

# Glossaire de SdF



## • **AEEL** - Analyse de Risque

L'Analyse des Effets des Erreurs du Logiciel est une analyse de risque issue d'une adaptation de l'AMDE au logiciel. Cette analyse permet de caractériser la criticité d'un logiciel au travers de ses constituants. Le but de cette méthode est d'exposer aux concepteurs les points critiques identifiés, et de permettre aux personnes chargées de la validation d'affiner leur démarche. L'AEEL doit être entreprise dès la phase de conception préliminaire afin que les propositions de modifications du logiciel soient prises en compte au plus tôt.

Un document PostScript décrivant les AEEL est disponible [ici](#).

---

## • **AMDE(C)** - Analyse de Risque

L'Analyse des Modes de Défaillance et de leurs Effets est une méthode d'analyse de risque systématique des causes et des effets des défaillances pouvant affecter les composants d'un système.

L'AMDE(C) est l'extension de l'AMDE à l'analyse de la criticité. Cette analyse consiste à déterminer l'importance de chaque mode de défaillance compte tenu de son influence sur le comportement normal du système ; elle permet d'évaluer l'impact des défaillances sur la fiabilité et la sécurité du système. L'objectif est d'identifier les barrières existantes pour limiter les effets des défaillances. A défaut de tels moyens de protection, il est proposé des mécanismes supplémentaires de détection / recouvrement pour tolérer les fautes. On peut également proposer des moyens de contournement basé sur l'utilisation de procédures opérationnelles. L'AMDE(C) peut-être réalisée à partir d'une représentation fonctionnelle du système ou du niveau structurel (décomposition du système en composants matériel / logiciel).

---

## • **Analyses de Risque**

Au cours de sa vie opérationnelle, un système contient inéluctablement des fautes de conception, quel que soit l'effort de validation fourni. Le zéro faute est une gageure et n'est pas un objectif réaliste étant donné les coûts de développement que cela induirait. Il est donc important, pour les systèmes dits critiques, d'évaluer le risque auquel sont soumis les utilisateurs de ces systèmes. Pour cela plusieurs méthodes sont généralement utilisées selon la phase du cycle de vie du système, l'origine des erreurs (matériel / logiciel) et le type de résultat attendu.

Les analyses de risque se fondent sur des techniques telles que les :

- AMDE(C) : Analyse des Modes de Défaillances et de leurs Effets (et Criticité)
  - AEEL : Analyse des Effets des Erreurs du Logiciel
  - APR : Analyse Préliminaire de Risque
- 

## ● **APR - Analyse de Risque**

L'Analyse Préliminaire des Risques est une méthode d'identification et d'évaluation des risques, de leurs causes, de leurs conséquences et de la gravité des conséquences. L'objectif de cette analyse de risque est d'en déduire les moyens et les actions correctives permettant d'éliminer ou du moins maîtriser les situations dangereuses et accidents potentiels mis en évidence. L'APR est particulièrement intéressante dès les premières phases du cycle de vie des nouveaux systèmes pour lesquels on ne peut pas s'appuyer sur le retour d'expérience.

---

## ● **Arbre de défaillance (arbre de faute)**

Diagramme logique utilisant une structure arborescente pour représenter les causes de défaillances et leurs combinaisons conduisant à un événement redouté (racine de l'arbre). La réduction des arbres de faute à partir du calcul des coupes minimales, permet d'identifier les chemins critiques. On en déduit les éléments matériel et logiciel du système dont la défaillance contribue le plus à la réalisation de l'événement redouté. Les arbres de faute peuvent être quantifiés, permettant ainsi de calculer l'indisponibilité ou la fiabilité du système modélisé.

---

## ● **Chaînes de Markov**

Les chaînes de Markov sont utilisées pour évaluer de façon quantitative la sûreté de fonctionnement des systèmes. Cette technique mathématique repose sur l'hypothèse que les taux de transition d'un système (taux de défaillance et taux de réparation) sont constants et que le processus stochastique modélisant son comportement est markovien (processus sans mémoire). Lorsque l'espace des états dans lequel peut se trouver le système est un ensemble discret, le processus markovien est appelé chaîne de Markov.

---

## ● **Criticité**

Voir Risque / Criticité .

---

## ● **Disponibilité**

La disponibilité est un des attributs de la sûreté de fonctionnement. C'est la propriété qu'à un système de

délivrer correctement le service (en terme de délai et de qualité) au moment où l'utilisateur en a besoin. La disponibilité est une mesure sans unité ; elle correspond à la proportion du temps de bon fonctionnement sur le temps total d'exécution du système.

---

### ● **Événement redouté (indésirable)**

Événement ne devant pas se produire ou devant se produire avec une probabilité peu élevée au regard d'objectifs de sûreté de fonctionnement.

Dans le cadre du programme Phidias-ODS France, l'événement redouté a été défini selon une approche « opérationnelle ». Il est tenu compte de la fréquence d'occurrence et de la durée à partir de laquelle un événement « indésirable » devient effectivement redouté.

---

### ● **Fiabilité**

La fiabilité est un des attributs de la sûreté de fonctionnement. Elle correspond à la continuité du service que le système doit fournir à ses utilisateurs, le système étant considéré comme non réparable. La fiabilité est également définie comme l'aptitude d'un système à accomplir une fonction requise, dans des conditions données, pendant une durée donnée. Toutes les défaillances de nature accidentelle sont prises en compte sans aucune discrimination vis-à-vis de leurs sévérités. Un exemple de mesure de fiabilité est le taux de défaillance, inverse du MTTF (Mean Time To Failure : temps moyen jusqu'à la première défaillance).

---

### ● **Fiabilité du logiciel - croissance de fiabilité**

Le logiciel contient inéluctablement des fautes de conception, aussi strictes que soient les règles de conception et de validation qui ont été appliquées au cours de son cycle de développement. L'aptitude d'un logiciel à fournir un service approprié en dépit de fautes résiduelles dans un environnement d'utilisation donné définit sa fiabilité. Pour un profil d'utilisation donné, l'amélioration progressive de cette aptitude à délivrer des services conformes aux spécifications traduit une croissance de fiabilité. Cette amélioration est généralement obtenue grâce aux corrections des erreurs apportées au logiciel. L'évaluation de la croissance de fiabilité du logiciel est réalisée à l'aide de modèles de fiabilité.

---

### ● **Fiabilité - Modèles**

Les modèles de fiabilité sont des modèles mathématiques qui permettent d'évaluer les mesures caractérisant la fiabilité du logiciel en évolution, à l'aide d'expressions relativement peu complexes. La plupart des modèles de fiabilités sont des modèles paramétriques. Le CENA utilise l'outil SoRel pour modéliser la fiabilité du logiciel.

---

## • **Maintenabilité**

La maintenabilité est un des attributs de la sûreté de fonctionnement. La maintenabilité d'un système traduit son aptitude aux réparations et aux évolutions, la maintenance devant être accomplie dans des conditions données avec des procédures et des moyens prescrits. Un exemple de mesure de maintenabilité est le MTTR (Mean Time To Recover : temps moyen de réparation ou de restauration du système dans l'état de bon fonctionnement).

---

## • **Réseaux de Petri**

Afin de d'analyser le comportement d'un système en présence de fautes, le CENA utilise les réseaux de Petri. Cette modélisation dynamique permet également d'obtenir des mesures de sûreté de fonctionnement, en assignant des valeurs numériques aux paramètres du modèle. Pour les besoins de ses modélisations, le CENA utilise deux types d'outils basés sur les réseaux de Petri :

- SURF-2, basé sur les réseaux de Petri stochastiques généralisés. Les transitions d'états sont régies par des lois exponentielles. Le calcul est effectué en transformant préalablement le réseau de Petri en une chaîne de Markov équivalente ;
- Miss-rdp (version non colorée et version colorée). Les calculs reposent sur le principe de la simulation de Monte-Carlo. On quantifie les temps moyens de séjour dans les différents états et les probabilités d'occurrence des événements à considérer. L'intérêt de cette approche est qu'elle permet de prendre en compte la plupart des lois de probabilité associées aux variables aléatoires des composants du système à modéliser. L'ajout de « couleurs » aux jetons augmente considérablement l'efficacité de la simulation.

---

## • **Risque / criticité**

La notion de risque est en quelque sorte une mesure d'un danger exprimée en fonction de l'occurrence d'un événement indésirable (probabilité, fréquence) et d'une mesure de ses effets ou de ses conséquences. Un danger est une situation dont les conséquences peuvent nuire à l'homme (blessure ou mort de personnes), à la société (perte de production, perte financière, etc.) ou à l'environnement (dégradation du milieu naturel et animal, pollution).

Une échelle de risque est souvent associée au danger afin de pouvoir les classer en niveaux de criticité. Pour cela, on considère le produit cartésien « fréquence d'occurrence x effets » de l'événement ou de la situation donnée.

---

## • **Safety Case**

La démarche safety case est une démarche d'assurance sécurité qui trouve son origine au Royaume-Uni. Elle est principalement utilisée dans le secteur du pétrole, chimie, nucléaire et transport ferroviaire. Elle peut également s'appliquer à n'importe quel type d'installations ou systèmes à risque, y compris les systèmes ATC. Cette démarche est utilisée par le concepteur et/ou l'exploitant pour démontrer à une autorité compétente que son installation ou système est sûr, et ce tout au long du cycle de vie du système. Durant la phase de conception, cette démarche facilite la traçabilité des études réalisées ; elle permet de bien gérer les évolutions de l'installation ou des systèmes en phase de vie opérationnelle.

Par extension, le safety case désigne aussi le dossier réglementaire regroupant l'ensemble des documents utilisés ou produits au cours de la démarche safety case.

---

## ● Sécurité

Il faut distinguer la sécurité-innocuité (safety, en anglais), de la sécurité-confidentialité (security, en anglais) :

- la sécurité-innocuité vise à se protéger des défaillances catastrophiques, c'est-à-dire celles pour lesquelles des conséquences sont inacceptables vis-à-vis du risque encouru par les utilisateurs du système ;
- la sécurité-confidentialité correspond à la prévention d'accès ou de manipulations non autorisées de l'information et concerne la lutte contre les fautes intentionnelles (virus, bombes logiques, chevaux de Troie, etc.). Elle vise également à garantir l'intégrité des informations fournies aux utilisateurs.

---

## ● Sûreté de fonctionnement

La sûreté de fonctionnement d'un système peut être définie comme étant la propriété qui permet à ses utilisateurs de placer une confiance justifiée dans le service qu'il leur délivre. L'utilisateur peut être un individu tel que l'Opérateur ou le Superviseur, ou un autre système matériel / logiciel ayant des interactions avec le système considéré. Selon les applications auxquelles le système est destiné, la sûreté de fonctionnement peut être vue selon des propriétés différentes mais complémentaires : fiabilité, disponibilité, sécurité, maintenabilité, confidentialité-intégrité. Des mesures ont été définies pour chacune d'entre elles.

Un document PostScript décrivant plus avant la sûreté de fonctionnement est disponible [ici](#)

---

## ● Tests de tendance

Les tests de tendance sont utilisés en fiabilité du logiciel pour obtenir des indicateurs de fiabilité, à partir des données de défaillances et déterminer les fluctuations de fiabilité dans le temps.

Pour cela on utilise des tests graphiques ou statistiques, appelés « tests de tendance ». Les tests

graphiques sont plus rudimentaires que les tests statistiques ; ces derniers offrent l'avantage d'indiquer pour certains, la tendance locale (au voisinage de t, temps d'observation) et globale de fiabilité. Le CENA dispose d'un outil d'évaluation de la fiabilité du logiciel, SoRel, permettant d'appliquer des tests de tendance à un échantillon de données de défaillances. SoRel a été développé par le LAAS-CNRS.

---

## ● **Tolérance aux fautes (erreurs)**

La tolérance aux fautes est mise en oeuvre par la détection et le traitement des erreurs. Le traitement des erreurs revêt deux formes :

- le « recouvrement » des erreurs, par l'utilisation de points de reprise. Le but est soit de ramener le système dans l'état où il se trouvait juste avant l'occurrence de la défaillance, soit de trouver un nouvel état à partir duquel le système peut fonctionner (généralement, en mode dégradé) ;
  - la « compensation d'erreur » grâce à l'utilisation de redondances pour permettre au système de continuer à fournir le service correct en dépit de fautes. Le principe consiste à dupliquer voire tripler les équipements ou composants logiciel les plus « critiques » et ceux contribuant aux services les plus importants à fournir aux utilisateurs. Une gestion de vote majoritaire ou de basculement sur des équipements « secours » est alors mise en oeuvre.
- 



Alain.Peytavin@cenatoulouse.dgac.fr

© CENA/SDF 1998