# Actuation Conflict Management Enabler for DevOps in IoT


**Thibaut Gonnin**
**Frank Dechavanne**
**Gérald Rocher**
**Stéphane Lavirotte**
**Jean-Yves Tigli**
University Côte d'Azur - CNRS
Sophia Antipolis, France
firstname.lastname@univ-cotedazur.fr

**Laurent Capocchi**
**Jean-François Santucci**
Université de Corse - CNRS
Corte, France
firstname.lastname@univ-corse.fr


## ABSTRACT


DevOps methodology is well known in the field of classical software development to improve the software quality and the frequency of the deliveries. However, DevOps has not yet been popularized in the field of IoT. The reasons of that are new challenges that require new enablers in the DevOps tools ecosystem. The European ENACT project supported under the H2020 programme aims to respond to these challenges by providing new enablers to apply DevOps methodology for IoT application[2]. We present, here, one of these enablers called Action Conflict Management Enabler (ACM enabler) which aims at anticipating and resolving at Devs Time all conflicts[1] and dysfunctions that might result from mismanagement of the IoT system actuators.


## CCS CONCEPTS


• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.



*IoT 2020, October 6-9, Malmö, Sweden*

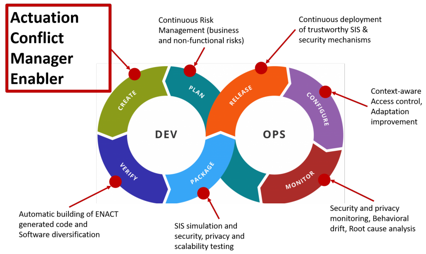


**Figure 1: Actuation Conflict Manager Enabler in the ENACT DevOps for IoT cycle**

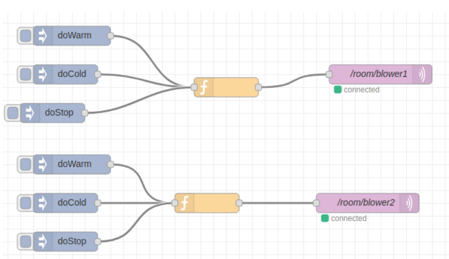


**Figure 2: Simple Node red implementation for the two fan coils in Kubik**

## KEYWORDS

Internet of things; DevOps; Actuation Conflict.

**ACM Reference Format:**
Thibaut Gonnin, Frank Dechavanne, Gérald Rocher, Stéphane Lavirotte, Jean-Yves Tigli, Laurent Capocchi, and Jean-François Santucci. 2020. Actuation Conflict Management Enabler for DevOps in IoT . In *Proceedings of ACM Woodstock conference (IoT 2020).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.475/123_4

## INTRODUCTION TO THE DEVOPS FOR IOT CHALLENGE

DevOps is a well-known methodology of best practices in the software industry.[4] DevOps seeks to improve collaboration between operators and developers by the promotion of automation and monitoring of all stages of software creation. If DevOps methodology and tools are common good practices in classical software development, this not yet the case for IoT. In particular, the DevOps methodology applies in homogeneous systems where the virtualization of the software infrastructure is facilitating the continuous integration of software applications on targets. However, IoT infrastructures are mainly heterogeneous due to embedded devices that are highly variable in software and hardware. Moreover, the possibilities for such devices to interact through complex actions to the physical environment defined by its own dynamic and behavioural processes can create complex situation to monitor[3]. The DevOps methodology applied to IoT applications thus faces new challenges that require new tools. The European ENACT project supported under the H2020 programme aims to respond to these challenges by providing new enablers (Figure 1) to apply DevOps methodology for IoT application[2]. We present, here, one of these enablers called Action Conflict Management Enabler (ACM enabler) which aims at anticipating and resolving at Devs Time all conflicts[1] and dysfunctions that might result from mismanagement of the IoT system actuators.

## CONFLICTS PATTERN AND THEIR IDENTIFICATION

We define a conflict as the access to a shared resource due to several non-communicating software application resulting in ambiguous or unproductive behaviour. It is not only a synchronization problem but an issue with the semantic resulting from two simultaneous actions. When direct conflict occurs, simultaneous antagonist try to access to a shared software resource (mostly actuators) leading to an ambiguity (should the lamp be "on" or "off"?). When an indirect conflict appears, two applications act on separate actuation whose effects interfere with each other through the physical environment. Effects resulting from these actuators may become non-predictable.

For instance, in the Tecnalia KUBIK smart building, fan coils manage heating, ventilation and cooling. If one considers two independent fan coils acting in the same room, it is possible that one fan coil is heating the room while the other one is cooling it. The resulting behaviour is obviously counterproductive and needs to be avoided.

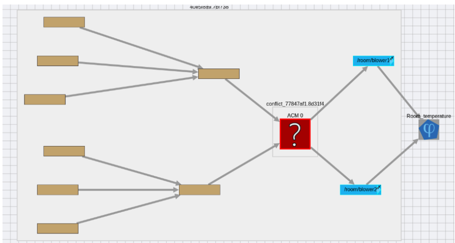

**Figure 3: WIMAC vizualisaton of the fan coil example after the conflict detection process.**

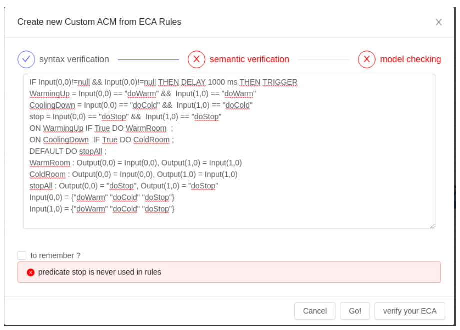


**Figure 4: ECA+ Customization interface with logical and temporal validation feedback**

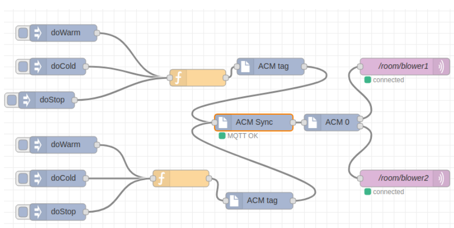


**Figure 5: New conflictless Node red flow with the custom ACM**

## MAIN FEATURES OF THE ACTUATION CONFLICT MANAGEMENT ENABLER

The first released of actuation conflict management enabler follows several innovative principles associated to the tools:

- A new language to describe a workflow and Interaction Model for Actuation Conflict Management (WIMAC). ACM Enabler requires a model used to detect and manage actuation conflicts. This model is generated automatically from the set of components models (including but not limited to Node-Red and ThingML) and a model of the impacts of the actuators on the physical environment.
- A generic tool for Actuation Conflict Detection and Solving. It consists in providing a tool for deploying ACM solution in a systematic manner by finding patterns in the WIMAC representation.
- A complete tool to design a custom and reliable custom Actuation Conflict Manager (ACM). In some particular cases, there is a particular necessity in creating your own ACM to solve your conflict. This tool helps in both creation and validation of custom ACMs.

## ACM ENABLERS ON REPRESENTATIVE EXAMPLE

The Tecnalia fan coils example will be used to show how the ACM enabler is working (Figure 2). This example is voluntary simple to make the core explanation clearer. The following steps msut be pursued create a conflictless IoT application.

- To detect conflicts in an IoT software, it is mandatory to import it into a WIMAC model equivalent. A transformation from Node-red to WIMAC and from ThingML to WIMAC is already implemented in ACM enabler. The remaining job for the designer is to define what impact each actuators have on physical properties of its environment. In the fan coil example, the two actuators access to the same physical property : the room's temperature.
- The detection phase allows to find conflict localization based on graph transformation algorithms. Detection and transformation are defined from a set of rules corresponding to some actuation conflict patterns. They are used to insert off-the-shelf ACM in the WIMAC model. In the fan coil example, the indirect conflict has correctly been found between the two actuators (Figure 3).
- Final step consists in choosing the correct behaviour to manage a conflict. Off-the-shelf ACM are available through a database and might provide a relevant ACM to most cases. If no suitable off-the-shelf ACM were too be found, ACM enabler provides a format close to Event Condition Action rules that has been especially adapted for ACM creation: ECA+ (Figure 4).
  Once the ACM has been created, it needs to be verified in order to guarantee the safety and quality of the IoT application. The end user just needs to express safety properties which refer to

**Figure 6: KUBIK Smart Building**

a state related to inputs and outputs that should never be accessible. Both logical and temporal verification are processed through model checking and simulation to achieve this goal. With the fan coil use case, the main property to validate is "whenever there is the command ***doCold*** for one fan coil, there shouldn't have ***doWarm*** for the other one".

## USE CASE IMPLEMENTATION

The ACM enabler as been used through many different use cases coming from various fields. From logical smart building (Figure 6) optimization to critical train sound management, ACM enabler has proven its efficiency by resolving several use cases. You can find most of the use cases video on the enact project website (https://www.enact-project.eu/videos.php)

## CONCLUSION

Included in a line of several DevOps tools for IoT, ACM allows you to manage unwanted and unpredictable behavior. Just like for classical DevOps methods, it is necessary to set up the architecture needed to implement the tools. This cost is greatly enhanced by the constant software quality provided by such tools. In the end, DevOps methodology can be adapted to IoT and have potential to help developers to provide quality IoT Application.

## ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Commission's H2020 Program under grant agreement numbers 780351 (ENACT).

## REFERENCES

[1] A. Al Farooq, E. Al-Shaer, T. Moyer, and K. Kant. 2019. IoTC2: A Formal Method Approach for Detecting Conflicts in Large Scale IoT Systems. In *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. 442–447.

[2] Nicolas Ferry, Arnor Solberg, Hui Song, Stéphane Lavirotte, Jean-Yves Tigli, Thierry Winter, Victor Muntés-Mulero, Andreas Metzger, Erkuden Rios Velasco, and Amaia Castelruiz Aguirre. 2019. ENACT: Development, Operation, and Quality Assurance of Trustworthy Smart IoT Systems. In *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment*, Jean-Michel Bruel, Manuel Mazzara, and Bertrand Meyer (Eds.). Springer International Publishing, Cham, 112–127.

[3] Stéphane Lavirotte, Gérald Rocher, Jean-Yves Tigli, and Thibaut Gonnin. 2020. IoT-based Systems Actuation Conflicts Management Towards DevOps: A Systematic Mapping Study. In *Proceedings of the 5th International Conference on Internet of Things, Big Data and Security, IoTBDS 2020, Prague, Czech Republic, May 7-9, 2020*, Gary Wills, Péter Kacsuk, and Victor Chang (Eds.). SCITEPRESS, 227–234. https://doi.org/10.5220/0009355102270234

[4] Leonardo Leite, Carla Rocha, Fabio Kon, Dejan Milojicic, and Paulo Meirelles. 2019. A Survey of DevOps Concepts and Challenges. *ACM Comput. Surv.* 52, 6, Article 127 (Nov. 2019), 35 pages. https://doi.org/10.1145/3359981