

La performance et ses méthode de calcul

L'objectif est de pouvoir aborder et comprendre les notions de performances au sein des système d'informations. Ces notions sont essentiels, car elle permette de pouvoir manipuler les leviers nécessaires afin d'éviter une perte d'argent lié à une qualité insuffisante du SI.

- les **temps de réponse** (response time)
- la **disponibilité** (availability)

On parle d'architecture à haute disponibilité à partir de 99,99 % de disponibilité.

Il est nécessaire de pouvoir chiffrer la disponibilité, pour ce faire nous devons définir certain concept :

- **l'uptime** : désigne le temps de bon fonctionnement ou temp écoulé depuis le dernier démarrage ou le dernier plantage
- **Le MTBF** (Mean Time Between Failures) : le temps moyen entre deux plantage. Il représente la mesure du taux de défaillances aléatoires, à l'exception des pannes systématique, dues par exemple aux défaut de fabrication ou de l'usure.
- **L'AFR** (Annualized Failure Rate) : il représente la proportion de composant à changer chaque année.
- **Le downtime** désigne le temps d'arrêt lié à un dysfonctionnement
- **Le MTTR** (Mean time to repair) : le temps moyen nécessaire au rétablissement du service.
- **L'AST** (Agreed Service Time) : exigence de continuité de service convenue avec la maîtrise d'ouvrage.

la **disponibilité** (A) peut se calculer de 3 manière différente :

$$A = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}$$

$$A = \frac{\text{UpTime}}{\text{UpTime} + \text{DownTime}}$$

$$A = \frac{\text{AST} - \text{DownTime}}{\text{AST}}$$

l'AFR en pourcentage se calcul de la manière suivante

$$\text{AFR}(\%) = \frac{8760}{\text{MTBF}} * 100$$

le MTBF peut être amélioré par les actions suivantes :

- redondance de l'infrastructure matérielle (router, firewall)
- redondance de l'infrastructure logiciel (cluster)
- redondance des sites de production

le MTTR peut être réduit par les actions suivantes :

- formation des équipes d'intégration et de production
- Définition de procédures d'exploitation standardisées
- Mise en place de solutions de monitoring et de supervision

Disponibilité	Indisponibilité / DownTime			
	secondes/an	minutes/an	heure/an	jours/an
90 %	3 155 760	52 596	877	37
98 %	631 152	10 519	175	7
99 %	315 576	5 260	88	4
99,8 %	63 115	1 052	18	
99,9 %	31 558	526	9	
99,99 %	3 156	53		
99,999 %	316	5		
99,9999 %	32			
99,99999 %	3			

Pour un système S composé de N élément E1 à En en **série** on calcule la disponibilité de la manière suivante :

$$A_s = A_{e1} * A_{e2} \dots * A_{en}$$

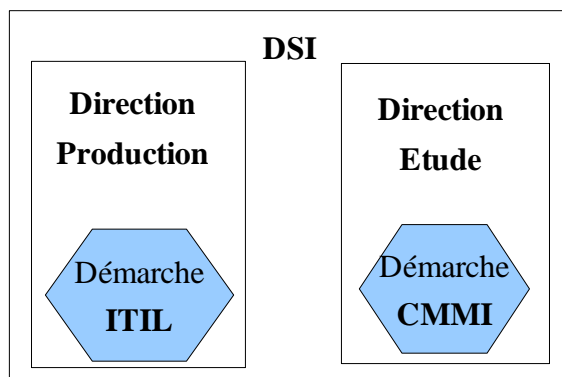
Pour un système S composé de N élément E1 à En en **parallèle** on calcule la disponibilité de la manière suivante :

$$A_s = 1 - (1 - A_{e1}) * (1 - A_{e2}) * \dots * (1 - A_{en})$$

- la **robustesse** (robustness)
- la **capacité à monter en charge** (scalability)

Organisation et Méthodologie des DSI

Comprendre les méthodes à la mode pour assurée la compétitivité dans l'entreprise



Gérer et optimiser la production

ITIL (Information Technology Infrastructure Library), un ensemble de bonne pratiques pour la fourniture de services informatique, propose un référentiel pour la gouvernance informatique. Cette dernière propose des pratiques sur

- la formalisation des contrat de service (SLA)
- le chiffrage des coût des service sous-jacent
- la gestion des demandes de service en amont des projets
- la gestion de la continuité de service
- la gestion de la capacité

ITIL définit des niveaux de maturité, permettant à l'entreprise utilisatrice de situer sa progression dans l'adoption de la démarche. Les cinq niveaux du référentiel sont :

- Le **niveau technologique** : ce niveau est basé sur des personnes clefs experte en technologies et irremplaçable.
- Le **niveau produit** ou service : les infrastructure sont documentées et des système d'alerte sont déployé.
- Le **niveau orienté client** : le suivi des infrastructures dans la durée permet d'anticiper les problème et les besoins.
- Le **niveau orienté business** : on est capable de définir et de gérer les SLA
- Le **niveau chaîne de valeur** : la maturité des infrastructure participant à la compétitivité de l'entreprise

ITIL fournit aussi un canevas pour formaliser les engagements auprès des maîtrise d'ouvrage :

- Le **Service level Requirement (SLR)** constitue l'expression des besoins utilisateur.
- Le **Service Level Agreement (SLA)** définit la convention de niveaux service
- L'**Operating Level Agreement (OLA)** définit les conventions de services entre les équipes de support. Il s'agit donc d'engagement entre équipe techniques, transparent pour l'utilisateur final.
- L'**Underpinning Contract (UC)** définit les contrats de sous-traitance.

Gérer et optimiser le cycle de vie du projet

L'approche **CMMI** (Capability Maturity Model Integration) fournit un guide des bonnes pratiques, permettant aux maîtrise d'oeuvre de déployer une méthodologie adaptée à leur contexte.

CMMI fournit 5 niveaux de maturité à l'instar d'ITIL:

- **Niveau initial** : l'aboutissement des projets repose sur quelques personnes compétentes.
- **Niveau reproductible** : les pratiques de gestion de projet sont documentées et partagées au sein de l'entreprise.
- **Niveau défini** : les solutions techniques sont normalisées et maîtrisées.
- **Niveau géré** : la performance des équipes est mesurée.
- **Niveau optimisé** : la performance des équipes suit un cycle d'amélioration permanente

Scalabilité du SI et Cluster

Il existe deux formes distinctes de clustering : la forme actif/passif et la forme actif/actif.

Actif/Passif

Les clusters Actif/Passif ou standby cluster, sont construits sur un principe de redondance passive qui consiste à doubler le serveur avec un second serveur en tout point similaire au premier. La particularité de ce mode réside dans le fait que, à un instant donné, seul un des serveurs du cluster travaille réellement (le serveur actif). L'autre serveur est dans un état passif. En cas de panne du serveur actif, le serveur deviendra actif et prendra le relais.

Les clusters Actif/Passif présentent les caractéristiques suivantes :

- amélioration du niveau de disponibilité puisque les serveurs sont montés en parallèle

- Pas de réel impact sur la scalabilité car seul un serveur est actif a un instant donné
- doublement des coût
- Relative simplicité de mise en oeuvre car pas d'accès concurrent à gérer

Actif/Actif

Les cluster Actif/Actif (takeover cluster) fonctionne sur un mode très simple : le travail est effectué par les serveurs actif. Si l'un d'eux tombe, il cesse de recevoir les requête. Les autres serveur actif s'occupent de récupérer la charge de travail.

Il est tout a fait possible de construire un cluster avec des configuration machine distinctes.

Les clusters Actif/Actif présente les caractéristiques suivantes :

- Amélioration de la capacité de montée en charge proportionnellement au nombre de serveurs qui sont ajouté au cluster.
- Amélioration de la disponibilité de l'application, montage de composant en parallèle
- Mise en oeuvre relativement complexe car ce type de configuration entraînent des difficultés technique, gestion de la concurrence pour les bases de données et gestion des sessions utilisateurs pour les serveurs d'application.

Les cluster Actif/Actif sont la meilleurs réponse à un problème de forte montée en charge. Si l'objectif n'est que d'améliorer la disponibilité cela n'a que peu d'intérêts.

Répartition de la charge sur les serveurs du cluster de manière dynamique, pour ce faire nous pouvons nous appuyer sur un composant appelé répartiteur de charge (Load Balancer).

cette approche présente les caractéristiques suivantes :

- les utilisateurs ne sont pas toujours reliés aux mêmes serveurs physiques.
- La répartition de charge se fait de manière précise.
- L'indisponibilité d'un serveur n'entraîne pas l'arrêt de l'application
- la répartition de charge se fait de manière transparente pour les utilisateurs

Session FailOver

Un point essentiel dans le cas d'un cluster Actif/Actif est le **session FailOver**, cela correspond à la capacité d'un cluster à ne pas perdre la session utilisateur en cas de défaillances d'un des serveurs. Cette fonctionnalité n'est pas obligatoires et de nombreux cluster ne l'implémente pas.

La mise en place d'un cluster Actif/Actif avec session FailOver nécessite une externalisation des

sessions. Le principe est le suivant : les requetes arrivent sur le répartiteur de charge et sont distribué entre les différents serveurs actifs sans tenir compte de la session utilisateur. Lorsqu'un serveur recoit une requete, il charge la session depuis un référentiel central qui contient toutes les sessions utilisateurs d'un cluster.

Cette forme de clustering permet de n'avoir aucune perte de session utilisateur.

L'externalisation des session peut se faire soit à l'intérieur d'une base de données soit en mémoire sur un serveur dédié. Le clustering n'étant pas décrit dans la normes J2EE, chaque éditeur est donc libre de proposer les solutions de son choix.

Il faut être attentif au serveur de session car celui ci devient un SPOF (Single Point Of Failure), c'est pour cela qu'il est nécessaire dans certains cas, de le mettre en cluster Actif/Passif. On constitue donc un cluster dans le cluster.

Il est intéressant de noter une propriété en matière de disponibilité, de la mise en place de cluster, en effet l'utilisation de plusieurs serveur physique pour une même application revient à un montage de composant en parallèle, la formule de calcul pour le taux de disponibilité s'applique donc ici, par exemple un cluster constitué de deux serveur ayant chacun une disponibilité de 98 % donne un dispo de :

$$A_s = 1 - (1 - 0,98)^2$$

$$A_s = 1 - 0,0004$$

$$A_s = 99,96 \%$$

La mise en place d'une solution de scalabilité horizontal e s'accompagne d'une augmentation sensible de la disponibilité. Bien entendu le principale objectif de ce type d'architecture reste essentiellement la suppression des SPOF (Single Point of Failure)

Il existe aussi des solutions Open Source comme **TerraCotta**, qui permettent une externalisation des sessions de façon transparente pour le serveur d'application.