

CASE ESZKÖZ OKTATÁSÁNAK TAPASZTALATAI A BME VIK MŰSZAKI INFORMATIKAI SZAKÁN

dr. László Zoltán, laszlo@fsz.bme.hu
BME Folyamatszabályozási Tanszék

Abstract

Due to the increasing impact of information technology on the production and service, the software engineering becomes one of the most important field of informatics. The software engineering deals with the methods, tools and techniques which are necessary to develop and maintain software systems. Because today the large software development project cannot be managed without using Computer Aided Software Engineering (CASE) tools, the importance of such tools is constantly increasing. Last decade at the Faculty of Electrical Engineering and Informatics at TU Budapest the staff has got extensive experience of practical application and teaching software technology. The last some years the technical background and the financial conditions make it possible to introduce some CASE courses in the regular education process. The purpose of this paper is to summarize the experience and practice of the CASE education.

Az információs technológiáknak a termelésre és a szolgáltatásra gyakorolt növekvő befolyása révén a szoftver mérnökség az egyik legfontosabb informatikai szakterületté vált. A tématerület a nagy szoftver rendszerek tervezéséhez és elkészítéséhez szükséges mérnöki módszerek, technikák és eszközök alkalmazásával foglalkozik. Mivel manapság a nagy szoftver rendszerek fejlesztése nem uralható számítógéppel támogatott szoftver technológiák (CASE rendszerek) nélkül, az ilyen eszközök jelentősége folyamatosan növekszik. A BME VIK Műszaki Informatikai szakán évtizedes múltja van a Szoftver technológiák alkalmazásának és oktatásának. Az elmúlt években megteremtődtek az anyagi feltételei a CASE eszközök oktatásba állításának. Az előadás célja annak a többéves előkészítésnek és oktatási tapasztalatnak a bemutatása, amely ezen a területen felhalmozódott.

Az első részben bemutatjuk, hogy a szoftver technológiák oktatása hol helyezkedik el az informatikus képzésben, majd azt követően megadjuk a CASE oktatására szolgáló labor tárgy tematikáját. Végezetül összefoglaljuk tapasztalatainkat és vázoljuk a továbbfejlesztés lehetőségeit.

1. A szoftver technológiák helye a képzésben

A VIK műszaki informatika szakán a képzés két fő részre bontható. Az első három éven a minden hallgató számára közös alapképzés megadja azt a háttérrel, amelyre az utolsó két év szakosodása épül. Az alapképzés tárgyai négy fő vonulatot vagy tárgycsoportot alkotnak.

Az első csoportot a közismereti, vagy humán tárgyak jelentik. Ennek keretében gazdaságtan, jog, szociológia stb. képzés folyik. A második vonulatba a matematika és az olyan, hozzá szorosan kapcsolódó elméleti tárgyak tartoznak, mint a "Formális nyelvek", az "Információelmélet" vagy a "Kódelmélet". A harmadik csoport a hagyományos villamosmérnöki ismeretekre koncentrál. Ebben szerepel többek között a "Digitális technika", a "Számítógép architektúrák" és a "Számítógépes hálózatok" témaköre is.

A negyedik vonulat a programozáshoz, általánosabban a szoftverhez kapcsolódik. A tárgycsoport felépítése az alábbi ábrán látható és belőle kiolvasható a tárgyak heti óraszámja szemeszterenként.

<i>tárgy szemeszter</i>	1	2	3	4	5	6
Programozás alapjai	2	2				
Programozás technológiája			4			
Matematikai logika				4		
Programozási paradigmák					4	
Operációs rendszerek					4	
Adatbázisok						4
Számítógép labor	2	2	2	2	2	2

A "Programozás technológiája" egy többé-kevésbé hagyományos "Software Engineering" kurzusnak felel meg, amely végigköveti a szoftver, mint termék életútját, ezen túlmenően részletesen foglalkozik a Jackson System Development [1] és az OMT [2] objektum orientált fejlesztési módszertanokkal. A tárgy keretében a hallgatóság egy rövid áttekintést kap a CASE eszközökről C. Gane munkája [3] alapján. A tárgyhoz kapcsolódó labor egy szemeszterrel később alkalmat ad a hallgatónak 3-4 fős csoportokban egy teljes fejlesztési projekt végigvitelére. Az utóbbi három alkalommal az OMT módszertan alkalmazásával kellett a feladatot megoldani. Ebben a laborban a hallgatók még nem, vagy csak korlátozottan használnak CASE eszközöket. Ennek oka egyrészt az, hogy itt a hallgatói létszám még 150 körül van és a rendelkezésünkre álló CASE licenszek ennek a létszámnak a töredékére elegendőek, másfelől nem akarjuk elvonni a hallgatók figyelmét magáról a feladatról a viszonylag bonyolult eszköz használatával. Összefoglalva elmondhatjuk, hogy a 4. félév végére a hallgatók megismerve és alkalmazva egy módszertant, felismerik a számítógépes támogatás jelentőségét, lehetőségét és céljait.

A 4. és 5. éven a képzés modul-rendszerben folyik. Kétféle modulból kell egyet-egyet választani a hallgatónak. Az ún. alfa modulok (legutóbb 12-t hirdettek meg) célja, hogy a hallgatók egy alkalmazási területről szerezzenek ismereteket. A modulok csaknem felét a BME többi kara adja, így például van gépész és közlekedés szakirány is. A béta modulok (8-at ajánlottak) célja, hogy a hallgatók az informatika valamely specifikus - nem alkalmazáshoz köthető - területét ismerjék meg. Ezek között szerepel a "Szoftver specifikáció és tervezés" modul, amelyben célul tűztük ki

- formális módszerek és eszközök alkalmazását
- a technikai aspektusokon kívül a menedzsment vonatkozások intenzívebb oktatását.

Természetesen nem csak az informatikus képzésben van helye a CASE eszközök oktatásának. A villamosmérnöki szakon folyó - a villamosmérnököket szoftveres ismeretekkel felruházó - "Szoftver technológia" modulon belül is indítottunk CASE kurzust, valamint a BKE-vel közösen informatikai rendszereket tervező közgazdász hallgatónak is tartunk ilyen tanfolyamot. Ismereteink szerint más modulokon belül is folynak kísérletek CASE eszközök oktatásával.

A már említett "Szoftver specifikáció és tervezés" modul" kialakítása TEMPUS együttműködés keretében zajlik. Együttműködő partnereink a Karlsruhei, Eindhoveni és Brightoni egyetemek.

Az első CASE eszközünk - a Teamwork PCSA rendszerének egy demo változata - a 80-as évek végéről származik. Ez az eszköz a hagyományos SA módszertannak egy PC-s implementációja, rengeteg korlátozással. Mindössze egyetlen alkalommal szerepelt az oktatásban demonstrációs céllal, mert időközben KFKI-s együttműködő partnereink segítségével beállítottuk az EXCELERATOR-t. Ez, az SSADM támogatását célzó eszköz kísérleti jelleggel két éven keresztül szerepelt a képzésben. Időközben pályázati pénzből megvettük a Teamwork egy OS/2-s változatát [4]-[7], amelyen diplomatervezés és tárgyfejlesztés folyt. Az eszköz gyártója, a CADRE érdeklődést mutatott az oktatásban való együttműködés kialakítására. Képviselője látogatást tett az egyetemen, aminek egyik eredménye egy nagyon előnyös licenz-szerződés megkötése volt. Időközben hasonló megállapodást kötöttünk az StP-re vonatkozóan is.

2. A CASE labor tematikája

A heti 2 órában tartott tárgy célkitűzése: elmélyíteni a strukturált módszertanban szerzett ismereteket, CASE eszközt és szabványos módszereket használni. Ezen belül célunk, hogy a hallgatók felismerjék a módszertan alkalmazásával járó előnyöket és hátrányokat. A félévben a hallgatók egy gyakorlati feladatot oldanak meg, amelynek során elvégzik a strukturált analízist, valamint logikai terveket készítenek, amelyet fizikai tervekkel alakítanak.

A tematika értékeléséhez szükséges egy rövid áttekintést adni az oktandó Teamwork CASE rendszerről. A rendelkezésünkre álló verzió száma 6.0.3, és a program egy DEC Alpha 2000-es szerveren fut, amelyhez X-terminálokra keresztül csatlakoznak a felhasználók. A licenszünk 15 felhasználó egyidejű belépését teszi lehetővé. Az alábbi ábrán a szoftver életciklus-modellen sötétítéssel ábrázoltuk azokat a részeket, amelyekre a licenszünk érvényes.

Analysis				Design		Coding
SA	RT	IM	SIM	SD	Ada	ASB
Project Database						CSB
IPSE						
DPI	Access		Toolkit		UM	

Az SA jelű analízis programmal a Yourdon/DeMarco jelöléstechnikának és módszertannak [8] megfelelő SA elemzés végezhető. Annak ellenére, hogy a Teamwork eredendően a strukturált módszertan támogatását célozta, az eszközt kibővítették egy real-time (RT) elemzővel, amelynek célja a vezérlési információk rögzítése és analízise volt. Az alkalmazott módszertan a Hatley/Pirbhai metódus [9]. Az információ modellezésre (IM) a Chen-féle entitás-relációs modellezést használja a rendszer. Nem áll rendelkezésünkre a SIM-mel jelölt csomag, amely az RT modellnek egy szimulált végrehajtását támogatja. A tervező eszközök közül csak a Yourdon/Constantine módszert relizáló SD blokkot kaptuk meg, az Ada csomagot nem. A kódolást támogató ASB (Ada Source Builder) és CSB (C Source Builder) egységek egyikével sem rendelkezünk. Ez a magyarázata, hogy a hallgatói feladatokat csak tervezési szintig vittük el.

Az Integrated Project Support Environment (IPSE) részei, a dokumentáció-generátor, a Document Production Interface (DPI), az adatbázis külső elérését biztosító Access adatbáziskezelő csomag, a Teamworkhoz tartozó segédprogramok (Toolkit) és a felhasználó által konfigurálható menü-rendszer, a User's Menu (UM).

Az alábbiakban megadjuk a kurzus anyagát órák szerinti bontásban.

1. Bevezetés

A Teamwork felépítése, komponensei, a könyvtár struktúrája, a rendszer használatának általános kérdései. A program indítása, megállítása. Help, on-line dokumentáció használata, felépítése. Mintafeladatok áttekintése, egy példa bemutatása.

2. Modell készítése

A Teamwork modell fogalma, annak vetületei. Az adatbázis felépítése, a modellek kapcsolata és elérésük. Modell mentése és visszaállítása. Modell készítése, törlése, átnevezése. A modell indexek. SA process modell, kontext-diagram készítése.

3. Adatfolyam diagram szerkesztése

A DFD (data flow diagram) szerkesztő lehetőségei, adatszótár definiálása, szintaktika. Szöveges és grafikus megjegyzések készítése, a modellhez kapcsolásuk. A státuszok szerepe.

4. Processzek specifikálása

A processz specifikáció (P-Spec) létrehozása, a DFD-hez kapcsolása, szintaktikája. Az SA modell ellenőrzése. Az ellenőrzés paraméterei. Szintek közötti kapcsolatok ellenőrzése (level balancing). Az adattárak (store) hierarchiája és elérésük.

5. SA Riportok generálása

A riport generáló makrók szintaktikája. A lekérdezések paraméterezhetősége.

6. Az RT modellezés

A vezérlés modellezése a Hatley/Pirbhay rendszerben. A vezérlő folyamat jelentősége, szerepe és szintaktikája. Vezérlő folyamatok, események, kettő és több értékű vezérlő jelek felvétele és szintaktikai ellenőrzése.

7. Vezérlések specifikálása

A vezérlő specifikáció (C-Spec) típusai, kapcsolata a DFD-vel. A C-Spec - P-Spec egyesített modell. A processz aktivációs tábla (PAT) definiálása, kitöltése, szintaktikai ellenőrzése.

8. Döntési- és állapotábrák

A táblaszerkesztő editorok parancsainak összefoglalása, a tábla ellenőrzése. A táblák (DT-Decision Table, SEM-State Event Matrix) szerepe a vezérlés leírásában. Mealy és Moore modell szerinti állapotábrák.

9. Az állapotgráf

Az állapotgráf (State Transition Diagram, STD) szerkesztő rendszer működése, parancsai. Az STD ábra szintaktikai ellenőrzése.

10. Az IM modellezés

Az entitás-relációs diagram (ER) szerkesztése, kapcsolata az SA és RT modellekkel. Az ER modell szintaktikai ellenőrzése. Relációk megvalósítása, kulcsok és referenciák ábrázolása az adatszótárban.

11. Konfigurációs menedzsment

A Teamwork modell származtatási rendszere. Az alapmodell (baseline) és inkrementális modellek kapcsolata. Párhuzamosan fejlesztett modellek konszolidálása.

12. Az SD tervezés I

A struktúra diagram szerkesztése, Modul Specifikáció (M-Spec) készítése. Modulok típusai és kapcsolatai, azok ábrázolása.

13. Az SD tervezés II

A struktúra diagram automatikus generálása DFD alapján. A Toolkit használata.

A labor első fél órájában rövid elméleti összefoglalást tartottunk, majd bemutattuk az elvégzendő lépéseket a rendszeren. Ezt követően a hallgatók az órai munka céljára általuk felvett vagy általunk adott példán, esetleg a rendszerhez tartozó mintapéldák valamelyikén gyakoroltak. Ezek a feladatok óráról órára változtak, közöttük általában nem volt összefüggés; legfontosabb céljuk az volt, hogy az adott anyagrészt elsajátítsák segítségükkel.

Az önálló hallgatói feladat megoldására az órai tevékenységgel párhuzamosan került sor. A hallgatók a feladatot a második héten megkapták és előre megadott ütemterv szerinti időpontokban az elkészült dokumentációkat be kellett adni. Az önálló feladat valamennyi hallgató számára egyforma volt. Az informatikus és villamosmérnöki szakokon a feladat egy intelligens bolti mérleg vezérlőprogramjának megtervezése volt. A közgazdász hallgatók számára egy iktatási rendszer tervezése volt a feladat.

Az első fázisban elkészült a rendszer definíciója és a követelmények elemzése, valamint a felhasználói felület specifikálása. A Teamwork használatával kellett elkészíteni a kontext diagramot és az azon szereplő adatok adatszótári definiálását. A második lépésben elkészítették a hallgatók az SA, RT és IM modelleket, a program- és vezérlő specifikációkat, valamint a felhasználói kézikönyvet. Az átadandó dokumentumok közé tartoztak az ellenőrzések eredményeit tartalmazó listák. A harmadik fázisban az ellenőrzött modul-tervek beadása volt a követelmény.

3. Eredmények, továbblépési lehetőségek

A CASE labor elindítása a hallgatók körében nagy tetszéssel találkozott. Ezt támasztja alá

- a hallgatók által készített dokumentációk kiállítása, minősége;
- az a tény, hogy a hallgatók a kötelező feladaton kívül más feladatoknál is a Teamwork-öt használták;
- az a körülmény, hogy a modult a következő évben is sokan választották;
- a hallgatói visszajelzések.

A tárggyal kapcsolatosan egyetértés alakult ki abban, hogy a kiadott önálló feladat látszólagos egyszerűsége ellenére lényegesen nagyobb munkát jelentett, mint azt előre becsültük. A hallgatók ezt nem mindenkor érezték tehernek. Ennek ellenére célszerűnek látszik a feladat egyszerűsítése. Problémát okozott a tematika és az önálló feladat ütemezése. A feladat megoldását a 2. héten kellett kezdeni és ekkor még nem rendelkezett a hallgatóság elegendő ismerettel. A későbbiekben célszerű lesz a lecsökkentett méretű feladatot teljes egészében a félév második részére áttenni. Az önálló feladat kizárta az együttműködést és így pontosan a Teamwork csapatmunkát támogató lehetőségei veszték el. Az eredmények értékelésének fényében érdemes lesz a konfigurációs menedzsment részt előre venni és nem önálló, hanem 3-4 fős csoportnak adott feladatokban gondolkodni. Viszonylag kevés idő jutott az IPSE által nyújtott szolgáltatások gyakorlására.

Összefoglalva megállapíthatjuk, hogy a CASE rendszerek oktatásba vitele sokéves alkalmazói és oktatásfejlesztői munka eredménye. Ezen munka legkritikusabb része az eszköz alapos megismerése, amihez csak kellő gyakorlattal lehetett eljutni. Ezt követően oktatási tapasztalatainkon alapulva kidolgoztunk egy olyan didaktikus tematikát rengeteg mintapéldával együtt, amely alkalmasnak látszik az eszköz használatának megtanítására. Munkánk során mindvégig olyan hallgatókat vontunk be a képzésbe, akik a módszereket jól ismerték, korábban több-kevesebb gyakorlatot szereztek annak alkalmazásában. Fontos jövőbeni kísérletnek tartjuk a képzés kiterjesztését módszertani előképzettség nélküli hallgatókra. Eredményeinket külföldi TEMPUS-partnereink is elismerték.

Irodalomjegyzék

- [1] Jackson, M.A. System Development, Prentice-Hall, 1983
- [2] Rumbaugh, J.et al. Object-Oriented Modeling and Design, Prentice-Hall, 1991.
- [3] Gane, Chris. Computer-Aided Software Engineering. The methodologies, the products, and the future. Prentice-Hall, 1990.
- [4] Cadre Technologies Inc. teamwork/SA teamwork/RT User's Guide 4.0, Cadre Technologies Inc. 1990.
- [5] Cadre Technologies Inc. teamwork/SD User's Guide 4.0, Cadre Technologies Inc. 1990.
- [6] Cadre Technologies Inc. teamwork/IM, Object-Oriented Real-Time Analysis with teamwork 4.0, Cadre Technologies Inc. 1990.
- [7] Cadre Technologies Inc. teamwork Environment Reference Manual 4.0, Cadre Technologies Inc. 1990.
- [8] DeMarco, Tom. Structured Analysis and System Specification. Yourdon Press, 1978.
- [9] Hatley, J. Pirbhai, I.A. : Strategies for Real-Time System Specification, Dorset House, 1988.