iDigi™ Dia

Software for Connecting Devices in an M2M World

Device connection software makes it easy to collect and use remote data, providing a powerful environment for rapid M2M solution creation.

Overview

iDigi Device Integration Application (iDigi Dia) is software that simplifies connecting devices (sensors, PLCs, etc.) to communication gateways. iDigi Dia includes a comprehensive library of plug-ins that work out-of-the-box with common device types and can also be extended to include new devices. Its unique architecture allows the user to add most devices in under a day.

iDigi Dia is a tested architecture that provides the core functions of remote device data acquisition, control and presentation between devices and information platforms. It collects data from any device that can communicate with a Digi gateway, and is supported over any gateway physical interface. iDigi Dia presents this data to upstream applications in fully customizable formats, significantly reducing a customer's time to market.

Written in the Python® programming language for use on Digi devices, iDigi Dia may also be executed on a PC for prototyping purposes when a suitable Python interpreter is installed.

iDigi Dia is targeted for applications that need to gather samples of data from a set of devices (ZigBee® sensors, wired industrial equipment, GPS devices, etc.). It is an integral component of the iDigi platform, which customers can deploy with iDigi Dia software to build flexible, robust solutions with unprecedented speed.

iDigi Dia is fully maintained and expanded by Digi. Visit www.idigi.com to learn more.

Application Highlight

ConnectPort™ X4
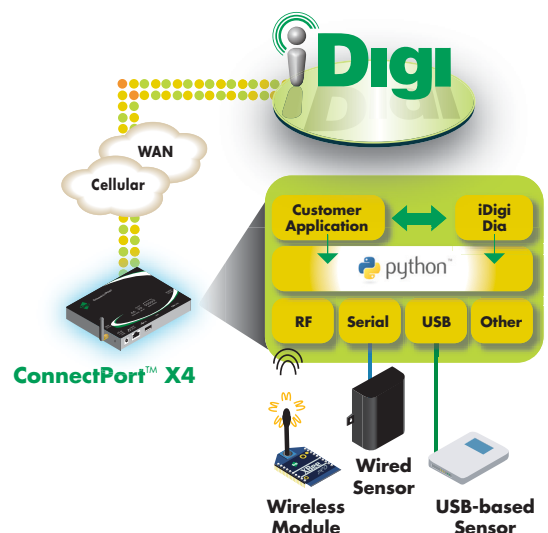
Features/Benefits

- Standardized, well-documented device connectors result in very short learning curve

- Abstraction of device connector internals to allow developers to focus on application functionality

- Scalable from one to dozens of addressable devices to suit the needs of the enterprise

- Extensible – Developers can enhance iDigi Dia with additional device types and presentation protocols

- Simplifies emerging technologies and handles complex features such as managing ZigBee sleeping nodes and sending alarms for low battery alerts

Why Use iDigi Dia?

Writing a data collection interface can be a daunting task. The interface (device driver) must be effective, efficient and resilient, while not interfering with other device and gateway functions. It needs to collect data from sensors and other devices, possibly transform that data into other labels (formats), log data, present data to the next application layer, and provide local device control. These interfaces must be created quickly; they are a critical requirement for deploying the application. iDigi Dia addresses these critical device collection requirements and works out-of-the-box. Weeks or months of interface design and testing are handled via iDigi Dia in under one day. iDigi Dia saves the M2M valuable time and expense that can be redirected to focus on their application.

How iDigi Dia Works – 3 Primary Components

1. Device Connectors: Convert real-world devices into sets of data quantities known as properties. A device connector (like a driver) instance will publish the values of these properties via a named channel known as a "channel property" in the form of a sample.

2. Device Channels: Manage the global area (namespace) of all channel names and give parties the ability to create new channels, remove existing channels, get samples, set samples, or subscribe to channel change notifications.

3. Presentations: Responsible for presenting samples to humans or machines running applications at the next highest layer. Presentations receive samples from the channel database either by polling or asking to be notified on changes using the channel publisher object's subscription interface.
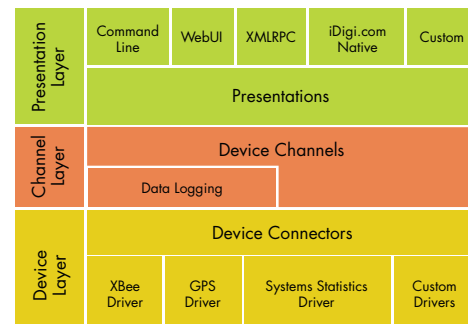
iDigi Dia Architecture

iDigi Dia runs on an intelligent communication gateway's operating system via Python. All connectors, channels and presentations are written in Python. Devices are organized by the iDigi Dia device manager. Device connectors acquire data and format the samples. Samples are injected into a set of channels, where they are logged. Presentation drivers then subscribe or poll channels to make the data available to the next highest layer application.

Extending iDigi Dia

iDigi Dia is much more than an I/O aggregator. It is an application with an extensible framework, ready for expansion. Expansion takes the form of modules, which are code libraries that extend iDigi Dia capabilities. Modules may be written to provide new or alternate methods for data collection. Modules may also be added to the presentation sub-system in order to introduce a new protocol, or to enhance an existing capability. Using Python to add and change modules means changes can be made quickly and tested in near real-time.
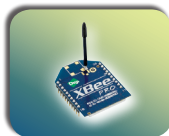
Diagram

Presentation Layer	Command Line	WebUI	XMLRPC	iDigi.com Native	Custom
	Presentations				
Channel Layer	Device Channels				
	Data Logging				
Device Layer	Device Connectors				
	XBee Driver	GPS Driver	Systems Statistics Driver		Custom Drivers

Related Products

Gateways   **Modules**   **Adapters**   **Python**   **iDigi Platform**

Get iDigi Dia Today.