

# La méthodologie ACIMOV pour l'intégration agile et continue des modules ontologiques

F.-Z. Hannou<sup>1</sup>, V. Charpenay<sup>2</sup>, M. Lefrançois<sup>2</sup>, C. Roussey<sup>3</sup>, A. Zimmermann<sup>2</sup>, F. Gandon<sup>4</sup>

<sup>1</sup> EDF R&D, Palaiseau, France

<sup>2</sup> Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, Saint-Étienne, France

<sup>3</sup> MISTEA, INRAE Centre Occitanie, Montpellier, France

<sup>4</sup> Inria, Univ. Cote d'Azur, I3S, CNRS, France

fatma-zohra.hannou@edf.fr, victor.charpenay@emse.fr, maxime.lefrancois@emse.fr, catherine.roussey@inrae.fr, antoine.zimmermann@emse.fr, fabien.gandon@inria.fr

## Résumé

*Ce travail décrit la méthodologie d'ingénierie d'ontologie Agile and Continuous Integration for Modular Ontologies and Vocabularies (ACIMOV) pour développer des ontologies et des vocabulaires. ACIMOV étend la méthodologie agile SAMOD pour mieux prendre en compte l'aspect modulaire des ontologies actuelles et le développement collaboratif entre différents types d'experts. ACIMOV adopte les principes de développement agile et DevOps. ACIMOV a été conçue pour être opérationnalisée à l'aide de plateformes de développement logiciel collaboratif Git et dotées de workflows d'intégration et de déploiement continus. Ces travaux ont été publiés dans un atelier international [4].*

## Mots-clés

*Méthodologie d'ingénierie des ontologies, ontologie modulaire, Agile, Git, intégration et déploiement continu*

## Abstract

*This work describes the Agile and Continuous Integration for Modular Ontologies and Vocabularies (ACIMOV) ontology engineering methodology for developing ontologies and vocabularies. ACIMOV extends the agile SAMOD methodology to better take into account the modular aspect of current ontologies and collaborative development between different types of experts. ACIMOV adopts the standard git-based approach for code version management, leveraging Agile and DevOps principles. It has been designed to be operationalized using collaborative software development platforms and tooling with continuous integration and continuous deployment workflows. This work has already been published in an international workshop [4].*

## Keywords

*Ontology engineering methodology, modular ontology, Agile, Git, Continuous Integration and Continuous Deployment*

## 1 Introduction

Ce travail décrit la méthodologie d'ingénierie d'ontologie Agile and Continuous Integration for Modular Ontologies and Vocabularies (ACIMOV) pour développer des ontologies et des vocabulaires, qui a été publiée à l'international [4]. ACIMOV étend la méthodologie agile SAMOD [5] pour (1) assurer l'alignement avec des ontologies de référence sélectionnées; (2) planifier des développements modulaires intégrant la gestion des dépendances; (3) définir les modules de l'ontologie qui peuvent être spécialisés pour des domaines spécifiques; (4) permettre une collaboration active entre les ontologues et les experts du domaine; (5) permettre aux développeurs d'applications de sélectionner des vues de l'ontologie pour leur domaine et des cas d'usages spécifiques. ACIMOV adopte l'approche standard de gestion des versions de code Git, et tire parti des principes Agiles et DevOps. Elle a été conçue pour être opérationnalisée à l'aide de plateformes de développement logiciel collaboratif telles que Github ou Gitlab, et est dotée de workflows d'intégration et de déploiement continus (workflows CI/CD) qui exécutent des contrôles syntaxiques et sémantiques sur le référentiel, valident les questions de compétence, spécialisent les modules, génèrent et publient les documentations de l'ontologie.

## 2 La méthodologie ACIMOV

Dans le contexte de nos travaux sur le déploiement du Web sémantique des objets (projet CoSWot<sup>1</sup>) et des agents (projet HyperAgents<sup>2</sup>), nous avons eu besoin de développer une ontologie pour standardiser l'échange de données. Nous souhaitons réutiliser plusieurs ontologies de références du domaine : W3C Thing Description (TD) [1], OGC&W3C Semantic Sensor Network (SSN) [3], ETSI Smart Applications Reference ontology (SAREF) [2]. Ces trois ontologies adoptent une conception modulaire. Nous identi-

1. <https://coswot.gitlab.io/>

2. <https://anr.fr/Projet-ANR-19-CE23-0030>

fions les besoins suivants pour notre ontologie : (O1) L'ontologie doit s'aligner sur les ontologies de référence de l'IoT; (O2) L'ontologie doit être modulaire et comprendre des modules qui couvrent les connaissances communes à tous les composants de la plate-forme IoT; (O3) L'ontologie doit réutiliser certaines ontologies identifiées pour les domaines d'application concernés; (O4) L'ontologie doit avoir une structure homogène et prévisible, de sorte que des concepts similaires pour différents domaines soient décrits de la même manière; (O5) Différentes représentations alternatives doivent être possibles pour tenir compte de la nécessité de manipuler des graphes de connaissances de petite taille dans des environnements contraints. (O6) On doit pouvoir sélectionner un sous-ensemble de l'ontologie (une vue) qui couvre les besoins d'une application spécifique.

Nous avons choisi de prendre comme point de départ la méthode SAMOD [5] et de l'adapter pour couvrir les besoins suivants : (M1) Les principes agiles doivent être adoptés pour améliorer la collaboration entre les ontologues, les experts du domaine et les *product owners*, avec des cycles courts et des incréments de travail; (M2) Des réunions régulières avec toutes les parties doivent être organisées pour aider à prioriser les exigences découlant des cas d'utilisation et à choisir la cible de la prochaine itération; (M3) Des réunions régulières entre les ontologues doivent être organisées pour aider à prioriser les modules sur lesquels travailler et pour s'assurer que le travail sur différents modules peut être mené en parallèle; (M4) Des plateformes de développement logiciel collaboratif avec versionnement du code et suivi des problèmes doivent être adoptées; (M5) Les principes DevOps doivent être adoptés pour permettre l'intégration et le déploiement continus des artefacts de l'ontologie (par exemple, les modules de l'ontologie, la documentation, les exemples).

ACIMOV cible les ontologues sensibilisés aux concepts et outils récents pour la gestion des version de code. La figure 1 donne un aperçu des sept étapes de la méthodologie ACIMOV. Elle comporte un cycle long de développement Agile impliquant les ontologues, les experts de domaine et les *product owners*, et deux cycles internes plus courts dédiés aux activités de développement menées par les ontologues. Les étapes sont les suivantes : (1) Collecter les exigences et identifier les ontologies de référence; (2) Organiser une réunion de bilan (un événement); (3) Sélectionner les modules pertinents dans les ontologies de référence; (4) Gérer le planning de développement des modules; (5) Organiser une réunion de développement des modules (un événement); (6) Développer et tester des modules; (7) Intégrer les modules et publier l'ontologie. ACIMOV est documentée en ligne <sup>3</sup> et des projets témoins facilitent son adoption et son outillage <sup>4</sup>. Différentes pistes de développement sont encore à l'étude, dont l'intégration dans des actions GitHub, ou la génération de représentations alternatives ou de vues de l'ontologie. Nous comptons également évaluer la mise en oeuvre d'ACIMOV sur différents projets.

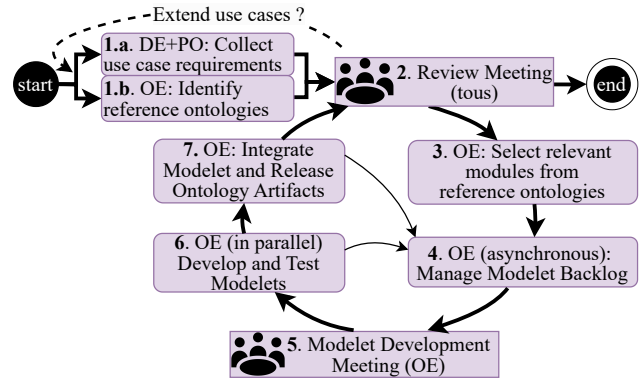


FIGURE 1 – Aperçu de la méthodologie ACIMOV. OE : Ontologue; DE : Expert de domaine; PO : *Product Owner*

## Références

- [1] V. Charpenay, M. Lefrançois, M. Poveda Villalón, and S. Käbisch. Thing Description (TD) Ontology, Editor draft, 10 May 2023. W3c working group draft, W3C, May 2023.
- [2] Raúl García-Castro, Maxime Lefrançois, María Poveda-Villalón, and Laura Daniele. The ETSI SAREF ontology for smart applications : a long path of development and evolution. In *Energy Smart Appliances : Applications, Methodologies, and Challenges*. Wiley, 2023.
- [3] Armin Haller, Krzysztof Janowicz, Simon JD Cox, Maxime Lefrançois, Kerry Taylor, Danh Le Phuoc, Joshua Lieberman, Raúl García-Castro, Rob Atkinson, and Claus Stadler. The SOSA/SSN ontology : a joint W3C and OGC standard specifying the semantics of sensors, observations, actuation, and sampling. *Semantic Web-Interoperability, Usability, Applicability an IOS Press Journal*, 56, 2019.
- [4] Fatma-Zohra Hannou, Victor Charpenay, Maxime Lefrançois, Catherine Roussey, Antoine Zimmermann, and Fabien Gandon. The ACIMOV Methodology : Agile and Continuous Integration for Modular Ontologies and Vocabularies. In *2nd Workshop on Modular Knowledge associated with FOIS 2023*, 2023.
- [5] Silvio Peroni. Samod : an agile methodology for the development of ontologies. In *Proceedings of the 13th OWL : Experiences and Directions Workshop and 5th OWL reasoner evaluation workshop (OWLED-ORE 2016)*, pages 1–14, 2016.

## Remerciements

Ce travail a été en partie soutenu par des subventions de l'Agence française de la recherche (ANR) sur les projets CoSWot (ANR-19-CE23-0012) et HyperAgents (ANR-19-CE23-0030), et le projet européen ACCORD (Horizon Europe R&I convention de subvention du programme n° 101056973).

3. <https://acimov.gitlab.io/>

4. <https://gitlab.com/acimov/>