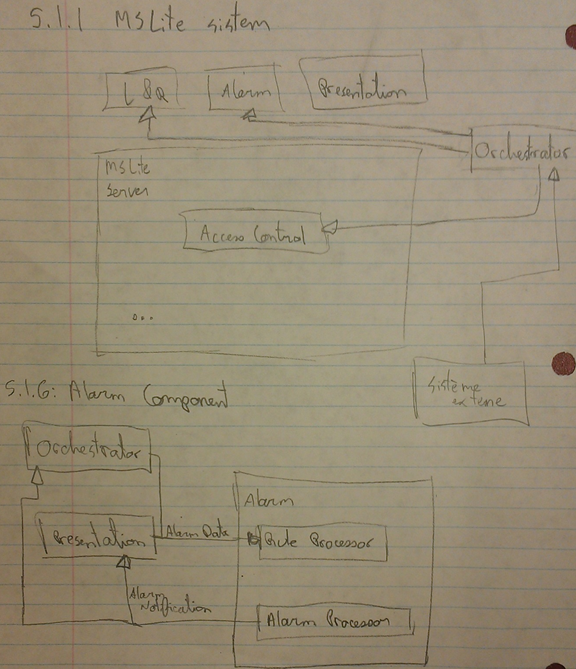
MSLite: Tactiques d'interopérabilité et de testabilité proposées par le groupe

# Nature de l’exercice

* Le groupe est divisé en petites équipes (3-6 personnes)
* Chaque équipe se voit attribuer un des deux attributs de qualité (ces attributs **ne sont pas** mentionnés dans l'exemple MSLite)
* Chaque équipe doit :
  1. **identifier et décrire** deux tactiques architecturales que vous utiliseriez pour supporter cet attribut dans MSLite;
  2. **justifier** votre choix de tactiques (pourquoi vous avez choisi celles-là, pourquoi pas d'autres tactiques pour le même attribut, ...);
     + vos réponses aux deux questions ci-haut haut doivent permettre d'établir que vous comprenez bien les tactiques retenues et leurs implications;
  3. **proposer des vues architecturales** modifiées démontrant l'implantation des tactiques dans l'architecture
     + **transmettez-moi vos vues** (fichiers électroniques, photos de dessins manuscrits, …) dès que possible afin que je les inclue ici

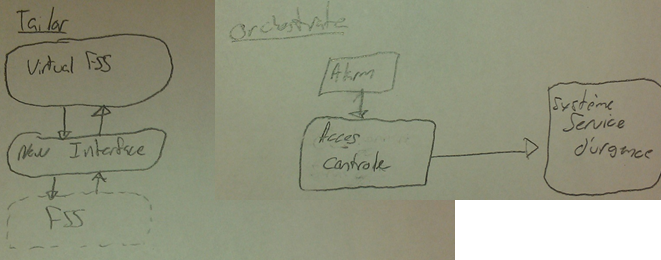
# Interopérabilité - Équipe 1

* **Discover service**: Le système a besoin d’un service capable de publier l’information sur les interfaces disponibles. Les systèmes externes désirant utiliser MSLite pourront faire appel à un annuaire pour connaître les services offerts. Pourrait être ajouté au composant Orchestrator (voir ci-bas).
* **Orchestrate**: Cette tactique permet au service externe d’accéder à une seule interface pour effectuer des actions complexes. Par exemple, l’action d’ajouter une alarme requière un accès à travers le Access Control et un possiblement un ajustement du module Présentation (GUI). Cette tactique est plus adaptée que le Tailor Interface parce que celle-ci est limitée à des composants définis. Comme le système MSLite contient plusieurs composants, une seule modification peut avoir un impact à plusieurs endroits. La tactique Orchestrate est très adaptée pour ce type de comportement. De plus, il est possible de vérifier les permissions d’un système externe grâce au Access Control avant de procéder à une quelconque modification. Ce composant peut même permettre de choisir les interfaces qui seront offertes.



# Interopérabilité - Équipe 3

* **Tailor interface**: permet d’ajouter ou d’enlever des fonctionnalités aux interfaces. Le système contient plusieurs interfaces pour permettre la gestion des événements. Cette tactique facilitera l’utilisation de ces interfaces lors de l’ajout d’alarme. Par exemple, lorsqu’il faut ajouter une nouvelle alarme après le lancement de l’application, ceci devrait faciliter l’intégration de nouvelles interfaces et la mise en commun des différents types de données. Ceci permet aussi de cacher les fonctions selon les droits de l’utilisateur.
* **Orchestrate:** permet de faire une interface globale qui gère plusieurs appels de services faits. Met en place un service qui permet de communiquer avec les services d’urgences. Par exemple, si la température d’une pièce est augmentée et l’alarme de feu est activé, communiquer avec le service incendie.



# Testabilité - Équipe 1

* **Specialized interfaces:** ajout d’une interface de tests entre le VFSS et le bus PubSub. Il s’agit d’accesseur (getters) sur diverses variables des VFSS. L’information est extraite du flux d’événements qui circule des VFSS vers Buffer. Condition nécessaire : les informations pertinentes sont présentes dans les événements. Pourrait être combiné avec « record/playback ». À considérer : impact sur le reste du système on n’a accès qu’aux variables présentes dans l’événement, pas celles du reste du système).
* **Sandbox:** Isolation des FSS et du Adapter Manager, afin de recréer des erreurs dans cette partie sans impact sur le reste du système. NOTE: Il n’est pas évident que ceci est possible avec l’architecture actuelle. Tout le serveur est actuellement dans un même processus. Considérer les impacts d’isoler le Adapter Manager dans un processus séparé.

## 

# Testabilité - Équipe 2

* **Sandbox:** << insérer description et justification ici >>
* **Tactique 2**
  + Justification :
  + Impact sur l'architecture:

## 