# What?

## What is it all about?

* The artifact shall have name for reference
* The artifact shall indicate its context of development (e.g., domain, problem, project)
* The artifact shall indicate the main functionalities supported (e.g., language support, code generation, model analysis)
* The artifact shall indicate its relation with its respective paper(s)

## What does it have?

* The artifact shall provide an inventory of files and folders
* The artifact shall include everything required for a replication (i.e., complete)
* The artifact shall include no more assets than necessary for replications (i.e., concise)
* The artifact shall include a preprint of its associated article
* What concepts and technologies underpin the artifact?
* The artifact shall indicate the theory that underpins the MDE tool (e.g., formalisms, semantics)
* The artifact shall indicate the modeling languages used by the MDE tool (e.g., UML, xtUML, SysML, BPMN)
* The artifact shall indicate the meta-modeling languages used by the MDE tool (e.g., EMOF, CMOF, ECore/EMF, KM3)
* The artifact shall indicate the standards and/or specifications used by the MDE tool (e.g., CMI, XMI, CWM, HUTN, JMI, DD, OCL)
* The artifact shall indicate the programming language used to develop the MDE tool
* The artifact shall indicate the libraries, dependencies and frameworks used to develop the MDE tool
* The artifact shall rely on open, well-maintained, and documented libraries or dependencies

# Why?

## Why it was created?

* The artifact shall report the motivation for its development
* The artifact shall report its objective/goal (e.g., verify claims, replicability, reusability, a whole new library/framework)
* The artifact shall indicate its advantages and/or novelty (e.g., algorithm, language, method)

# Where?

## Where is it hosted?

* The artifact shall be hosted in an open, public repository (e.g., GitHub, BitBucket)
* The artifact shall be hosted in an indexed repository (i.e., findable by search engines)
* The artifact shall be archived for long-term, permanent access (e.g., Zenodo, Figshare)

## Where should I cite?

* The artifact shall provide an explicit format for citation (e.g., in a CITATION or README file)
* The artifact shall provide a bibtex file for citation
* The artifact shall provide an URL for citation
* The artifact shall provide a DOI for citation
* Where to find related work?
* Authors shall give credit to data obtained from other sources (e.g., authors, paper , repository, benchmark)
* It shall provide references about key concepts (e.g., papers, surveys, wiki, reports)
* It shall provide references to studies using the artifact (e.g., known uses, integrated with)
* It shall provide references to related projects or artifacts
* It shall provide a bibtex file with the references cited in the paper
* It shall provide references using in-code citation (e.g., code header)

# Who?

## Who could use it?

* The artifact shall be deposited under an explicit open license (e.g., reported in a LICENSE file)
* Files shall be made available using open/non-proprietary format (e.g., JSON, XML, ODT)
* The artifact shall indicate potential users of the artifact (e.g., professionals, researchers, industry sectors)
* The artifact shall indicate the target audience (i.e., academia, industry, or both)

## Who are the authors?

* The artifact shall provide a communication channel for interacting with tool creators and community (e.g., forum, mailing list, issue tracker, IRC, Slack, Discord)
* The artifact shall indicate the names of its authors/collaborators
* The artifact shall indicate the institution of its authors/collaborators
* The artifact shall indicate the contact details of its authors/collaborators (e.g., email address, ResearchGate, Linkedin, website)
* The artifact shall indicate the ORCID of its authors/collaborators
* The artifact shall indicate the level of experience of its authors/collaborators (e.g., bio, degree, position)

## Who funded this project?

* The artifact shall indicate the funding agencies that supported the project (e.g., ERC, NWO, CNPq, DFG, EPSRC, NSF)
* The artifact shall indicate the universities that mostly contributed with the project

# When?

## When changes are implemented?

* The artifact shall indicate the version of the platform used to develop the MDE tool (e.g., Eclipse/Netbeans/JetBrains releases)
* The artifact shall indicate the version of every library or dependency used by the MDE tool
* The artifact shall indicate the version of standards or specifications used by the MDE tool
* The artifact changes shall be tracked using version control (e.g., GitHub, GitLab, BitBucket)
* The artifact shall allow to reference/retrieved specific versions using tag and/or release identifiers
* Every change shall be small (e.g., conciseness, cohesion, clear edit)
* Every change shall be coherent (i.e., non-breaking)
* Every change shall be explained and documented (e.g., in CHANGELOG.txt, via commit messages)

## When changes shall happen?

* The artifact shall provide a timeline for future goals and planned updates (e.g., frequency, next steps, future work plans)
* The artifact shall be open for change requests and receiving feedback from users (e.g., bug fixes, pull requests, collaboration)

# How?

## How is it organized?

* Experiment design shall be reported following a standard format (e.g., taxonomy, feature-based classification)
* File formats shall be explicitly documented (e.g., file type/extension)
* Tabular data files shall follow analysis-friendly formats (e.g., column is variable, row is observation, data dictionary, meaning of column/row headers)
* Files and folders shall have self-explaining names matching content, mearning and human abstractions (e.g., doc/, src/, results/, src/, bin/)
* The artifact shall report followed norms and standards (e.g., naming or code conventions, guidelines/checklists)
* Useful metadata shall be used as part of filenames for pattern matching (e.g., yyyymmdd)
* The artifact shall be compliant with ICT accessibility standards (e.g., Section 508, WAI)
* The experiment workflow shall be broken-down into small and simple procedures to facilitate reuse (e.g., scripts, functions)
* The source code shall have minimum amount of duplicated code (i.e., no copy-paste or clone-and-own)
* The artifact shall include a proper website
* The artifact shall include a wiki-page

## How to setup a running environment?

* The artifact shall provide download instructions
* The artifact shall provide the open source code of the MDE tool
* The artifact shall provide a binary/compiled version for the MDE tool
* The artifact shall provide a container for freezing dependencies and quickly setting up an environment for the MDE tool (e.g., VM, Docker)
* The artifact shall provide a step-by-step tutorial of how to build the source code
* The artifact compilation shall rely on build automation tools (e.g., make, ant)
* The artifact compilation shall rely on dependency management tools (e.g., maven, pip)
* The artifact shall provide instructions to install the MDE tool
* The artifact shall provide instructions to obtain libraries or dependencies
* Once downloaded, the artifact shall work without network connection
* The artifact shall provides means to verify file integrity (e.g., SHA-256, MD5)

## How to get started?

* The artifact shall include a small test data and configuration files/parameters as running example (e.g., low hanging fruit, quick run, smoke test)
* The artifact shall include a simple benchmark test data (e.g., reproducibility with reasonably modest resources or time constraints)
* The artifact shall indicate the most relevant and interesting parts of the source code/artifact
* The artifact shall include a document with a step-by-step for running the artifact (e.g., README, PDF)
* The artifact shall include a video tutorial with a step-by-step for running the artifact (e.g., Youtube, Vimeo)
* Experiment execution shall be controlled by if-else structures and parameters, rather than code (un)commenting

## How to replicate the experiment?

* It shall provide instructions for manual/automated pre-processing of raw data for experiments (e.g., bash, python, Rscript)
* It shall provide instructions for manual/automated replication of the complete (or at least a subset) experiment as in the paper (e.g., bash, python, Rscript)
* Experiment workflow shall be fully automated from raw data processing, over experiment execution, to figures plotting
* The artifact shall provide means to evaluate the scalability of the MDE tool with large-scale models (e.g., realistic/synthetic, incremental size/complexity)
* The artifact shall include the experiment results shown in its associated paper in tabular and machine-readable format (e.g., csv, tab)
* The artifact shall include the figures/diagrams shown in its associated paper (e.g., PNG, JPG, PDF, SVG)
* The artifact shall include log files produced by the MDE tool
* The artifact shall include the complete set of test models analyzed

## How to run the analysis of results?

* The artifact shall include scripts for the automated analysis of results as in the paper (e.g., Rscripts, python scripts, Jupyter notebooks)
* The artifact shall includes scripts for drawing figures and/or plots as in the paper (e.g., Rscript, python scripts, Jupyter notebook)
* The artifact shall provide a clear description of the measurement procedures and metrics used in the paper

## How it could be repurposed?

* The artifact shall indicate suggestions for contributions (e.g., notes.txt, todo.txt, ways it could be repurposed, wishlist)
* The artifact documentation shall be designed considering users with minimal expertise
* The artifact shall provide details about ethical concerns in replications
* Experiment workflow shall allow to produce intermediate files to ease rework (e.g., increasing levels of cleanliness or task specificity)
* The artifact source code shall be documentated (e.g., in-code comments, javadocs)
* The artifact shall provide means for integration/chaining with commercial tools (e.g., Matlab, DOORS)
* The artifact shall describe other potential applications or problems that could be addressed
* The artifact shall report known issues, bugs and limitations (e.g., issue tracker)

# How much?

## How much resource does it need?

* The artifact shall indicate the system and environment where it was successfuly evaluated (e.g., OS, CPU, RAM, GPU, Disk)
* The artifact shall indicate the minimum system and environment requirements for usage (e.g., OS, CPU, RAM, GPU, Disk)
* The artifact shall indicate the skills and/or settings required for usage (e.g., team configuration, users' skills)
* The artifact shall indicate the approximate amount of time needed to replicate the experiment