LabVIEW INTERFACE FOR MADOCA II
WITH KEY-VALUE STORES IN MESSAGES
Takahiro Matsumoto, Yukito Furukawa, Yousuke Hamada, Tomohiro Matsumita
JASRI/SPRING-8, Hyogo, Japan

Abstract

We redesigned MADOCA II–LabVIEW interface for easy to use to spread the benefits of MADOCA II into many LabVIEW users in SPring–8. LabVIEW VIs were built with Dynamic Link Libraries for better maintainability. Message and various data in MADOCA II are easily accessed with key-value stores and messaging processes are simply decomposed with the VI. MADOCA II applications can be easily built with the new LabVIEW interface. We plan to apply MADOCA II–LabVIEW in control applications in SPring–8 such as image monitoring at experimental stations.

Developed VI components of MADOCA II

- Messaging processes are simply decomposed for clients and servers (Equipment Manager)
- Easily programmed with 23 VIs in total
  - Open/Close, Send/Receive, Parameter, Error
  - Message Interface (SVOC message + Various data)

Key Value Stores

- Improved for the flexibilities with
  - Various data (image data etc.) attached in a message
  - Control applications on Windows
- Implemented into SPring–8 and SACLA DAQ system in 2013

MADOCA II LabVIEW Interface

- Benefits to have unified control framework in SPring–8
  - Many LabVIEW users especially in experimental stations
  - Developed easy to use interface
- Dynamic Link Library (DLL) is used to build LabVIEW VIs
  - Easily upgraded by replacing the DLL
  - Python interface is also available from the DLL
- Key-Value Stores are used to easily manage message and various data in MADOCA II
  - An internal buffer in the DLL to intermediate the information

Available Data Format

- Data can be accessed through Key–Value Stores
  - Value, Array, Hierarchical structure
- MessagePack to serialize the data in the messaging
- Data format is defined for each case
  - Image data, Waveform data, Camera controls etc.

ex.) Data format for the image data

<table>
<thead>
<tr>
<th>Key</th>
<th>Data type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>image_data_type</td>
<td>string</td>
<td>“MONO”, “RGB”, “RGBA”</td>
</tr>
<tr>
<td>image_width</td>
<td>int32_t</td>
<td></td>
</tr>
<tr>
<td>image_height</td>
<td>int32_t</td>
<td></td>
</tr>
<tr>
<td>image_depth</td>
<td>int32_t</td>
<td></td>
</tr>
<tr>
<td>image_num_type</td>
<td>string</td>
<td>“uint8_t”, “uint16_t”, “uint32_t”, “float”, “double”</td>
</tr>
<tr>
<td>image_data</td>
<td>defined by</td>
<td></td>
</tr>
<tr>
<td>image_pixel_order</td>
<td>string</td>
<td>“lefttop”, “leftbottom”</td>
</tr>
</tbody>
</table>

- Applied to image data in two–dimensional interferometer with MADOCA II
  - A. Kiyomichi et. al, Proc. of ICALPCS 2013, p.78

Camera Image Viewer with MADOCA II LabVIEW Interface