FORMATION

PyMoDAQ – Pourquoi, pour qui, comment ?

Formation **EN PRESENTIEL** assurée par **Sébastien WEBER**
Soutenue par la Mission pour les Initiatives Transverses et Interdisciplinaires (MITI)

Dates : lundi 9 et mardi 10 septembre 2024 à RENNES

Pré-requis : savoir coder en python et connaitre la notion de programmation orientée

Lien d'inscription : https://lime.dr17.cnrs.fr/index.php/873341?lang=fr

Date limite inscription : 12/07/2024

Nombre de participants : 10-12 maximum

****

DESCRIPTION ACTION :

PyMoDAQ est une suite logicielle libre, écrite dans le langage python et utilisée pour l'acquisition automatique de données en fonction de paramètres expérimentaux multiples.
Conçu pour être utilisé sans qu'il soit nécessaire de programmer, il possède une interface graphique générique pour le contrôle des instruments avec des extensions spécifiques qui lui assurent une grande modularité. Il intègre en natif un stockage des données compatible avec les défis de l'open data et de la traçabilité.
Lors de ces journées, nous aborderons deux points essentiels au passage à PyMoDAQ :

- son installation/utilisation
- l'ajout de nouveau matériel à sa bibliothèque d'instrument.

PROGRAMME DETAILLE :

Jour 1:

1. Démonstration et utilisation "classique" de PyMoDAQ
   a. dashboard et preset
   b. extension: scan
   c. exploration des données

2. Installation/utilisation des fonctionnalités
   a. Introduction

   b. (TP): Installation
      i. Vérification que chacun peut lancer PyMoDAQ
      ii. les différentes façons de lancer les applis PyMoDAQ

   c. Les Modules de Contrôles
      i. DAQ_Move
      ii. (TP): MockTau
      iii. DAQ_Viewer
      iv. (TP): DAQ2D Mock
      v. DashBoard
      vi. (TP) Configurer un preset avec 2 actuateurs MockCamera: Xaxis, Yaxis et 1 détecteur 2D MockCamera: leur donner un nom caractéristique

d. Les extensions: le DAQ_Scan
   i. (TP) Faire différents Scan: visu live?
   ii. ROIs
   iii. Overshoot
   iv. (TP) Application

e. Les options du Dashboard
   i. Configuration
   ii. ROI Modes
   iii. Remote/shortcuts
   iv. Extensions
   v. Updates

Jour 2 :

3. Les plugins
   a. Liste des plugins et le PluginManager

   b. Utilisation de GitHub
      i. cloner le repos pymodaq_plugins_template
      ii. forker et cloner le repo: pymodaq_plugins_teaching
      iii. l'installer en mode éditeur (pip install -e .) instructions

   c. (Présentation) Présentation de la structure des plugins

   d. (Présentation) Utilisation des templates

   e. (TP) Ecriture de plugin simulant un spectromètre (wrapper Spectrometer)
      i. écrire un plugin actuateur DAQ_Move_Monochromator
      ii. Ajout de Settings: info, grating et tau
      iii. écrire un plugin détecteur 1D DAQ_1DViewer_Spectrometer
      iv. Ajout de Settings: info, lambda0 et width

   f. (Présentation et TPs) Notion de plugins maître/esclaves. Comment les configurer. Utilisation des plugins écrit précédemment dans un dashboard et avec le DAQScan

   g. Notions sur les wrappers instruments
      i. driver fournisseur
      ii. google est mon meilleur ami
      iii. Bibliothèques de driver: instrumental-lib, pylablib, pymeasure, instrbuilder, instrumentkit, …
      iv. Je m'inspire d'autres plugins

CNRS
Délégation Bretagne Pays de la Loire
Parc Alcyone – 1 rue A. et Y Meynier
35069 RENNES Cedex
www.cnrs.fr

Service des Ressources Humaines
Pôle FORMATION
Karine ROBIN
T. : 02 99 28 68 19
formation@dr17.cnrs.fr