. 63–76.
3. Гантер, Р. Методы управления проектирования программного обеспечения. Москва, „Мир“, 1981.

АВТОМАТИЗАЦИЯ ОДНОЙ ИЗ ФАЗ СОФТУЭРНОГО ПРОИЗВОДСТВА

РЕЗЮМЕ

Предлагается модель двух самых существенных функций – сопровождения и технического обслуживания – фазы использования программного продукта. На базе модели создана и внедрена информационная система, которая облегчает и совершенствует эту фазу софтуерного производства.

COMPUTERIZATION OF A SOFTWARE PRODUCTION PHASE

SUMMARY

A model of the two most important functions (those of maintenance and support) of the Use Phase of the program product is offered. Based on the model, an information system is built and implemented which facilitates and improves this phase of software production.