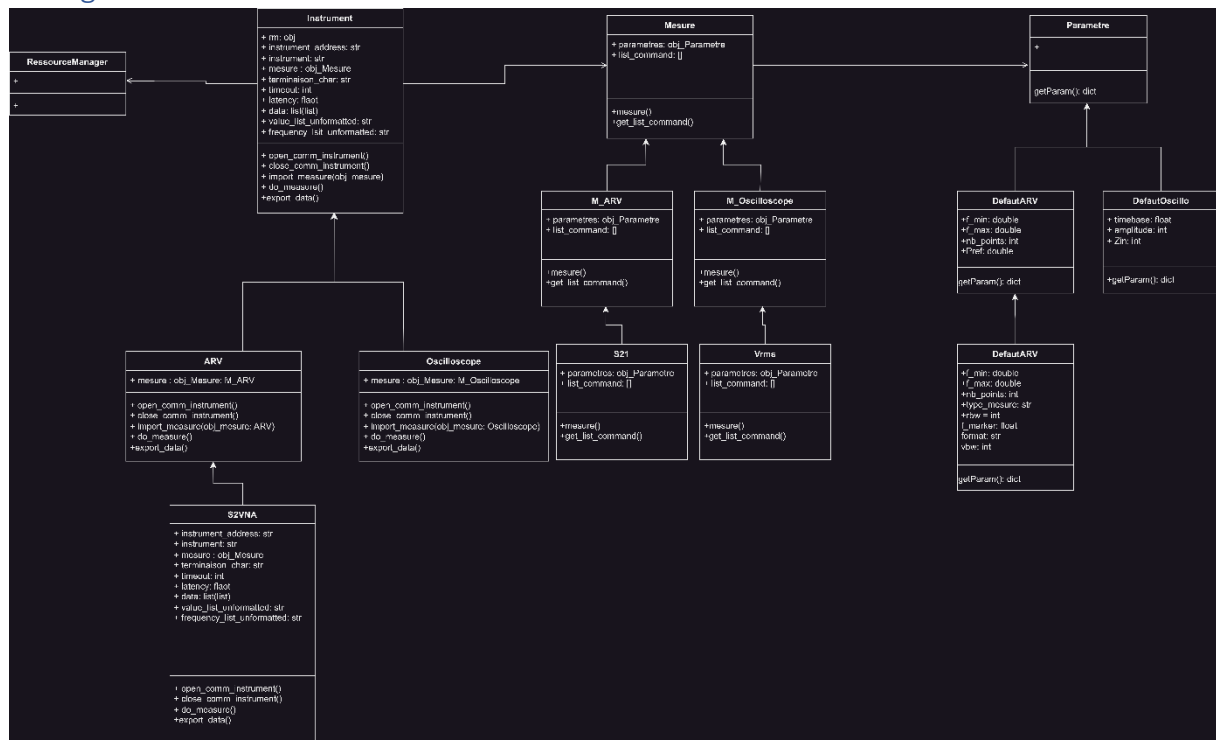


MAURY Nicolas

GROSSET Victor

Compte rendu SAE POO : Automatisation de mesure sur instrument en utilisant la bibliothèque Pyvisa suivant le paradigme de la programmation orientée objet.

1 Diagramme de classe :



2 Explication

Dans ce diagramme de classe, nous avons utilisé le fait qu'un appareil utilise des paramètres (représenté par la classe mère Paramètre) ainsi que des commandes matérialisant la mesure via VISA (représenté par la classe Mesure). Nous avons, au maximum, essayé de faire des héritage d'interdace afin de limiter le couplage mère/fille et ainsi simplifier le processus de maintenance/modification du code.

La classe Instrument gère tout ce qui est inhérent à l'instrument. En d'autres termes, la classe gère uniquement la communication avec l'appareil.

Chaque classe mère, est abstraite pour forcer l'implémentation par les classes filles. Le diagramme est encore incomplet, mais nous pensons séparer les classes mère en sous classes de famille d'instrument.

Par exemple, la classe fille ARV hérite d'Instrument et ne peut être associé qu'à des objet héritant des classes filles de Mesure ne concernant que les ARV. Cependant, cela nous demande un temps de réflexion supplémentaire afin de choisir une méthode qui ne nuit pas à la facilité de modification du code. Voici en tout cas comment nous pensions faire cette implémentation.

Un problème qui subsiste dans le changement d'objet mesure dans Instrument. Il faudrait simplifier le processus pour enchaîner des mesures sur un même équipement.

3 Et après ?

Plusieurs choses restent à faire pour améliorer cette implémentation :

1. Instrument devrait être composé de Mesure et de Parametre et non composé de Mesure lui-même composé de Parametre.
2. Actuellement, dans le code et non dans le diagramme, la classe Instrument s'occupe du formatage des données retournées. Il faudrait ajouter une classe Traitement composée d'Instrument.
3. Un Encoder/Decoder JSON devrait être implémenter pour charger des Parametre et Mesure dans Instrument.
4. Vérifier l'accès en lecture écriture des fichier exportés.
5. Faire une **documentation** !

4 Conclusion

Au bout des ces 6 heures de SAE, nous avons pu implémenter une esquisse (pas parfaite mais fonctionnelle) d'un code d'automatisation de mesure sur instrument en utilisant la bibliothèque Pyvisa suivant le paradigme de la programmation orientée objet. L'ajout de nouveaux objets de type Mesure, Parametre et Instrument est suffisamment aisé et le code assez clair à lire. Cependant, le technicien ne peut s'en remettre qu'au code pour vérifier ses manipulations. Il doit rester maitre des opérations.

Il reste des améliorations à effectuer, certes mais il nous a permis de manipuler la POO sur un projet concret qui ouvre de nouveaux horizons (interface graphique par exemple pour interagir avec nos classes).