

NGB_IN003_1 SZE 2014-15/2 (8)

Szoftver- minőségbiztosítás

Szoftvertesztelési folyamat
(folyt.)

Szoftvertesztelési ráfordítások

- ❖ (Perry 1995)
- ❖ Tesztelésre fordítódik
- ❖ a projekt költségvetés 24%-a
- ❖ a projekt menedzsment költségeinek 32%-a
- ❖ az időkeret 27%-a

A tesztelés mint folyamat

- ❖ Formális folyamat:
 - ❖ specializált teszt-team
 - ❖ a vizsgálat a **szoftver futtatásával** történik:
 - ❖ jóváhagyott eljárásokkal
 - ❖ jóváhagyott tesztesetek végrehajtásával

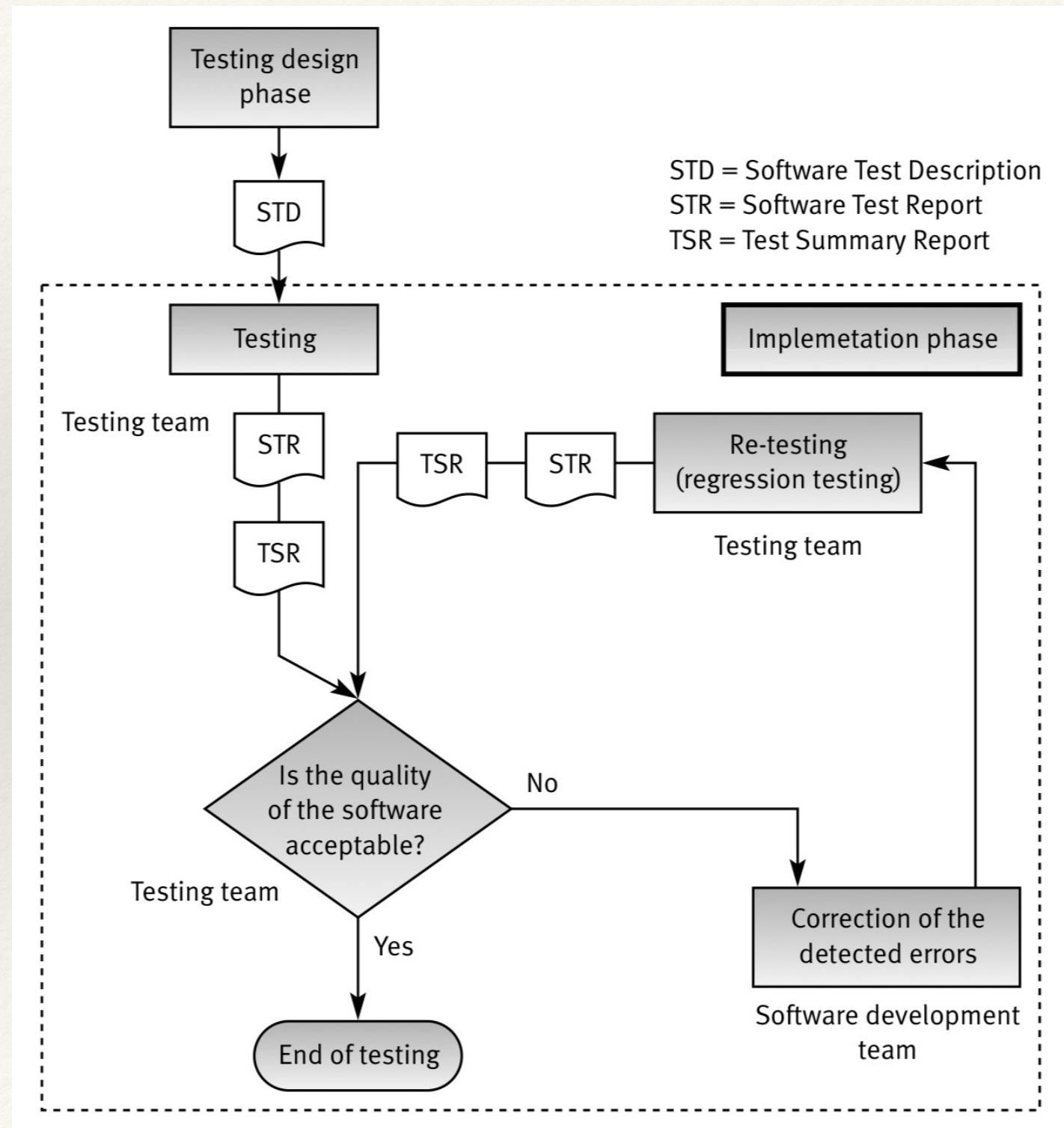
Önálló teszt-team

- ❖ speciális képzés
- ❖ **hatékony és elfogultságtól mentes tesztelés**
- ❖ fejlesztők mint tesztelők nem hatékonyak (nem képesek azokat a hibák megtalálni, melyeket korábban nem azonosítottak a fejlesztéskor)
- ❖ de a fejlesztők effektívek lehetnek egységtesztelésnél

Tesztelési formalitás

- ❖ A tesztelés formalitásának mértéke függ:
 - ❖ a szoftver felhasználásától
 - ❖ a fejlesztő szervezet kultúrájától (fejl. folyamat érettsége)
 - ❖ az időkorlátoktól (dokumentáció alapossága)

Tesztelési tevékenység



Tesztelési feltételek

- ❖ **Mit kell tesztelni?**
- ❖ Információ források:
 - ❖ rendszerkövetelmények
 - ❖ technikai specifikáció
 - ❖ kód
 - ❖ vállalati folyamatok
- ❖ => **tesztbázis** kialakítása (=tesztelési feltételek, tesztkészlet, tesztelési leltár)
 - ❖ a tesztek hatásos és hatékony részhalmaza
- ❖ **szelektálási koncepció/stratégia**
 - ❖ => műszaki tesztervezés

Tesztelési feltételek (folyt.)

- ❖ Stratégia kialakítása
 - ❖ kockázat, hibamodellek, megfelelőségi követelmények, szakértői tanácsok, heurisztikák
- ❖ **Nyomonkövethetőségi (traceability) cél**
 - ❖ tesztelési feltétel visszavezethetősége forrásaikra
 - ❖ követelmények változása
 - ❖ teszteléskor felmerülő problémák elemzése
 - ❖ tesztelés kiterjedésének vizsgálata
- ❖ Prioritások meghatározása és rögzítése

Műszaki tesztterv specifikáció tartalma

- ❖ **(IEEE 829)**
- ❖ Műszaki tesztterv specifikáció azonosítója
- ❖ Tesztelendő jellemzők
- ❖ A tesztelési megközelítés kifejtése
- ❖ Tesztmeghatározás
- ❖ Jellemzők (features) sikerességi / bukási feltételei

Teszteset specifikáció

- ❖ (IEEE 829)
- ❖ Teszteset specifikáció azonosítója
- ❖ Tesztelem
- ❖ Bemenet specifikáció
- ❖ Kimenet specifikáció
- ❖ Környezeti elvárások
- ❖ Speciális eljárások
- ❖ Esetek közötti függőségek

Tesztelési eljárás specifikáció

- ❖ Tesztesetek ésszerű **csoportosítása** (végrehajtáshoz)
 - ❖ funkcionalitás, feature, regressziós készlet
- ❖ **Végrehajtási sorrend** (előfeltételek)
- ❖ =>tesztelési eljárás / tesztszript (manuálisan végrehajtott tesztekhez is)
- ❖ Ütemterv (mikor, ki hajtja végre) [**időszükséglet**]



Tesztelési eljárás specifikáció (folyt.)

- ❖ (IEEE 829)
- ❖ Tesztelési eljárás specifikáció azonosítója
- ❖ Cél
- ❖ Speciális követelmények
- ❖ Az eljárás lépései

Műszaki teszttervezési technikák

❖ Statikus

- ❖ Informális felülvizsgálat
- ❖ Átvizsgálás
- ❖ Technikai felülvizsgálat
- ❖ Inspekció
- ❖ Statikus elemzés
 - ❖ adatfolyam
 - ❖ vezérlésfolyam

❖ Dinamikus

- ❖ Specifikáció alapú
- ❖ ekvivalencia osztályok
- ❖ határérték elemzés
- ❖ döntési táblák
- ❖ állapotátmenet
- ❖ Struktúra alapú
 - ❖ utasítás
 - ❖ döntés (ág)

❖ feltétel (bool)

- ❖ út

❖ Tapasztalat alapú

- ❖ hibasejtés
- ❖ felderítőteszt

A technikák alkalmazása

- ❖ **Specifikáció alapú** technikák minden tesztelési szinten, ahol van specifikáció
- ❖ **Struktúra alapú** tesztelés alapvetően kód tesztelése, de integrálás során (spec. alapú tesztekhez is)
- ❖ **Tapasztalat alapú** technikák kiegészítésként alkalmazhatók (de inkább alacsony kockázatú rendszerek esetén)

Szoftvertesztelési stratégiák

- ❖ Szoftver tesztelése **teljes egészében**
- ❖ Szoftver tesztelése **kisebb részenként**
 - ❖ modulok csoportjai
 - ❖ integrációs tesztek
 - ❖ **inkrementális tesztelés** (modulok lépésenkénti hozzáadása teszteléskor)

Inkrementális tesztelés

- ❖ Tételezzük fel, hogy a szoftver egészét **hierarchikus** modulok építik fel
 - ❖ Lépésenkénti integrálási stratégiák (modulok hívási gráfja alapján)
 - ❖ **fentről lefelé (top-down)**
 - ❖ **lentől felfelé (bottom-up)**
 - ❖ Kevésbé tiszta hierarchia esetén
 - ❖ **szendvics integrálás**
 - ❖ páronkénti integrálás
 - ❖ szomszédsági integrálás

Függőségi mátrix

- ❖ Dependency Structure Matrix (Design Structure Matrix)
- ❖ Szoftver modulok, komponensek egymástól való függésének (hivatkozás, hívás) leírására
- ❖ Rétegzettség vizsgálatára
- ❖ Integrálási stratégia meghatározásához

		1	2	3	4
Task A	1	.		X	X
Task B	2		.	X	
Task C	3	X		.	X
Task D	4				.

Figure 1: A Simple DSM

		1	2	3	4
Task D	1	■			
Task A	2	X	■	X	
Task C	3	X	X	■	
Task B	4			X	■

Figure 2: Block Triangular DSM after Partitioning

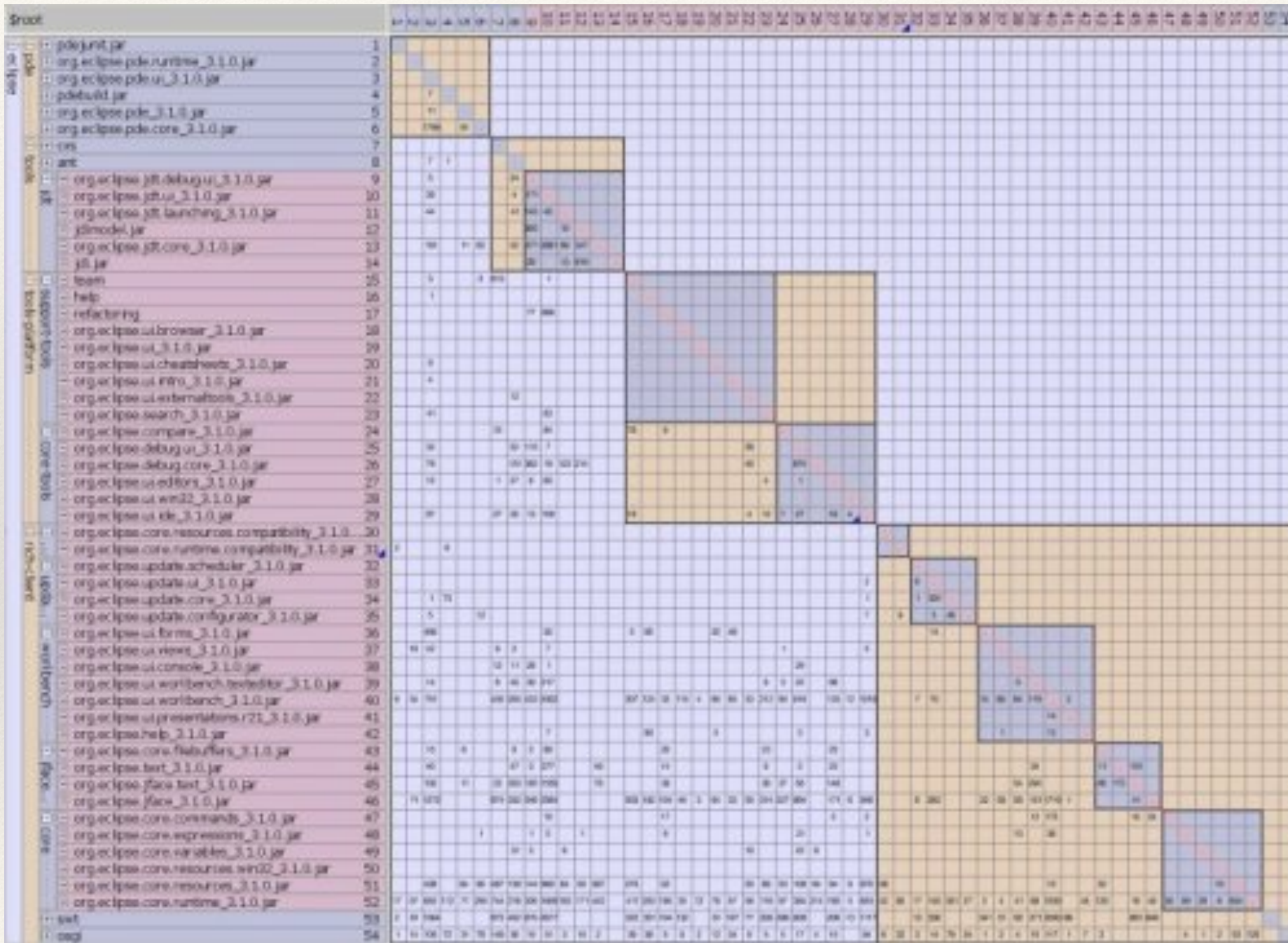
		1	2	3
Task D	1	■		
Task A-C	2	X	■	
Task B	3		X	■

Figure 3: Lower Triangular DSM

		1	2	3	4
Task D	1	■			
A-C	Task A	2	X	■	X
	Task C	3	X	X	■
Task B	4			X	■

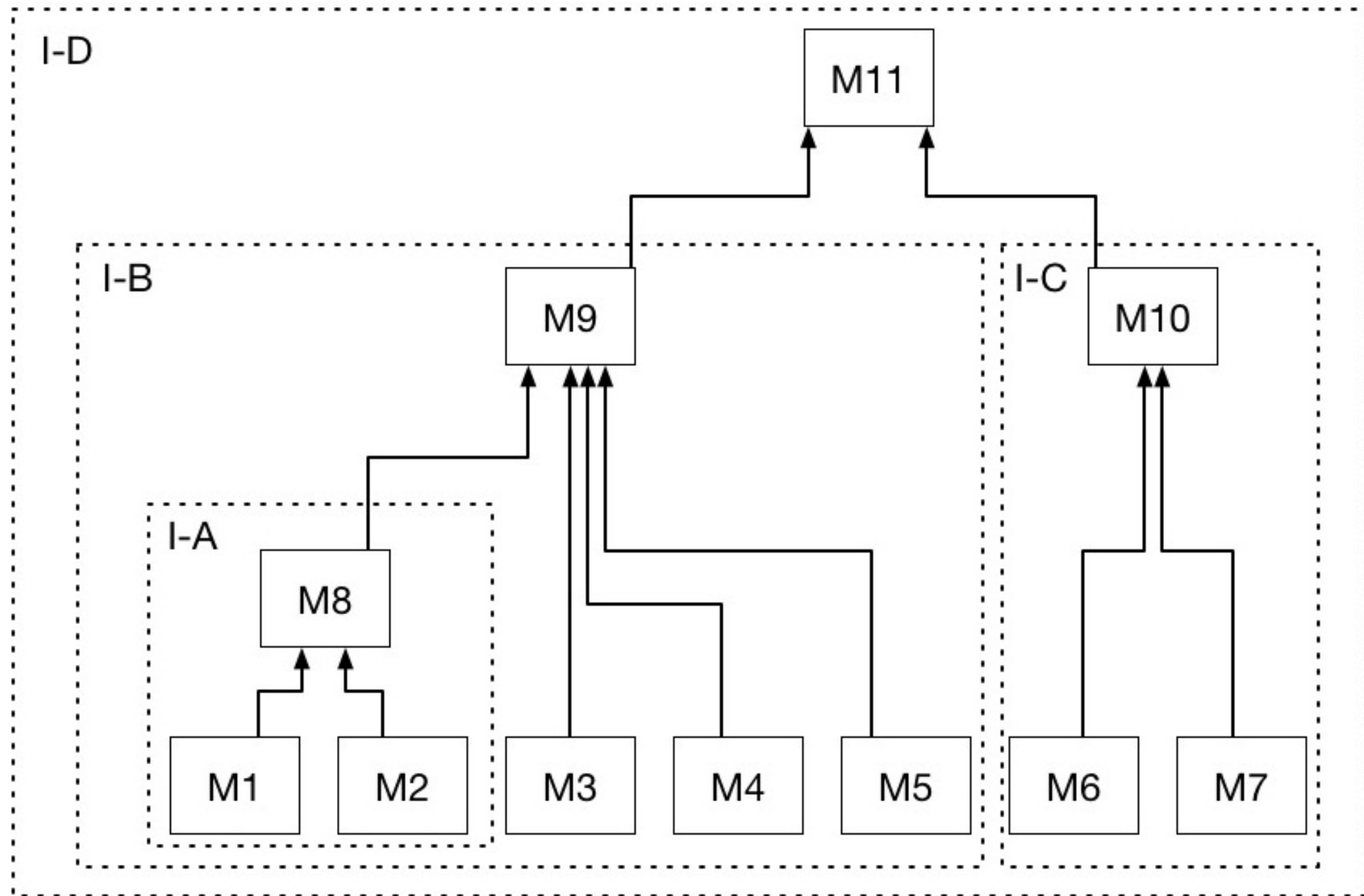
Figure 4: Hierarchical DSM

Függőségi mátrix (pl.)

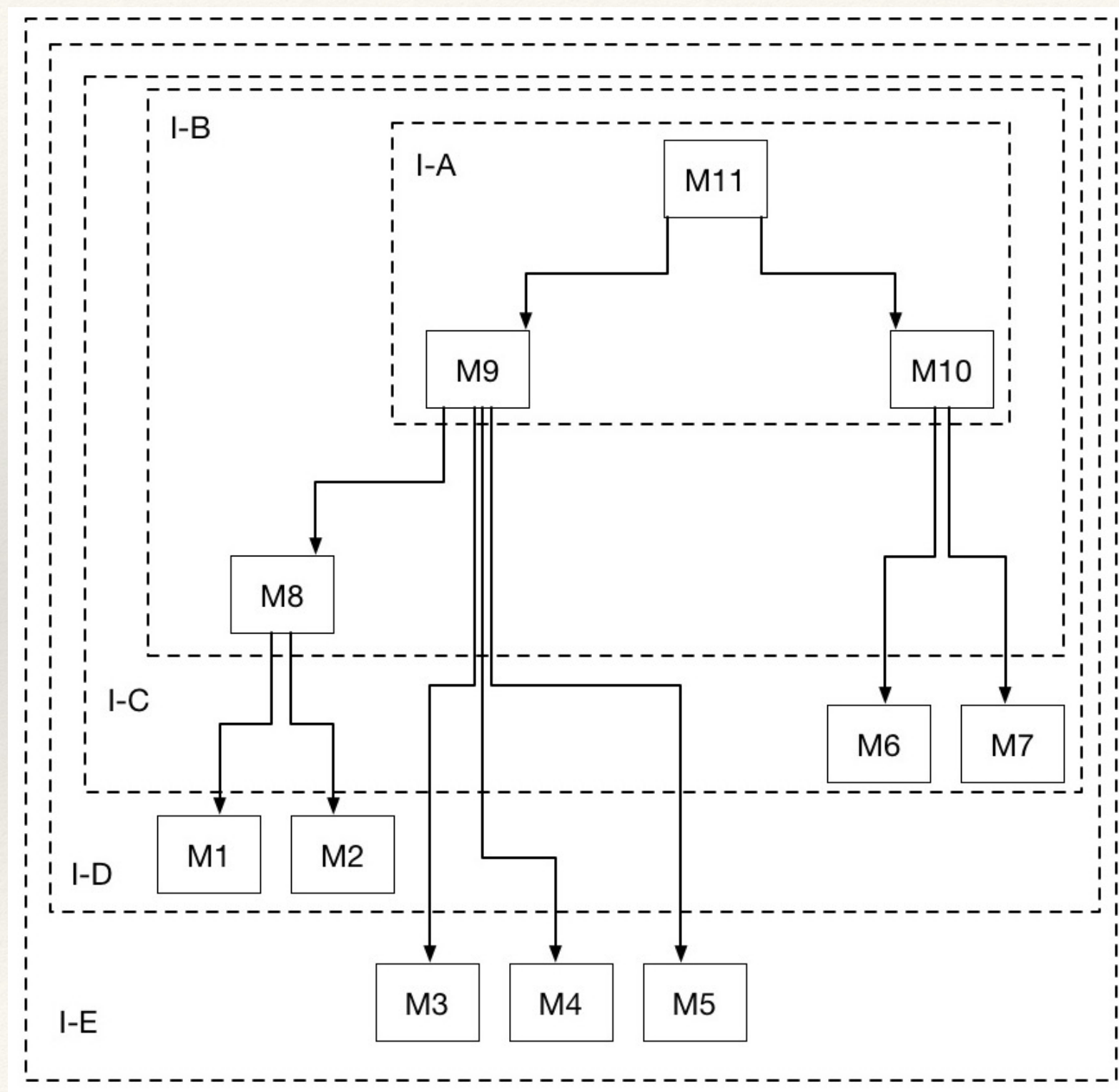


Eclipse platform (Version: 3.1.0, Build id: I20050627-1435)

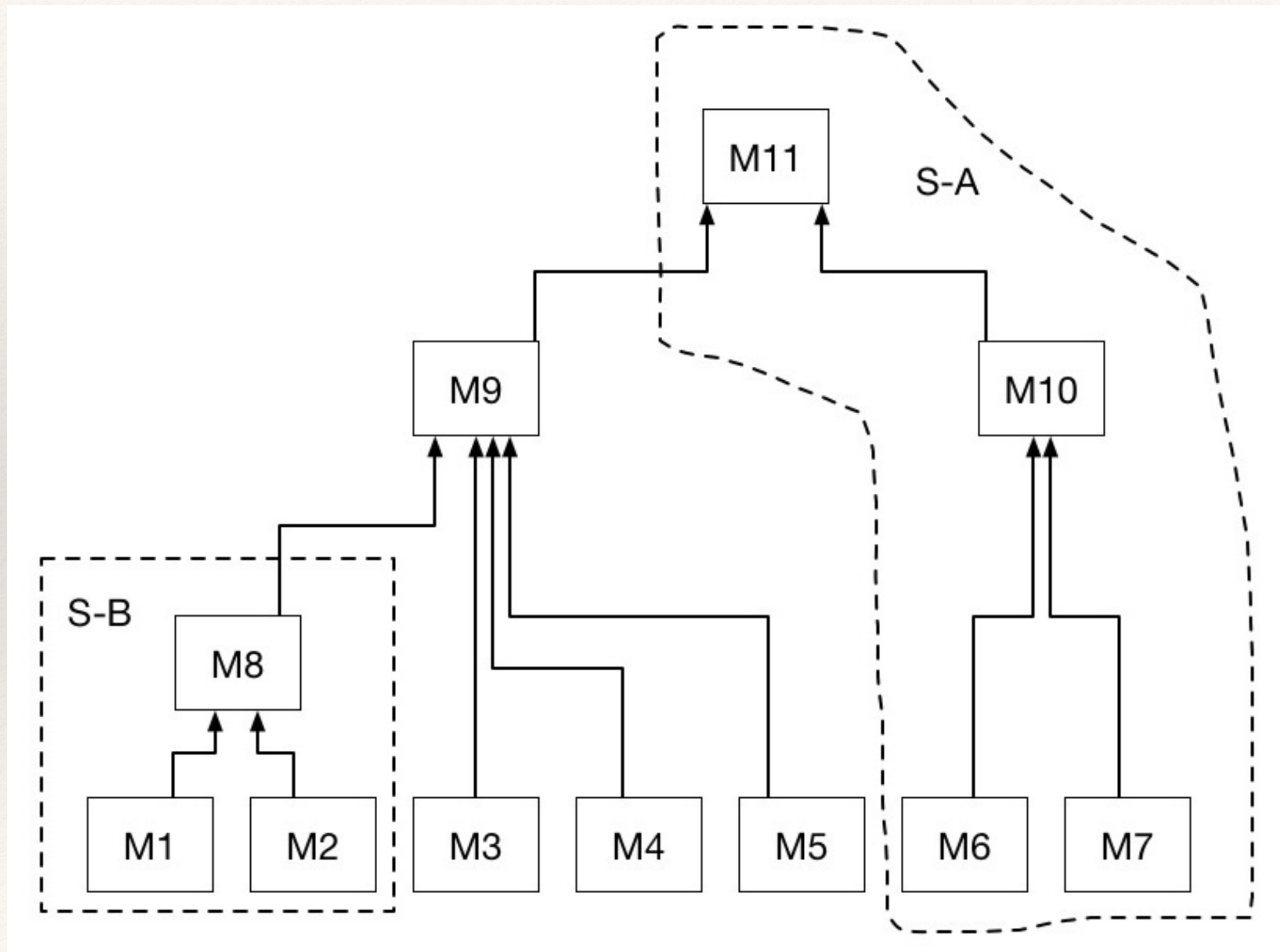
Lentről felfelé integrálás



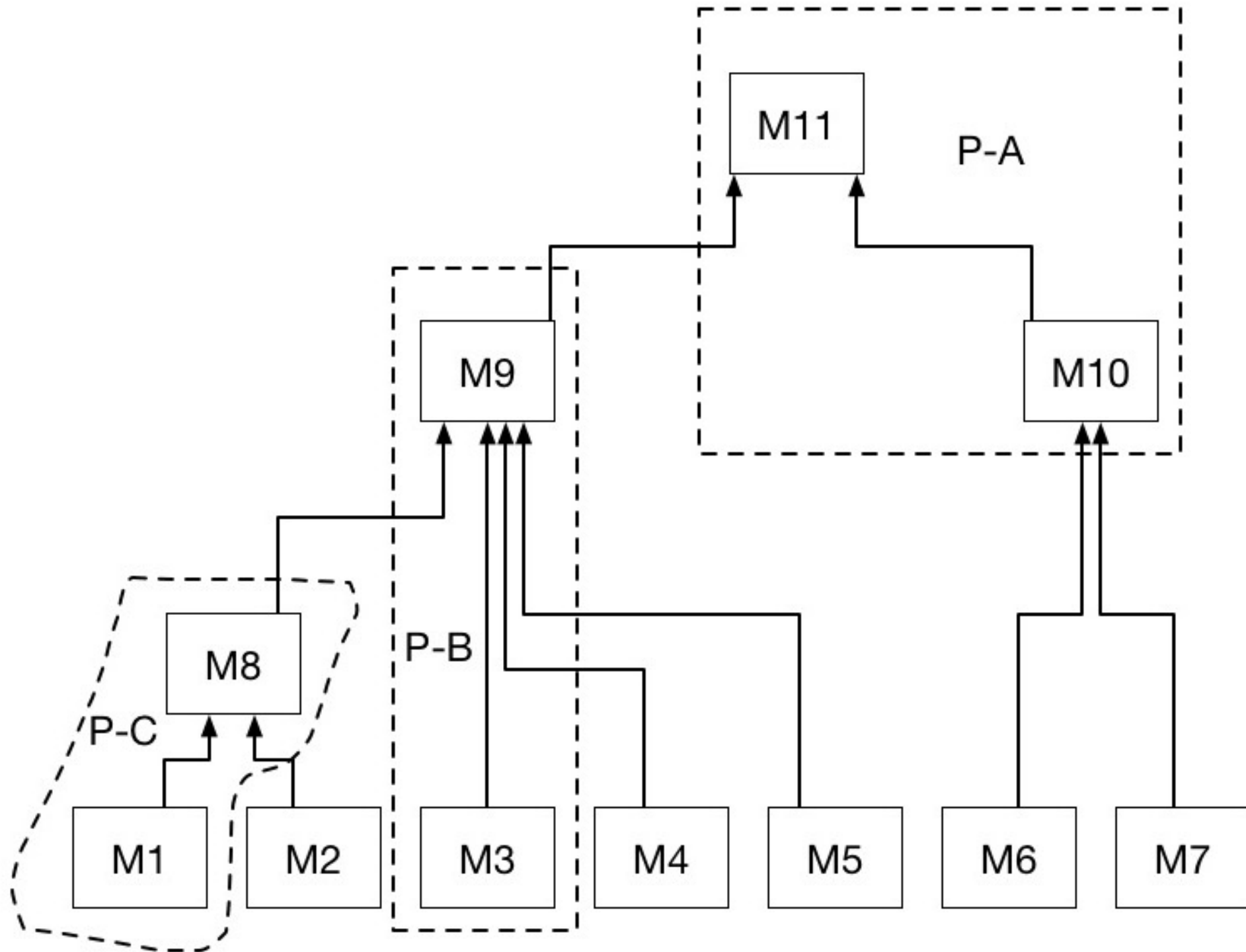
Fentről lefelé integrálás



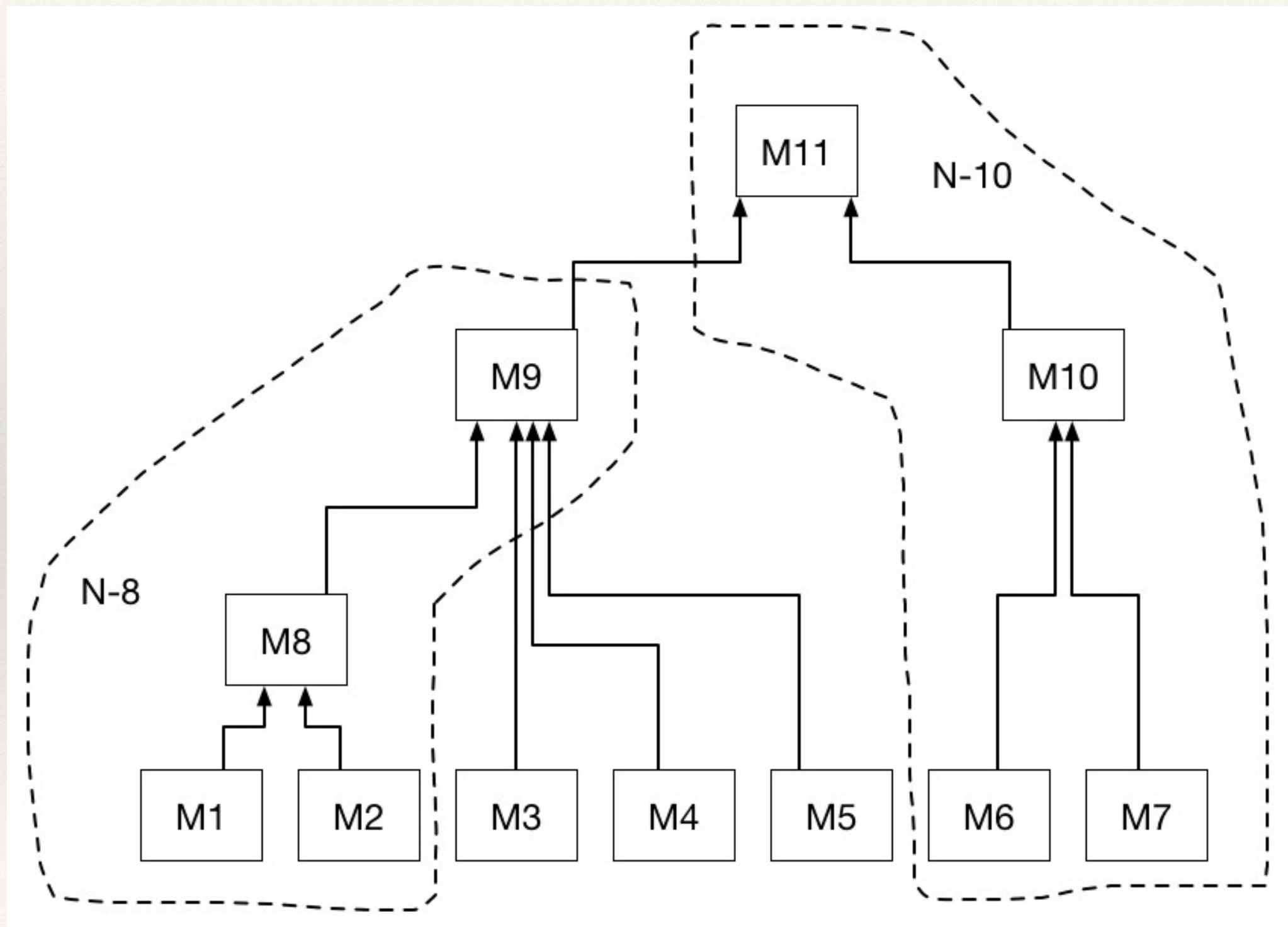
Szendvics integrálás



Páros integrálás



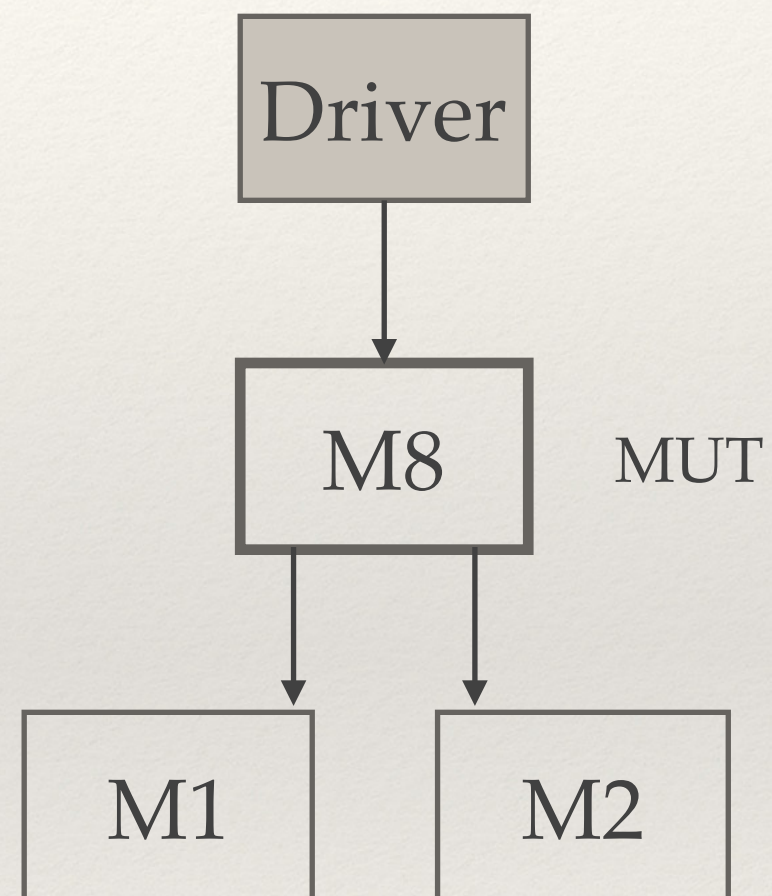
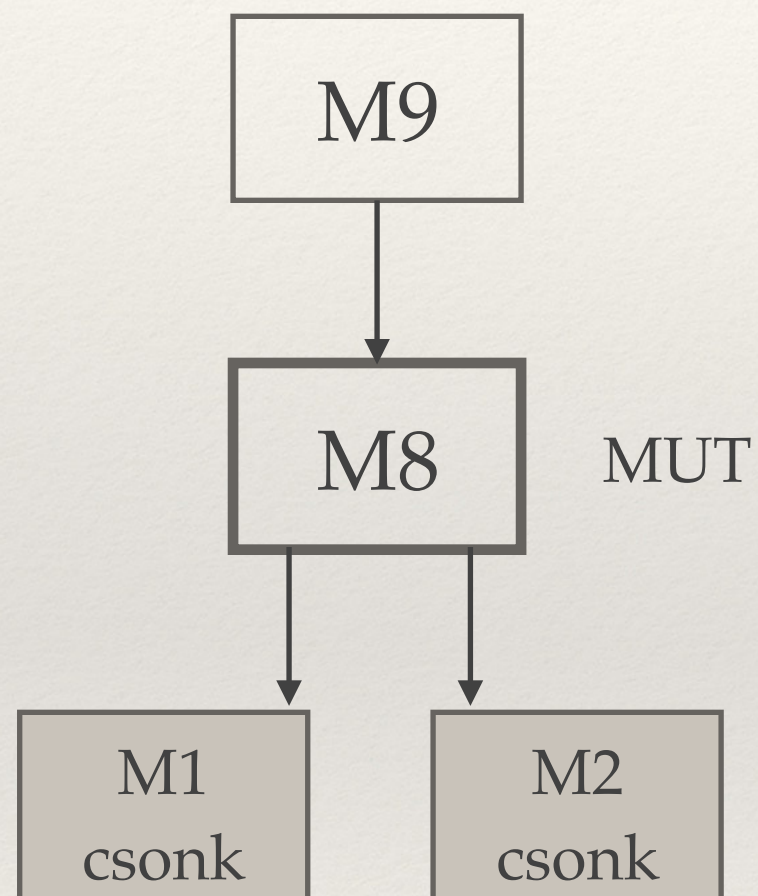
Szomszédsági integrálás



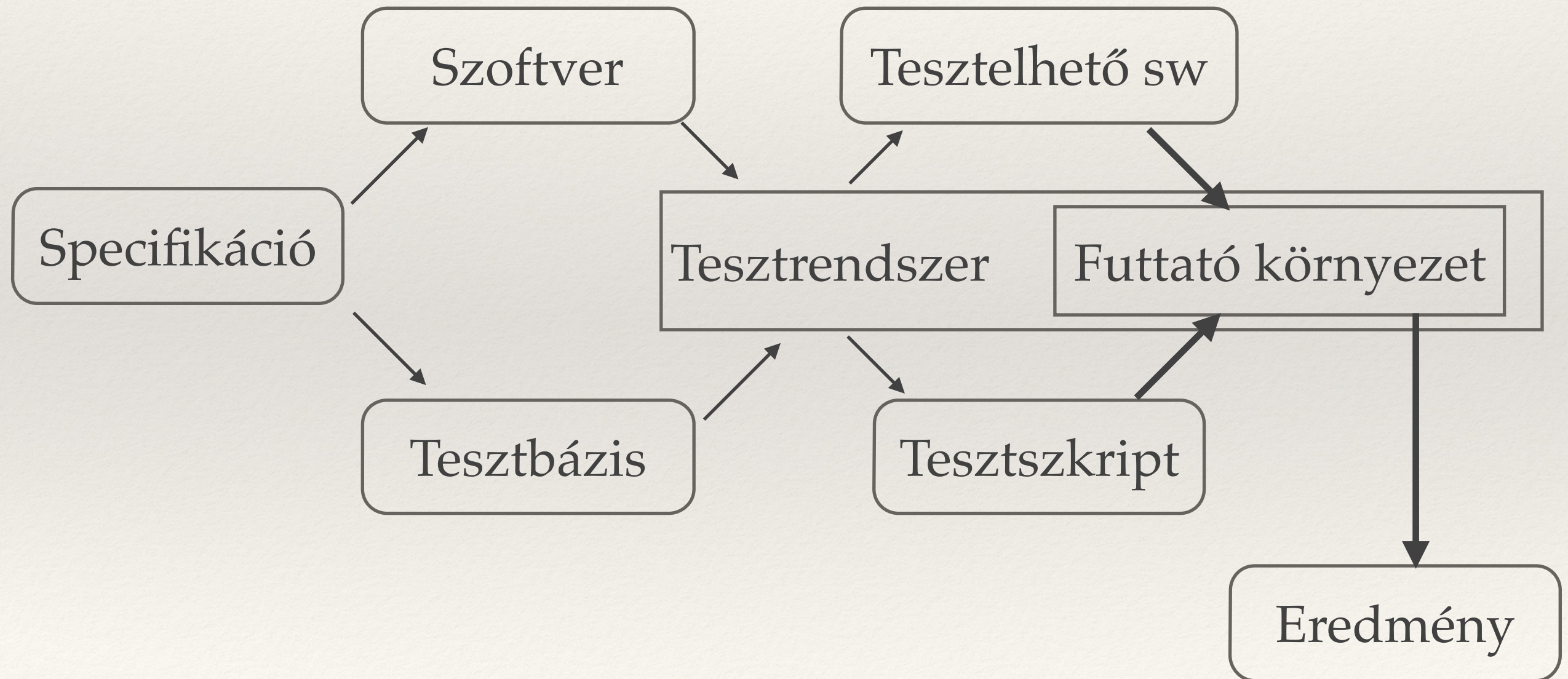
Meghajtó és csonkok

- ❖ Nem használható modulok helyettesítésére
- ❖ **Csonk (stub)**
 - ❖ alacsonyabb szintű modulok helyettesítésére
 - ❖ hiba injektálási lehetőség
 - ❖ hardver szimuláció
- ❖ **Meghajtó (driver)**
 - ❖ magasabb szintű, hívó modul helyettesítése (futtatás)
 - ❖ input bevitele
 - ❖ kimenet begyűjtése

Meghajtó és csonkok (folyt.)



A teszt (végrehajtó) környezet



A teszt (végrehajtó) környezet

