

# Examen SOA : Services & Intégration

- Date : 13 novembre 2017
- Durée : 13h30 – 17h30 (4 heures)
- Aucun document autorisé ; barème donné à titre indicatif.

**Lisez tout le sujet (2 pages) avant de commencer à répondre aux questions ;**  
**Les questions sont identifiées en gras dans le texte du sujet ;**  
**Les différentes parties (au nombre de 3) sont indépendantes ;**  
**Pour toutes les questions, la justification de vos réponses**  
**est tout aussi importance que votre proposition.**

Vous devez répondre aux questions posées dans le sujet sur la copie d'examen fournie, avec d'éventuelles feuilles intercalaires. Vous ne pouvez pas sortir de la salle d'examen durant la première heure (avant 14h30), ni durant le dernier quart d'heure (après 17h15). **Toute fraude identifiée sera systématiquement transmise au conseil de discipline de l'UNS.**

Partie 1	Partie 2	Partie 3			
/4	/4	Q1 /3	Q2 /5	Q3 /2	Q4 /2

## Partie #1 : Prise de recul (0h45)

Le cours traite d'Architectures orientées services (SOA), avec un focus plus particulier en projet sur les piliers des architectures micro-services.

**Proposez une note de synthèse (1 page, 300 mots) décrivant les fondements de SOA, ceux des micro-services, puis analysez les synergies et points de divergences entre les deux.**

## Partie #2 : Idempotent Receiver (0h45)

Le texte suivant (extrait du livre de référence « *Enterprise Integration Patterns* » de la bibliographie du cours, p470) décrit le patron d'intégration « *Idempotent Receiver* ».

***Design a receiver to be an Idempotent Receiver - one that can safely receive the same message multiple times.*** The term idempotent is used in mathematics to describe a function that produces the same result if it is applied to itself, i.e.  $f(x) = f(f(x))$ . In Messaging this concept translates into a message that has the same effect whether it is received once or multiple times. This means that a message can safely be re-sent without causing any problems even if the receiver receives duplicates of the same message.

**Proposez une analyse critique et synthétique de ce patron (environ 1 page, 300 mots.** Pistes que vous pouvez suivre dans votre analyse (non exhaustives) : Quel est son

intérêt, quelles sont les différentes possibilités pour le mettre en œuvre dans une architecture logicielle, à quels types de scénarios peut-il répondre, ...

### Partie #3 : *Music Corp*<sup>1</sup> (2h30)

*Music Corp* est un disquaire en ligne (initialement un disquaire « physique » implanté à Albuquerque) ayant vocation à faire évoluer son marché en proposant à ses clients de découvrir de nouveaux artistes, sur la base de leurs goûts musicaux. La société souhaite automatiser cette découverte à l'aide d'algorithmes de *machine learning*, qui permettront aux clients n'ayant pas l'opportunité de visiter la boutique en Arizona de profiter de leur expérience. Les domaines couverts par le système vont du catalogue de produits offerts (CD, vinyles, matériel d'écoute) à l'achat en ligne et au suivi des stocks, en passant par la recommandation d'artistes et de chansons et la facturation.

Music Corp vous embauche pour définir une architecture micro-services permettant de mettre en place ce système.

1. **Définissez en pseudo-code les interfaces des services** (objets métiers inclus) dont vous avez besoin pour répondre à cette étude de cas. **Justifiez** le (ou les) paradigme(s) de conception de services choisis
2. Parmi ces services, **quels services sont implémentés par des flots d'intégration** ? Définissez en pseudo-code Camel ou via le langage graphique des EIP l'implémentation de ces flots d'intégration.
3. **Comment se déploie cette architecture** micro-services ? Quel intérêt pour le passage à l'échelle de Music Corp ?
4. **Quel rôle joue le bus de message** (s'il existe) dans cette architecture ?

---

<sup>1</sup> *Music Corp* est l'exemple fil rouge du livre "Building Fine grained systems with micro-services" cité dans la bibliographie du cours.