

Αιτιώδεις Σχέσεις και Χρονισμός

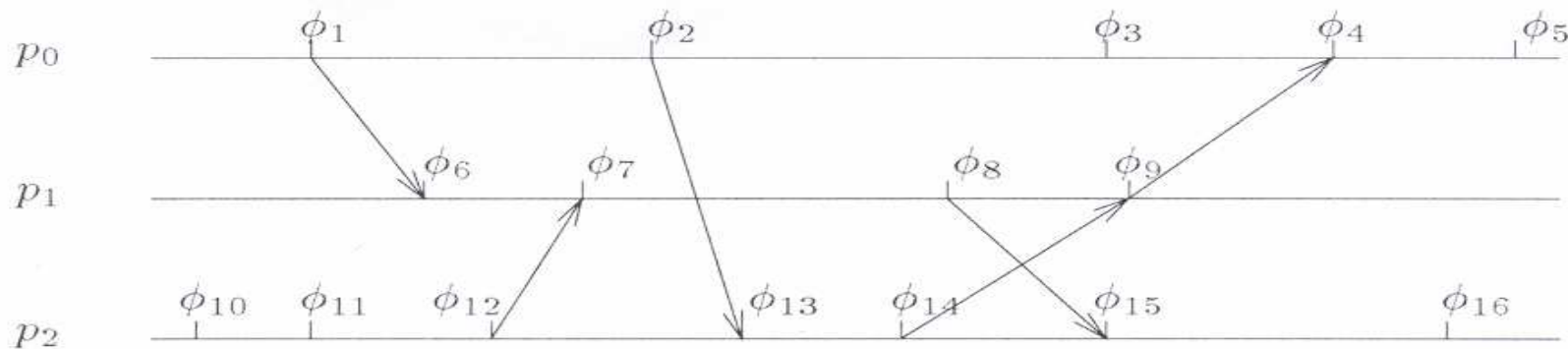
Η Σχέση Happens-Before (Συμβαίνει-πριν)

- ◇ Οι εκτελέσεις, ως ακολουθίες γεγονότων, καθορίζουν μια καθολική διάταξη σε αυτά.
- ◇ Ωστόσο είναι δυνατό δύο υπολογιστικά γεγονότα από διαφορετικούς επεξεργαστές να μην επηρεάζουν το ένα το άλλο, παρότι η εκτέλεση καθορίζει μια (αυθαίρετη) διάταξη αυτών.
- ◇ Έτσι, η δομή αιτιότητας μεταξύ των γεγονότων χάνεται.

Η Σχέση Happens-Before (Συμβαίνει-πριν)

Έστω a μια αυθαίρετη εκτέλεση.

- Έστω ότι ϕ_1 και ϕ_2 είναι 2 γεγονότα της a που εκτελούνται από τον ίδιο επεξεργαστή. Το ϕ_1 **επηρεάζει αιτιωδώς** (casually influences) το ϕ_2 , αν το ϕ_1 προηγείται του ϕ_2 στην a .
- Έστω ότι ϕ_1 και ϕ_2 είναι 2 γεγονότα της a που εκτελούνται από διαφορετικές διεργασίες p_i και p_j , αντίστοιχα. Το ϕ_1 **επηρεάζει αιτιωδώς** το ϕ_2 αν το ϕ_1 είναι γεγονός στο οποίο αποστέλλεται μήνυμα m από την p_i στην p_j ενώ το ϕ_2 είναι το γεγονός παραλαβής του m από την p_j .
- Επίσης, ισχύει η μεταβατική ιδιότητα (αν το ϕ_1 επηρεάζει αιτιωδώς το ϕ_2 και το ϕ_2 επηρεάζει αιτιωδώς το ϕ_3 , τότε το ϕ_1 επηρεάζει αιτιωδώς το ϕ_3).

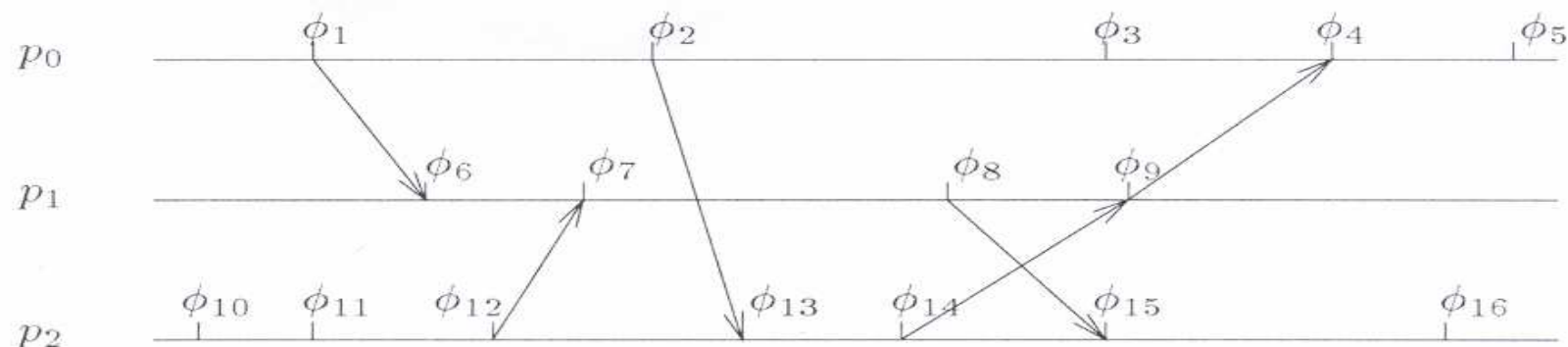


Η σχέση Happens-Before (Συμβαίνει-πριν)

Πιο φορμαλιστικά:

Δεδομένων δύο γεγονότων ϕ_1 και ϕ_2 στην a , το ϕ_1 *Συμβαίνει-πριν* (happens before) από το ϕ_2 , το οποίο συμβολίζεται $\phi_1 \rightarrow \phi_2$, αν ισχύει μια από τις ακόλουθες συνθήκες:

- τα ϕ_1, ϕ_2 είναι γεγονότα που εκτελούνται από την ίδια διεργασία p_i και το ϕ_1 προηγείται του ϕ_2 στην ακολουθία a ,
- το ϕ_1 είναι γεγονός στο οποίο αποστέλλεται ένα μήνυμα m από μια διεργασία p_i σε μια άλλη p_j , και το ϕ_2 είναι το γεγονός παραλαβής του m από την p_j ,
- υπάρχει γεγονός ϕ τέτοιο ώστε $\phi_1 \rightarrow \phi$ και $\phi \rightarrow \phi_2$.



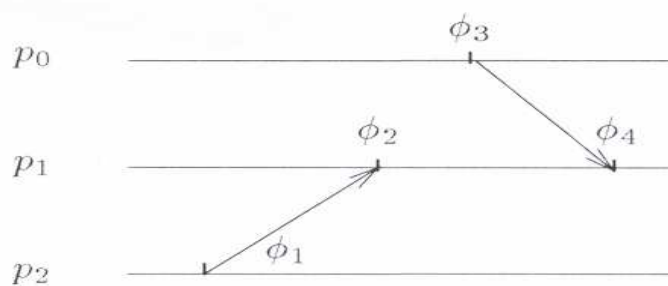
Αιτιώδεις Αναδιατάξεις (Casual Shuffles)

Η σχέση *Συμβαίνει-πριν* χαρακτηρίζει τις αιτιώδεις σχέσεις μιας εκτέλεσης.

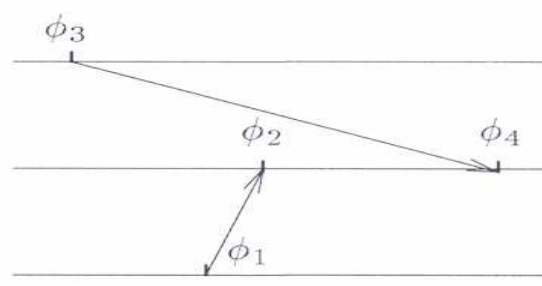
Αν τα γεγονότα μιας εκτέλεσης αναδιαταχθούν χωρίς ωστόσο να καταστρατηγούν τη σχέση *Συμβαίνει-πριν*, το αποτέλεσμα εξακολουθεί να είναι έγκυρη εκτέλεση και οι δύο αυτές εκτελέσεις είναι παρόμοιες σε όλες τις διεργασίες.

Ορισμός: Έστω ότι $a = \text{exec}(C, \sigma)$ είναι ένα τμήμα εκτέλεσης. Μια μετάθεση (permutation) π του χρονοδιαγράμματος σ είναι μια *αιτιώδης αναδιάταξη* (casual shuffle) του σ αν:

- ο για κάθε j , $0 \leq j \leq n-1$, $\sigma|_j = \pi|_j$, και
- αν ένα μήνυμα m αποστέλλεται από μια διεργασία p_j κατά την εκτέλεση του γεγονότος ϕ της a , τότε στην π το ϕ προηγείται του γεγονότος παραλαβής του m .



(a) $\text{exec}(C, \sigma)$



(b) $\text{exec}(C, \pi)$

○

Αιτιώδεις Αναδιατάξεις (Casual Shuffles)

Λήμμα 1

Έστω ότι $a = \text{exec}(C, \sigma)$ είναι ένα τμήμα εκτέλεσης. Οποιαδήποτε καθολική διάταξη των γεγονότων του σ είναι συνεπής με τη σχέση Συμβαίνει-πριν της a , είναι μια αιτιώδης αναδιάταξη (casual shuffle) της σ .

Λήμμα 2

Έστω ότι $a = \text{exec}(C, \sigma)$ είναι ένα τμήμα εκτέλεσης. Έστω ότι π είναι μια αιτιώδης αναδιάταξη (casual shuffle) του σ . Τότε, $a' = \text{exec}(C, \pi)$ είναι ένα έγκυρο τμήμα εκτέλεσης παρόμοιο (indistinguishable) με το a σε όλες τις διεργασίες.

Λογικά Ρολόγια (Logical Clocks)

Πως μπορούν οι διεργασίες να παρατηρήσουν την σχέση Συμβαίνει-πριν σε μια εκτέλεση α;

Με κάθε γεγονός φ , συσχετίζουμε μια χρονοσφραγίδα $LT(\varphi)$ (Logical Time of φ).

Χρειαζόμαστε μια μερική διάταξη $<$ στις χρονοσφραγίδες, έτσι ώστε για κάθε ζεύγος γεγονότων φ_1 και φ_2 :

$$\text{αν } \varphi_1 \rightarrow \varphi_2 \text{ τότε } LT(\varphi_1) < LT(\varphi_2)$$

Απλός Αλγόριθμος για ανάθεση και διατήρηση χρονοσφραγίδων

- Κάθε διεργασία p_j διατηρεί μια τοπική μεταβλητή LT_j , που ονομάζεται λογικό ρολόι (logical clock). Το LT_j αποθηκεύει μη-αρνητικούς ακεραίους και έχει αρχική τιμή 0.
- Ως μέρος κάθε υπολογιστικού γεγονότος φ , η p_j αυξάνει το LT_j ώστε να είναι μεγαλύτερο κατά μία μονάδα από το μέγιστο της τρέχουσας τιμής του LT_j και της μεγαλύτερης χρονοσφραγίδας κάθε μηνύματος που παραλαμβάνεται στο φ .
- Σε κάθε μήνυμα που αποστέλλεται στο φ ανατίθεται ως χρονοσφραγίδα το LT_j .
- Η χρονοσφραγίδα, $LT(\varphi)$, που συσχετίζεται με το γεγονός φ της διεργασίας p_j , είναι η νέα τιμή LT_j που υπολογίστηκε κατά τη διάρκεια εκτέλεσης του γεγονότος.

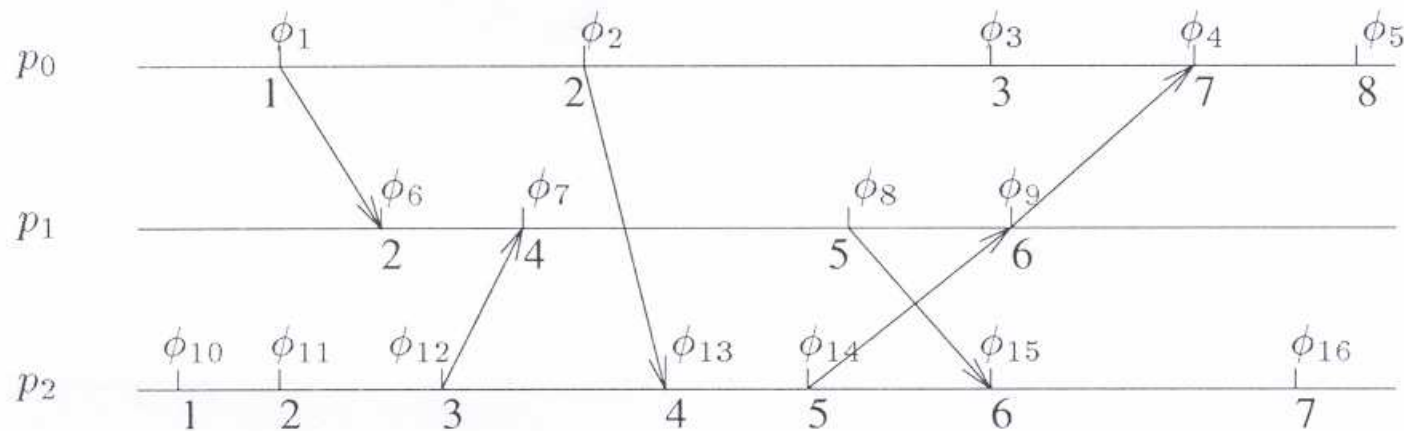
Λογικά Ρολόγια (Logical Clocks)

Η μερική διάταξη των χρονοσφραγίδων καθορίζεται από την γνωστή σχέση $<$ των ακεραίων.

Για κάθε διεργασία p_j , το LT_j είναι γνήσια αύξων.

Τότε, και οι τρεις συνθήκες της σχέσης *Συμβαίνει-πριν* ικανοποιούνται.

Θεώρημα: Έστω ότι a είναι μια αυθαίρετη εκτέλεση και ας υποθέσουμε ότι ϕ_1 και ϕ_2 είναι δύο οποιαδήποτε γεγονότα της a . Αν $\phi_1 \rightarrow \phi_2$, τότε $LT(\phi_1) < LT(\phi_2)$.



Διανυσματικά Ρολόγια (Vector Clocks)

Αρνητικά Λογικών Ρολογιών

Αν $LT(\varphi_1) \geq LT(\varphi_2)$, τότε γνωρίζουμε ότι το φ_1 δεν συμβαίνει πριν το φ_2 .

Είναι το αντίστροφο αληθές:

ΟΧΙ! Είναι δυνατό $LT(\varphi_1) < LT(\varphi_2)$ αλλά να μην ισχύει $\varphi_1 \rightarrow \varphi_2$.

Το πρόβλημα είναι ότι η σχέση *Συμβαίνει-πριν* ορίζει μερική διάταξη, ενώ οι λογικές χρονοσφραγίδες είναι ακέραιοι και η $<$ ορίζει καθολική διάταξη.

Διανυσματικά Ρολόγια (Vector Clocks)

Ορισμός: Τα γεγονότα φ_1 και φ_2 *συμβαίνουν ταυτόχρονα* σε μια εκτέλεση a , το οποίο συμβολίζεται $\varphi_1 \parallel_a \varphi_2$, αν δεν ισχύει ούτε ότι $\varphi_1 \rightarrow \varphi_2$, ούτε ότι $\varphi_2 \rightarrow \varphi_1$.

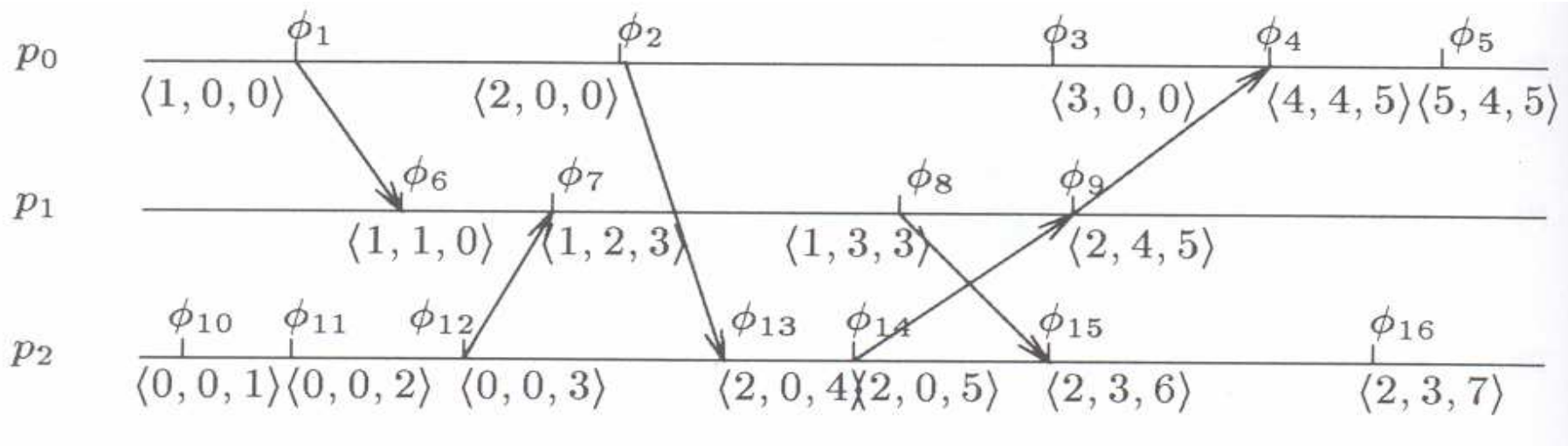
Αν $\varphi_1 \parallel_a \varphi_2$, τότε υπάρχουν 2 εκτελέσεις a_1 και a_2 , και οι δύο παρόμοιες (indistinguishable) με την a , τέτοιες ώστε το φ_1 συμβαίνει πριν από το φ_2 στην a_1 , και το φ_1 συμβαίνει μετά το φ_2 στην a_2 ή το αντίστροφο.

\Rightarrow οι διεργασίες δεν γνωρίζουν αν το φ_1 συμβαίνει πριν από το φ_2 ή το αντίστροφο στην a , ενώ ότι από τα δύο και αν συμβαίνει οι διεργασίες δεν παρατηρούν κάποια διαφορά.

Διανυσματικά Ρολόγια (Vector Clocks)

- ✚ Κάθε διεργασία p_j διατηρεί μια τοπική μεταβλητή VC_j (Vector Clock) που ονομάζεται *διανυσματικό ρολόι* της p_j . Το VC_j είναι ένας πίνακας n στοιχείων, κάθε στοιχείο του οποίου είναι ένας μη-αρνητικός ακέραιος με αρχική τιμή 0.
- ✚ Ως μέρος κάθε υπολογιστικού γεγονότος φ μιας διεργασίας p_j , η p_j ενημερώνει το VC_j ως εξής:
 - το $VC_j[j]$ αυξάνεται κατά 1
 - Για κάθε $i \neq j$, το $VC_j[i]$ παίρνει τιμή ίση με το μέγιστο της τρέχουσας τιμής του και της μεγαλύτερης τιμής της εγγραφής i όλων των διανυσματικών χρονοσφραγίδων των μηνυμάτων που ελήφθησαν κατά τη διάρκεια εκτέλεσης του φ .
 - Σε κάθε μήνυμα που αποστέλλεται κατά τη διάρκεια εκτέλεσης του φ ανατίθεται ως χρονοσφραγίδα η νέα τιμή του VC_j .
- ✚ Η διανυσματική χρονοσφραγίδα του φ , $VC(\varphi)$, είναι η τιμή του VC_j στο τέλος της εκτέλεσης του φ .

Διανυσματικά Ρολόγια (Vector Clocks)



Πρόταση: Για κάθε διεργασία p_j , σε κάθε προσβάσιμη καθολική κατάσταση, ισχύει ότι $VC_j[j] \geq VC_i[j]$, για κάθε i , $0 \leq i \leq n-1$.

Διανυσματικά Ρολόγια (Vector Clocks)

Μερική Διάταξη Διανυσματικών Ρολογιών

Έστω ότι v_1 και v_2 είναι δύο διανύσματα των n ακεραίων το καθένα. Τότε, $v_1 \leq v_2$ αν και μόνο αν για κάθε j , $0 \leq j \leq n-1$, $v_1[j] \leq v_2[j]$; $v_1 < v_2$ αν και μόνο αν $v_1 \leq v_2$ και $v_1 \neq v_2$.

Τα διανύσματα v_1 και v_2 είναι μη συγκρίσιμα αν δεν ισχύει ούτε ότι $v_1 \leq v_2$, ούτε ότι $v_2 \leq v_1$.

Λέμε ότι οι διανυσματικές χρονοσφραγίδες εκφράζουν την εκτέλεση ταυτόχρονων γεγονότων (δηλαδή την παραλληλία), αν για κάθε ζεύγος γεγονότων φ_1 και φ_2 , $\varphi_1 \parallel \varphi_2$ αν και μόνο αν τα $VC(\varphi_1)$ και $VC(\varphi_2)$ είναι μη-συγκρίσιμα.

Διανυσματικά Ρολόγια (Vector Clocks)

Θεώρημα

Έστω ότι a είναι μια αυθαίρετη εκτέλεση και έστω ότι φ_1 και φ_2 είναι δύο οποιαδήποτε γεγονότα της a . Αν $\varphi_1 \rightarrow \varphi_2$ τότε $VC(\varphi_1) < VC(\varphi_2)$.

Απόδειξη

Έστω ότι τα φ_1, φ_2 είναι γεγονότα του ίδιου επεξεργαστή και ας υποθέσουμε ότι το φ_1 προηγείται του φ_2 . *Γιατί ισχύει ο ισχυρισμός για τα φ_1 και φ_2 σε αυτή την περίπτωση;*

Έστω ότι το φ_1 είναι γεγονός στο οποίο αποστέλλεται κάποιο μήνυμα m και φ_2 είναι το γεγονός παραλαβής του μηνύματος. *Γιατί ισχύει ο ισχυρισμός για τα φ_1 και φ_2 σε αυτή την περίπτωση;*

Ισχύει η μεταβατική ιδιότητα για τη σχέση \leq σε διανύσματα.

Διανυσματικά Ρολόγια (Vector Clocks)

Θεώρημα

Έστω ότι a είναι μια αυθαίρετη εκτέλεση και έστω ότι φ_1 και φ_2 είναι δύο οποιαδήποτε γεγονότα της a . Αν $VC(\varphi_1) < VC(\varphi_2)$ τότε $\varphi_1 \rightarrow \varphi_2$.

Απόδειξη

- ◇ Έστω ότι φ_1 και φ_2 είναι δύο γεγονότα που συμβαίνουν ταυτόχρονα, το φ_1 από τη διεργασία p_i και το φ_2 από τη p_j , $p_i \neq p_j$.
- ◇ Ας υποθέσουμε ότι $VC_i[i](\varphi_1) = m$. Ο μόνος τρόπος το i -οστό στοιχείο του διανυσματικού ρολογιού μιας διεργασίας p_j να αποκτήσει τιμή τουλάχιστον m είναι μέσω μιας αλυσίδας μηνυμάτων που ξεκινούν από την p_i (είτε στο γεγονός φ_1 ή σε επόμενο γεγονός).
- ◇ Μια τέτοια αλυσίδα θα συνεπαγόταν ότι τα φ_1 , φ_2 δεν συμβαίνουν ταυτόχρονα. Άτοπο!
- ◇ Άρα, $VC_j[i](\varphi_2) < m \Rightarrow VC_i(\varphi_1)$ δεν μπορεί να είναι μικρότερο από $VC_j(\varphi_2)$.