

Intro to iDigi Dia

Jordan Husney, Digi International



Digi ESP IDE for Python & iDigi Dia



iDigi Dia: What is it?

- Extensible framework for remote data acquisition, control, and presentation of device data on ConnectPort X gateways
- Collects data from any device which can communicate with a Digi gateway
- Re-presents this data to upstream applications, including the iDigi platform, in a fully customizable way

iDigi Dia: Why is it needed?

- Most users need to do a variation of the same thing:
 - Collect data
 - Transform data
 - Log data
 - Present layer to next application over a network
 - Utilize some local control
- The iDigi Dia makes this easy

A Channelized View of the World

- Devices are made up of properties
- Device/property combinations become *channels* and are named *device.property*
- Channels are stored in the Channel Database and become available for logging
- For example, a GPS device may be composed of:
 - gps0.latitude
 - gps0.longitude



A View of iDigi Dia's Channels

```
=>> channel_dump

Device instance: aio_adapter_1

Channel                Value      Unit      Timestamp
-----
channel1_value         0.17556152 V    2008-10-15 15:05:38
channel2_value         0.17556152 V    2008-10-15 15:05:38
channel3_value         0.17556152 V    2008-10-15 15:05:38
channel4_value         0.17556152 V    2008-10-15 15:05:38

Device instance: lth_sensor_1

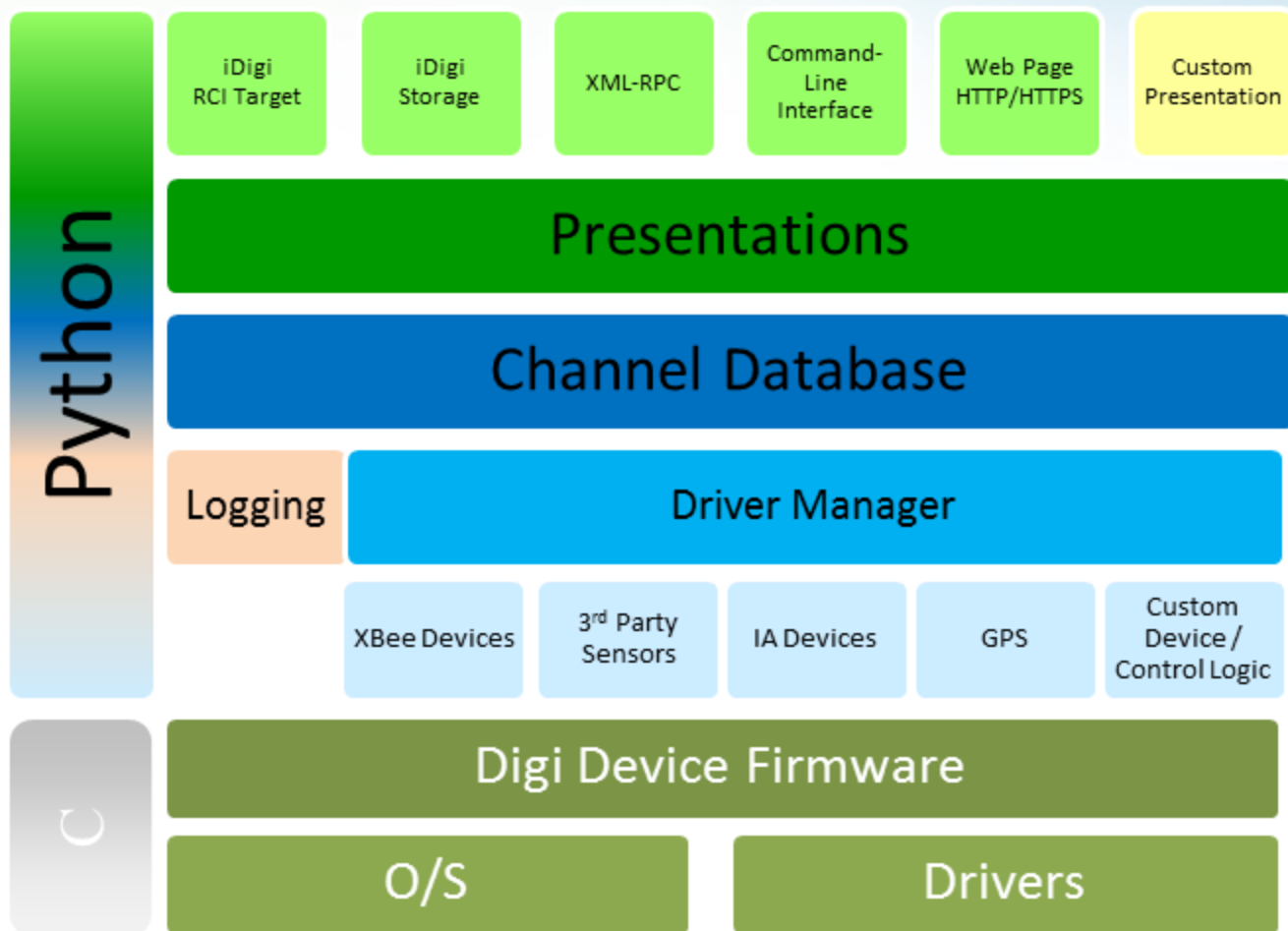
Channel                Value      Unit      Timestamp
-----
humidity               46.0890387 %    2008-10-15 15:05:39
light                  422.287390 lux   2008-10-15 15:05:39
temperature            19.0791788 C    2008-10-15 15:05:39
```

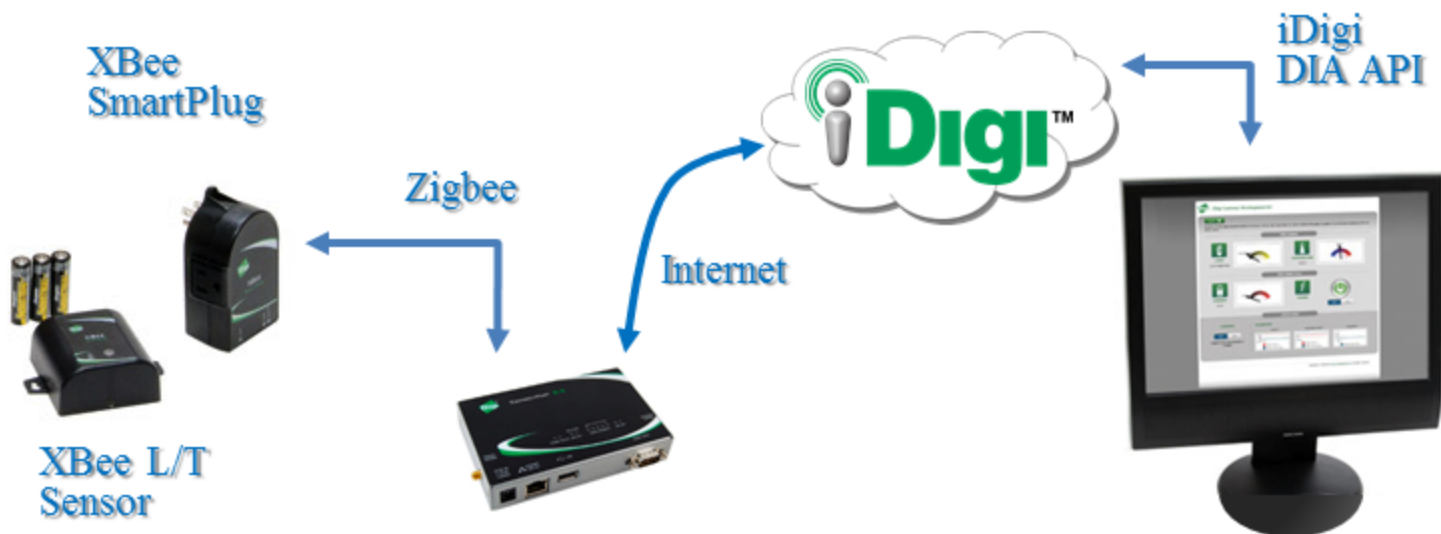
XBee Analog Adapter

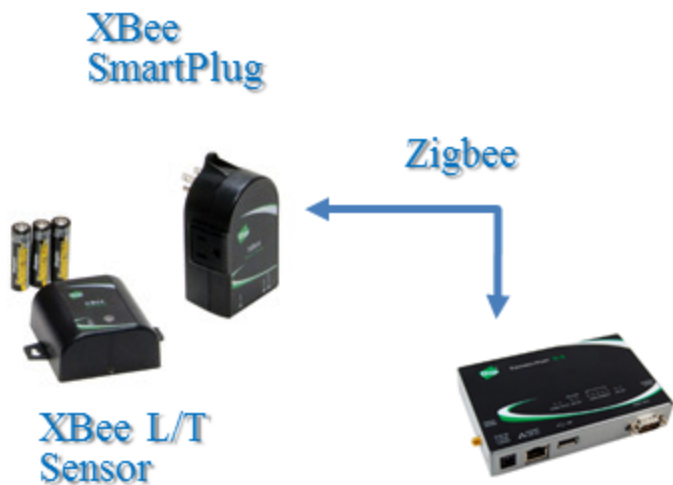
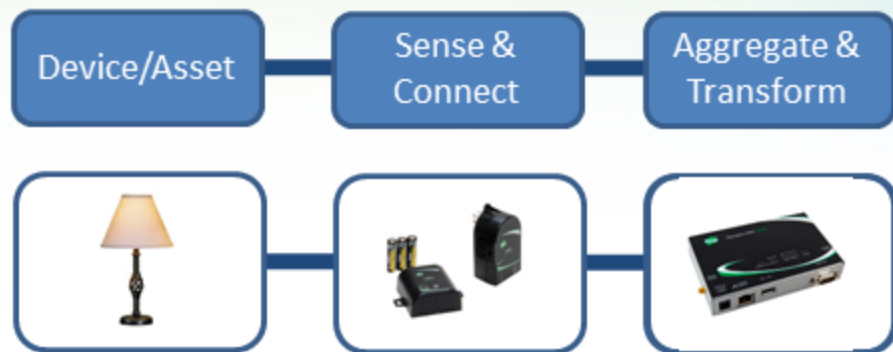
XBee Sensor L/T/H

More output follows...

iDigi Dia Architecture

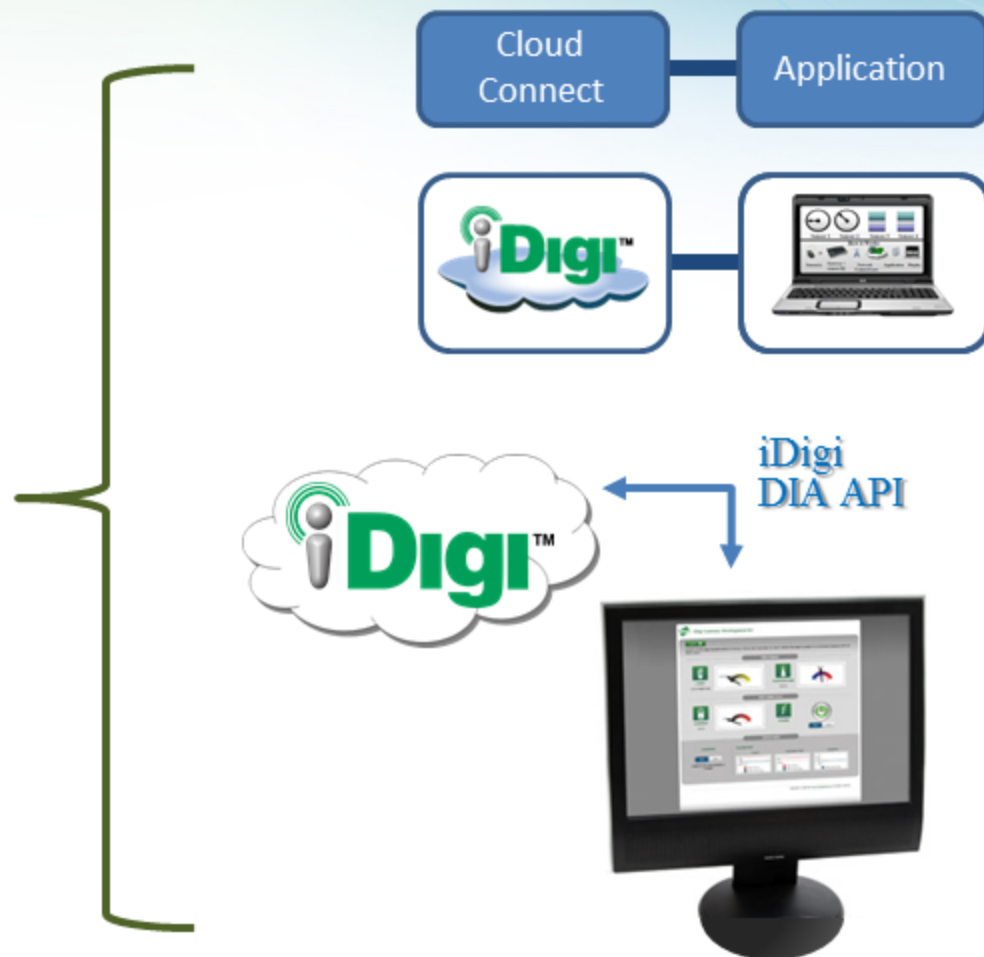




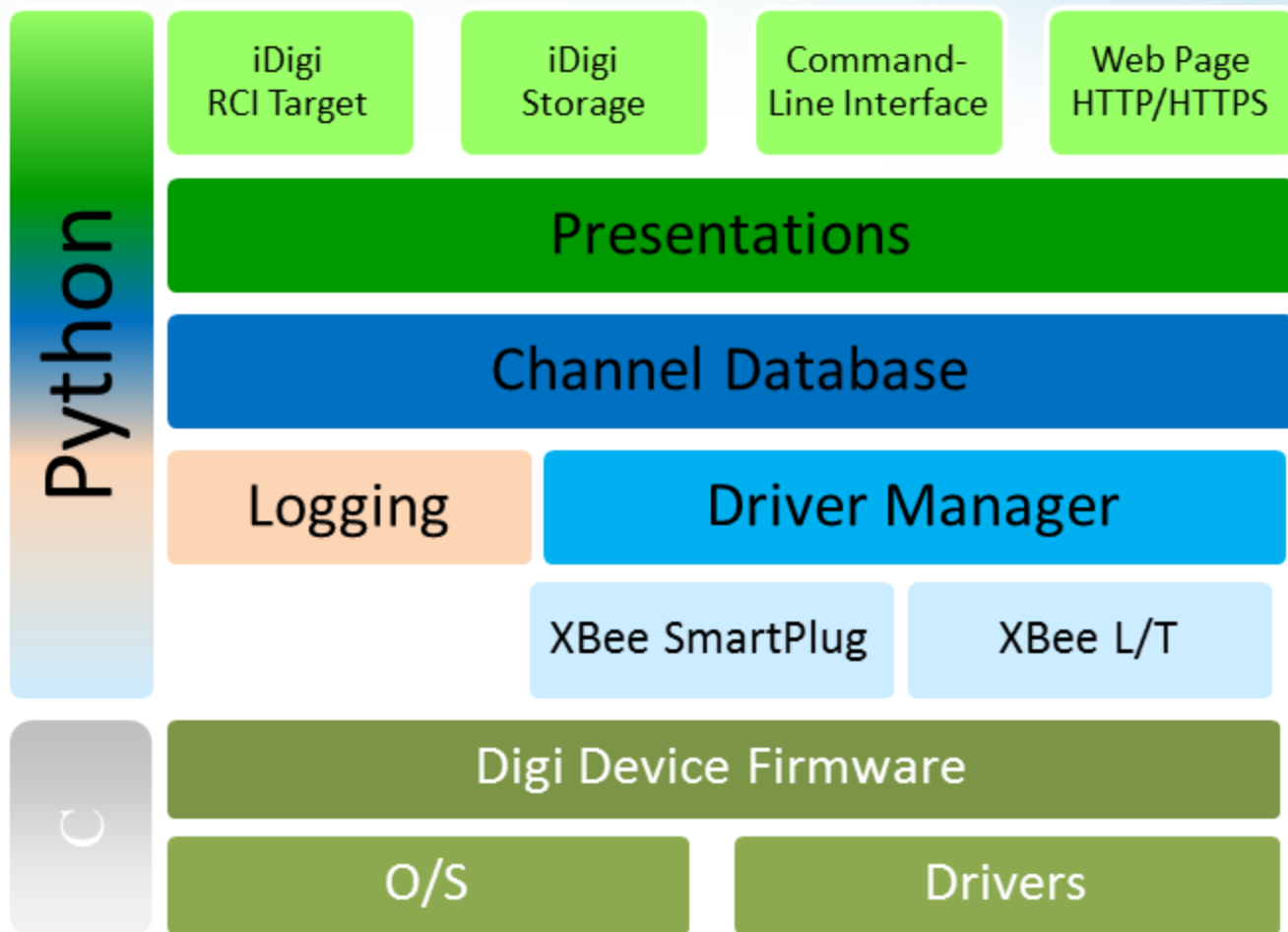


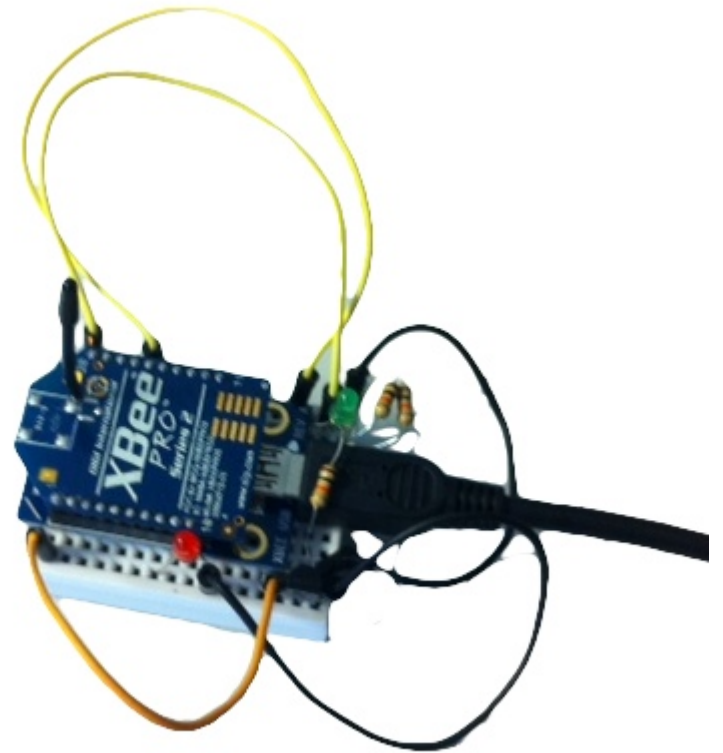
Collect
The
Data

Interface The App

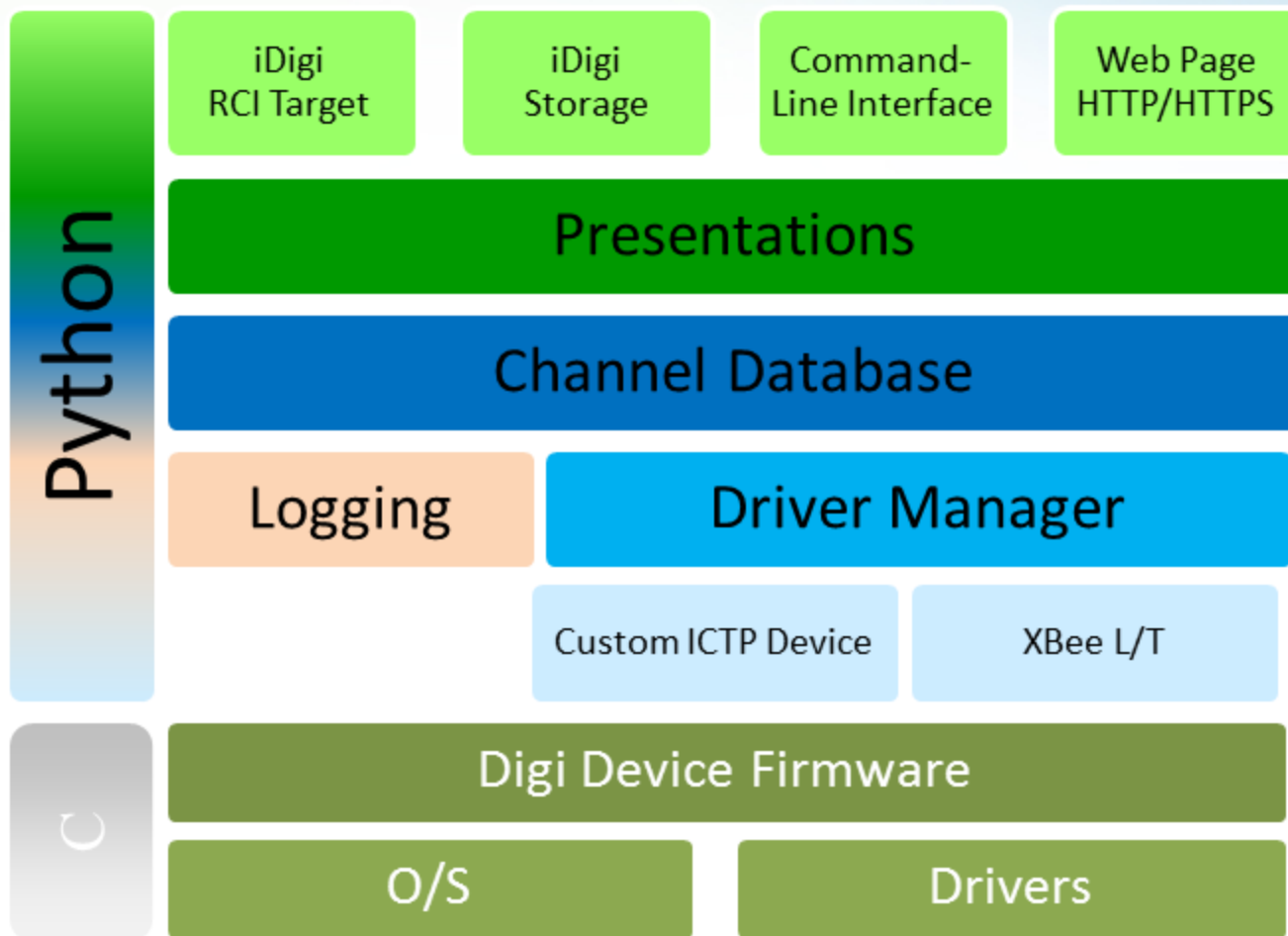


iDigi Gateway Kit Architecture





iDigi Dia Demo Architecture



Gardening, the iDigi Way



- Sick of the Minnesota winters and eager to build something, we made this:



The Crops



Irrigation Zones



Irrigation Zones



Irrigation Zones



Irrigation Control

Digi

Irrigation Control

- Pump & Filter
- XBee DIO for valve control
- Pressure Regulator
- Pressure Accumulator
- 3 Zone Controls



Irrigation Tanks



Irrigation Tanks

- Filled from well
- Gravity boosted pressure to garden
- Level measured wirelessly with XBee



Solar Power System



Solar Power

- 140 watt solar panel
- Charge controller
- 20 Ah battery
- XBee Analog I/O monitored charge state

Soil Sensoring



Soil Moisture Sensors

- Decagon devices
- Integrates 900 MHz XStream radio
- Communicates to 900 MHz XBee-PRO XSC in ConnectPort

Garden Dashboard

Digi



The Same iDigi Architecture

Digi



Garden Intelligence

- Execute monitoring tasks:
 - Check if the well is empty
 - Check if a garden zone is dry and manage the watering schedule

- If the well is empty
 - Start the pump, open the refill valve
 - Stop filling when the tanks are full

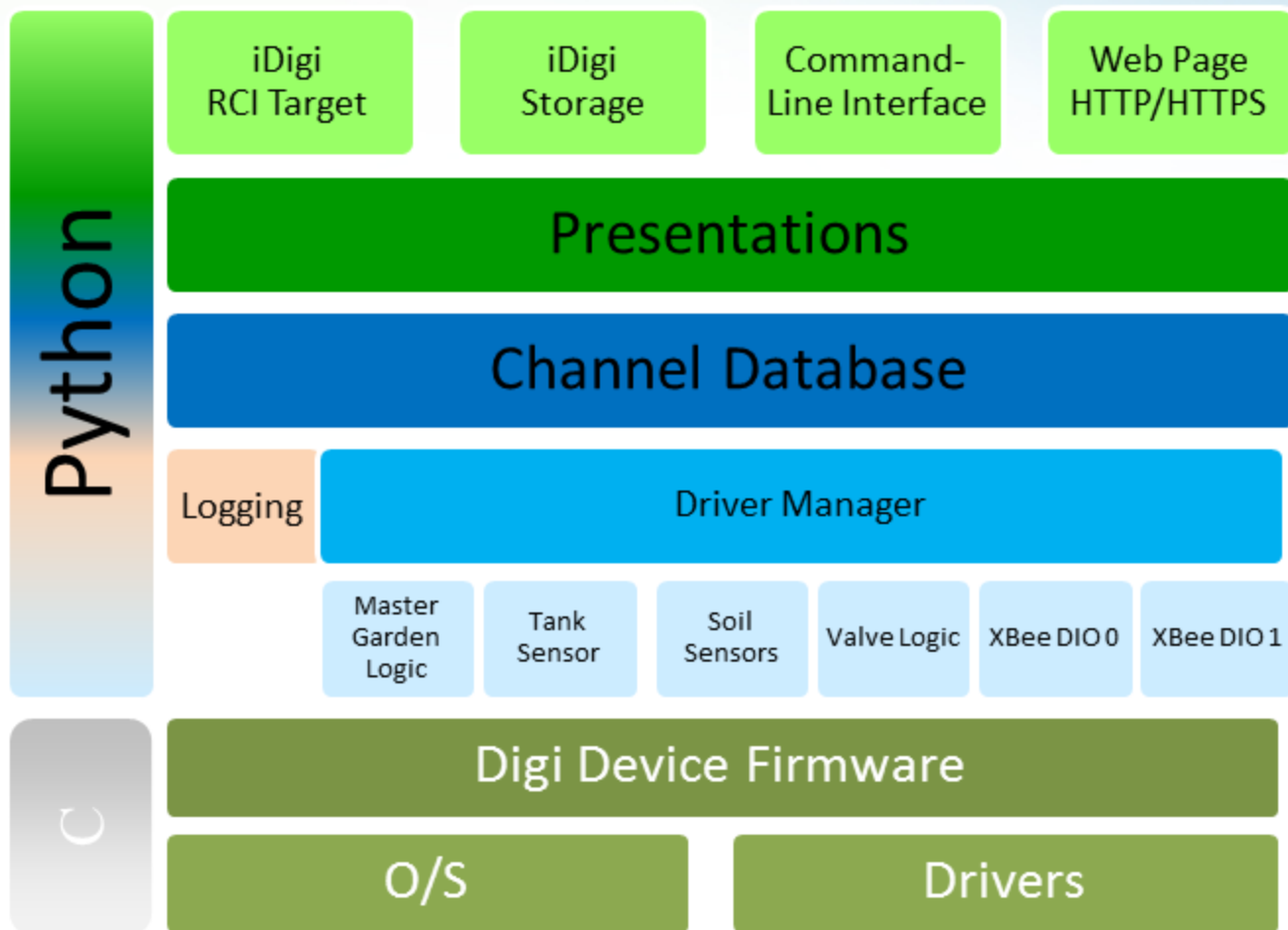
- If a zone is ready for watering
 - Start the pump and open the valve to the proper zone
 - Stop watering after a time period or when the tanks are empty

How iDigi Dia Application Was Architected



- Do as much as possible with configuration:
 - Outputs of Analog I/O devices transformed to real-world units
 - Flexible configuration syntax developed for defining garden zones
- Create custom device drivers where needed:
 - Custom valve driver which starts pump, waits, and delivers pulses to open and close valves
 - Central logic state machine called the IrrigationSystemDevice

iDigi Garden Architecture



Jordan Husney

jordan.husney@gmail.com

@jordan_husney

<http://jordan.husney.com>