

Examen SOA1 : Services & Intégration

- Date : 18 novembre 2013
- Durée : 13h30 – 16h30 (3 heures)
- **Aucun document autorisé**
- **Barème donné à titre indicatif**

Vous devez répondre aux questions posées (texte en orange) 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 16h15). Toute tentative de fraude identifiée sera systématiquement transmise au **conseil de discipline de l'UNS**.

Lisez tout le sujet avant de commencer à répondre aux questions.

Les exercices 1, 2 et 3 sont indépendants les uns des autres.

Description du contexte : *My Broadcasting Company*

Vous venez d'être recruté(e) architecte par la société « *My Broadcasting Company* » (MBC¹), après un passage de plusieurs années au sein de la société MisterDiscount.

Le cœur de métier de MBC est la création d'écrans de diffusion d'information à destination de lieux public (voir Fig 1). Le produit principal développé par MBC est un outil permettant de se connecter à diverses sources d'informations (*e.g.*, Twitter, Facebook, flux RSS, prévisions météo) en fonction des préférences des usagers: quel *hashtag* lire chez Twitter, quel album photo récupérer chez Picasa, quel flux RSS doit-on lire, ... Vous êtes en charge de la partie arrière du produit, *i.e.*, la partie permettant la récupération des informations auprès des différentes sources et leur livraison à l'afficheur d'information développé par une autre équipe de MBC (application Java « classique »).



Figure 1 - Écrans de diffusion d'informations à Vaison-La-Romaine (bigair 8m², écran led 50")

¹ Ce sujet est très librement inspiré du projet YourCast, mis en œuvre depuis 2005 au sein des équipes de recherche MODALIS (Université Nice – Sophia Antipolis, CNRS) et ADAM (Université Lille 1, CNRS, Inria Lille-Nord Europe), en partenariat avec les sociétés LudoTIC et SupraLog. Plus d'informations sur <http://www.yourcast.fr>

Les sources d'informations disponibles, tout comme les paramètres de chacune de ces sources d'informations, diffèrent d'un client de MBC à l'autre. A l'heure actuelle, MBC redéveloppe une application J2EE complète pour chacun de ses clients. Lorsqu'un client a un besoin spécifique, MBC conçoit et met en œuvre un nouveau composant (*e.g.*, un assemblage d'EJBs). Par exemple, pour diffuser l'emploi du temps des étudiants de Polytech dans le hall principal du bâtiment Templiers Ouest, MBC a produit un assemblage d'EJBs déployés sur le serveur de Polytech pour interagir avec la base de données *HyperPlanning* déjà utilisée par l'école.

Exercice #1 : « SOA, or not SOA, that is the question. »

/ 5

Fier de sa lecture de la presse spécialisée, votre Président Directeur Général à vaguement entendu parler de l'approche « orientée service ». Il vous demande votre avis pour connaître les **avantages** et **inconvénients** pour MBC de migrer son architecture actuelle vers une approche à base de services, puisque vous étiez en charge d'une SOA à votre ancien poste.

Rédigez une note à l'intention de votre PDG d'une vingtaine de lignes maximum donnant votre point de vue sur l'approche SOA dans le contexte de la société MBC. Identifiez très clairement les gains et les risques de la migration vers une SOA si celle-ci venait à être décidée par MBC.

Exercice #2 : Architecture d'une preuve de concept

/ 12

La migration est en passe d'être décidée par la direction de MBC. Afin de limiter les risques, vous êtes chargé(e) de mettre en œuvre une preuve de concept (PoC) pour valider la faisabilité de la migration. De retour d'un déjeuner avec la direction, votre directeur technique (*Chief Technical Officer, CTO*) vous transmet les spécifications de la PoC à mettre en œuvre par votre équipe, dessinée sur une serviette en papier (Fig 2).

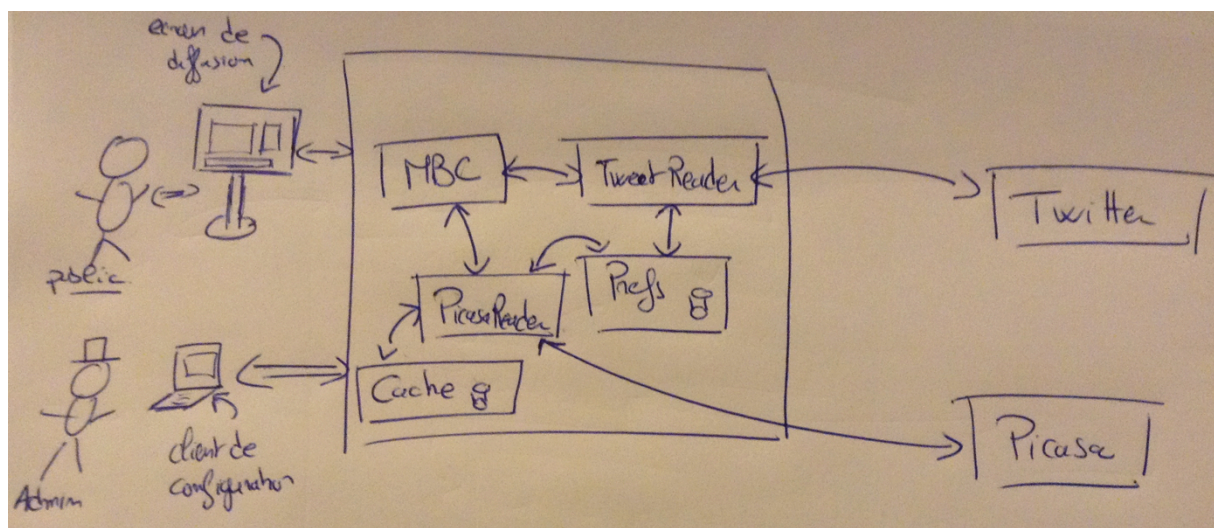


Figure 2 - Spécifications techniques de la PoC (photo d'une esquisse sur une serviette en papier)

Twitter et *Picasa* sont des entités externes, exposant des services permettant respectivement (i) la lecture et l'envoi de messages courts et (ii) le stockage et la lecture d'images. Au sein de votre système, votre CTO envisage 5 entités internes. D'un point de vue non fonctionnel, vous aurez besoin d'un système de mise en *Cache* pour les

informations, ainsi qu'un service de stockage des *Préférences* des utilisateurs (*e.g.*, quel album photo lire chez Picasa). L'entité *TweetReader* utilise les préférences stockées dans le système pour invoquer le service fourni par *Twitter Inc.* avec les bons paramètres (*e.g.*, je veux récupérer les hashtags «#PolytechSophia» et «#UNS»). Il en va de même pour l'élément *PicasaReader*, qui récupère les préférences stockées dans le système (*e.g.*, je veux récupérer l'album «Fête de la science 2013», mon login est «polytechnsa») pour interroger le service de *Google*. Finalement, une entité MBC sert de point d'entrée au client lourd utilisé pour diffuser les informations auprès du public. Un administrateur joue le rôle d'éditeur pour administrer le jeu de préférences stockées dans le système.

Exercice #2.1 : Visibilité métier des entités à définir

/1

Parmi les entités à définir dans votre PoC, certaines doivent être exposées à l'extérieur de la société MBC afin que le client lourd déployé chez le client puisse y accéder. Identifiez quels éléments doivent être exposés et donnez les raisons d'une telle exposition.

Exercice #2.2 : Analyse critique d'une interface de services existantes

/2

Vous disposez de la description d'interface (en pseudo-code) suivante pour le service *Picasa*.

```
service Picasa {  
  
  path: /service/picasa/v1.0  
  
  /** Structures de données et alias pour les types primitifs **/  
  struct UserInformation {  
    user : String ;  
    password : CypheredString ;  
  }  
  
  alias BigInteger ≡ AlbumId, PictId  
  alias Date ≡ ExpirationDate  
  
  /** Opérations disponibles dans le service **/  
  
  operation login(info: UserInformation) → ExpirationDate ;  
  operation logout();  
  
  // Les opérations suivantes doivent être précédées d'une invocation  
  // réussie de l'opération login, non encore expirée.  
  
  operation createAlbum(albumName: String) → AlbumId;  
  operation addPicture(id: AlbumId, pict: Byte[]) → PictId;  
  operation addComment(id: PictId, comment: String);  
  
  operation getAlbumContent(id: AlbumIdentifier) → PictId[];  
  operation getPictureURL(id: PictId) → URL;  
  operation getPictureComment(id: PictId) → String;  
}
```

Quels problèmes identifiez-vous dans cette interface ?

Exercice #2.3 : Enrobage (« Wrapping ») de l'interface de Picasa

/4

Dans un monde idéal, vous seriez en mesure de demander à Google de changer son interface, étant donné tous les problèmes que vous avez identifiés en #2.2. Ceci n'étant pas possible dans le monde réel, vous devez proposer un « enrobage » du service Picasa.

Proposez 3 versions différentes de l'interface idéale (dans un contexte SOA) du service Picasa : une approche RPC classique, une approche message et une approche ressource.

Exercice #2.4 : Caractérisation des éléments constituant la PoC

/5

Vous disposez des 3 scénarios suivants fournis par votre CTO (inscrits au dos de la serviette en papier de la Figure 2) :

- *En tant qu'administrateur, je peux déclarer des écrans qui sont identifiés par des noms unique (e.g., « Ecran du hall, Polytech Ouest », « Ecran K-Fèt étudiants »)*
- *En tant qu'administrateur, je peux visualiser (et éditer) les préférences stockées dans le système pour chaque écran;*
- *En tant que client lourd, je peux envoyer l'identifiant de l'écran sur lequel je m'exécute et obtenir du système les informations à afficher.*

Identifiez parmi les entités de la Figure 2 ce qui doit être mis en œuvre par (i) un service ou (ii) par un flot d'intégration. Pour chaque entité :

- **S'il s'agit d'un service, décrivez son interface en utilisant le pseudo-langage de la question #2.2 (RPC, message ou REST). Vous décrirez aussi brièvement pour chaque opération son comportement ;**
- **S'il s'agit d'un flot d'intégration, donnez la signature de l'opération implémentée et dessinez le en décrivant les étapes clés flot (en utilisant les symboles de Mule Studio et/ou ceux fournis en annexe).**

Exercice #3 : Exploitation de l'ESB

/ 3

Le passage à une architecture SOA est maintenant décidé. Vous êtes désigné volontaire pour la responsabilité de l'équipe de développement de la solution MBC-SOA. Fort de votre expérience passée chez MisterDiscount, votre CTO vous demande d'être innovant afin de proposer une exploitation « au mieux » de l'ESB déployé dans l'entreprise.

Exercice #3.1 : Diversification des entrées

/ 2

Un de vos clients reçoit chaque semaine le menu de sa cafétéria sous la forme d'un fichier Excel. Il souhaite avoir à disposition dans ses écrans de diffusion le menu du jour à la cafétéria.


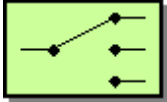
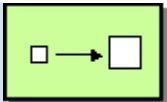
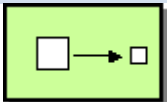
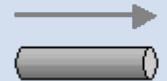
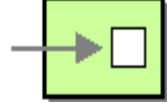







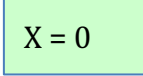
**Comment mettriez-vous en œuvre cette fonctionnalité en exploitant l'ESB ?
Donnez les interfaces des services et/ou les flots d'intégration nécessaires.**

Exercice #3.2 : Diversification des sorties

/ 1

Comment exploiteriez-vous le bus pour proposer de nouveaux moyens de diffusion d'informations à vos usagers ? Quel serait l'impact sur l'architecture de MBC ?

Appendix: Integration Components

Symbol	Name	Description
	Aggregator	How do we combine the results of individual, but related messages so that they can be processed as a whole?
	Content-based Router	How do we handle a situation where the implementation of a single logical function (e.g., inventory check) is spread across multiple physical systems?
	Content Enricher	How do we communicate with another system if the message originator does not have all the required data items available?
	Content Filter	How do you simplify dealing with a large message, when you are interested only in a few data items?
	Channel	How does one application communicate with another using messaging?
	Endpoint	How does an application connect to a messaging channel to send and receive messages?
	Database	How can we store data in a persistent way?
	Message	How can two applications connected by a message channel exchange a piece of information?
	Message Filter	How can a component avoid receiving uninteresting messages?
	Message Translator	How can systems using different data formats communicate with each other using messaging?
	Multi-channel (All)	How can the sender broadcast an event to all interested receivers?
	Scheduled Message	How can we send a given message at a given time?
	Splitter	How can we process a message if it contains multiple elements, each of which may have to be processed in a different way?
	Variable	How can we store a temporary value inside an integration flow?

Reference: <http://www.eaipatterns.com/toc.html>

Appendix: MBC deployed at Polytech

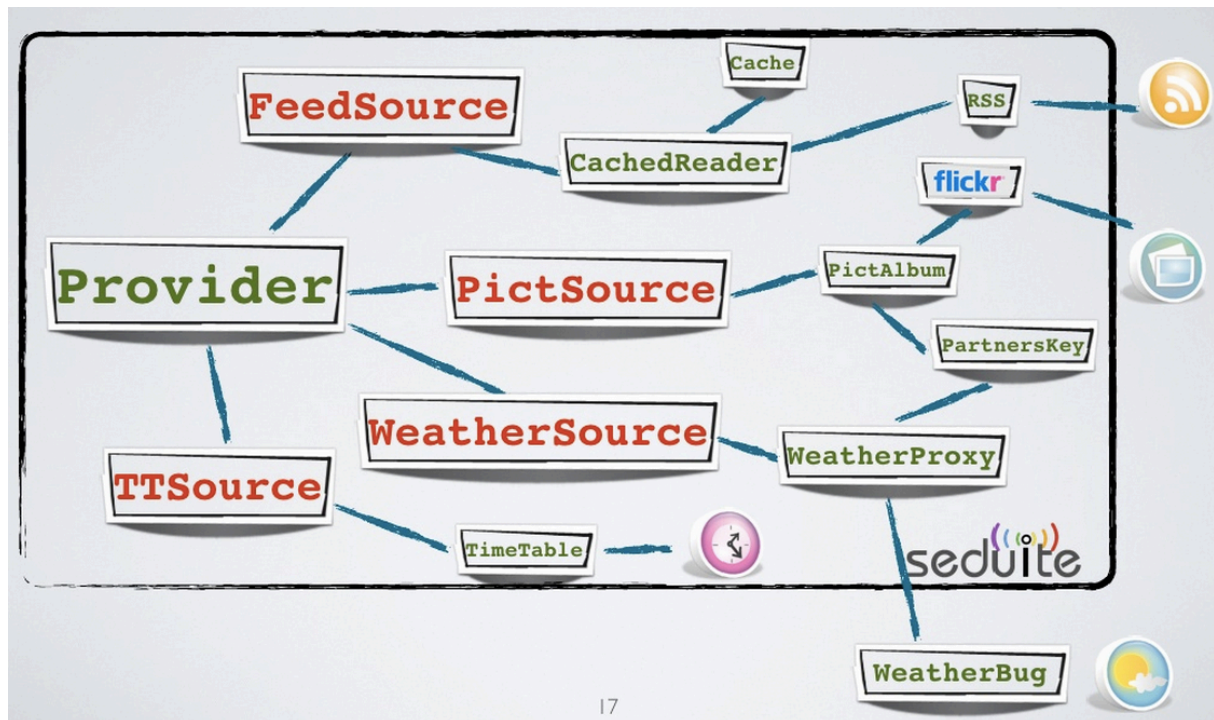


Figure 3 - Example of MBC system deployed at Polytech

The entry point of the component assembly is called a “Provider”. It consumes 4 sources of information:

- **FeedSource:** to collect RSS feeds contents from news providers;
- **PictSource:** to collect pictures from external services (e.g., Flickr);
- **WeatherSource:** to retrieve weather forecast data;
- **TTSource:** to connect to *HyperPlanning* and extract schedules.

Several business components are deployed to support the information retrieval process:

- **RSS:** a component to read remote RSS feeds;
- **PictAlbum:** a component to collect picture album contents from Flickr;
- **WeatherBug:** a component to connect to a remote forecast provider;
- **TimeTable:** a component to interact with <http://edt.polytech.unice.fr>

Several technical components are necessary to support this deployment:

- **Cache:** a cache server to save network usage;
- **PartnerKey:** a database to store external partners credentials (e.g., login).