MoCoP: Towards a Model Clone Portal

Önder Babur
Dept. of Mathematics & Computer Science
Eindhoven University of Technology
Eindhoven, The Netherlands
o.babur@tue.nl

Matthew Stephan
Dept. of Computer Science & Software Engineering
Miami University
Oxford, Ohio, USA
stephamd@miamioh.edu

Abstract—Widespread and mature practice of model-driven engineering is leading to a growing number of modeling artifacts and challenges in their management. Model clone detection (MCD) is an important approach for managing and maintaining modeling artifacts. While its counterpart in traditional source code development, code clone detection, is enjoying popularity and more than two decades of development, MCD is still in its infancy in terms of research and tooling. We aim to develop a portal for model clone detection, MoCoP, as a central hub to mitigate adoption barriers and foster MCD research. In this short paper, we present our vision for MoCoP and its features and goals. We discuss MoCoP’s key components that we plan on realizing in the short term including public tooling, curated data sets, and a body of MCD knowledge. Our longer term goals include a dedicated service-oriented infrastructure, contests, and forums. We believe MoCoP will strengthen MCD and more. This necessitates scalable techniques for model management; in particular, organizing, searching, reusing, maintaining, analyzing, synthesizing, and visualizing these large and complex set of artifacts and ecosystems.

I. INTRODUCTION

The increasing volume and complexity of software systems has led academia and industry to explore methodologies to better facilitate the engineering and maintenance of those systems. Model-driven engineering (MDE) is one such methodology. However, as MDE practices mature, the volume and complexity of its corresponding artifacts, notably models, transformations, and modeling languages is increasing in kind. This manifests itself in industrial practice in individual companies [1], [2], as well as in the open source domain, and academic repositories [1], [3], [4]. Taking also model evolution into account, large-scale MDE practitioners have to deal increasingly with legacy models and ecosystems [5]. This necessitates scalable techniques for model management; in particular, organizing, searching, reusing, maintaining, analyzing, synthesizing, and visualizing these large and complex set of artifacts and ecosystems.

Management of MDE artifacts has manifested itself as a crucial focus of both emerging model repositories and model management research. The latter involves a wide range of techniques. Model management is a high-level concept in which entire models and their relationships can be manipulated using operators to achieve useful outcomes [6], [7] or made consistent with respect to other models [8]. Issues within model management are addressed by researchers through increasingly popular model portals/repositories, which are an important step and challenge in furthering model-driven solutions [9]. Model repositories allow convenient web-based interfaces to store, version control, retrieve, and analyze MDE artifacts.

Model clone detection (MCD) is one form of model management. Models, or their fragments, which are similar according to some similarity measure, are identified by model clone detectors to infer valuable insights into understanding underlying ecosystems, their quality, maintainability, and more. The analogous source code clone detection (CCD) has been experiencing popularity for over two decades [10]. CCD, an established form of source code analysis and manipulation, is very important in software development. In contrast to CCD, there has been only limited volume of research and a lack of much available, usable, and mature tooling for MCD. There has even been a drop in MCD research in recent years despite the benefits it has provided to industry [11].

There are several challenges hampering the advancement and adoption of MCD techniques. These challenges include,

- A lack of available, usable, mature, open source tooling for MCD
- Approaches/tools that are specific to certain modeling languages, and difficult to apply to different types of models
- Having no curated data sets, standard corpora or benchmarks, and only a limited set of comparative studies
- There being no repository or portal allowing others to experiment/use/evaluate model clone detectors, nor any body of knowledge for MCD research
- End user usability and researcher reproducibility

In this paper, we propose tackling these challenges and promoting MCD research and practice through creation of a model clone portal (MoCoP) for disseminating MCD tools, data sets, knowledge, and more. We begin the paper with Section II by providing background on MCD concepts, literature, and tools. Section III presents related model repositories and initiatives. We describe the important features of MoCoP and provide illustrations of it in Section IV. We conclude the paper and discuss next steps in Section V.

II. BACKGROUND - MODEL CLONE DETECTION

MCD is a form of model comparison that involves identifying sets of similar model elements [12]. Model clone detectors measure similarity in a variety of ways including graphical
of the results involved code clone detection, and therefore were excluded by us in our survey, we suspect code clone detection literature is increasing in volume. MCD literature emerges in 2006, has relatively small volume, and experiences decline after its peak in 2014. We also marked the introduction of prominent tools along the years. We identify some adoption obstacles with the tools: CloneDetective/ConQAT (2008) is discontinued and replaced with a commercial tool for non-academic use; ModelCD (2009) is not available at all; Simone (2012) is available upon request and through a FreeBSD license; MQlone (2013) is available closed-source and with very limited functionality in the MACH toolset; and SAMOS (2018) is not published nor mature yet.

III. RELATED WORK - SOFTWARE MODEL PORTALS AND OTHER INITIATIVES

In this section we describe other portals and related initiatives from which we draw inspiration.

A. Software Model Portals

- The open models initiative aims to provide a platform to develop conceptual and other types of models collaboratively for open public development, use, and distribution [26]. It acts as a central hub for the modeling community with an infrastructure supporting model development, teaching, and research. The lack of open source software development practices in modeling is observed by the authors. They plan to mitigate “reinventing the wheel” by providing better tried and tested models, domain expert knowledge, easy integration and reuse. The open model approach also serves as a test-bed for investigating the effects of conceptual modeling and open models on open source software development.
• MDEForge is an extensible modeling framework to support the discovery and reuse of existing modeling artifacts such as models, transformations, and domain specific languages [4]. The lack of these in current MDE practice has been identified by the authors as an issue hampering wide adoption of MDE, leading to the development of these artifacts from scratch. They present an extensible service-based repository for storing, querying, managing and reusing modeling artifacts in a convenient manner via a REST API. It supports the modeling community with the modular and collaborative nature of the repository, where new artifacts and tools, in the form of services, can be integrated easily into the repository.

• Repository for Model-Driven Development (ReMoDD) is a repository for improving MDE research, productivity, and learning [27]. It incorporates MDE case studies using graphical and specification models. It includes other related artifacts such as reusable transformations, source code of models’ implementations, example models reflecting good and bad practices, exercises and other pedagogical materials for teaching modeling, and benchmark models for the testing and evaluation of MDE techniques. ReMoDD further provides a web-based interface for searching and browsing the repository, as well as a forum for the community interaction.

• models-db provides a UML repository for UML models, images, and design metrics with search functionalities via the web interface. It is intended to aid researchers in performing empirical experiments on the models. It currently hosts the Lindholmen dataset [3].

B. Other Initiatives

• Apromore process analytics portal: Apromore addresses the increasing number of process models being created by organizations, and the challenges in dealing with consulting, updating, and reusing these collections of models over long periods of time by various stakeholders [28]. It provides facilities to analyze, maintain, and exploit process models using advanced techniques such as MCD. It has an open source service-oriented architecture for usability and extendability by the modeling community.

• CCD benchmarks: Benchmarks are considered as means to advance the state of the art in software engineering [29]. The idea of using curated data sets and benchmarks in CCD research for comparing and evaluating the tools has been explored to some extent by various researchers as well [30]. Several benchmarks exist in the literature, ranging from one by Bellon et al. [31] to BigCloneBench [32] from Roy et al. The latter contains more than eight million manually validated clones within over 25,000 open-source Java systems. The former triggered various empirical studies, such as the one by Charpentier et al. on clone classification and manual labelling [33].

• Data Science and Machine Learning portals: Another related platform is kaggle\(^1\), a very popular crowd-sourced platform to attract, train, and challenge data scientists. It provides data sets, contests, and an overall medium for researchers and practitioners to interact, cooperate, and compete in solving problems in data science, machine learning, and predictive analytics. It has tremendous success in the data science community, including industry, with over one million members, numerous challenges and solutions, and high community interaction.

IV. MoCoP: The Model Clone Portal

We propose the MoCoP to address the issues we identified in MCD and to foster MCD research and practice. We intend for it to act as a central hub for MCD research with practical information, pointers to literature, tooling, and a platform for community interaction. Although inspired by existing model repositories and initiatives, our vision is very much focused on MCD, yet going beyond the scope of individual repositories. Hence we advocate a separate portal. We plan to have the following features in the short and medium term,

1) Different tools along with configurations and documentation, presented in a convenient way. For example, dockers for easy deployment and use.

2) Facilitation of continuous growth of knowledge and tooling through implementation and revival of relevant techniques, ranging from graph isomorphism approaches and other unavailable tools from the literature.

3) Exploration of how and to what extent the state-of-the-art code clone detection tools can be adapted and enhanced for operating on models.

4) An investigation feature for existing tools for different modeling languages, concrete syntaxes and file formats, e.g. via model transformations, conversions, or bridges.

5) A repository of curated datasets, standard corpora, benchmarks for different types of models and tools.

We have made some preliminary progress towards MoCoP. Based on these features, we provide our sketch of the envisioned portal in Figure 2. We are constructing an initial body of knowledge and temporarily publishing it through our model management and analytics website\(^2\).

Longer term features include challenges, along the lines of Kaggle competitions, transformation tool contests, and model matching challenges. Another feature is a forum for discussing various MCD topics, sharing publications, models and tools, which we believe will be very beneficial. While MoCoP is more than just a model repository, we acknowledge the specific challenges for model repositories in general as identified by Basciani et al. [34]. Thus we also wish to tackle some of them in the long run including a service-based infrastructure, in particular model clone detection as a service: advanced querying and searching for the artifacts; supporting heterogeneity and scalability; and licensing and artifact management supporting open science and reproducibility. We are in contact with the Data Science Center at Eindhoven University of Technology, and will continue to investigate infrastructure support and inter-disciplinary collaboration opportunities by the center.

\(^1\)https://www.kaggle.com/

\(^2\)https://modelanalytics.wordpress.com/mocop/
In this paper, we address challenges in MCD and propose a portal to help tackle them, foster MCD research and practice, and support the MCD community. The immediate next step is to realize the portal with the basic components, that is, the data, tooling, documentation and body of knowledge. As a central hub with additional input from other MCD researchers, we wish to advance the state-of-the-art in MCD through MoCoP. We are in the process of forming a body of knowledge for MCD, and will enhance it with cross-domain insights, for instance, code clone detection, approaches for other modeling languages, and technical spaces such as systems and knowledge engineering, data mining, and machine learning domains. Researchers will also be able to benefit from MoCoP in advanced studies into other important aspects of MCD, such as annotation, validation, ranking and actionability, usability, and applicability of existing techniques in industrial contexts. We are optimistic about the opportunities such a portal will afford for advancing MCD research, and look forward to receiving and incorporating feedback from the workshop and the modeling community in general.

REFERENCES