

Curs 4
Maparea conceptelor în
Rețele Petri

Cuprins

- De ce rețele Petri?
- Concepte ale Workflow-urilor
- Modelarea Proceselor cu Rețele Petri
- Exemplu: Tratarea unei plangeri
- Rutarea
- Task- work item- activity
- Triggere

De ce modelare cu rețele Petri?

- Avantaje:
 - – Tehnica grafică ușor de aplicat
 - – Concepte ilustrate elegant folosind rețele Petri
 - – Baza formală: semnificația fiecărui proces definită precis (ex. Existența 2 OR-splituri)
 - – Stările sunt indicate precis (diferența dintre OR-splituri implicite și explicite)
 - Soluții pentru dezavantaje:
 - Rețele Petri clasice + extensii
- de culoare
de timp
ierarhice

Concepte ale Workflow-urilor

- Cazul

- Task-ul

- Conditia

- Trigger-ul


- Rutarea

- **Procesul**

Modelarea Proceselor

- Proces \sqsubset {taskuri, conditii, subprocese} => defineste ordinea de executie a task-urilor

Element proces	Mapare retea Petri
-Task	-Tranzitie
-Conditie	-Loc
-Aparitia Cazului	-Token in Locul Start
-Finalizarea Cazului	-Token in Locul End



Cerinte proces:

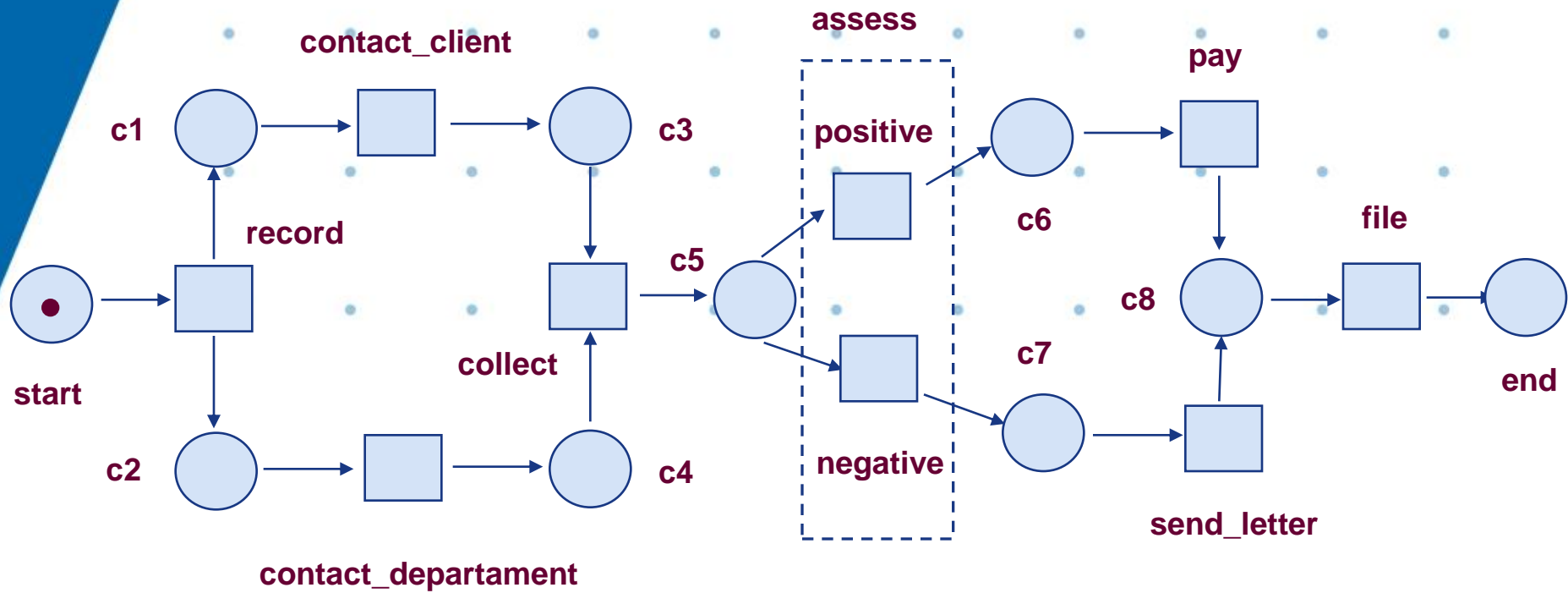
1. In orice moment este posibil sa se ajunga –dupa executia unui nr de taskuri- la starea in care este un token in locul End
2. Cand un token e in End, nici un alt token nu mai este in alt loc => finalizare dupa executia tuturor task-urilor

Exemplu: Tratarea unei plangeri

Etape

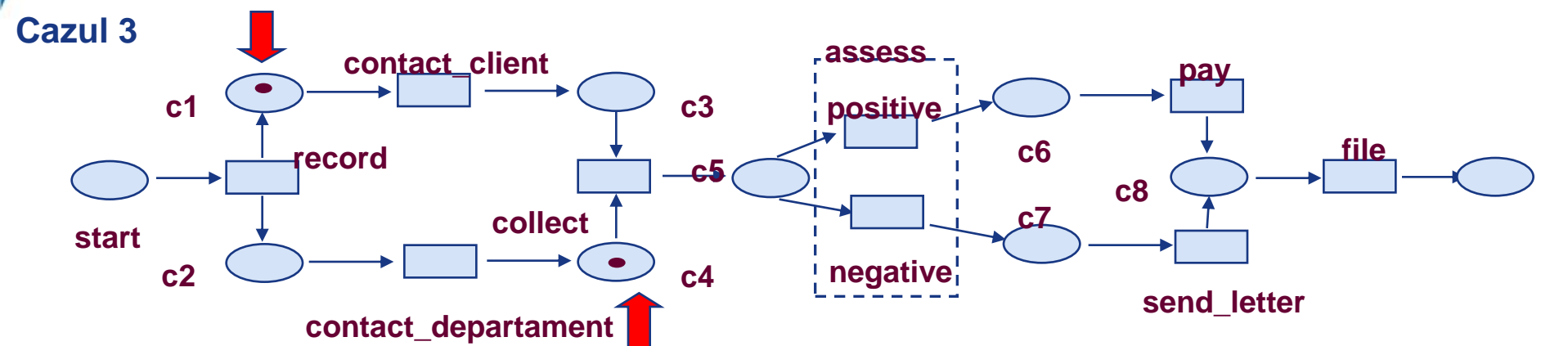
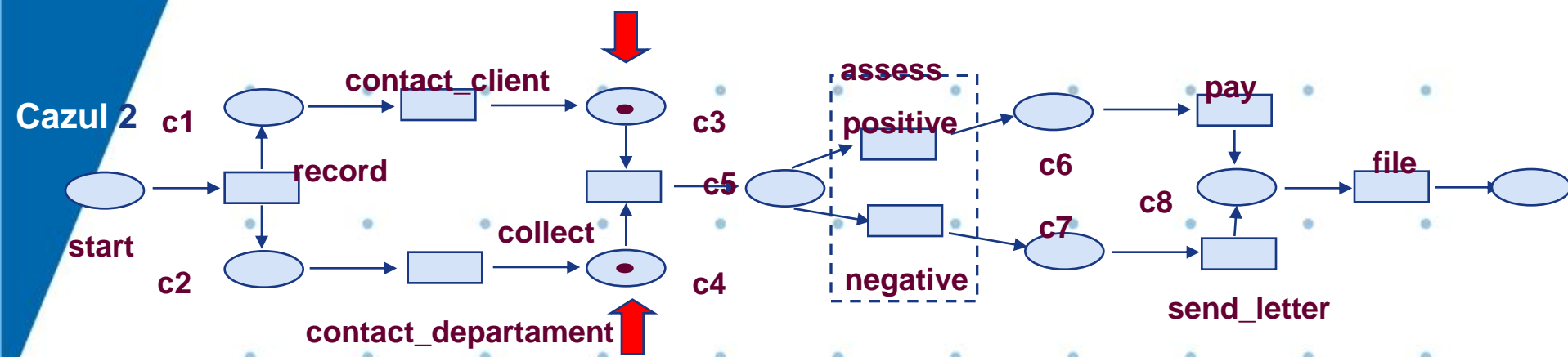
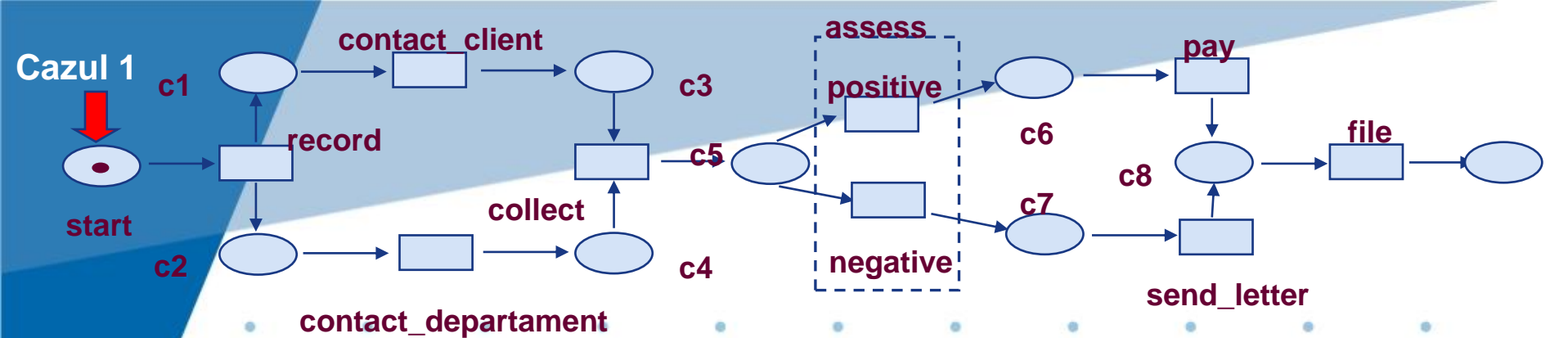
- 1. Inregistrarea plangerii
- Taskuri paralele** {
 - 2. Contactarea clientului care a depus plangerea pentru a furniza detalii
 - 3. Instiintarea departamentului afectat si e intrebat de reactia initiala
- 4. Strangerea datelor si luarea unei decizii:
 - Taskuri selective** {
 - platirea unei compensatii
 - trimiterea unei scrisori clientului
- 5. Arhivarea plangerii.

Modelare cu rețele Petri

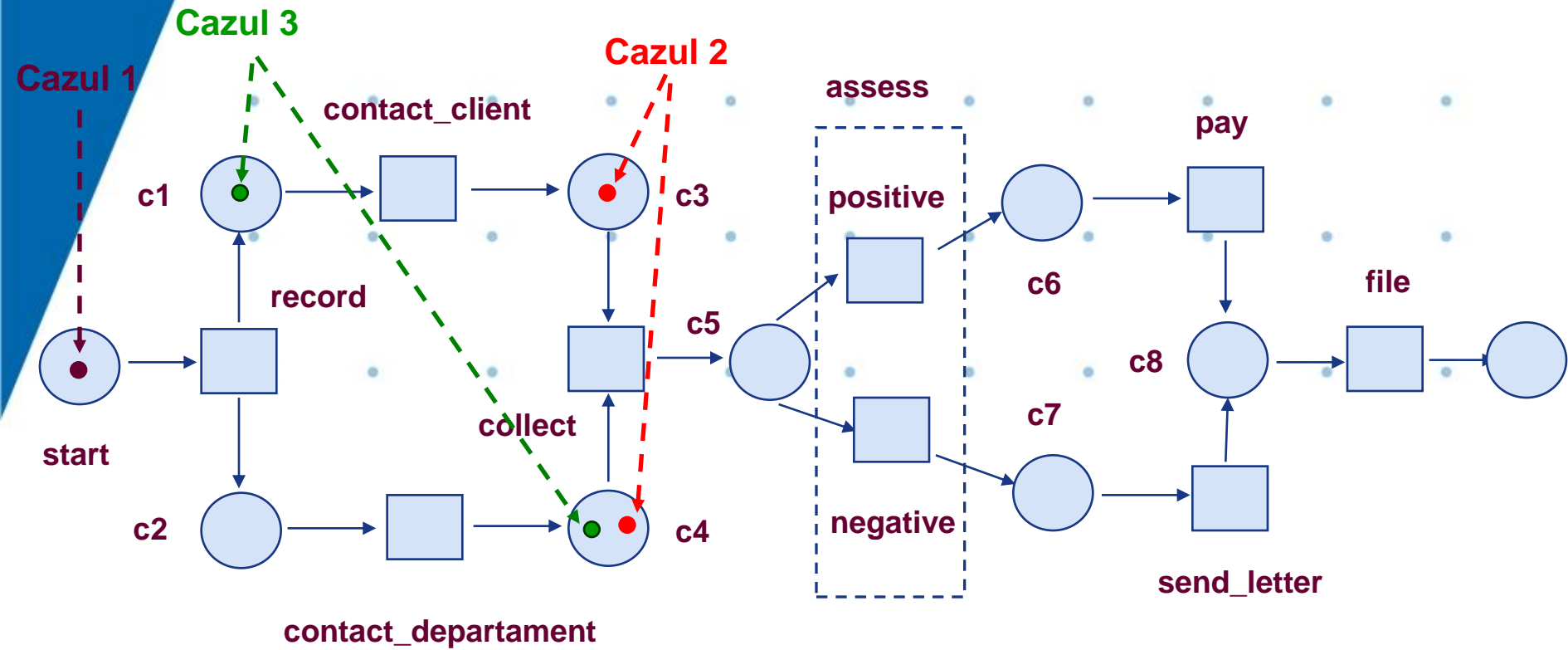


Tratarea cazurilor particulare intr-un proces

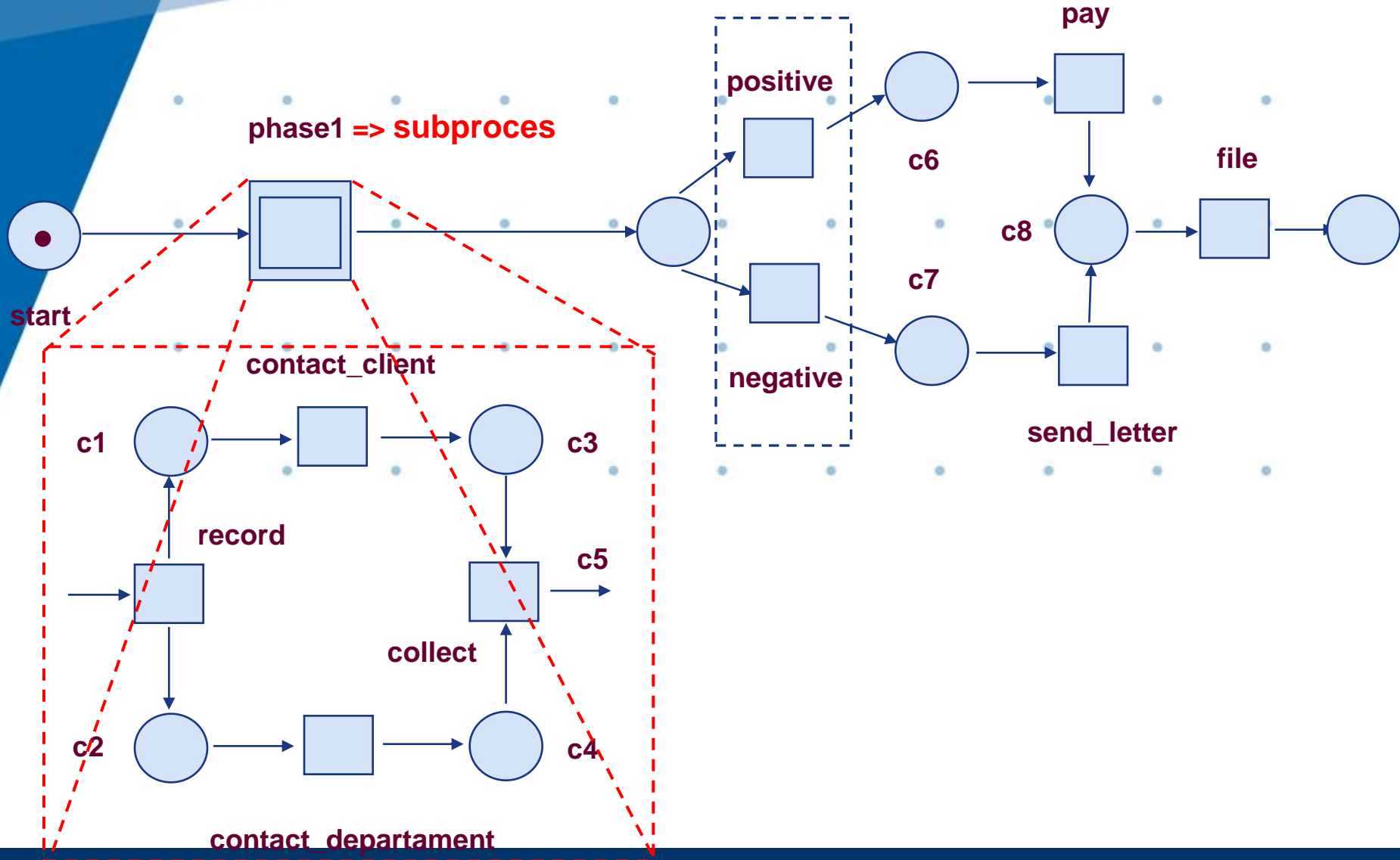
- Procesul modeleaza o **categorie** de cazuri.
 - Tokenurile care corespund cazurilor particulare sunt tinute separate => nu pot interfera
 - Abordari:
 - 1. Existenta unei replici a retelei Petri initiale pentru fiecare caz in evolutie.
 - 2. Utilizarea retelelor Petri extinse pe culori.
- OBS!! Fiecare tranzactie contine preconditionia ca numai tokenuri din acelasi caz pot fi consumate la executie.



Retele Petri extinse – color extend



Retele Petri ierarhice

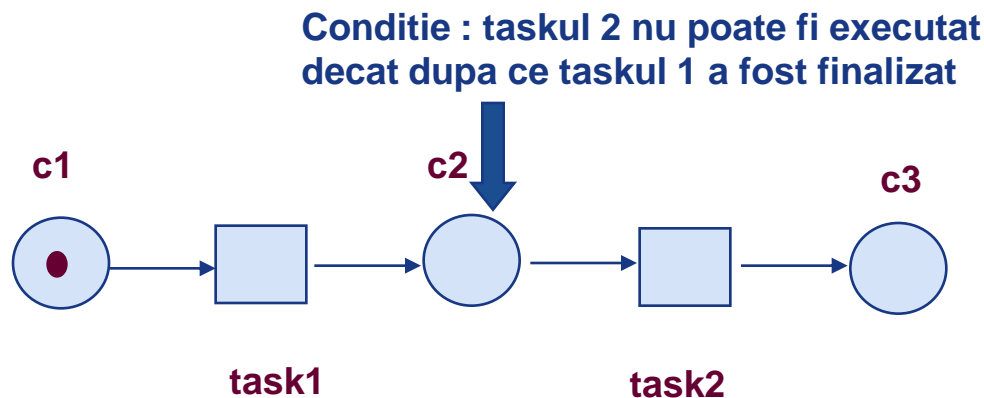


Rutarea

- Tipuri de rutare
 - 1. secventiala
 - 2. paralela
 - 3. selectiva
 - 4. iterativa

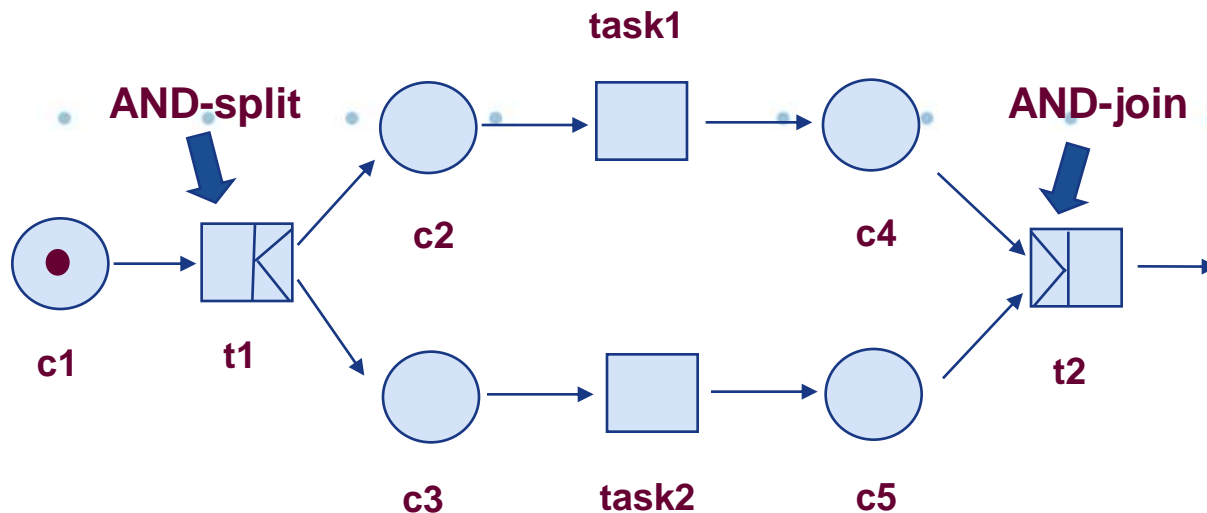
1. Rutarea secventiala

- Taskurile trebuie executate secvential
- Exista interdependenta intre ele => outputul primului poate fi intrare pentru urmatorul task
- **Corespondenta modelare Petri : legarea a doua tranzitii printr-un loc**



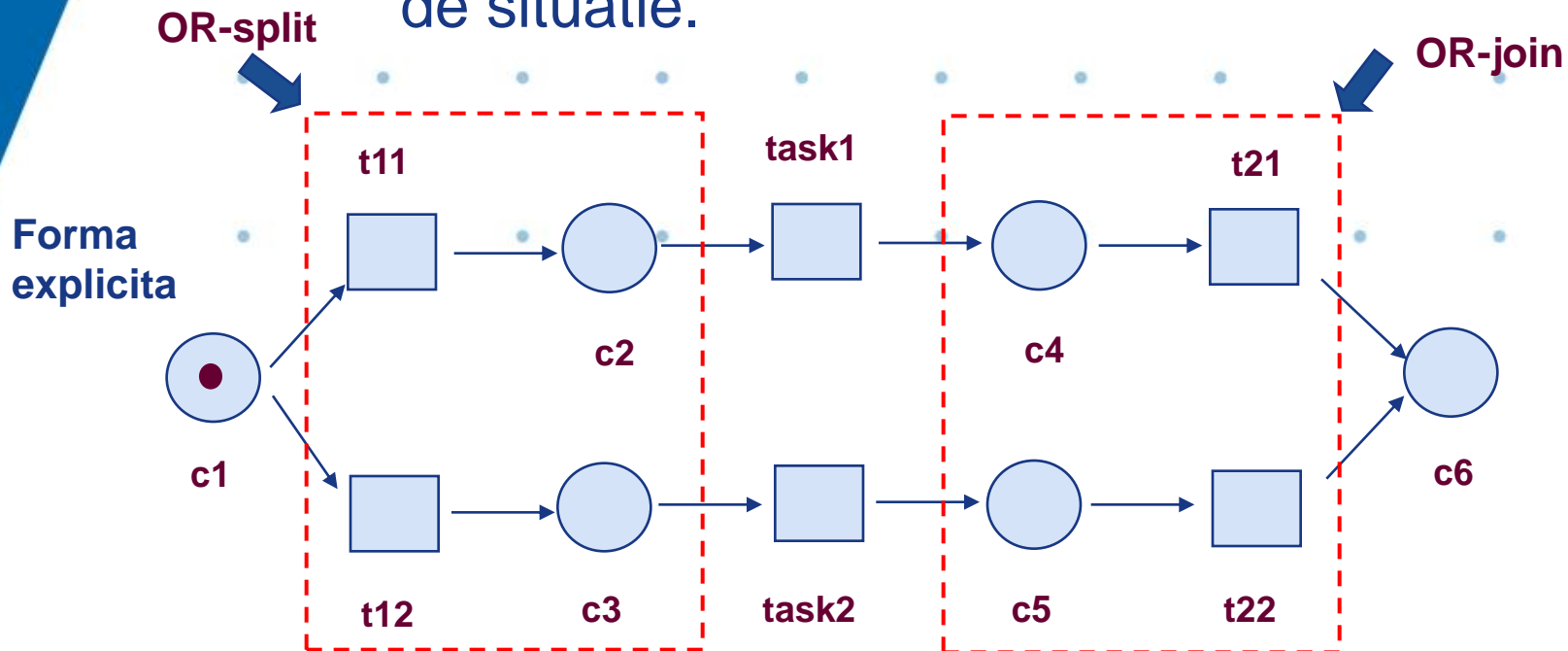
2. Rutarea paralela

- Mai multe taskuri pot fi executate simultan sau in orice ordine
- **AND-split** si **AND-join** = management tasks

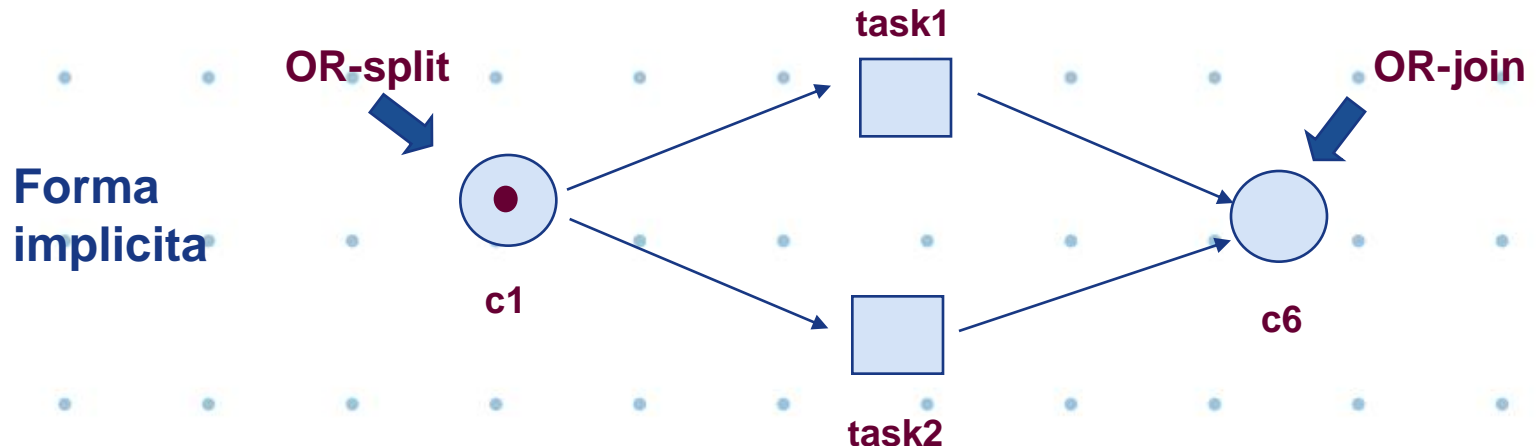


3. Rutarea selectivă

- Intr-un proces, rutarea poate fi lasata cazurilor specifice, cai diferite putand fi urmate in functie de situatie.



3. Rutarea selectiva 2

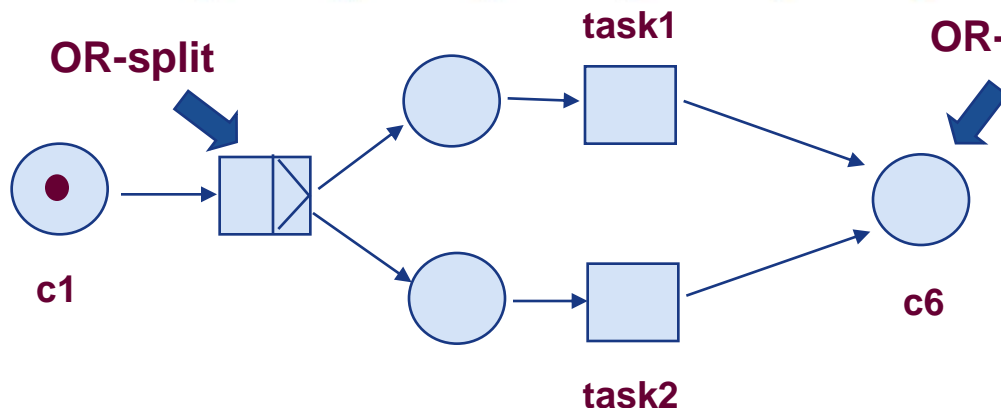


Diferente intre forma implicita si explicita - Modelarea OR-split :

- modelarea explicita => decizia asupra caili se ia in momentul in care exista jeton in c1
- modelarea implicita => decizia se ia mai tarziu, cand trebuie executat fie task1, fie task2

3. Rutarea selectiva 3

- Cazurile anterioare –alegerea intre cele 2 cai este nondeterministica => la latitudinea **mediului workflowului**
- Pentru ca decizia sa se ia in conformitate cu attributele cazului => extensia culorii tokenului



1. Regula de decizie

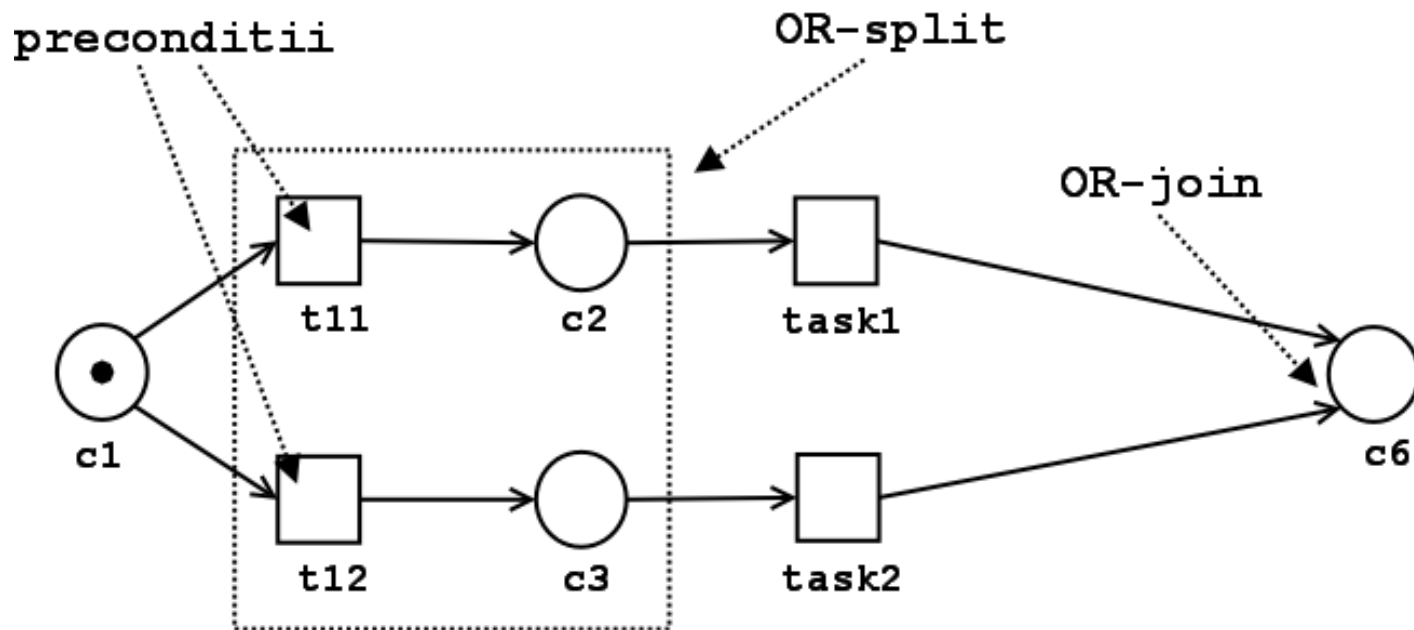
- valoarea jetonului **c1**
- regula de decizie **t1**

2. Preconditii

- Bazate pe culoarea tokenului de consumat => gardieni ai tranzactiilor

3. Rutarea selectiva 4

- În figura precedentă numărul jetoanelor produse de către tranziția $t1$ este variabil (0 sau 1)
- Putem implementa această alegere și prin folosirea unor precondiții (de tip restrictiv)



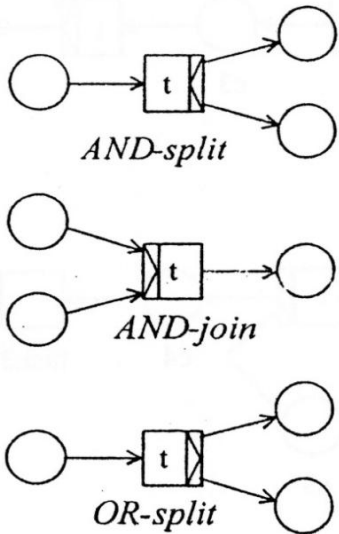
3. Rutarea selectiva 5

- În reprezentarea fără precondiții alegerea era efectuată pe baza atributelor de caz ale jetonului din $c1$ și a regulii decizionale din $t1$
- Precondițiile din $t11$ corespund cerințelor necesare pentru a executa $task1$, iar cele din $t12$ corespund cerințelor necesare pentru a executa $task2$
- Dacă precondiția din $t11$ reprezintă de fapt negarea dpdv logic a celei din $t12$, atunci cele două reprezentări sunt echivalente

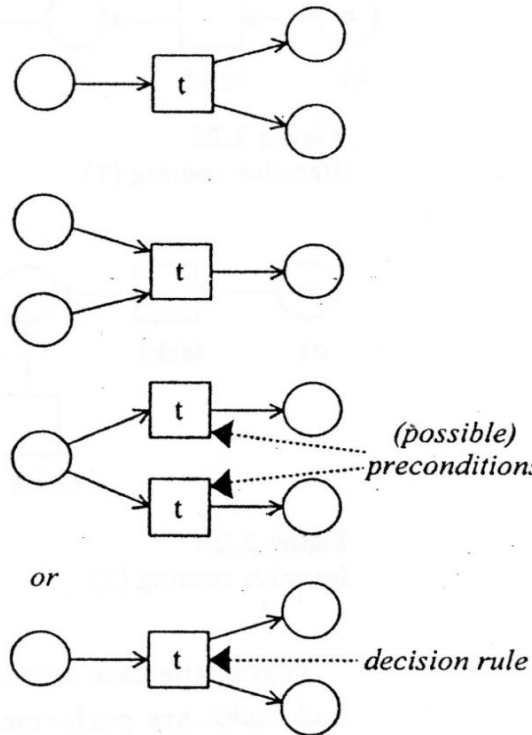


3. Rutarea selectiva 6

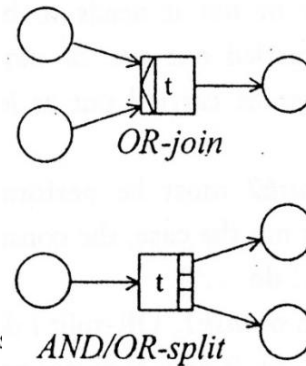
Notation



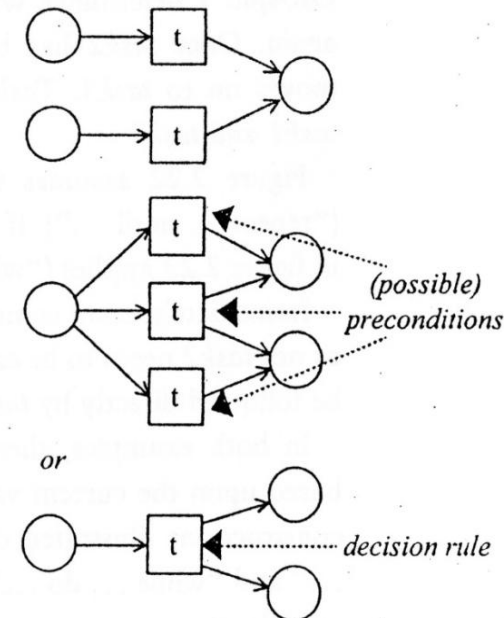
Meaning



Notation



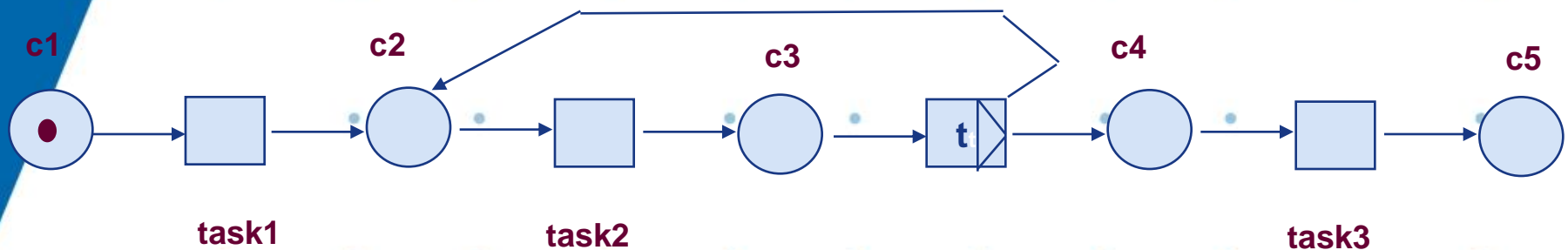
Meaning



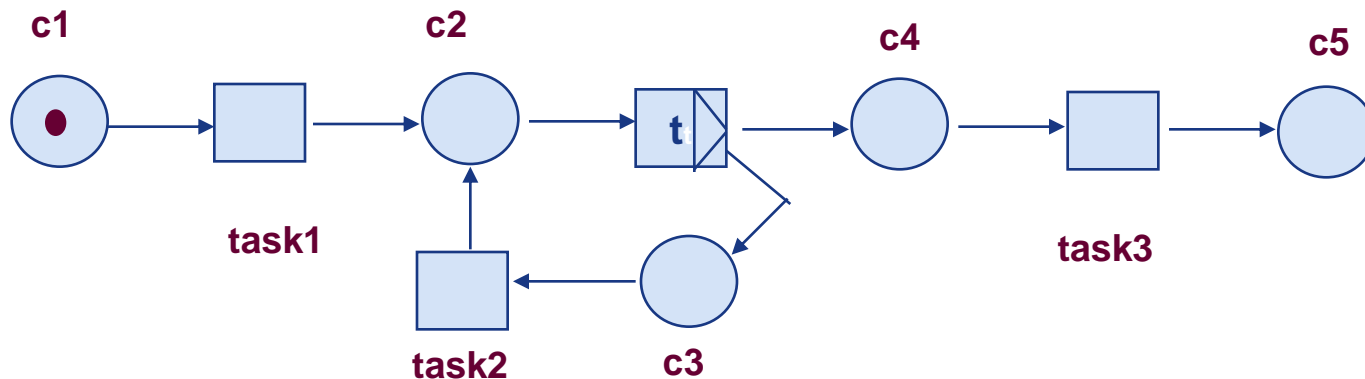
4. Rutarea iterativa

- Executia repetata a unui task particular

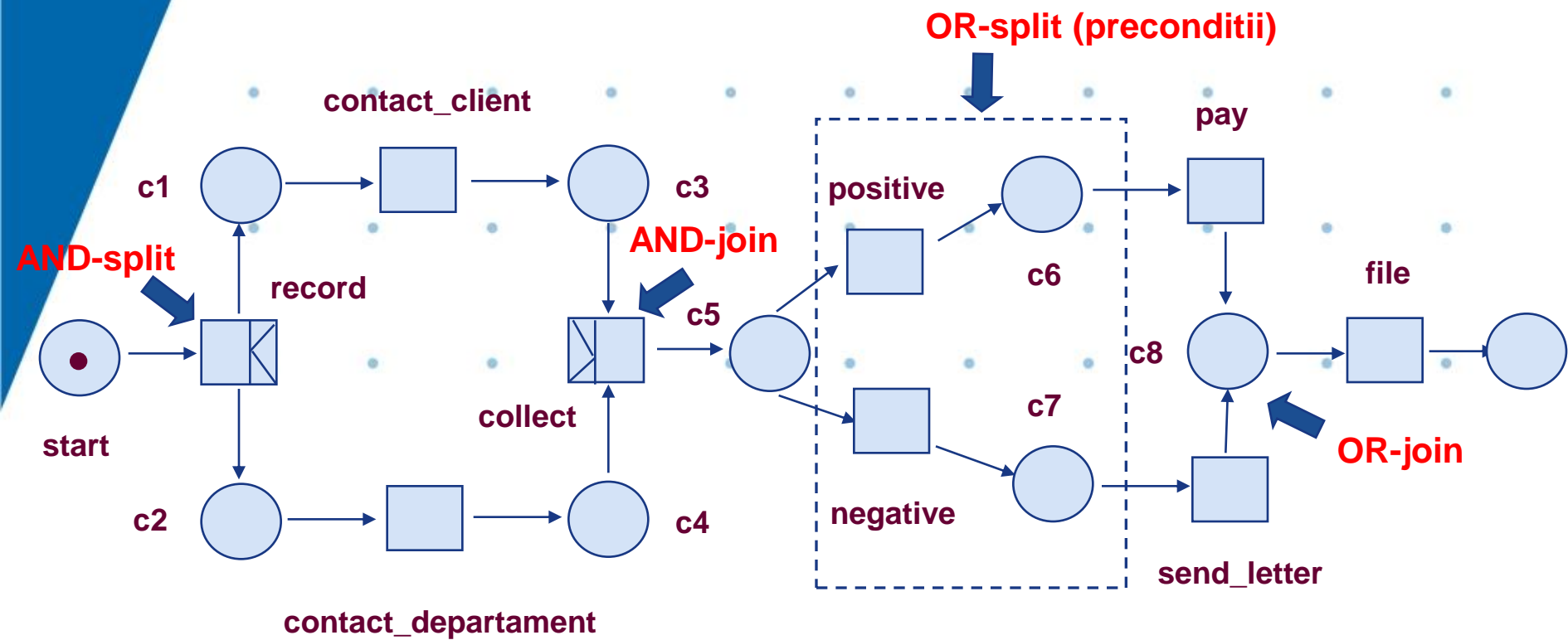
Repeat...until – task2 se executa cel putin o data



While...do – OR-split determina daca se executa task2 sau nu



Exemplu rutari



Punerea în aplicare

- Procesul: taskuri, condiții, subprocese și relații
- Un proces este conceput să se ocupe de o categorie particulară de cazuri.
- Un task nu este specific unui anumit caz. Când un caz este realizat de proces, taskurile sunt efectuate pentru el.
- Pentru a evita confuziile dintre task și performanțele sale pe un caz anume: “**element de lucru**” și “**activitate**”.



Punerea în aplicare

- Un *element de lucru* este o combinație între caz și un task care urmează a fi efectuată.
- Termenul de *activitate* se referă la performanțele unui element de lucru.
- Când un element de lucru este în faza de execuție el se transformă într-o activitate.
- Trebuie precizat că față de un task, atât elementul de lucru cât și o activitate sunt legate de un anumit caz.

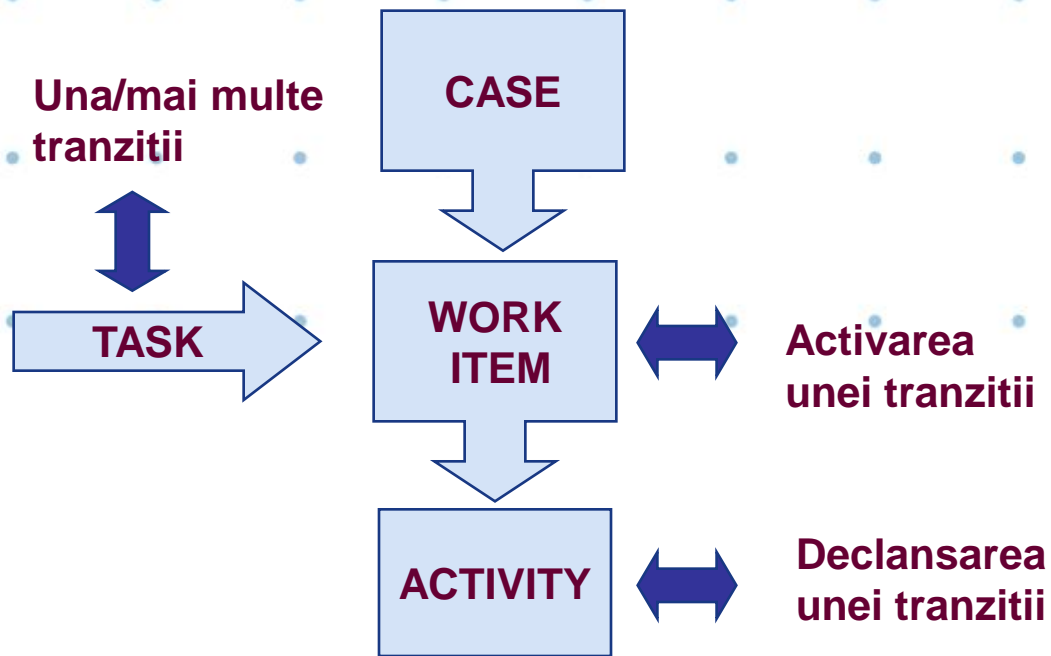


Punerea în aplicare

- Distincția dintre un task, un element de lucru și o activitate devine foarte clară când translatăm totul în limbajul rețelelor Petri.
- Un task corespunde cu una sau mai multe tranziții.
- Un element de lucru corespunde cu o tranziție ce se poate executa.
- Activitatea este asociată cu acțiunile asociate unei tranziții.



Task- work item- activity



Punerea în aplicare

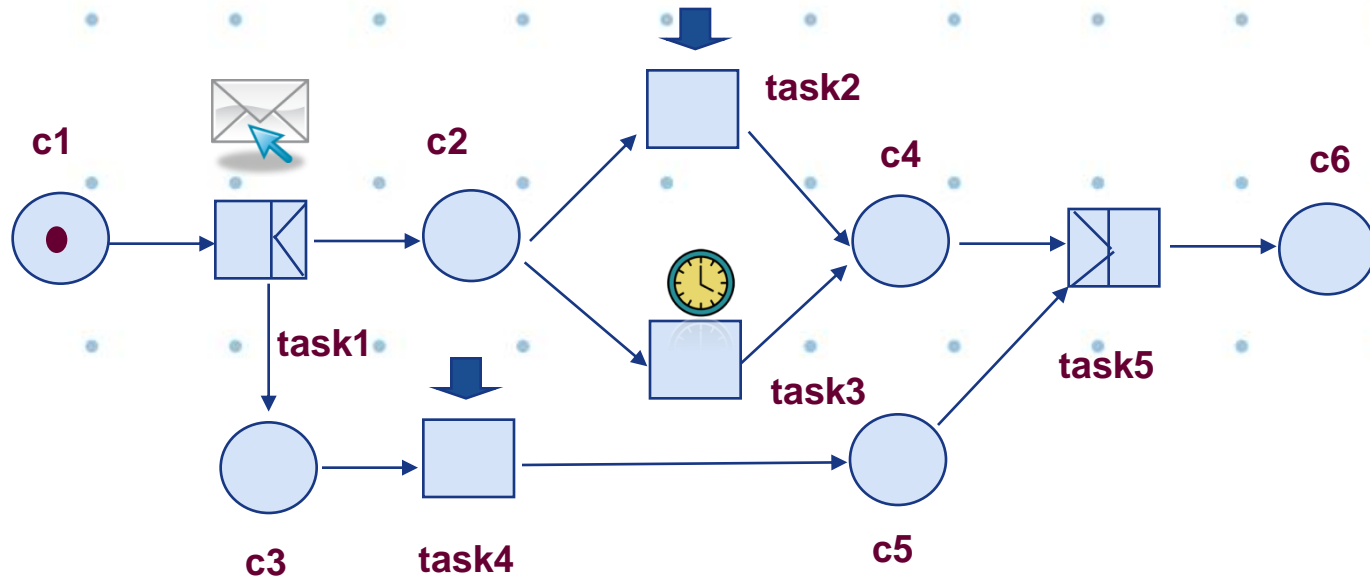
- Tranzițiile în rețele Petri sunt “nerăbdătoare”: se execută instantaneu după ce au fost activate.
- Un element de lucru este realizat doar când o persoană a luat inițiativa. Acesta este motivul pentru care recunoaștem existența declanșării.
- Anumite elemente de lucru se transformă în activități doar după ce au fost declanșate.



Triggere

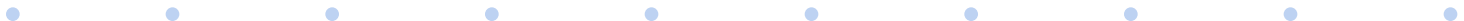
- Anumiti itemi de lucru nu pot fi transformati in activitati decat prin declansarea unui trigger.
- Tipuri- initiati de :
 - – 1. o resursa
 - – 2. un semnal de timp
 - – 3. un eveniment extern
- => mediul controleaza evolutia cazului, nu sistemul
- Fiecarei tranzitii unui task care necesita un trigger, i se adauga un loc de intrare. Un token in acest loc reprezinta triggerul.

Triggere 2



Punerea în aplicare

- Taskurile care sunt declanșate de o *resursă* sunt simbolizate folosind o **săgeată** cu fața în jos.
- Cele care sunt declanșate de un *eveniment extern* sunt simbolizate printr-un **pachet**.
- Iar cele care sunt *dependente de timp* sunt simbolizate cu un **ceas**.



Punerea în aplicare

- Mecanismul de declanșare indică că temporizarea unei alegeri OR-split este crucială.
- OBS: determinarea sincronizării dintre taskul 2 și taskul 3 este greu de făcut.
- Odată ce condiția c2 devine validă avem 2 posibilități:
 - execuția taskului 2 până la momentul de timp specificat în taskul 3
 - taskul 2 este în așteptare



Punerea în aplicare

- În primul caz se execută taskul 2 , iar în cel de al doilea se execută taskul 3
- Astfel posibilitatea de a alege între cele două este întârziată până în momentul în care se primește primul declanșator. Deoarece nu se știe ce task va fi activat, OR-splitul implicit al lui c2 nu poate fi înlocuit cu un OR-split explicit sub forma unor tranziții adiționale.



Punerea în aplicare

- OR-split:
 - implicit
 - explicit

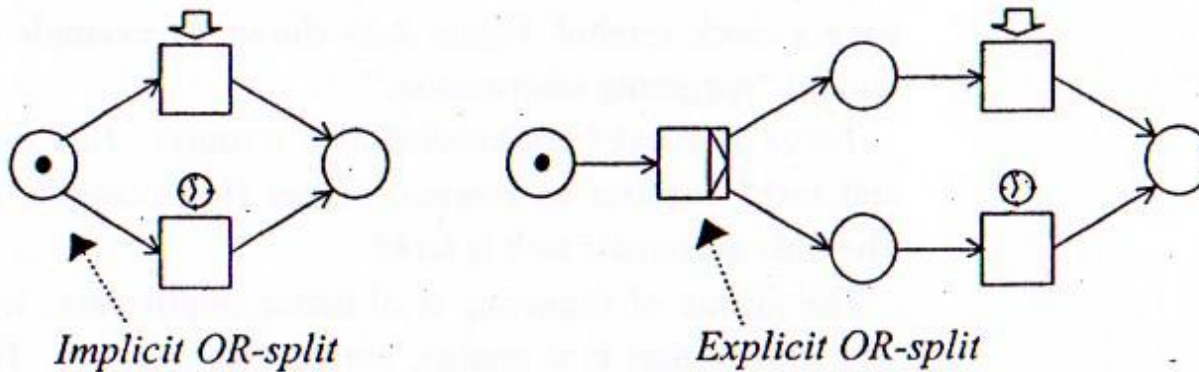


Figure 2.26

There is an essential difference between the implicit and explicit OR-split

Agentie de voiaj

Organizarea unei călătorii:

- Se înregistrează clientul.
- Un angajat caută soluții care sunt comunicate către client.
- Clientul va fi contactat pentru a valida cererea făcută pentru o călătorie și pentru alte alternative.
- Avem 3 posibilități:
 1. Clientul nu este interesat.
 2. Clientul ar dori să vadă alte alternative.
 3. Clientul selectează propunerea făcută .

Agentie de voiaj

- Dacă clientul selectează propunerea făcută atunci călătoria este rezervată.
- Se pregătesc una sau două tipuri de asigurări preplătite.
 - Clientul poate primi asigurări pentru pierderea bagajelor și/sau anularea călătoriei.
 - Clientul poate decide dacă are nevoie de asigurare și de ce tip.
- Documentele sunt trimise cu două săptămâni înaintea călătoriei.
- O călătorie se poate anula oricând.

Agentie de voiaj

