**Fault Tolerance**

Με τη τεράστια εξάπλωση του cloud computing καθώς και τη ραγδαία αύξηση της χρήσης του σε ποικίλες περιπτώσεις, η ανάγκη διατήρησης του QoS (Quality of Service), της σταθερότητας, της αποτελεσματικότητας και την ασφάλειας, είναι συνεχώς αυξανόμενη. Σε μια τέτοια πραγματικότητα, η εμφάνιση των ανεπιθύμητων λαθών και συμπεριφορών αυξάνεται αντίστοιχα. Η δυνατότητα ενός συστήματος να αντιμετωπίσει αποτελεσματικά τα προβλήματα αυτά ονομάζεται **Fault Tolerance**.

Γενικότερα, ονομάζουμε μια προσέγγιση ως προσέγγιση «γενικού σκοπού», όταν αυτή στοχεύει να αντιμετωπίσει τα προβλήματα ενός συγκεκριμένου stack της εφαρμογής, αποκρύπτοντας πλήρως τα προβλήματα αυτά αλλά και τον μηχανισμό επίλυσης, από τον τελικό χρήστη. Εδώ, έχουν σημασία δύο παράμετροι:

* Τα σφάλματα προκύπτουν σε άγνωστο και τυχαίο χρόνο
* Το σύστημα έχει σχεδιαστεί ανεξάρτητο, μη λαμβάνοντας υπόψη τα ρίσκα αποτυχίας και πιθανών σφαλμάτων.

Μια τεχνική γενικού σκοπού, αποτελεί το **Process Checkpointing**. Ο στόχος της τεχνικής αυτής αποτελεί η **σωστή, ασφαλής και σταθερή αποτύπωση ενός στιγμιότυπου της κατάστασης του συστήματος**, τη δεδομένη χρονική στιγμή. Στη πραγματικότητα, μια εφαρμογή αποτελείται από πολλά threads σε επίπεδο χρήστη αλλά και σε επίπεδο συστήματος, δηλαδή λειτουργεί σε παράλληλα επίπεδα. Η συνήθης υλοποίηση του process checkpointing περιέχει έναν μηχανισμό προσωρινού κλειδώματος, παγώνοντας έτσι στιγμιαία όλα τα εκτελούμενα threads, ώστε να αποτυπώσει και να αποθηκεύσει τη συγκεκριμένη κατάσταση του συστήματος. Σε ένα μελλοντικό επερχόμενο σφάλμα, η εκάστοτε εφαρμογή μπορεί να επιστρέψει στη προηγούμενη ασφαλή κατάσταση της και να συνεχίσει κανονικά την εκτέλεση της. Όλα αυτά γίνονται σε πολύ μικρό χρονικό διάστημα, μη αισθητό από τον χρήστη, δίνοντας έτσι την εντύπωση πως το σφάλμα δεν συνέβη ποτέ. Σε περίπτωση που γίνει κάτι τέτοιο αντιληπτό, τότε αφορά κάποιο σοβαρότερο σφάλμα, όμως η ασφάλεια τον δεδομένων καθώς και η ταχύτατη επιστροφή στη κανονική λειτουργία είναι εγγυημένη. Τέλος, μια process checkpointing τεχνική διακρίνεται με βάση 3 παραμέτρους:

* Σε ποιο stack της εφαρμογής υλοποιείται
* Με ποιο τρόπο υλοποιείται στο συγκεκριμένο stack
* Με ποιο τρόπο πραγματοποιείται η αποθήκευση του checkpoint

[Book] : Fault Tolerance Techniques for High-Performance Computing

By Thomas Herault & Yves Robert / Springer