

A tool to automate device programming for IoT applications

December 10 2020



Credit: Pixabay/CC0 Public Domain

The Internet of Things (IoT) has ushered in a new era, with everyday items evolving into what we now refer to as cyber-physical systems. These systems are physical mechanisms controlled or monitored by computer algorithms and deeply intertwined through the internet. Such

systems have pierced their way into industry and are being deployed and used above all to manage and control industrial processes, thus giving rise to the so-called Industry 4.0. ICREA research professor Jordi Cabot and researcher Abel Gómez, two members of the Systems, Software and Models (SOM) Research Lab at the Universitat Oberta de Catalunya (UOC) Internet Interdisciplinary Institute (IN3), in collaboration with the IKERLAN technology research center, have designed an innovative new tool for automating and streamlining the creation of systems that employ asynchronous event-driven communication, one of the most widely used computer architectures in this sector. The tool is the first to use the recently published AsyncAPI specification, which standardizes work with this type of architecture. The prototype, initially developed as part of the MegaMa@Rt2 project, is open-source and thus available for free online.

In IT infrastructures where numerous devices have to communicate with each other, such as in factories with different machinery to be monitored and controlled, the entirety of information is usually managed by a central node. In order to prevent these infrastructures from collapsing due to a faulty component, event-driven asynchronous architectures are deployed. Among the advantages of these architectures is that a breakdown in one component does not trigger a full system crash. One of the most popular paradigms is called the publish-subscribe architecture, where messages are not sent to specific receivers. According to Abel Gómez, "a channel shared by all the devices in a network is set up, and when one element, whether a computer, server or other type of device, wants to receive certain information, all it has to do is subscribe to a specific category of messages. This way, when another device is able to provide the information, it simply publishes it in the shared channel under the agreed category, and only the subscribed devices will receive the messages published on this topic."

Although these distributed architectures are particularly scalable and

flexible, they are not without problems since there is still no established programming standard like there are for website creation, for instance. Therefore, the sector still needs to agree on the message categories, as well as their internal format and structure. The researcher said: "As there is no common language and they are such distributed infrastructures, the likelihood that each element is programmed by a different person is high, meaning messages may vary between devices. As a result, if there is any divergence in the topic names or format used, the receivers will either not receive or not know how to decipher the content."

A new solution is now on the table, seeking to standardize the programming of event-driven architectures: the AsyncAPI specification. This specification allows users to define all the relevant information needed to design and run IoT devices in these environments. However, AsyncAPI is still in the early stages of development and therefore the tools that support it remain in short supply. Despite this, the researchers have developed a tool based on this new proposal that allows users to automate the creation of messages in the appropriate format, as well as the sending and receiving of these messages. Abel Gómez said: "Much of the work that goes into implementing a program for an IoT device involves creating messages in the format that subscribers to the channel expect and also "translating" messages from other devices in order to process the information. A large amount of code must therefore be programmed and, when done manually, this can be a source of errors."

Gómez said, "By adopting this new tool, we can significantly shorten the amount of [time](#) needed to develop and launch programs, which favors interoperability, improves code quality and in turn limits the number of errors in the software development life cycle."

A model for optimizing programs based on time series data

Another challenge posed by the integration of cyber-physical systems in Industry 4.0 is the need to improve computerized management of [time series data](#), such as temperature or other data collected on a continuous basis. These series of historical data are key to monitoring system runtimes, improving [industrial processes](#) and pinpointing possible sources of error after a catastrophic failure. In this area, the UOC researchers have teamed up with an Austrian research group to design a theoretical model that combines model-based engineering and time series databases to automate part of the development process.

Time series databases support the storage and analysis of massive amounts of historical data, such as the temperature reading of an industrial refrigerator at regular intervals. This information then allows different operations to be carried out, such as calculating the maximum or average temperature over a specific period of time. Above all, the project aims to integrate time series calculations such as these into a model, i.e. a representation of a certain computer system which automates the whole process and removes the need to code its functions repeatedly for different cases. Abel Gómez said, "We have come up with a model that allows us to optimize historical queries by taking advantage of time series databases. This model outlines the preferred query language and the appropriate structure of the time series databases that would support that model."

According to the researchers, this proposal is beneficial in that it would make it possible to automate the programming process and circumvent the risk of potential coding errors, since the model would specify all the information necessary for it to run properly. "This model would reduce programming time and the prevalence of errors. By generating the code automatically from the [model](#), you don't have a programmer doing it by hand who can make mistakes," Gómez said.

More information: Alexandra Mazak et al. Temporal Models on Time

Series Databases., *The Journal of Object Technology* (2020). [DOI: 10.5381/jot.2020.19.3.a14](https://doi.org/10.5381/jot.2020.19.3.a14)

github.com/SOM-Research/asyncapi-toolkit

Provided by Universitat Oberta de Catalunya

Citation: A tool to automate device programming for IoT applications (2020, December 10) retrieved 25 April 2024 from

<https://techxplore.com/news/2020-12-tool-automate-device-iot-applications.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.