

Kockázat és kockázatkezelés a szoftvervásárlásban

A szoftvervásárlás vonzó alternatívává vált a saját szoftverfejlesztéssel szemben, mert nagyságrendi megtakarítást jelent. Ugyanakkor viszont ezzel alapvetően megváltozik a kockázatok szerkezete.

A szoftverfejlesztésre specializált cégeknek olyan rutinjuk van, ami az információs részlegekben semmiképpen sem található meg. A határidőktől való lemaradás is csökkenthető a vásárolt szoftverrel. A készen vett szoftver hozzáigazítása a helyi viszonyokhoz még mindig hatékonyabb megoldás anyagilag, mint a saját szoftverfejlesztés.

Persze, amíg a kész szoftver vásárlása egyes kockázati tényezőket mérsékel, addig másokat fokoz. Ezek közül a legjelentősebb annak a kockázata, hogy tényleg megfelel-e speciális igényeinknek a vásárolt szoftver. Vásárlásunkban csak a dokumentációra támaszkodhatunk, el vagyunk zárva attól a belső diskusziótól, amely a valódi működési zavarokat felszínre hozza. Ha a kielégítetlen jellemzőket a szoftver utólagos testre szabásával korrigáljuk, az megint újabb kockázati tényezőket hoz felszínre.

Az ellenőrizhető kockázati tényezőket mindenesetre fel kell tárnunk, és fel kell becslnünk, hogy a telepítés során kezelhetők legyenek. Különösen vonatkozik ez a működési hibák kockázataira.

Csökkenő és növekvő kockázati tényezők

A szoftverfejlesztés és -használat kockázati tényezőit az 1. táblázatban tekinthetjük át. E kockázati tényezők egy részét csökkenti, más részét fokozza, ha saját fejlesztés helyett kész szoftvert szerzünk be.

1. táblázat

A szoftverfejlesztés és -használat kockázati tényezői

Fejlesztési kockázat	Működtetési kockázat
Ütemezési kockázat	A működési hibák kockázatai
Műszaki kockázat	Működőképességi kockázat
Konfliktusok kockázata	Minőségi kockázat
	A módosítás kockázatai
	Karbantartási kockázat
	Adaptációs kockázat
	Portabilitási kockázat
	Pénzügyi kockázat
	Szervezeti kockázat
	Konfliktusok kockázata

A fejlesztés kockázatai közül mérséklődik az *ütemezési kockázat*, vagyis annak a kockázata, hogy kiszaladunk a határidőkből vagy a költségkeretből, és a *műszaki kockázat*, vagyis annak a kockázata, hogy kiszaladunk a műszaki lehetőségek korlátai közül.

Persze, a szoftver saját igényeinkhez igazítása is jelent némi kockázatot határidőben és költségben, egy teljes rendszer megtervezése azonban ennél mindenképpen nagyobb kockázatot jelent. Nőhet a *konfliktusok kockázata*, ha a belső szoftverfejlesztő részleg a megélhetési lehetőségei elleni támadásnak tekinti a kívülről történő szoftverbeszerzést. Ez a kockázat a döntési folyamatba való bevonásukkal kezelhető.

A *működtetés kockázatai* közül a legfontosabb a *működési hibák kockázata*, amely maga is két tényezőre bontható: a *működőképesség kockázata*ra és a *minőség kockázata*ra. Az előbbiben az tükröződik, hogy alkalmas-e a szoftver arra, amire kell. Mivel a vásárolt szoftvernek rendszerint vannak már korábbi felhasználói is, a szoftverminőség kockázata várhatóan alacsonyabb, mint saját fejlesztésű szoftver esetén. A működőképesség kockázata viszont csak akkor csökkenthető, ha a felhasználók igazán megértik, hogyan működik majd a rendszer.

Ha első felhasználók vagyunk, az jelentősen mérsékli a vásárolt szoftver hagyományos előnyét, a működés előre látható voltát, különösen akkor, ha az alkalmazási környezet nagyon eltér a szokásostól. Nő a működőképesség kockázata, ha a szoftver nem minden igényünket elégíti ki, ami a készen vett szoftvernél előfordul. A szoftver igényeinkhez igazítása sérülékenyebbé tesz bennünket, mint a saját fejlesztés esetében, így nő a minőségi kockázat.

Ami a *szoftvermódosítás kockázatát* illeti, itt három elemet különböztethetünk meg: a *karbantartási kockázatot*, az *adaptációs kockázatot* és a *portabilitási kockázatot*. A szoftvervásárlás új kooperációs kapcsolatot hoz létre. Ha a karbantartás az eladó dolga, lehet, hogy újra akarja tárgyalni a szerződést, növelve a karbantartás díját, vagy elutasítva a további karbantartási kötelezettséget. A más rendszerre való áttérés igen nagy költsége jelentős szerződési kockázattal, és az eladóhoz való láncolódás veszélyével jár. Ráadásul, ha a vásárló engedi visszafejlődni felkészültségét, teljesen elveszítheti a képességét a belső karbantartásra, adaptálásra, a rendszer más környezetbe való átvitelére.

A *pénzügyi kockázat*, vagyis annak a veszélye, hogy a rendszer nem eredményezi a befektetés megfelelő megtérülését, csak akkor csökkenthető a szoftvervásárlással, ha a vevő igazán megérti a szoftver méretre igazításának és a nagyobb rendszerbe integrálásának a követelményeit, és precízen képes felbecsülni az ehhez szükséges erőfeszítéseket.

A *szervezeti kockázat* akkor jelentkezik, ha megváltozik a környezet, és ettől a meglévő rendszer értéke csökken. Ez a kockázat nőhet, ha a szoftver eladója olyan információhoz jut, amelyet azután a tárgyaláson ellenünk fordíthat.

A szoftvervásárlás döntési pontjai

Bár a szoftvervásárlás kockázatának minden eleme aktívan kezelendő, most elsősorban a működési hibák kockázataira összpontosítunk, amelyek a szoftvervásárlás és -telepítés döntési pontjai között gyakran csak áttételesen szerepelnek. Mind a költség, mind a haszon vizsgálendő a rendszer kiválasztásakor. Olyan rendszert kell választanunk, amely nemcsak kielégíti igényeinket, de a működési hibák kockázata is a legkisebb vele az új környezetben.

A feladat megoldásának képessége mindig abban a környezetben vizsgálendő, amelyben a szoftver működni fog. Ha túl sok igényt nem teljesít a szoftver, az igényekhez igazítás és a rendszerbe integrálás csaknem akkora erőfeszítést igényelhet, mint a saját fejlesztés. A rendszer más telepítéseinek működés közbeni megtekintéséből jelzéseket kaphatunk arra, hogy mit várhatunk.

Formális értékelő módszereket használhatunk akkor, ha konkurens rendszerek közül kell választanunk. Ezek a módszerek általában arra koncentrálnak, hogy mire képes a rendszer, hogyan elégíti ki a követelményeket, és mennyire javítja az intézmény teljesítőképességét. A kockázati tényezők ezekben az eljárásokban súlyozhatók, vagy egyes kockázati tényezők preferálhatók. Nem mérhető viszont velük egzaktnak az egyes kockázati tényezőkhöz tartozó potenciális veszteség.

Értékelnünk kell, mi romolhat el. A szükséges módosításokkal járó kockázat döntő lehet a szoftverrendszerek közötti választásban.

A beszerzést a telepítés tervezése követi. Ha a szoftver egy korábbi rendszer helyére lép, döntenünk kell az átmeneti párhuzamos működtetés és a drasztikus lecserélés lehetősége között. Egyes esetekben értelme van egy kis részrendszeren a próbatelepítésnek.

A telepítési döntésben figyelembe kell vennünk a költségeket is és a hiba kockázatát is. A párhuzamos működtetés költséges és nehezen kezelhető, a drasztikus lecserélés viszont elfogadhatatlan kockázattal járhat. Drasztikus lecserélés előtt különösen nagy jelentősége van a próbajáratásnak, viszont a kisebb rendszer nem feltétlenül hozza ki a hibákat.

A szoftvernek rejtett hibái is lehetnek, ezért döntő, hogy mennyi időt szánunk a próbajáratásra. A kockázat mértékétől függ, hogy megéri-e a hosszas próbajáratás az ezzel járó költségeket.

A működési hibák kockázatának becslése

A működési hibák kockázatának becslését a *lehetőleges veszteségek becslésével* kell kezdenünk. Ehhez fel kell tárnunk minden olyan kockázatot, amely a rendszer telepítése után veszteségekre vezethet. Minden telepítésnek egyedi környezete van, saját sze-

mélyzettel és folyamatokkal. Ez egyedi üzemzavar-lefutásokat és egyedi üzemzavarmódokat eredményez. A szoftver kiválasztás során kívülről kell elemeznünk a rendszert, milyen üzemzavarai lehetnek. Ennek során külön koncentrálnunk kell arra, milyen gyenge pontjai vannak a gépi rész és az emberi rész összeillesztésének, valamint arra, milyen olyan változások várhatók a jövőben, amelyek befolyásolják a rendszer működését. Minden olyan tényezőt fel kell tárnunk, amely kockázatot jelezhet előre, és mindegyiknek meg kell becsülnünk a bekövetkezési valószínűségét. Gondosan elemeznünk kell a rendszert és a környezetét. A nem azonosított kockázat vagy hibamód a lehetséges veszteségek alábecslésére vezet.

Mivel a veszteséget okozni képes környezeti tényezők száma gyakorlatilag korlátlan, a veszteségek várható értékével kell operálnunk.

A j kockázattal járó lehetséges veszteség a T működési időtartam függvénye: $L_j(T)$.

A második lépés felbecsülni a *modulok hibájának kockázatát*. Ez az adott M modul egyszeri hibájából eredő $E^M(T)$ veszteség és a modul meghibásodásai száma $N^M(T)$ várható értékének szorzata:

$$R^M(T) = E^M(T) \cdot N^M(T).$$

Az $E^M(T)$ tényező becsléséhez a lehetséges veszteségek becslését modulról modulra el kell végeznünk. A keresett érték:

$$E^M(T) = \sum_i p(U_i^M) \sum_j p(H_j / U_i^M) L_j(T),$$

ahol $p(U_i^M)$ annak a valószínűsége, hogy az M modult i módon használjuk, $p(H_j / U_i^M)$ pedig annak a feltételes valószínűsége, hogy ha az M modult i módon használjuk, *akkor* a j kockázat előáll. A lehetséges veszteség ugyanis függ attól, hogy a modult milyen módon használjuk. Mivel minden felhasználó másképp használja a rendszert, a $p(U_i^M)$ eloszlás telepítésről telepítésre más és más.

A $p(H_j / U_i^M)$ feltételes valószínűségi eloszlás meghatározásához meg kell vizsgálnunk, hogy a modul egyes lehetséges hibái milyen forgatókönyvre vezetnek. Mivel a szoftverhibákból téves vagy elmaradt eredmény következik, a modul hibái akkor jelentenek kockázatot, ha ilyen eredményért felelősek. A specifikációk visszaelemzésével meghatározható, hogy mely modulokra fordulhat ez elő. A kockázatok feltételes valószínűségi eloszlását tekinthetjük egyenletesnek, vagy a kockázatok komolyságával súlyozottnak.

Az $N^M(T)$ tényezőt a szoftvermegbízhatósági modellel határozhatjuk meg:

$$N^M(T) = \mu(1 - e^{-\theta t(T)}),$$

ahol μ a hibák átlagos száma a modulban, θ annak a valószínűsége, hogy egy szoftverhiba egységnyi CPU-idő alatt meghibásodásra vezet, $t(T)$ pedig a T működési idő alatt felhasznált CPU-idő.

Ehhez a modellhez meg kell becsülnünk a szoftverhibák számát, valamint annak a valószínűségét, hogy egy szoftverhiba adott idő alatt meghibásodásra vezet. Mindkettő a modul jellegének és fejlesztési szintjének függvénye, és vásárolt szoftver esetén nem biztos, hogy rendelkezésünkre áll. Kérhetjük a szoftverfejlesztőtől a fejlesztési problémákról szóló jelentést, de nem biztos, hogy megkapjuk, az is lehet, hogy irreálisan optimista jelentést kapunk.

A tesztelés és a betanulás drága ugyan, de csökkenti a hibás működés kockázatát. Egyrészt a tesztelés és betanulás költségei, másrészt a hibás működés költségei között igyekeznünk kell megtalálni az optimumot.

Következtetések

A vásárolt szoftvertermékek egyre terjedő használata speciális kockázatot eredményez, amely abból ered, hogy a meglévő szoftvert hozzá kell igazítani az új környezethez, és be kell építeni abba. A szoftver

meghibásodási kockázatának megismerése iránymutató lehet a szoftver beszerzésében és telepítésében.

Míg a szoftvervásárlás hagyományos szemléletmódja a rendszer képességeire és korlátaira koncentrált, addig a kockázatelemzés feltárja, mivel járulhat hozzá a vásárolt szoftver a befogadó rendszer kockázataihoz. Ezzel egyrészt többletet ad a hagyományos kiválasztási eljáráshoz, másrészt felhívja a figyelmet a rendszer későbbi működési problémáira és kockázati forrásaira. A működési kockázatok explicit elemzése a kockázatelemzés legfontosabb eleme.

A modulonkénti kockázatelemzés irányt szabhat a telepítési munkának, azokra a részekre fókuszálva a figyelmet, amelyek a legnagyobb kockázatot hozzák. Segít eldönteni, mennyi erőt kell fordítani a tesztelésre és a betanulásra.

/SHERER, S. A.: Purchasing software systems. Managing the risc. = Information & Management, 24. köt. 5. sz. 1993. p. 257–266./

(Válas György)

CD-ROM vendorok az online piacon

Nemrégiben a *Dataware* új szoftverrel és ehhez kapcsolódó szolgáltatásokkal jelentkezett a piacon. A szoftver az Interneten publikált anyagokhoz BRS/SEARCH, Gopher és Mosaic lehetőségeket kínál. A *Dataware* törekvése, hogy minél változatosabb szolgáltatásokat nyújtson felhasználóinak, nem egyedüli. Hasonló szándékot jelez, hogy a CD Plus megvette a BRS Online-t, illetve a SilverPlatternek a CARL Systemmel való megállapodása is. A SilverPlatter törekvéseiről *Christopher MacPhail* beszélt az 1994. május közepén Londonban megrendezett *Internet World & Document Delivery World International* elnevezésű konferencián.

MacPhail elmondta, hogy a SilverPlatter 9 évvel ezelőtti megalakulása óta 1 Mbájt adat telefonvonalakon történő továbbításának költsége – a gyorsabb modemeknek és az egyre olcsóbb vonalaknak köszönhetően – 1/15-ödére csökkent. 1992 végén, „amikor az Internet kereskedelmi használata lehetővé vált”, a távoli kapcsolódások gazdaságossága 20-szorosára nőtt. A 300-szoros fejlődés stratégiájuk átgondolásának szükségességét vetette fel a vállalatnál. Mára a SilverPlatter évi 40–60%-os fejlődést ért el azzal, hogy megelőzte a vevőit: érdemesebb az egész adatbázist helyileg használni, s meg lehet feledkezni a modemekről és telefonvonalakról.

A SilverPlatter egyik legradikálisabb alkalmazása a hálózaton keresztüli, térítéses szolgáltatások bevezetése volt. Ennek kulcsa egy *SPIRS* nevű visszakereső szoftver, melynek fejlesztés alatt álló kliens-szerver verziója – az Internethez hasonlóan – a TCP/IP protokollt használja arra, hogy két fél kommunikálni tudjon egymással. (A *SPIRS* Internet-verziója az első tesztek fázisában van, ehhez 7, díjmentesen kereshető adatbázist helyeztek el egy bostoni szerveren. Még ez évben elkészül a szoftver DOS-, Windows- és Macintosh-változata is.)

A cél által használt anonim ftp és a WWW lehetőséget kínál arra, hogy folyamatosan tájékoztathassák ügyfeleiket, s ezzel kutatási, postai és nyomtatási költséget is megtakarítanak. Saját belső cc:mail e-mail rendszerüket is hozzákapcsolták az Internethez. Ez biztosítja számukra a vevőkkel való levelezés egyszerű módját, azonnali hozzáférést a disztribútorok listájához, belső jelentésekhez, bármiféle emberi beavatkozás nélkül.

A SilverPlatter mellett még két cég kért szót a konferencián. A Springer Verlagtól *Amoud de Kemp* a cég már ismert projektjeit mutatta be, hozzátéve, hogy a Springer számos elektronikus újság kibocsátását is tervezi.

A másik felszólaló *Mary McCall* volt a Dow Jones képviselőjében. A cég az Internetet egyedülállóan globális, megbízható, nyilvános kommu-