

8. Rendszermodellek

Kérdések

- Miért kell a rendszer kontextusát már a követelménytervezés során modellezni?
- Mi a viselkedési modell, az adatmodell és az objektum-modell?
- Milyen jelöléseket tartalmaz az UML (Unified Modeling Language)?
- Hogyan segítik a CASE munkapadok a rendszermodellezést?

Tartalom

- Kontextus modellek
- Viselkedési modellek
- Adat-modellek
- Objektum-modellek
- CASE munkapadok (workbench)

Rendszermodellezés

- A rendszer modellezése egyrészt segíti a rendszerelemzőt a rendszer funkcionalitásának megértésében, másrészt modellek segítségével a megrendelővel is lehet kommunikálni.
- A rendszert különböző nézőpontokból különféle modellek reprezentálhatják
 - A *külső nézőpont* a rendszer kontextusát és környezetét mutatja be;
 - A *működési nézőpont* a rendszer működését mutatja;
 - A *strukturális nézőpont* a rendszer vagy az adatszerkezetek felépítését mutatja.

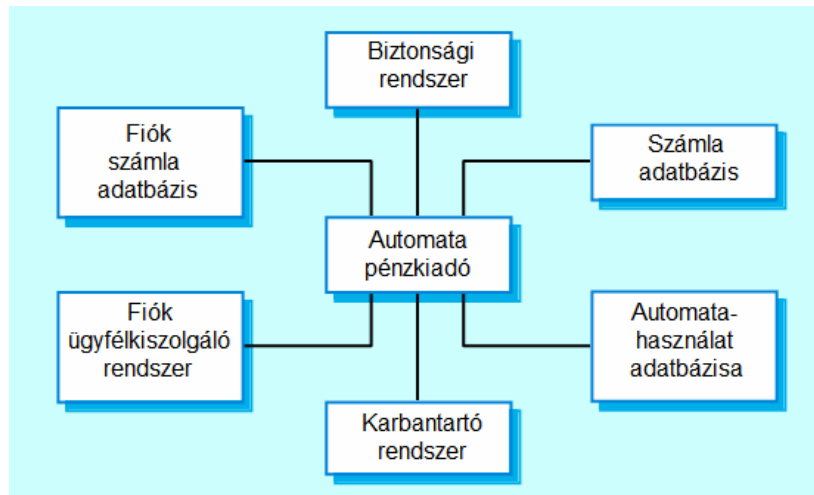
A modellek típusai

- *Adatfeldolgozás modell*: az adat feldolgozásának lépései különböző szinteken.
- *Kompozíció (aggregáció) modell*: egyes entitások hogyan épülnek fel más entitásokból.
- *Architektúrális modell*: a legfontosabb alrendszereket mutatja be.
- *Osztály-modell*: Az entitások közös jellemzőit mutatja be.
- *Gerjesztés/válasz modell*: a rendszernek különféle eseményekre adott reakcióit mutatja.

Kontextus modellek

- A kontextus modellek a rendszer működési környezetét mutatják be: mi van a rendszer határain kívül.
- Társadalmi és szervezeti érdekek befolyásolhatják a döntést, hogy hová helyezzük a rendszer határait.
- Achitektúrális modellekkel a rendszert és annak más rendszerekkel való kapcsolatát mutatjuk be.

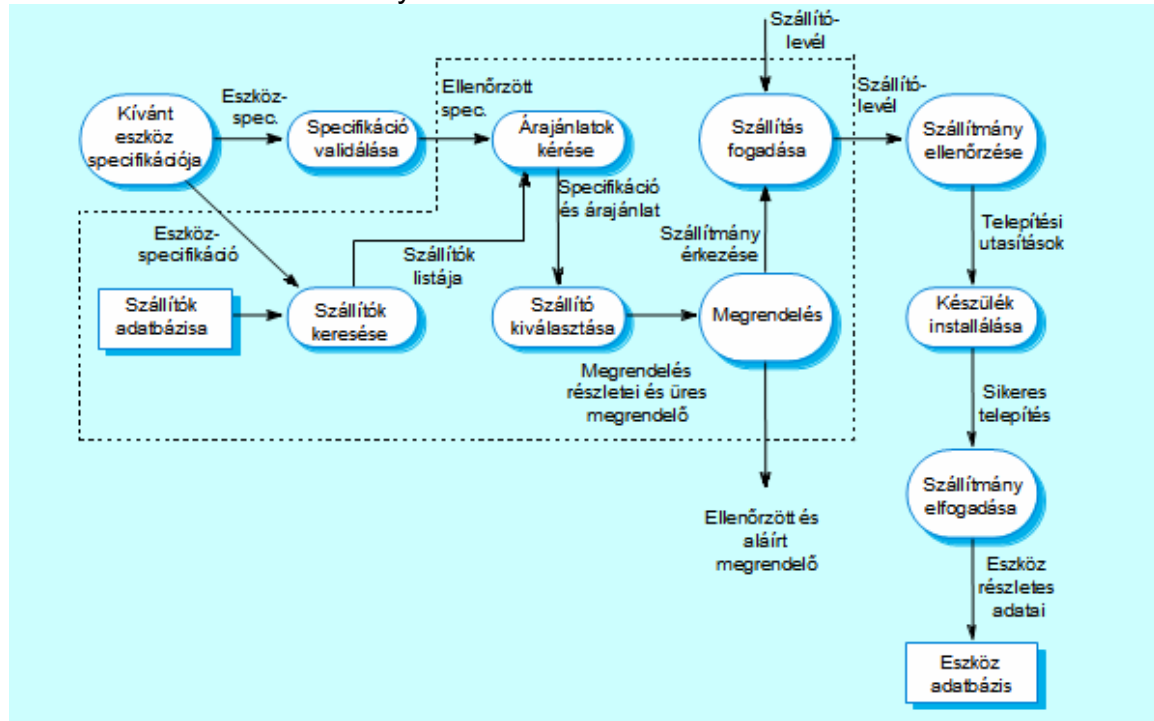
Példa: banki automata kontextusa



Folyamat modellek

- A folyamatmodellek a teljes folyamatot, és ezen belül a rendszer által támogatott folyamatokat írják le.
- Az adatfolyam modellek a folyamatokat és a folyamatok közötti információ-áramlást mutatják.

Példa: eszközbeszerzés folyamata



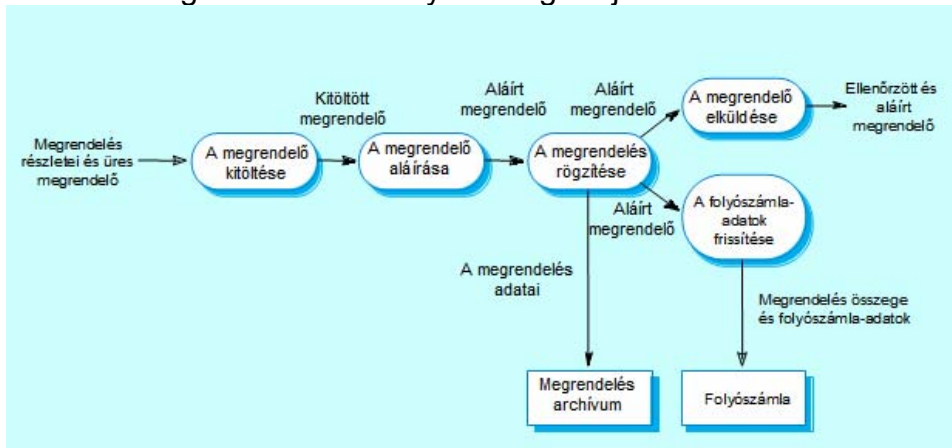
Viselkedési modellek

- A viselkedési modellek a rendszer viselkedését írják le.
- Két típusa:
 - Adatfeldolgozó modellek: az adatok feldolgozásának leírása a rendszeren való áthaladásuk során;
 - Állapotgép modellek: a rendszer eseményekre való válaszát írják le.
- Ezek a modellek különböző perspektívákból mutatják be a rendszert, így mindkettő szükséges a viselkedés leírásához.

Adatfeldolgozó modellek

- Adatfolyam-diagramok (AFD) jól használhatók a rendszer adatfeldolgozó funkcióinak leírásához.
- A feldolgozás lépéseit mutatják, ahogy az adat halad a rendszeren át.
- Az AFD-ok számos analízis módszer lényeges részét alkotják.
- Egyszerű és intuitív jelölés, a megrendelő is megérti.
- Az adat feldolgozását mutatja a bementettől a kimenetig.

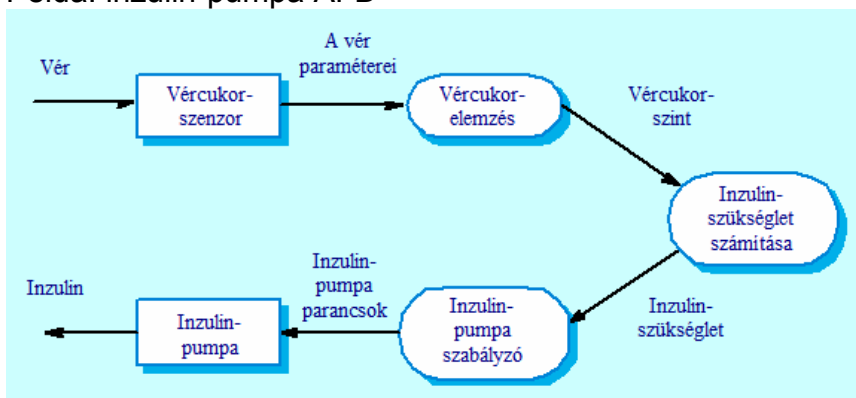
Példa: A megrendelés adatfolyam-diagramja



Adatfolyam-diagramok

- Az AFD-ok a rendszert egy funkcionális nézőpontból mutatják.
- Az adatok és folyamatok kapcsolatának összerendelése és követése hasznos a rendszer működésének megértéséhez.
- AFD-ok használhatók még a rendszer és annak környezetében lévő más rendszerek közötti adatcsere bemutatására.

Példa: inzulin-pumpa AFD



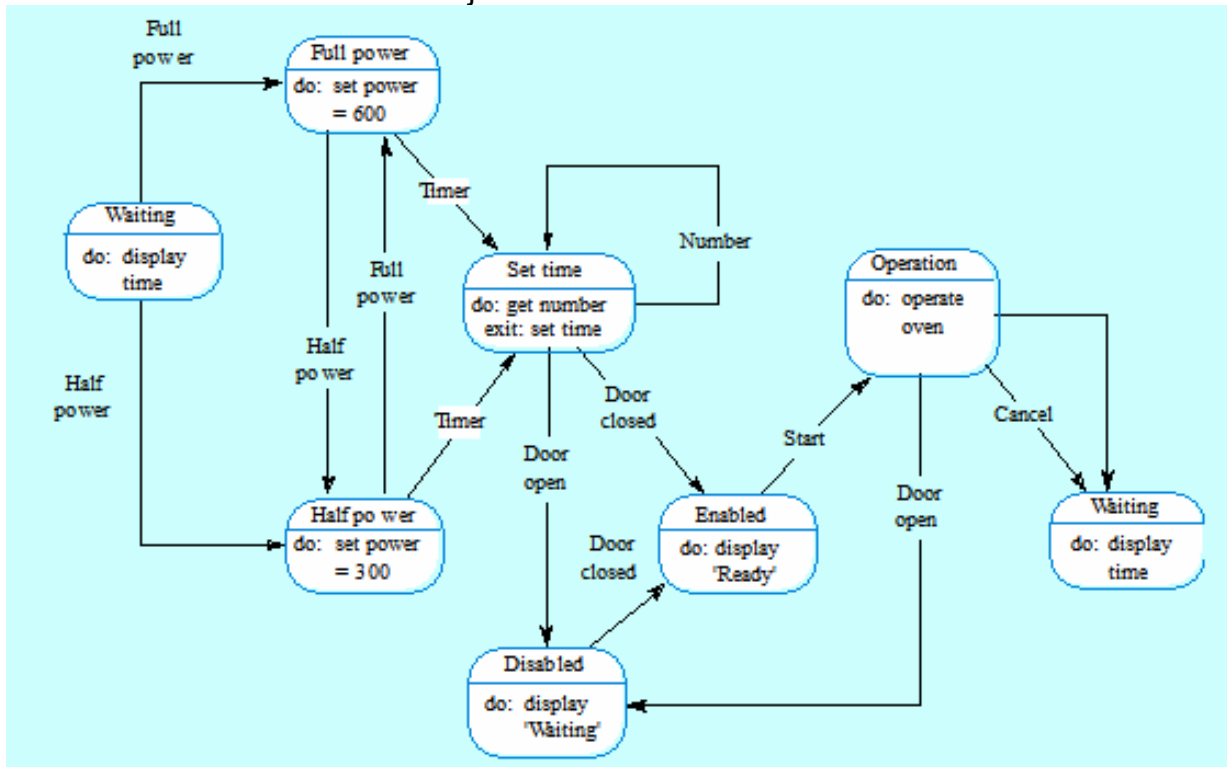
Állapotgép modellek

- A rendszer viselkedését modellezzük annak különböző külső és belső eseményekre adott válaszain keresztül.
- Mivel a rendszernek gerjesztésekre, ingerekre adott válaszait modellezzük, gyakran használjuk valós idejű rendszerek modellezésére.
- Állapotgépek: a rendszer állapotai a csomópontok, köztük futó irányított élek pedig az események. Egy esemény bekövetkezésekor a rendszer egyik állapotból a másikba megy át.
- Az állapot-diagramok (*statechart*) az UML fontos részét alkotják. Ezzel állapotgépeket írhatunk le.

Állapot-diagramok

- A modellnek kisebb részekre, al-modellekre bontását teszi lehetővé (dekompozíció).
- Az akciók rövid leírása az egyes állapotokban a 'do' utasítás után található.
- Kiegészíthető az állapotok és a gerjesztések leírását tartalmazó táblázatokkal.

Példa: Mikrohullámú sütő modellje



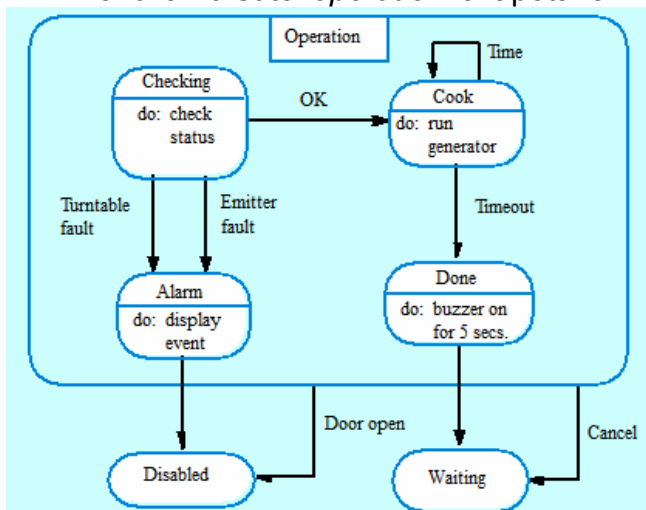
Mikrohullámú sütő állapotai

Állapot	Leírás
Waiting	A sütő bemenetre vár. A kijelzőn a pontos idő látható.
Half power	A sütő teljesítménye 300W. A kijelzőn a 'Half power' üzenet látható.
Full power	A sütő teljesítménye 600W. A kijelzőn a 'Full power' üzenet látható.
Set time	A főzési idő a felhasználó által megadott értékre állítva. A kijelző az éppen beállított időt mutatja.
Disabled	A sütő működése biztonsági okokból nem engedélyezett. A belső lámpa be van kapcsolva. A kijelzőn a 'Not ready' üzenet látható.
Enabled	A sütő működése engedélyezett. A belső lámpa ki van kapcsolva. A kijelzőn a 'Ready to cook' üzenet látható.
Operation	A sütő működik. A belső lámpa be van kapcsolva. A kijelzőn a visszafelé számláló időzítő látható. A főzés befejezése után egy 5 másodperces hangjelzést ad. A lámpa be van kapcsolva. A kijelzőn a hangjelzés ideje alatt a 'Cooking complete' üzenet látható.

Mikrohullámú sütő gerjesztései

Gerjesztés	Leírás
Half power	A felhasználó megnyomta a 'half power' gombot
Full power	A felhasználó megnyomta a 'full power' gombot
Timer	A felhasználó megnyomta az egyik időzítő gombot
Number	A felhasználó megnyomott egy szám-billentyűt
Door open	A sütő ajtaja nyitva
Door closed	A sütő ajtaja zárva
Start	A felhasználó megnyomta a 'start' gombot
Cancel	A felhasználó megnyomta a 'cancel' gombot

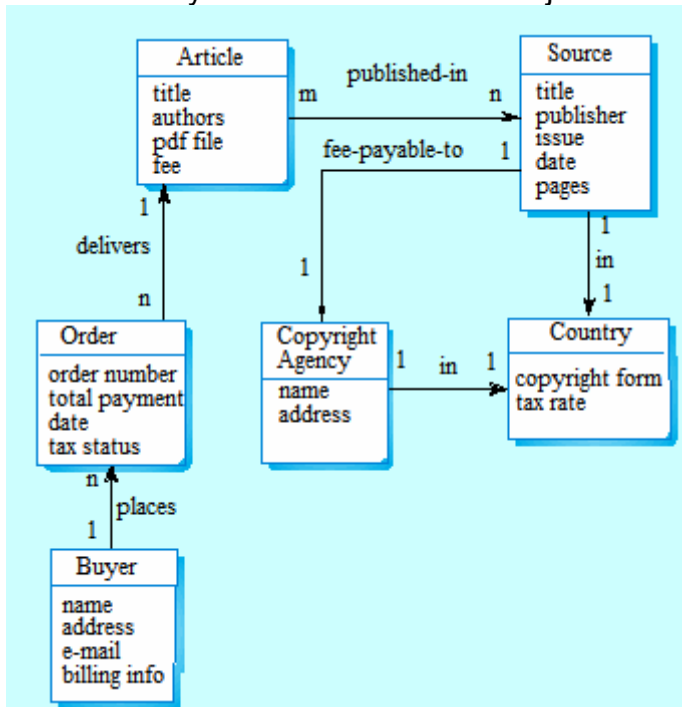
A mikrohullámú sütő 'operation' állapotának modellje



Szemantikus adatmodellek

- A rendszer által feldolgozott adatok logikai struktúráját írja le.
- Az *Entitás – Reláció – Attribútum* diagram a rendszerben használt entitásokat, az ezek közötti relációkat és a entitások attribútumait mutatja be.
- Adatbázisok modellezésénél széles körben használt. Relációs adatbázisokkal könnyen implementálható.
- Az UML nem nyújt specifikus jelölésrendszert, de a az objektumok és az asszociációk használhatók e célra.

Példa: A könyvtár szemantikus modellje



Adat-szótárak

- Adat-szótár: a rendszer-modellekben használt valamennyi név listája. Az entitások, kapcsolatok és attribútumok leírását is tartalmazza.
- Előnyei
 - Támogatja a név-menedzsmentet és segít az ütközések elkerülésében;
 - Fontos szervezési tudástár: összeköti az elemzés, tervezés és implementáció során összegyűlt információkat;
- Sok CASE munkapad támogatja az adat-szótárak kezelését.

Példa: adat-szótár bejegyzések

Név	Leírás	Típus	Dátum
Article	Egy LIBSYS rendszeren keresztül megrendelhető cikk adatai.	Entitás	30.12.2002
authors	A cikk szerzőinek neve, akik a kifizetett díjból részesülhetnek.	Attribútum	30.12.2002
Buyer	A személy vagy szervezet, aki a cikk egy példányát megrendelte.	Entitás	30.12.2002
fee-payable-to	Egy 1:1 reláció az 'Article' és a 'Copyright Agency' között, aki a szerzői jogdíjat beszedi.	Reláció	29.12.2002
address (Buyer)	A megrendelő címe. Ezt az információt a számlázás során használják.	Attribútum	31.12.2002

Objektum modellek

- Az objektum modellek a rendszert az objektum-osztályok és ezek kapcsolatán keresztül mutatja be.
- Egy objektumosztály egy olyan objektumhalmaz absztrakciója, amelynek elemeinek közös attribútumai és szolgáltatásai (operációi) vannak.
- Lehetséges objektummodellek:
 - Öröklési modellek;
 - Aggregáció modellek;
 - Interakció modellek.
- A rendszer által befolyásolt valós entitások reprezentálására természetes módon alkalmazható
- Absztraktabb entitások modellezése ezzel a módszerrel nehezebb
- Az objektum-osztályok meghatározása általában nehéz, az alkalmazási környezet mélyebb ismeretét igénylő folyamat
- Az alkalmazási környezet entitásait ábrázoló objektum-osztályok újra felhasználhatók más rendszerekben

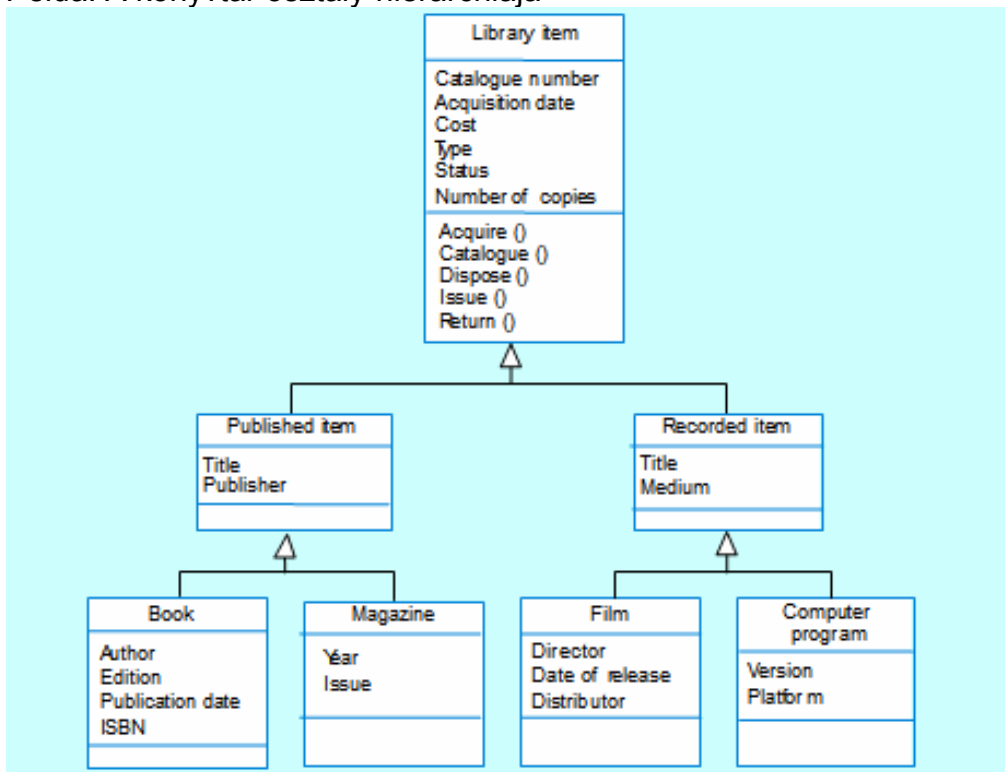
Öröklési modellek

- Az objektum-osztályok hierarchiába szervezésére.
- A hierarchia csúcsán lévő osztályok az összes osztály közös tulajdonságait reprezentálják.
- Az objektum-osztályok az attribútumaikat és szolgáltatásaikat egy vagy több szuper-osztálytól öröklik. Ezeket aztán igény szerint specializálni lehet.
- Az osztály-hierarchia tervezése nehéz folyama, ha a különböző ágakon előforduló ismétlődéseket el akarjuk kerülni.

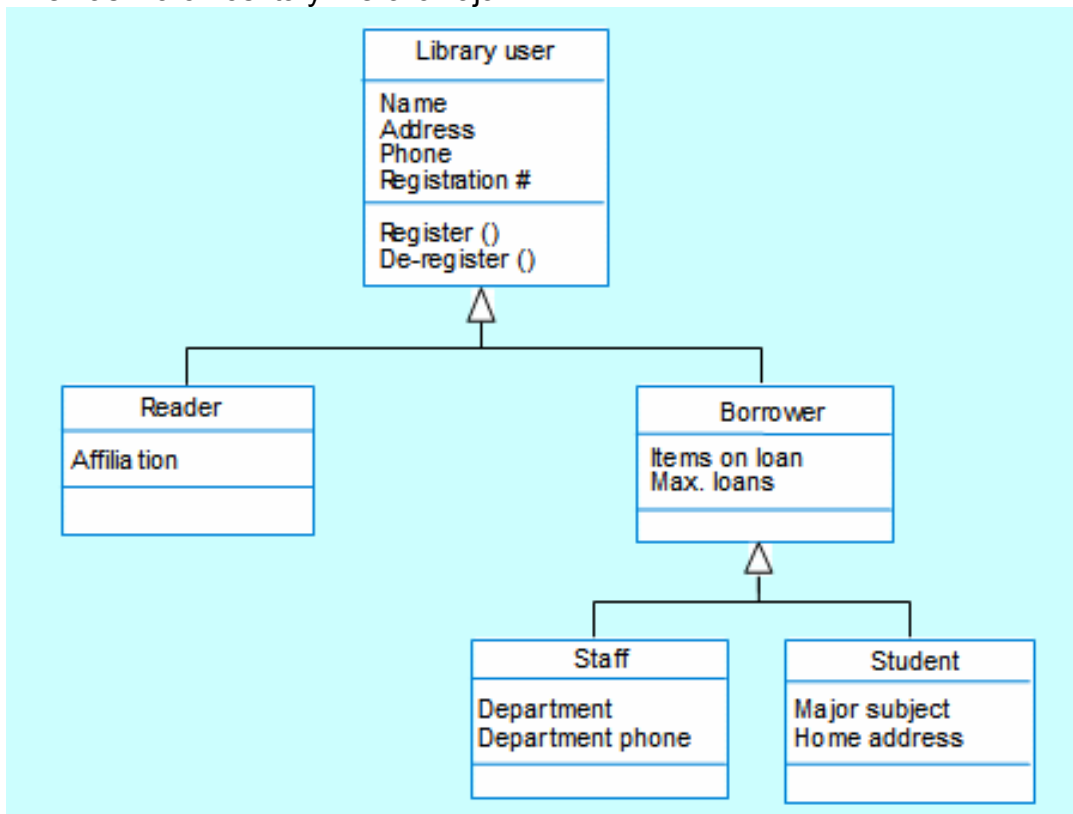
Objektum modellek és az UML

- Az UML egy szabványos leíró nyelv, amelyet széles körben használt objektum-orientált analízis és tervezési módszerek tervezői hoztak létre.
- Az objektum-orientált tervezés hatékony szabványává vált.
- Jelölés
 - Az objektum osztályokat téglalapok jelképezik, amelyben a név felül, a tulajdonságok középen, az operációk pedig az alsó részben helyezkednek el;
 - Az objektumok közti relációkat (melyeket asszociációnak is neveznek) az objektumokat összekötő vonalak jelképezik;
 - Az öröklést itt *általánosítás*nak nevezik és a hierarchiában nem „lefelé”, hanem „felfelé” olvasandó.

Példa: A könyvtár osztály-hierarchiája



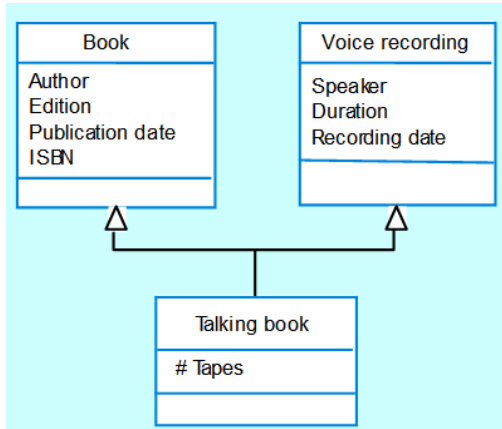
A felhasználók osztály-hierarchiája



Többszörös öröklés

- A többszörös öröklés lehetővé teszi objektum-osztályoknak attribútumok és szolgáltatások több (és nem csak egy) szuper-osztálytól való öröklését.
- Ez szemantikus konfliktusokhoz vezethet, ha attribútumok vagy szolgáltatások több szuper-osztályban ugyanolyan névvel, de különböző szemantikával szerepelnek.
- A többszörös öröklés az osztály-hierarchia átrendezését nagyon megnehezíti.

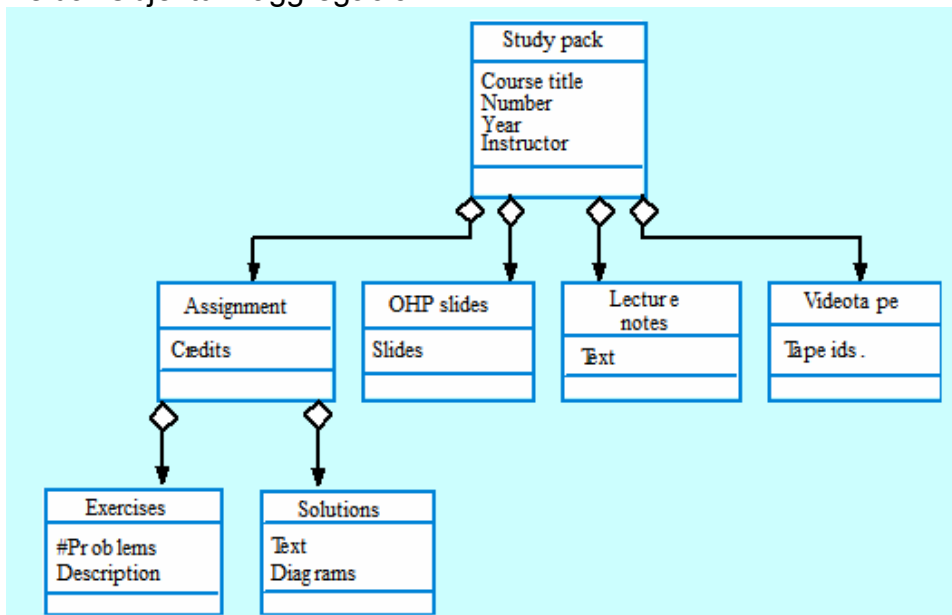
Példa: többszörös öröklés



Objektum aggregáció

- Az aggregációs modell azt mutatja, hogy összetett osztályok hogyan állnak össze más osztályokból.
- Az aggregációs modellek hasonlóak a szemantikus adatmodellek „tartalmaz” relációjához.

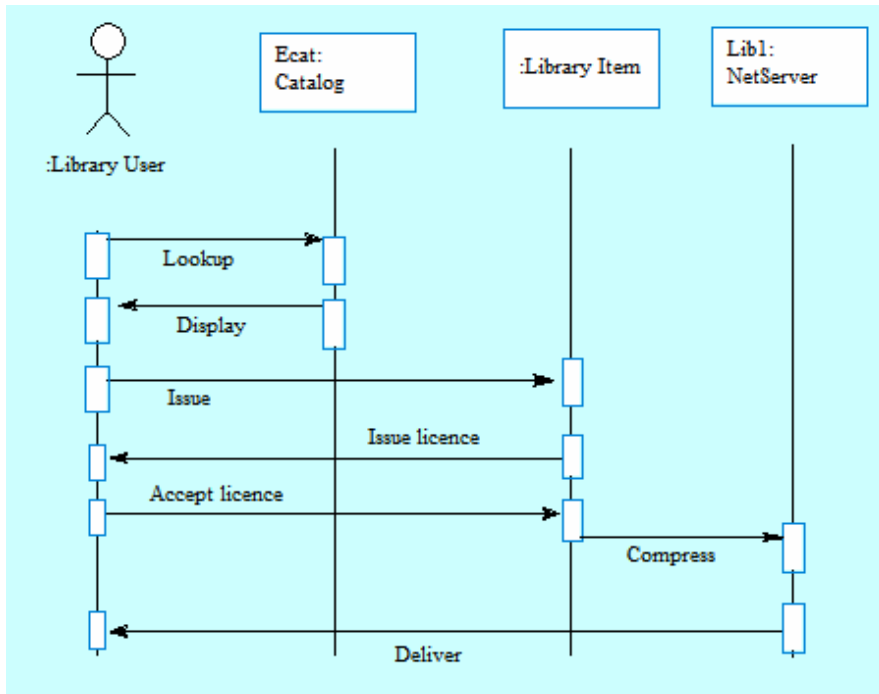
Példa: Objektum aggregáció



Objektumok viselkedés-modellezése

- A viselkedési modellek az objektumok közötti interakciókat mutatják be a rendszer valamely funkciója során, amelyet esettanulmány (use case) specifikál.
- A szekvencia-diagramokat (vagy együttműködési diagramokat) az UML-ben az objektumok közötti interakciók modellezésére használjuk.

Példa: Elektronikus dokumentumok letöltése



Strukturált módszerek

- A strukturált módszerek fontos eleme a rendszermodellezés.
- Ezek a módszerek definiálnak egy modellhalmazt, egy eljárást ezen modellek meghatározására, valamint a modellekre vonatkozó szabályokat és ajánlásokat.
- CASE eszközök támogatják a rendszermodellezést egy strukturált módszer részeként.

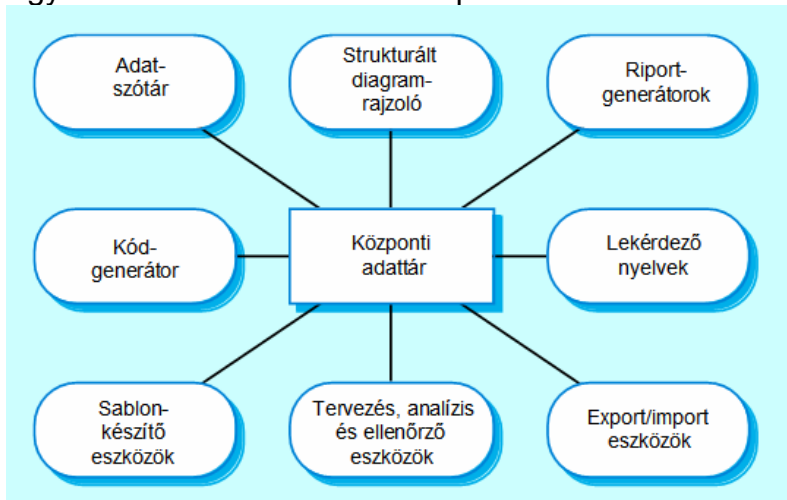
A módszerek hiányosságai

- Nem modellezik a nem funkcionális rendszerkövetelményeket.
- Általában nem tartalmaznak arról információt, hogy a módszer alkalmazható-e az adott problémára.
- Túl sok dokumentációt eredményeznek.
- A rendszermodellek néha túl részletesek és a felhasználók számára nehezen érthetőek.

CASE munkapadok

- Eszközök koherens halmaza, amelyek a szoftvergyártás egyes tevékenységeit támogatják, mint az analízis, tervezés és tesztelés.
- Az analízis és tervezés munkapadok támogatják a rendszermodellezést mind a követelménytervezés, mind a rendszertervezés fázisokban.
- Ezen munkapadok néha csak egyetlen tervezési módszert, mások többfajta rendszermodell létrehozását is támogatják.

Egy analízis és tervezés munkapad



Analízis munkapad komponensei

- Diagram szerkesztők
- Modell analízis és ellenőrző eszközök
- Adattár és kapcsolódó kereső nyelv
- Adat-szótár
- Riport definíciós és generátor eszközök
- Sablon definíciós eszközök
- Import/export transzlátorok
- Kódgenerátor eszközök

Összefoglalás

- A modell a rendszer egy absztrakt nézete. Az egymást kiegészítő különféle modell típusok különböző rendszer-információkat tartalmaznak.
- A kontextus modellek a rendszer helyét definiálják a környezetben, valamint más rendszerek és folyamatok között.
- Az adatfolyam-modellek az rendszerben folyó adatfeldolgozás menetét mutatják.
- Az állapotgép-modellek a rendszer viselkedését mutatják külső és belső események hatására.
- A szemantikus adatmodellek a rendszerbe importált és az abból exportált adatok logikai struktúráját mutatják.
- Az objektum-modellek a rendszer entitásait, ezek osztályait és aggregációját ábrázolják.
- A szekvencia-modellek az aktorok és az általuk használt rendszerobjektumok közötti interakciókat mutatják.
- A strukturált módszerek rendszermodellek kidolgozásához biztosítanak keretrendszereket.