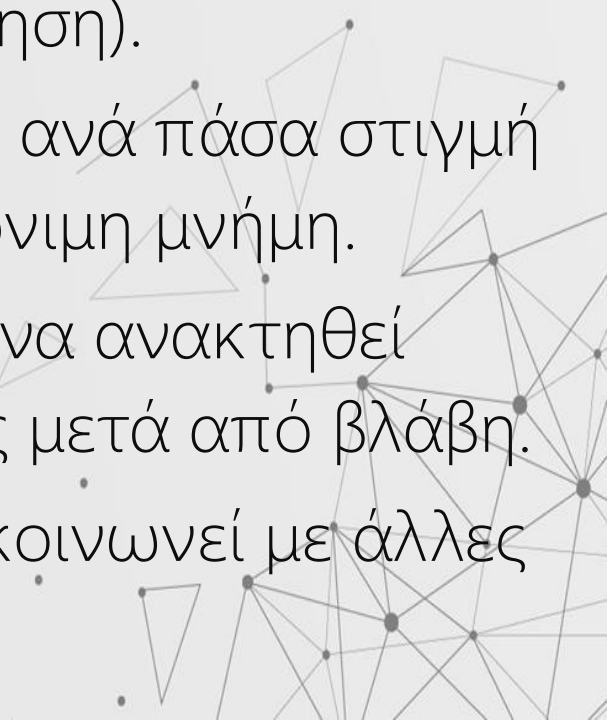


17

Κατανεμημένα Συστήματα

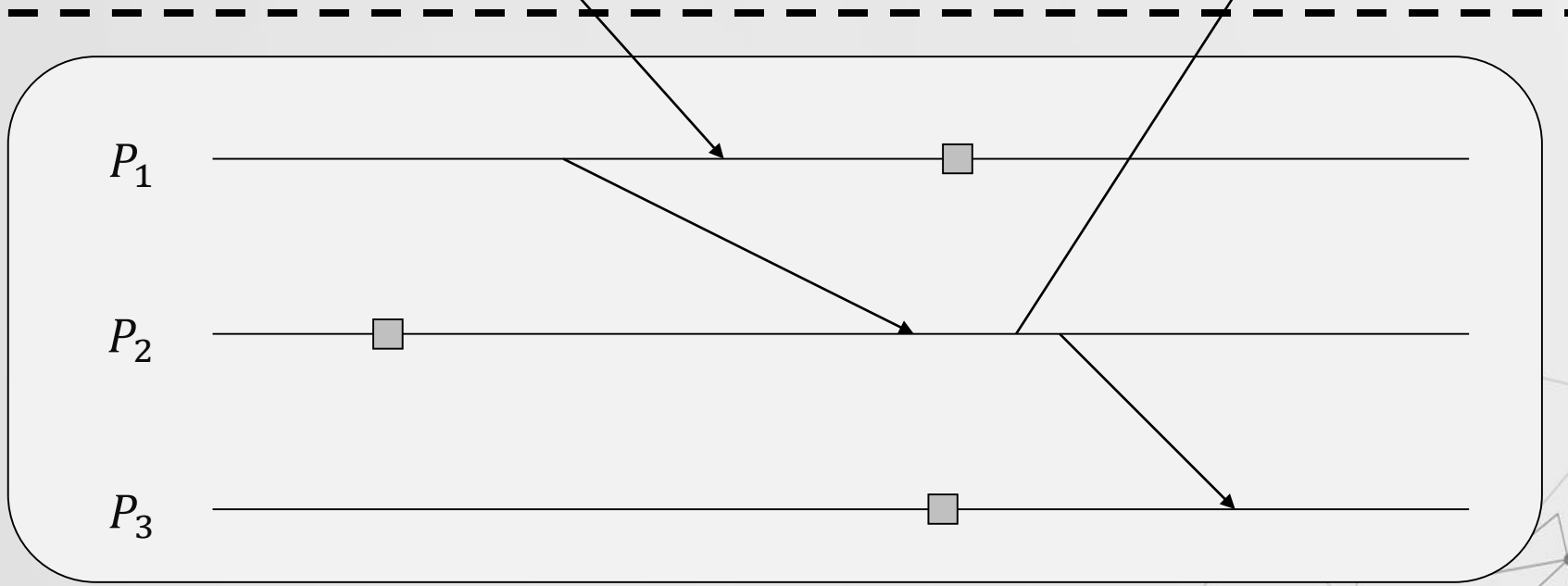
Ανοχή σε σφάλματα
(Fault tolerance)

Μοντέλο συστήματος

- Ένα κατανεμημένο σύστημα με N διεργασίες που επικοινωνούν μεταξύ τους με μηνύματα.
 - Θεωρούμε πως το δίκτυο δεν χάνει μηνύματα.
 - Οι διεργασίες εκτελούν μαζί κάποιον υπολογισμό.
 - Οι διεργασίες μπορεί να παρουσιάσουν βλάβες, αλλά και να ανακάμψουν από αυτές (επανεκκίνηση).
 - Οι διεργασίες μπορεί να καταγράψουν ανά πάσα στιγμή την (σημαντική) κατάσταση τους σε μόνιμη μνήμη.
 - Η καταγεγραμμένη κατάσταση μπορεί να ανακτηθεί κατά την επανεκκίνηση της διεργασίας μετά από βλάβη.
 - Το σύστημα/πρόγραμμα μπορεί να επικοινωνεί με άλλες «εξωτερικές» οντότητες.
- 
- A decorative network diagram in the bottom right corner, consisting of a series of interconnected nodes (dots) and lines, forming a complex web-like structure.

Μοντέλο συστήματος

εξωτερικό
περιβάλλον



κατανεμημένο σύστημα

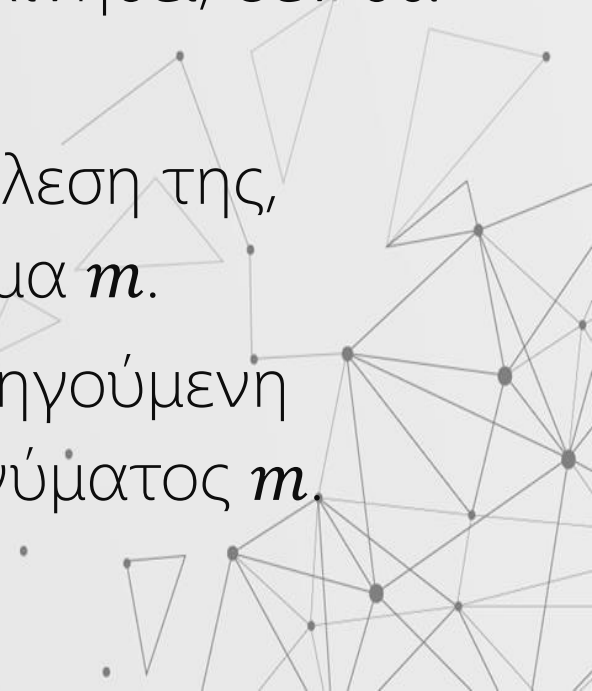
□ checkpoint

Ζητούμενο

- Επιθυμούμε το σύστημα να ανέχεται βλάβες χωρίς να αναγκάζουμε κάθε διεργασία να καταγράφει την κατάσταση της συνεχώς, σε κάθε βήμα της εκτέλεσης.
- Το σημείο επανεκκίνησης πρέπει να επιλέγεται έτσι ώστε οι διεργασίες να συνεχίζουν την εκτέλεση με συνέπεια (1) μεταξύ τους και (2) ως προς το εξωτερικό περιβάλλον.
- Πρόβλημα: μια διεργασία που επανεκκινείται μπορεί να ακολουθήσει μια διαφορετική ροή εκτέλεσης από αυτήν που είχε ακολουθηθεί πριν την βλάβη.
- Πρόβλημα: μια διεργασία που επανεκκινείται μπορεί να «χάσει» μηνύματα που έλαβε πριν την βλάβη.

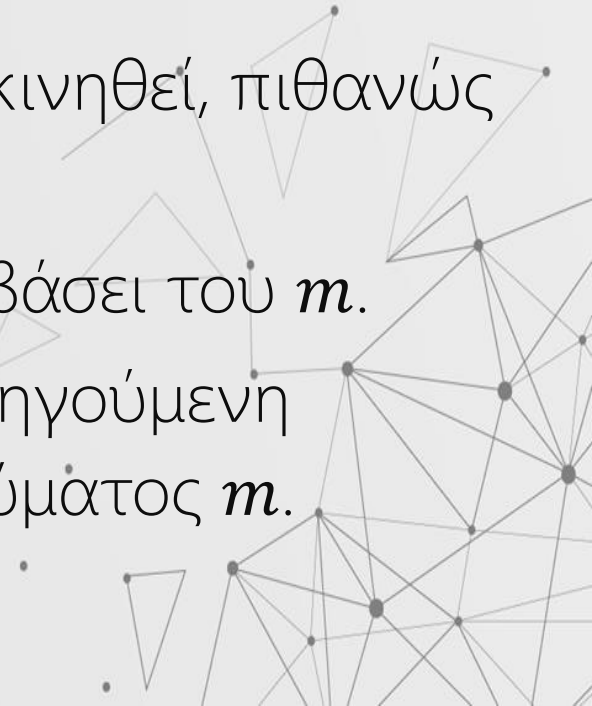
Πρόβλημα (1)

- Έστω ότι η P_1 στέλνει ένα μήνυμα m στην P_2 , και στην συνέχεια καταγράφει την κατάσταση της.
- Όσο η P_2 δεν καταγράφει την κατάσταση της, συνολικά, έχει καταγραφεί η αποστολή του m , αλλά όχι η παραλαβή του.
- Αν η P_2 παρουσιάσει βλάβη και επανεκκινηθεί, δεν θα λάβει υπόψη της ποτέ το μήνυμα m .
- Όμως η P_1 συνεχίζει κανονικά την εκτέλεση της, θεωρώντας ότι η P_2 έχει λάβει το μήνυμα m .
- Η P_1 πρέπει να οπισθοδρομήσει σε προηγούμενη κατάσταση πριν την αποστολή του μηνύματος m .

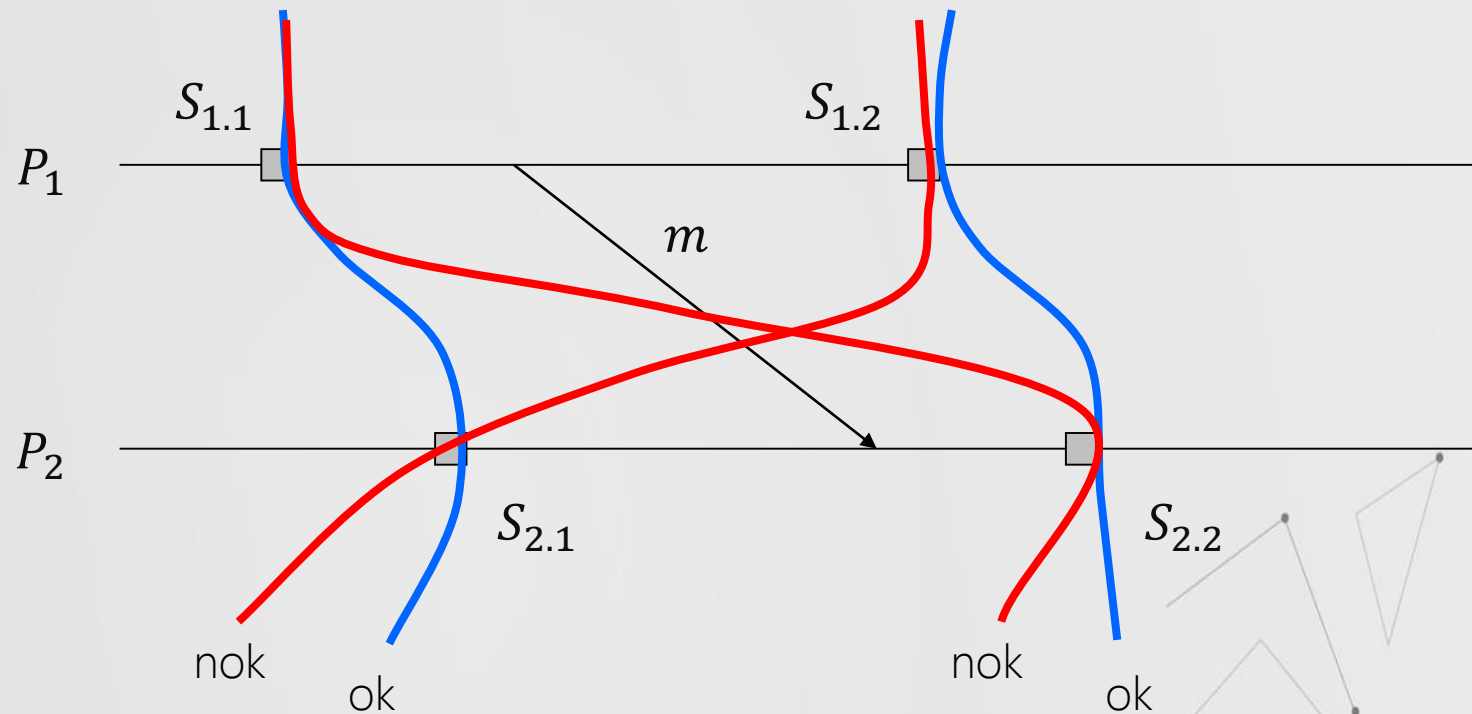


Πρόβλημα (2)

- Έστω ότι η P_1 στέλνει ένα μήνυμα m στην P_2 , και ότι στην συνέχεια η P_2 λαμβάνει το μήνυμα και καταγράφει την κατάσταση της.
- Όσο η P_1 δεν καταγράφει την κατάσταση της, συνολικά, έχει καταγραφεί η παραλαβή του m , αλλά όχι η αποστολή του.
- Αν η P_1 παρουσιάσει βλάβη και επανεκκινηθεί, πιθανώς να μην δημιουργήσει το ίδιο μήνυμα m .
- Όμως η P_2 συνεχίζει την εκτέλεση της βάσει του m .
- Η P_2 πρέπει να οπισθοδρομήσει σε προηγούμενη κατάσταση πριν τη παραλαβή του μηνύματος m .



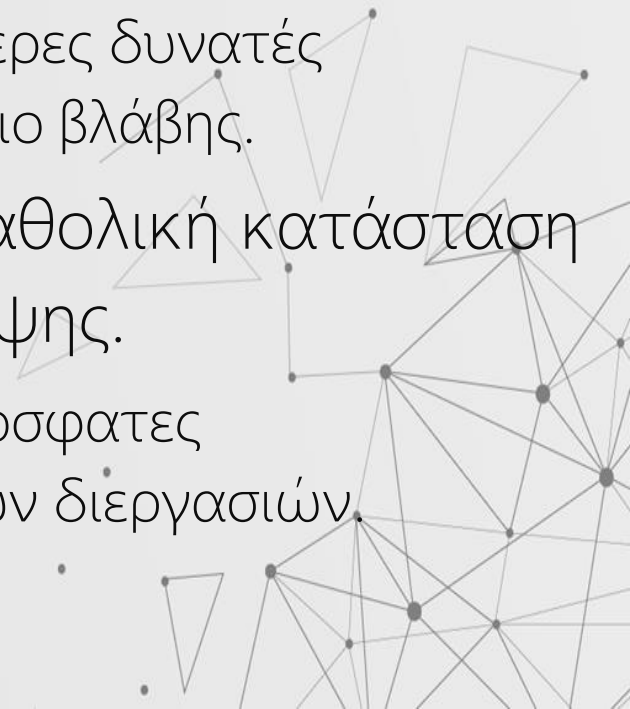
Γραμμή ανάκαμψης



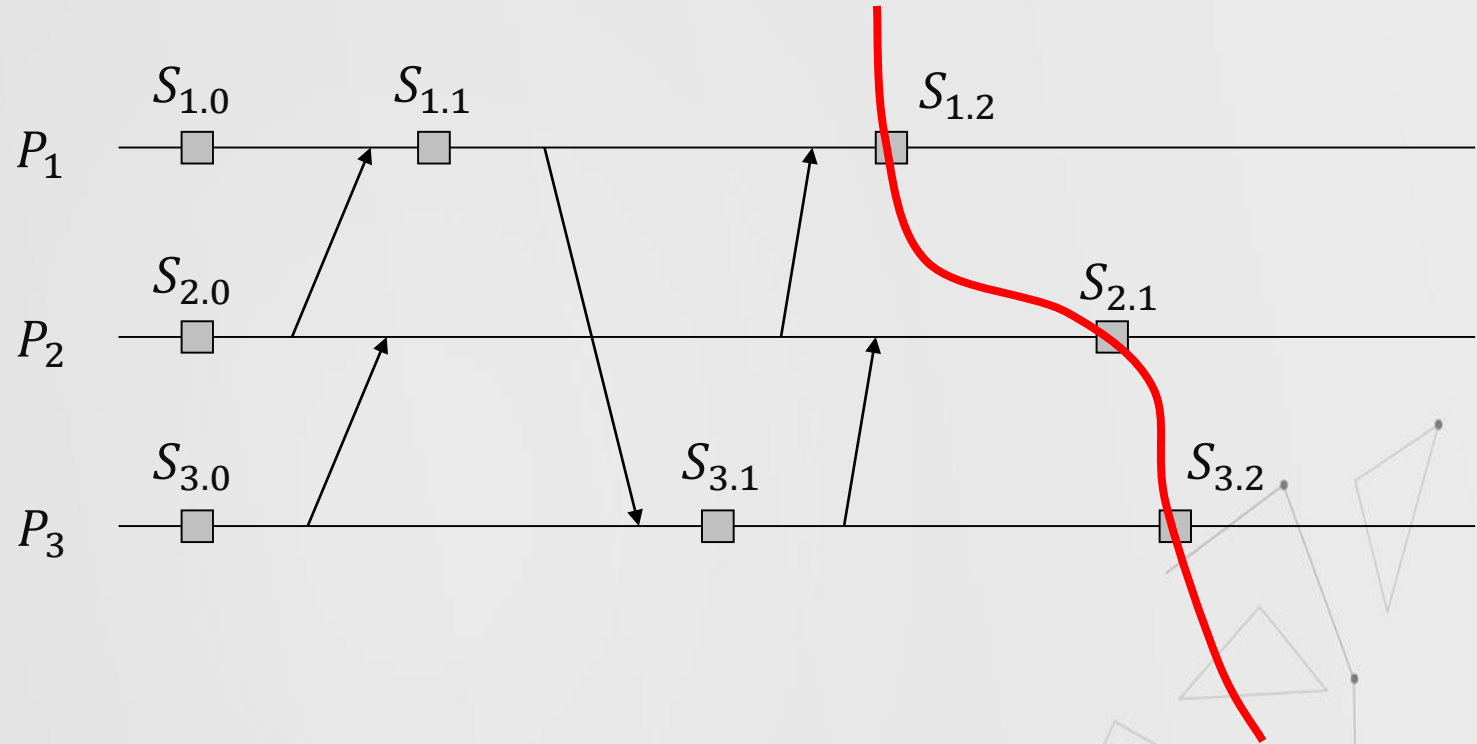
τομές που «επιτρέπουν» την συνεπή ανάκαμψη ενός κατανεμημένου υπολογισμού σε περίπτωση βλάβης

Γραμμή ανάκαμψης

- Γραμμή ανάκαμψης: η πιο πρόσφατη τομή της κατανεμημένης εκτέλεσης, που δεν τέμνει την μετάδοση ενός μηνύματος.
 - αν συμπεριλαμβάνεται η αποστολή ενός μηνύματος, τότε συμπεριλαμβάνεται και η παραλαβή του – και το αντίστροφο.
 - ορίζει το σημείο από το οποίο μπορεί να συνεχιστεί ο υπολογισμός με συνέπεια και με τις λιγότερες δυνατές οπισθοδρομήσεις, για οποιοδήποτε σενάριο βλάβης.
- Μέγιστη ανακάμψιμη κατάσταση: η καθολική κατάσταση που αντιστοιχεί στην γραμμή ανάκαμψης.
 - δεν αντιστοιχεί (απαραίτητα) στις πιο πρόσφατες καταγεγραμμένες τοπικές καταστάσεις των διεργασιών



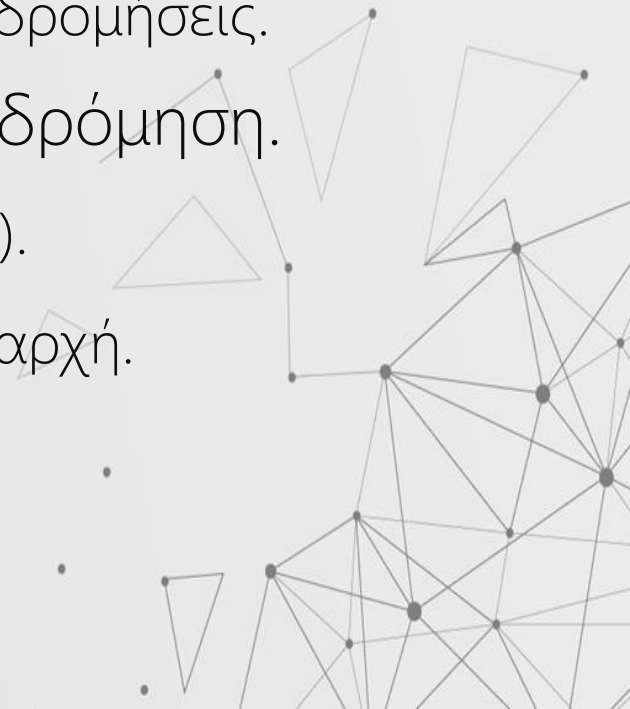
Γραμμή ανάκαμψης



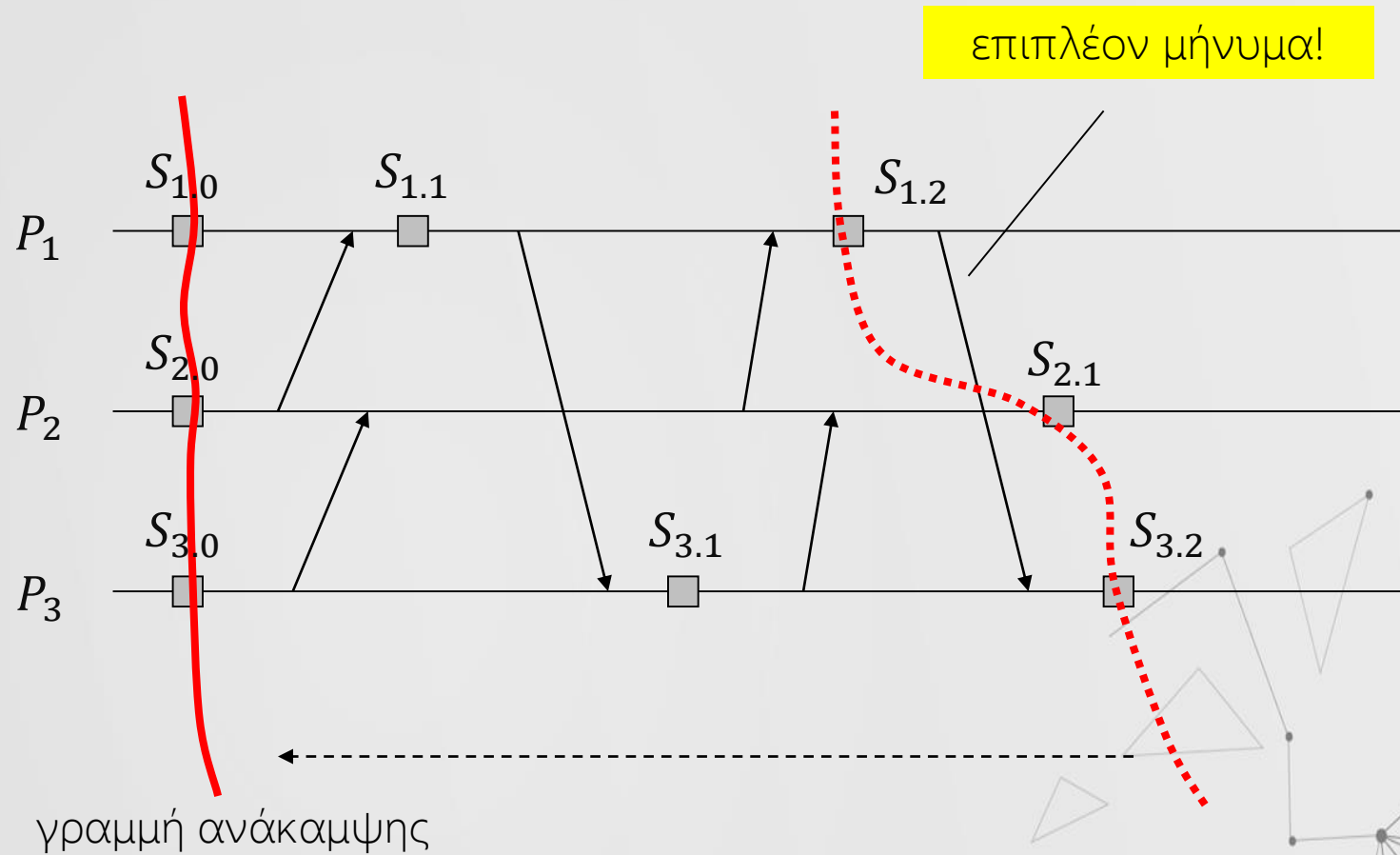
Γραμμή ανάκαμψης

Αλυσιδωτές οπισθοδρομήσεις

- Η επανεκκίνηση μιας διεργασίας P από την τελευταία καταγεγραμμένη κατάσταση μπορεί να μην είναι εφικτή.
 - λόγω ασυνέπειας (μη καταγεγραμμένων μηνυμάτων).
- Τότε, η P πρέπει να οπισθοδρομήσει σε μια προηγούμενη καταγεγραμμένη κατάσταση.
 - αυτό μπορεί να επιφέρει επιπλέον οπισθοδρομήσεις.
- Χειρότερη περίπτωση: πλήρης οπισθοδρόμηση.
 - αλυσιδωτή οπισθοδρόμηση (domino effect).
 - επανεκκίνηση του υπολογισμού, από την αρχή.

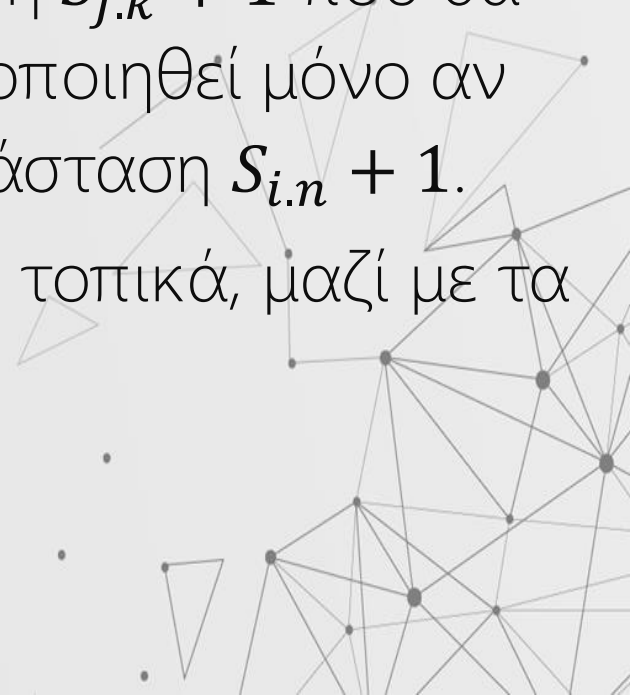


Αλυσιδωτές οπισθοδρομήσεις

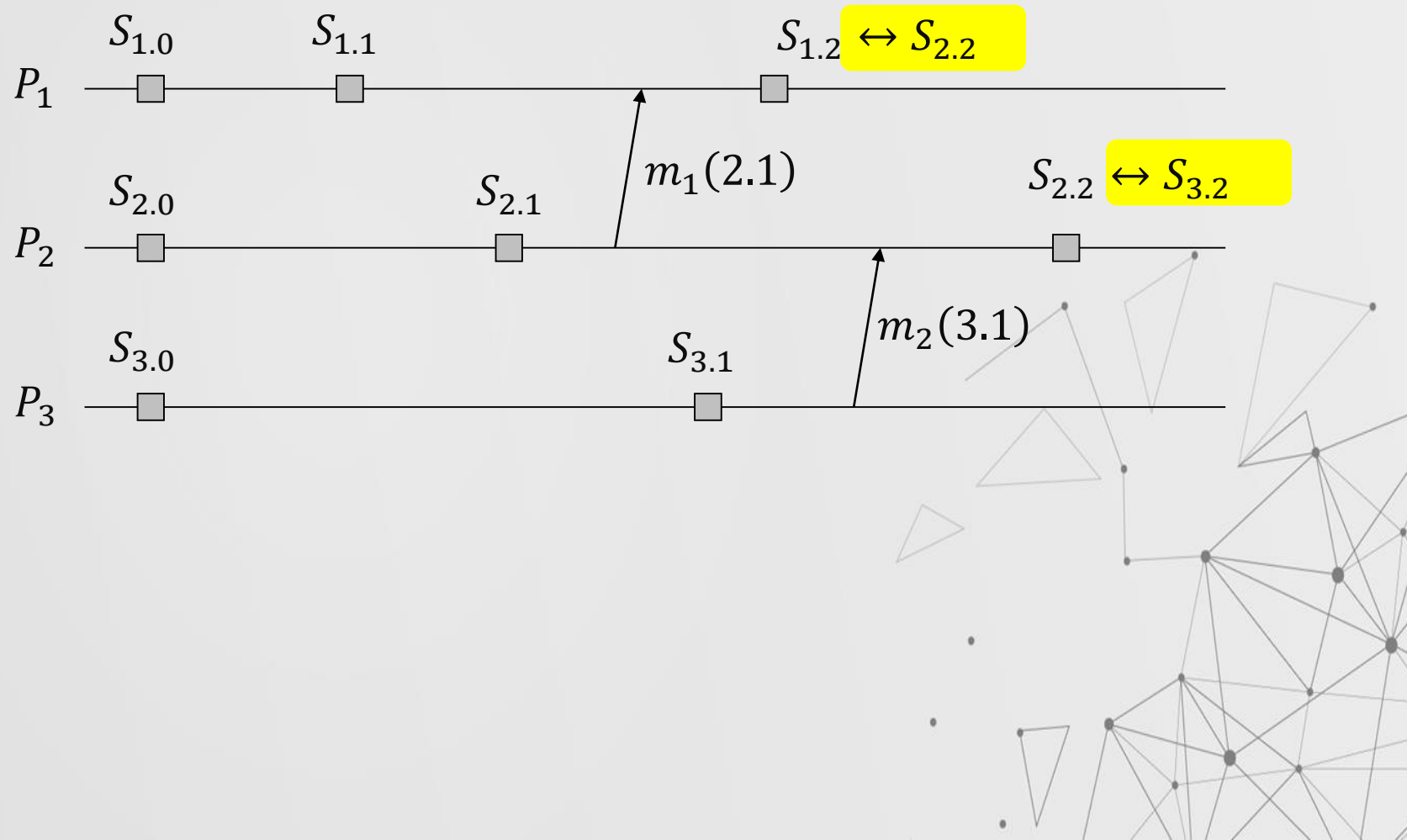


Καταγραφή χωρίς συγχρονισμό

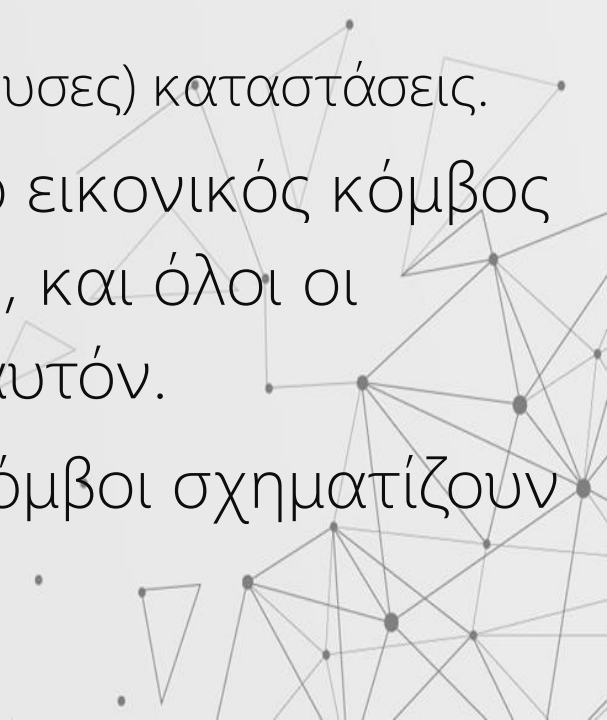
- Κάθε διεργασία καταγράφει (περιοδικά) την τοπική της κατάσταση χωρίς συνεννόηση με τις υπόλοιπες.
- Όταν η P_i στέλνει μήνυμα m στην P_j , επισυνάπτει τον αριθμό της τελευταίας τοπικά καταγεγραμμένης κατάστασης $S_{i.n}$.
- Η P_j γνωρίζει ότι η επόμενη κατάσταση $S_{j.k} + 1$ που θα καταγράψει τοπικά μπορεί να χρησιμοποιηθεί μόνο αν και η P_i καταγράψει την επόμενη κατάσταση $S_{i.n} + 1$.
- Αυτές οι «εξαρτήσεις» καταγράφονται τοπικά, μαζί με τα περιεχόμενα των καταστάσεων.



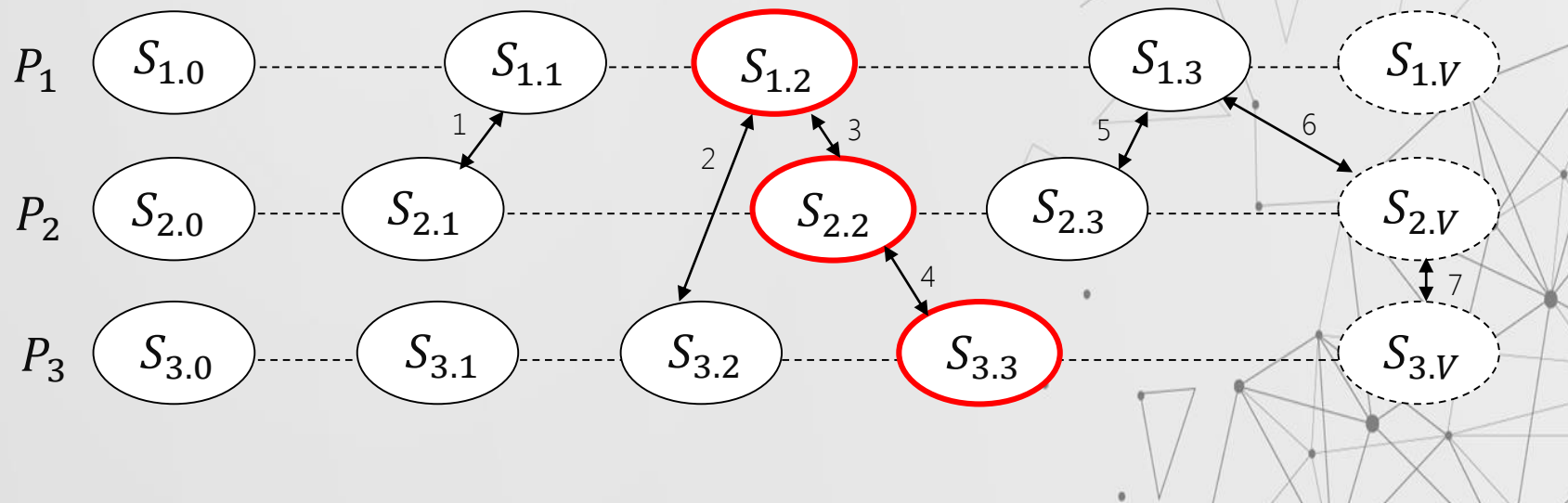
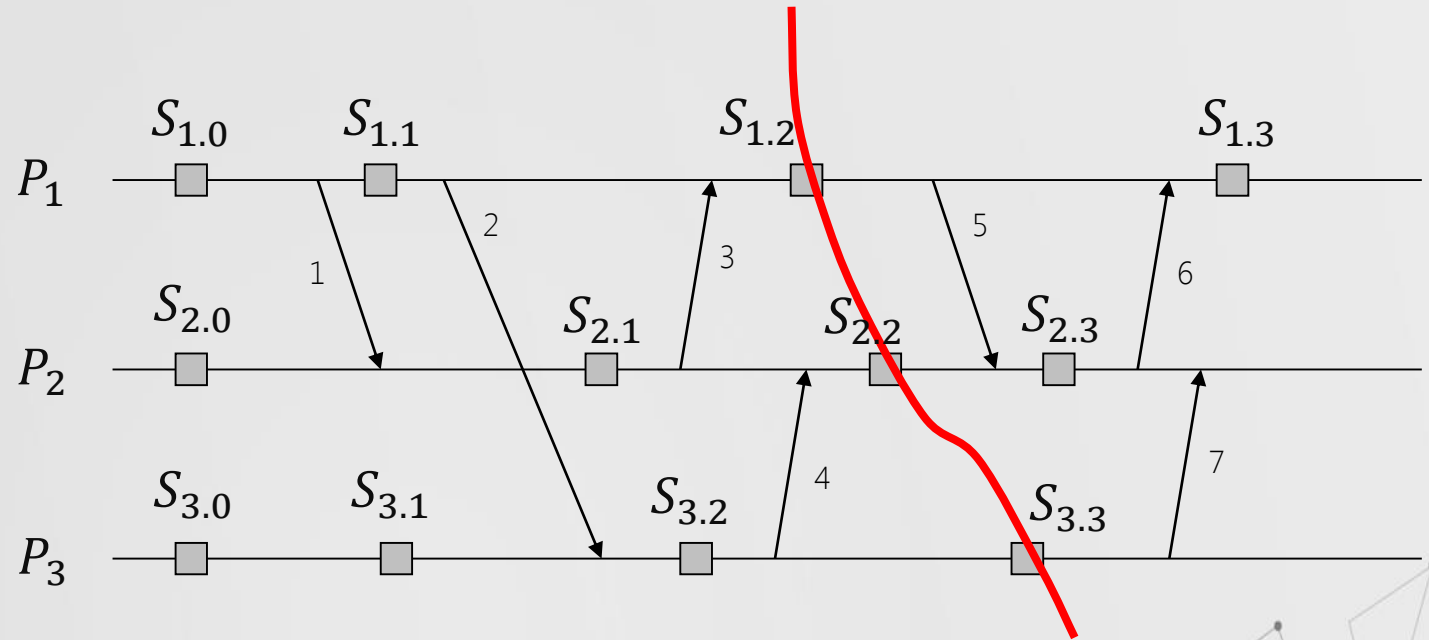
Καταγραφή χωρίς συγχρονισμό



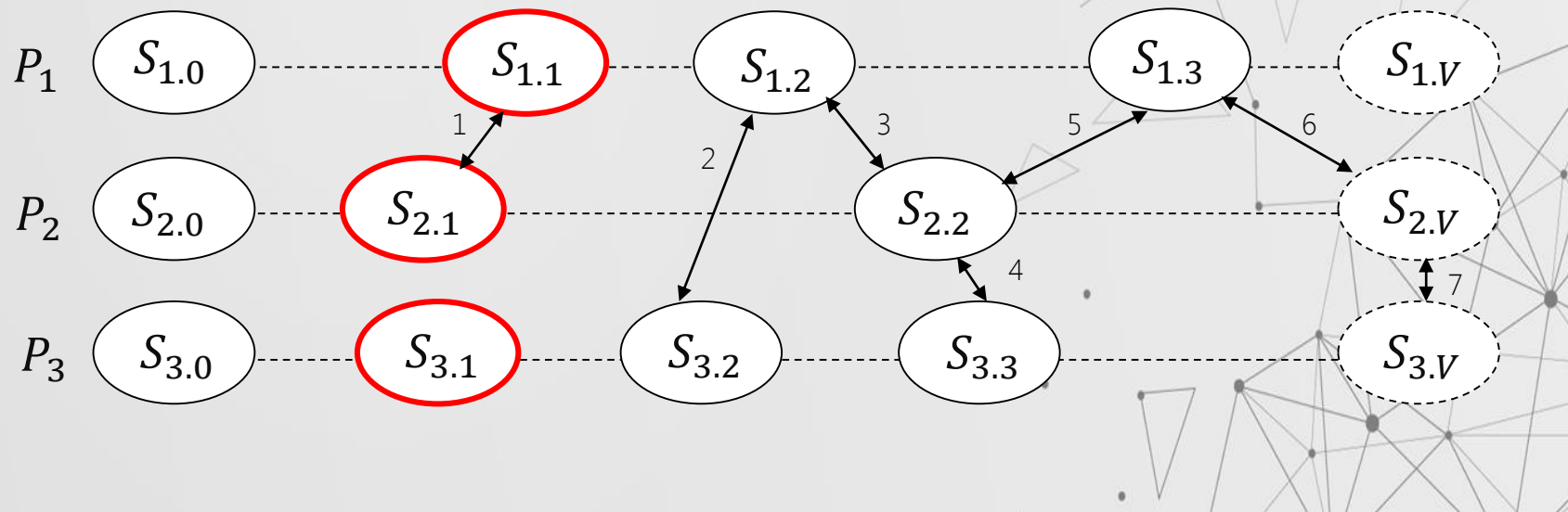
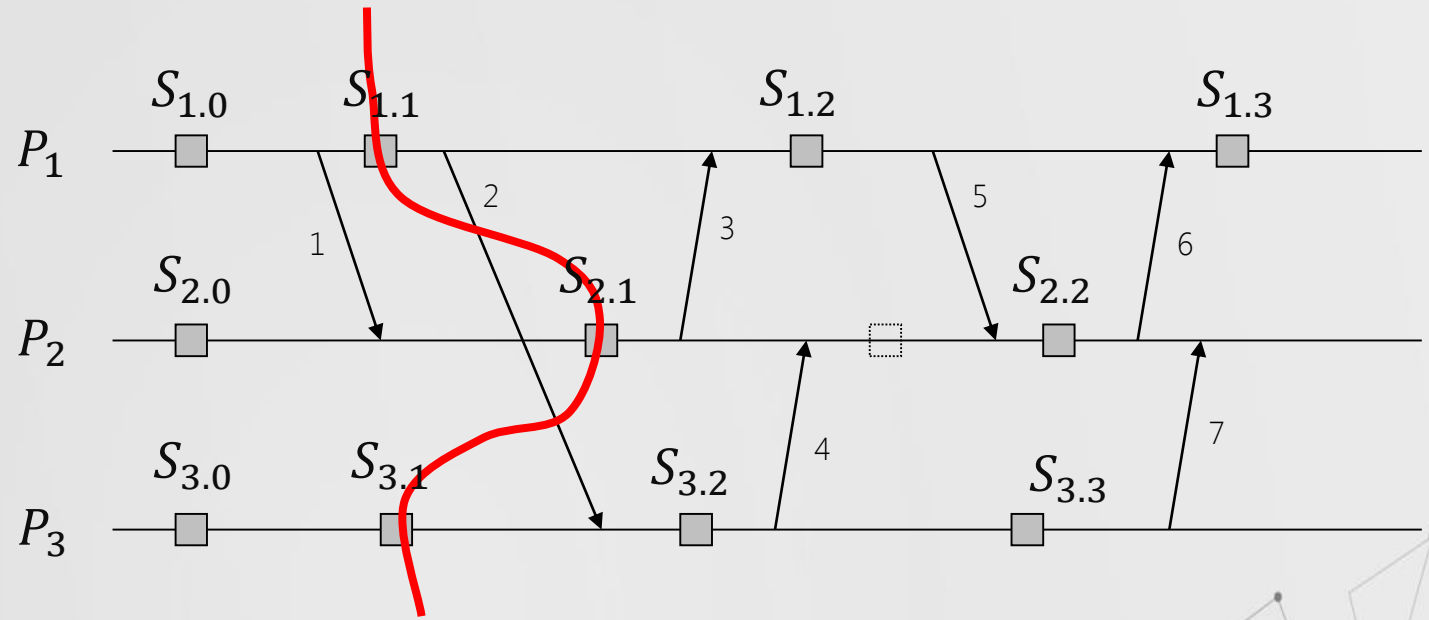
Μέγιστη ανακάμψιμη κατάσταση

- Η πληροφορία εξάρτησης συλλέγεται από μια διεργασία (περιοδικά ή μετά από ανάκαμψη από βλάβη).
 - Σχηματίζεται ένας γράφος εξάρτησης.
 - ένας κόμβος για κάθε καταγεγραμμένη κατάσταση.
 - αν υπάρχει εξάρτηση ανάμεσα σε $S_{i.n}$ και $S_{j.k}$, τότε εισάγεται μια ακμή $S_{i.n} \leftrightarrow S_{j.k}$.
 - εισάγονται (εικονικοί) κόμβοι για τις (τρέχουσες) καταστάσεις.
 - Για κάθε σενάριο βλάβης, μαρκάρεται ο εικονικός κόμβος της διεργασίας που παρουσίασε βλάβη, και όλοι οι κόμβοι που είναι προσπελάσιμοι από αυτόν.
 - Οι πιο πρόσφατοι μη μαρκαρισμένοι κόμβοι σχηματίζουν την μέγιστη κατάσταση ανάκαμψης.
- 

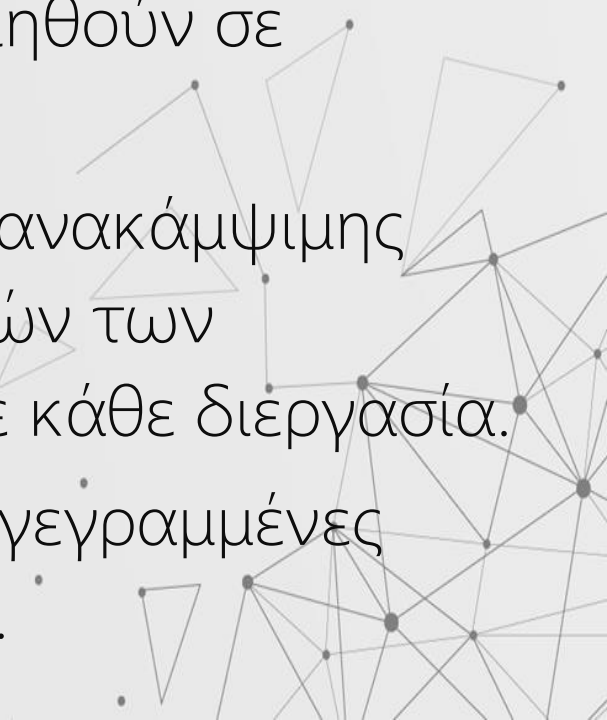
Μέγιστη ανακάμψιμη κατάσταση



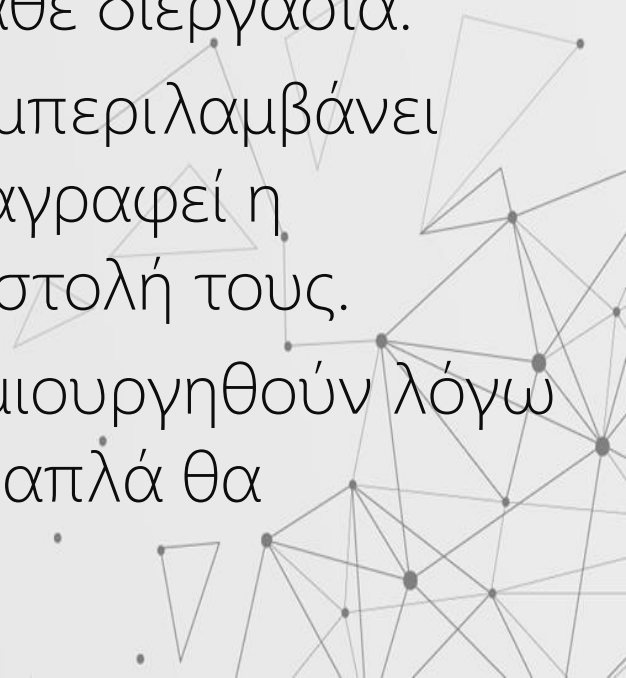
Μέγιστη ανακάμψιμη κατάσταση

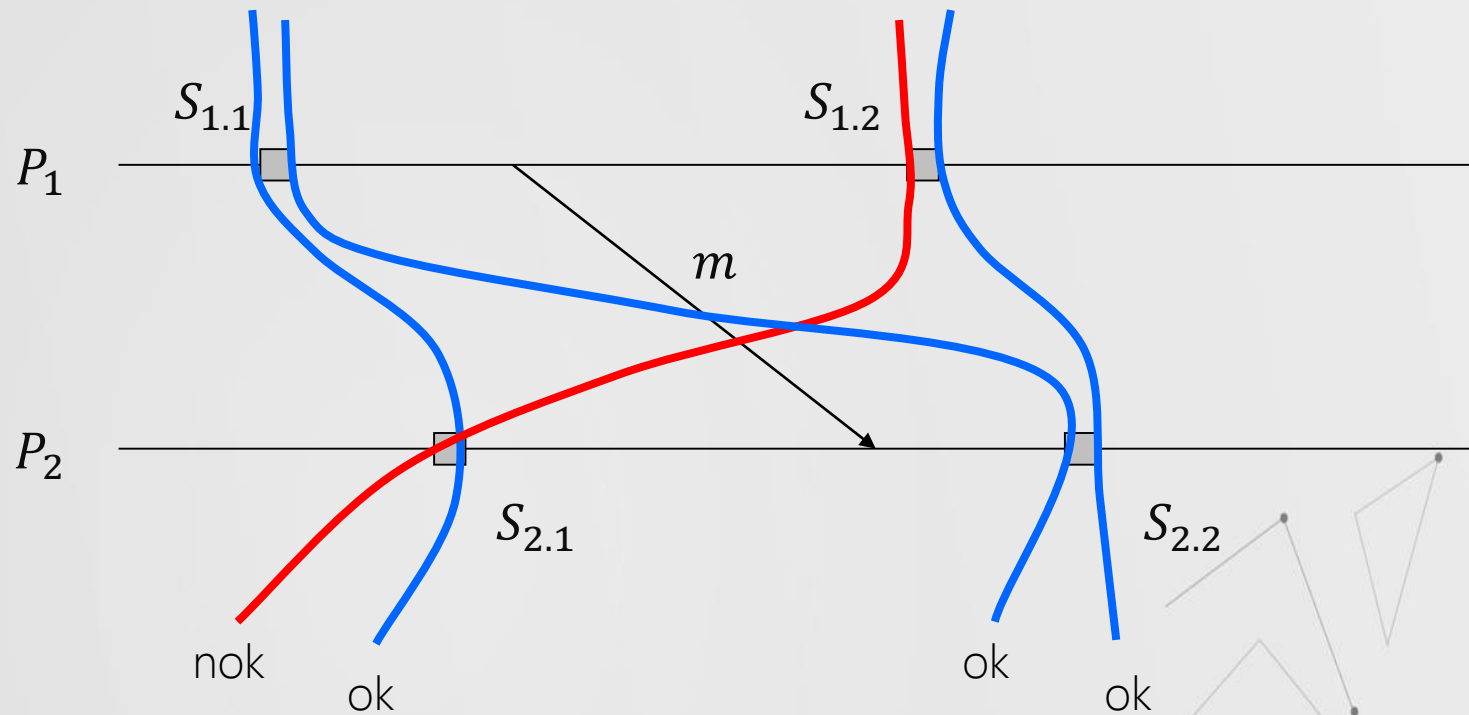


Συλλογή καταστάσεων

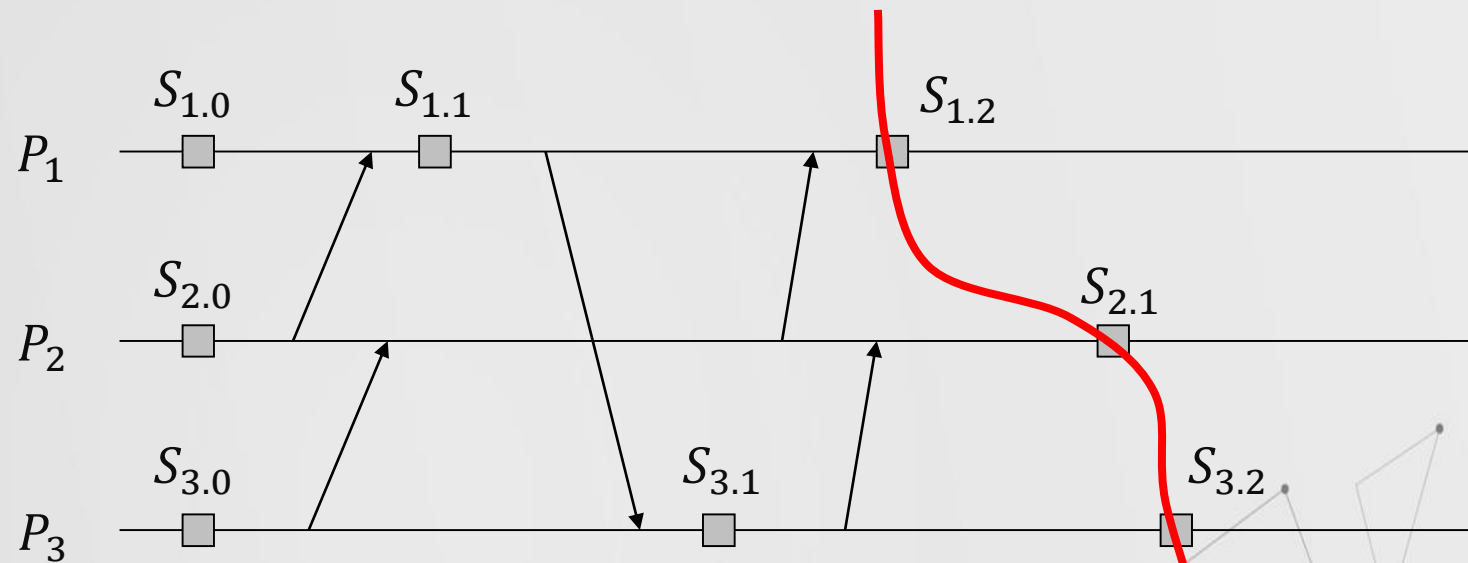
- Ο αριθμός των καταγεγραμμένων καταστάσεων μεγαλώνει διαρκώς καθώς εξελίσσεται η εκτέλεση.
 - Πρέπει να απομακρυνθούν οι καταστάσεις που είναι σίγουρο ότι δεν έχουν (πλέον) κάποια χρησιμότητα.
 - Όλες οι καταστάσεις πίσω από την γραμμή ανάκαμψης.
 - Δεν υπάρχει περίπτωση να χρησιμοποιηθούν σε επανεκκίνηση.
 - Περιοδικός υπολογισμός της μέγιστης ανακάμψιμης κατάστασης, και αποστολή των αριθμών των αντίστοιχων τοπικών καταστάσεων σε κάθε διεργασία.
 - Κάθε διεργασία σβήνει τις τοπικά καταγεγραμμένες καταστάσεις με μικρότερους αριθμούς.
- 

Ντετερμινιστικές διεργασίες

- Αν η P με κατάσταση S οπισθοδρομήσει σε μια προηγούμενη κατάσταση S' , θα πραγματοποιήσει ακριβώς την ίδια εκτέλεση, και θα πάει από την S στην S' , εφόσον η προηγούμενη εκτέλεση δεν συμπεριλαμβάνει την παραλαβή ενός μηνύματος.
 - Κάθε διεργασία καταγράφει τους σειριακούς αριθμούς των μηνυμάτων που λαμβάνει από κάθε διεργασία.
 - Η μέγιστη ανακάμψιμη κατάσταση συμπεριλαμβάνει πλέον και καταστάσεις όπου έχει καταγραφεί η παραλαβή μηνυμάτων χωρίς την αποστολή τους.
 - Τα «διπλότυπα» μηνύματα που θα δημιουργηθούν λόγω οπισθοδρόμησης, θα εντοπιστούν και απλά θα αγνοηθούν από τον παραλήπτη.
- 

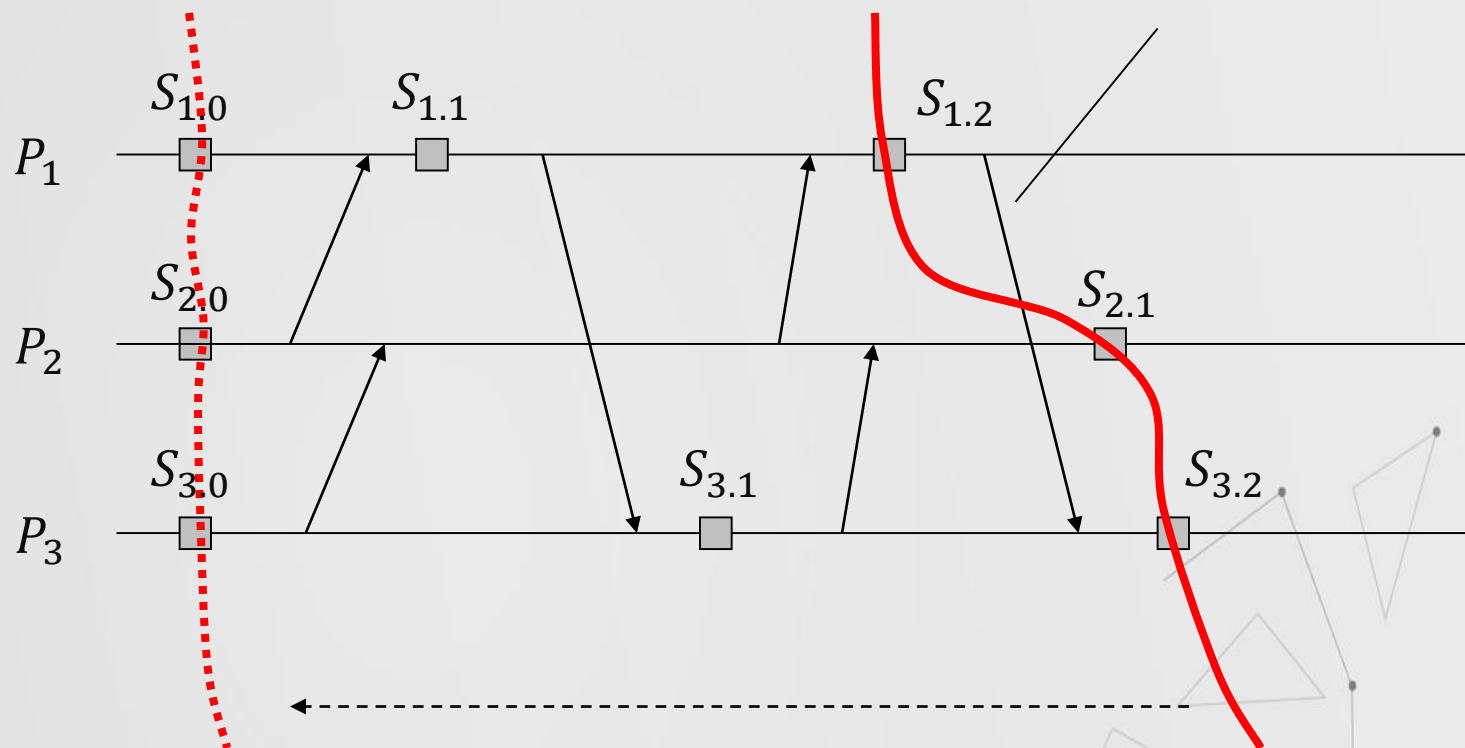


τομές που «επιτρέπουν» την συνεπή ανάκαμψη ενός κατακεκομένου υπολογισμού σε περίπτωση βλάβης, στην περίπτωση **ντετερμινιστικών** διεργασιών



γραμμή ανάκαμψης

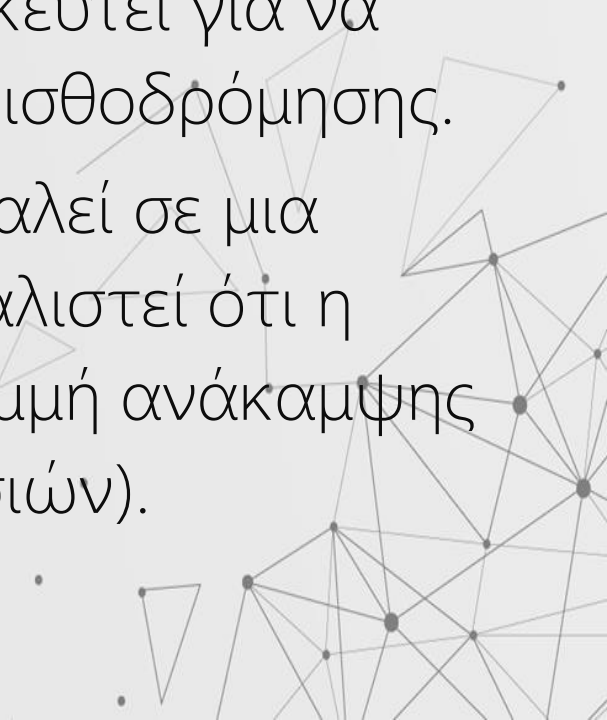
επιπλέον μήνυμα!



γραμμή ανάκαμψης για μη
ντετερμινιστικές διεργασίες

γραμμή ανάκαμψης για
ντετερμινιστικές διεργασίες

Input-output commit problem

- Το κατανεμημένο σύστημα μπορεί να επικοινωνεί και με κάποιες «εξωτερικές» οντότητες.
 - Ιδανικά, πρέπει να παρέχεται διαφάνεια βλαβών και σε ένα τέτοιο «εξωτερικό» περιβάλλον.
 - Input commit: αν ένα μήνυμα παραληφθεί από μια εξωτερική οντότητα, πρέπει να αποθηκευτεί για να ληφθεί (ξανά) υπόψη σε περίπτωση οπισθοδρόμησης.
 - Output commit: προτού ένα μήνυμα σταλεί σε μια εξωτερική οντότητα, πρέπει να εξασφαλιστεί ότι η τρέχουσα κατάσταση ανήκει στην γραμμή ανάκαμψης (απαιτείται συγχρονισμός των διεργασιών).
- 

Καταγραφή κατανεμημένων στιγμιότυπων

- Οι διεργασίες επικοινωνούν (περιοδικά) μεταξύ τους για να κατασκευάσουν κατανεμημένα στιγμιότυπα.
 - π.χ., με βάση τον αλγόριθμο των Chandy/Lamport.
- Κάθε συνεπές κατανεμημένο στιγμιότυπο αντιστοιχεί σε μια μέγιστη ανακάμψιμη κατάσταση.
 - οι καταγεγραμμένες τοπικές καταστάσεις σχηματίζουν μια συνεπή καθολική κατάσταση.
 - επιπλέον έχουν καταγραφεί όλα τα μηνύματα που βρίσκονταν καθοδόν την ώρα της καταγραφής.
 - τα οποία πρέπει να ληφθούν υπόψη αν γίνει επανεκκίνηση με βάση αυτή τη κατάσταση.
- Με το που κατασκευαστεί το επόμενο στιγμιότυπο μπορεί να σβηστεί το προηγούμενο.
 - δεν υπάρχει περίπτωση οπισθοχώρησης πίσω από αυτό.

