

Le Manifeste Réactif

Publié le 16 Septembre 2014 (v2.0)

Dans différentes branches de l'industrie informatique, les organisations découvrent toutes indépendamment les unes des autres des modèles similaires de conception de logiciels. Ces systèmes sont plus robustes, plus résistants, plus flexibles et mieux positionnés pour répondre aux nouvelles demandes.

Ces changements sont causés par l'évolution rapide de ce que l'on attend des systèmes informatiques. Il y a seulement quelques années, une application de grande taille était composée de dizaines de serveurs, avait des temps de réponse de l'ordre de la seconde, plusieurs heures de maintenance et plusieurs gigaoctets de données. Les applications actuelles sont déployées sur des plates-formes variées, allant des simples appareils mobiles jusqu'à des grappes de serveurs distribuant leur charge à des milliers de processeurs multicoeurs. Les utilisateurs attendent des temps de réponse de l'ordre de la milliseconde et une disponibilité continue des services, les données sont dorénavant mesurées en petaoctets.

Notre conviction est qu'il faut une approche cohérente de l'architecture des systèmes modernes et que tous les aspects requis pour cela sont déjà reconnus individuellement: nous parlons de systèmes qui doivent toujours être disponibles pour répondre, être robustes, souples et orientés message. Nous les appelons Systèmes Réactifs.

Des systèmes développés selon ces directives sont plus flexibles, à couplage faible et [extensibles](#). Cela les rend plus simples à développer et plus propices au changement. Ils ont une plus grande tolérance vis-à-vis des [erreurs](#), et en cas d'erreur, réagissent de façon plus élégante, évitant ainsi des pannes catastrophiques. Ayant des temps de réponse rapide, les systèmes réactifs permettent aux [utilisateurs](#) une interaction accrue et ainsi une plus grande satisfaction.

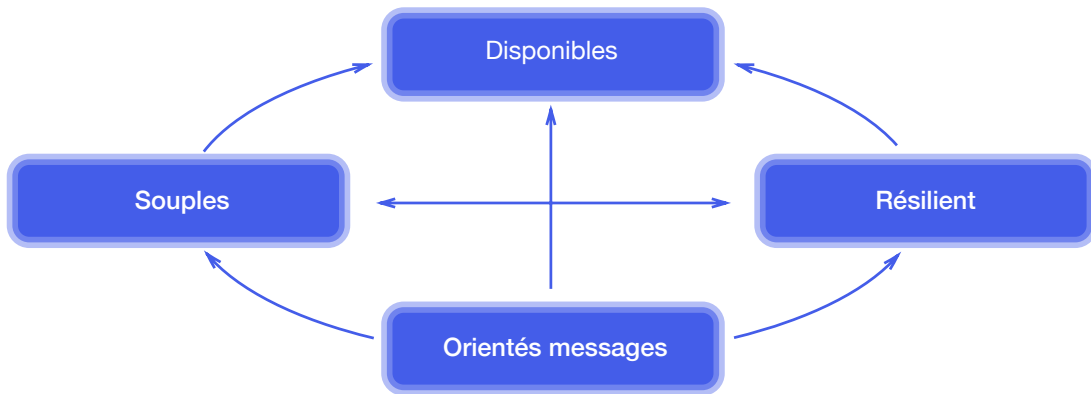
Les Systèmes Réactifs sont:

Disponibles (en anglais: responsive): Le [système](#) répond rapidement en toutes circonstances, si cela est possible. Cette disponibilité à répondre est primordiale pour le bon fonctionnement et l'utilisation d'un système, mais par-dessus tout, elle est basée sur la capacité d'un système à détecter rapidement des erreurs et à y répondre de façon effective. Ce type de système est en mesure d'assurer une qualité de service constante, reposant sur des temps de réponse minimaux et cohérents. C'est ce comportement cohérent qui permet de traiter plus simplement les cas d'erreurs et de gagner la confiance des utilisateurs, encourageant ainsi une interaction accrue avec le système.

Résilient (en anglais: resilient): Le système reste disponible en cas d'[erreur](#). Ce comportement ne se limite pas aux systèmes critiques - du moment qu'un système n'est plus en mesure de répondre après une panne, il ne s'agit pas d'un système robuste. Cette robustesse peut être atteinte grâce à la [réplication](#) de fonctionnalité, l'atténuation de cas d'erreurs, l'[isolement](#) de [composants](#) et la [délégation](#) de responsabilité. Les erreurs sont ainsi limitées à chaque composant, empêchant leur propagation à travers le système et permettant à chaque composant de récupérer son comportement normal sans autant nuire au système entier. Le processus de récupération est délégué à un autre composant (externe) et la disponibilité est garantie par de la réplication si nécessaire. L'utilisateur d'un composant n'a pas besoin de se soucier de gérer d'éventuelles erreurs.

Souples (en anglais: elastic): Le système reste disponible quelle que soit la charge de travail. Les Systèmes Réactifs sont en mesure de répondre au changement en terme de débit d'entrée en ajustant les [ressources](#) allouées à la réponse à ces entrées. Cela requiert une conception du système sans points de contention, nécessaire au partitionnement de données ou à la réplication de composants et permettant ainsi la distribution des demandes parmi ces composants. Les Systèmes Réactifs sont capables de mesurer et de faire évoluer leur distribution en temps réel en utilisant des algorithmes prédictifs et réactifs. Leur nature [élastique](#) leur permet d'utiliser des fonctionnalités en terme de plates-formes hardware et logicielles et ainsi de réduire les coûts opérationnels.

Orientés messages (en anglais: message-driven): Le système utilise le [passage de messages asynchrones](#) entre ses composants afin de garantir leur couplage faible, leur isolation, la [transparence de localisation](#) ainsi que la délégation d'[erreurs](#) à d'autres composants. L'utilisation de communication explicite à base de messages permet aux Systèmes Réactifs de bénéficier de la répartition de charge, d'élasticité et de contrôle de flux grâce à la surveillance et à l'ajustement des queues de messages dans le système, en appliquant de la [contre-pression](#) si nécessaire. La transparence de localisation utilisée dans le contexte du passage de message permet de gérer les erreurs avec le même modèle sémantique, peu importe si le système est réparti à travers une grappe de serveurs ou s'il s'agit d'un système local sur une seule machine. La communication [non-bloquante](#) donne la possibilité aux récepteurs de messages de n'utiliser des [ressources](#) que quand ils sont actifs, amenant à des systèmes plus efficaces.



De grandes applications sont toujours composées de plus petits composants et par conséquent dépendent des propriétés réactives de chaque composant. Cela veut dire que les Systèmes Réactifs emploient des méthodes de conception qui permettent d'appliquer les quatre propriétés réactives à toute échelle, les rendant composables selon ces quatre aspects. Les plus grands systèmes informatiques dans le monde dépendent d'architectures construites basées sur ces propriétés et répondent au besoin de milliards de personnes au quotidien. Il est temps d'appliquer ces principes de conception systématiquement dès le début de la conception de nouveaux systèmes au lieu de les redécouvrir à chaque fois.