

Sisyphus (1548–49)  
by Titian, Prado  
Museum, Madrid,  
Spain



# MPS-Sisyphos

A build tool for  
modular MPS projects

Arjan Oortgiese and Johan Blok



3X10



Seville 17-19 October, 2024

<https://langdevcon.org>

# What has this to do with a build tool?

1. It is boring, so plenty of time to think about the build problem
2. At our work we have a big monolithic MPS plugin (ALEF)
3. Questions that arise
  - Why are MPS extensions monolithic?
  - Is Gradle or Maven the solution?
  - MPS build language
4. Splitting this is hard
  - Open Source vs Private edition vs Extensions you are going to make
5. Software Bill Of Material (SBOM)



# What has this to do with a build tool?

1. You need to be creative
2. You need to work together



Is this project open source?



# What has this to do with a build tool?

The screenshot shows the Overheid.nl website interface. The browser address bar displays 'wetten.overheid.nl'. The page title is 'Wet open overheid'. The main content area includes a search bar with options for 'Eenvoudig zoeken', 'Uitgebreid zoeken', and 'Zoeken in EU-richtlijnen'. Below the search bar, the breadcrumb 'U bent hier: Zoeken / Regeling' is visible. The main content area is divided into a left sidebar and a main content area. The sidebar contains a search bar, a section for 'Wet open overheid' with a date range 'Geraadpleegd op 08-10-2024. Geldend van 01-10-2024 t/m heden', and an 'Inhoudsopgave' (Table of Contents) with expandable sections for 'Hoofdstuk 1 Rechten (Artikel 1.1)', 'Hoofdstuk 2 Algemene bepalingen (Artikelen 2.1-2.5)', and 'Hoofdstuk 3 Openbaarmaking uit eigen beweging'. The main content area displays the title 'Wet open overheid' and a summary: 'Wijziging(en) op nader te bepalen datum(s); laatste bekendgemaakt in 2021. Zie het [overzicht van wijzigingen](#). Geraadpleegd op 08-10-2024. Geldend van 01-10-2024 t/m heden'. Below this, there are sections for 'Hoofdstuk 1. Rechten' and 'Artikel 1.1. Recht op toegang', followed by a paragraph: 'Eenieder heeft recht op toegang tot publieke informatie zonder daartoe een belang te hoeven stellen, behoudens bij deze wet gestelde beperkingen.' The page also features a table of contents for 'Hoofdstuk 2. Algemene bepalingen' and 'Artikel 2.1. Begripsbepalingen'. A large red watermark 'NOT APPLICABLE' is overlaid diagonally across the page.

# Let's work together

- We started a proof of concept
- Available at: <https://github.com/mps-sisyphus/mps-sisyphus>
- License: Apache license 2.0



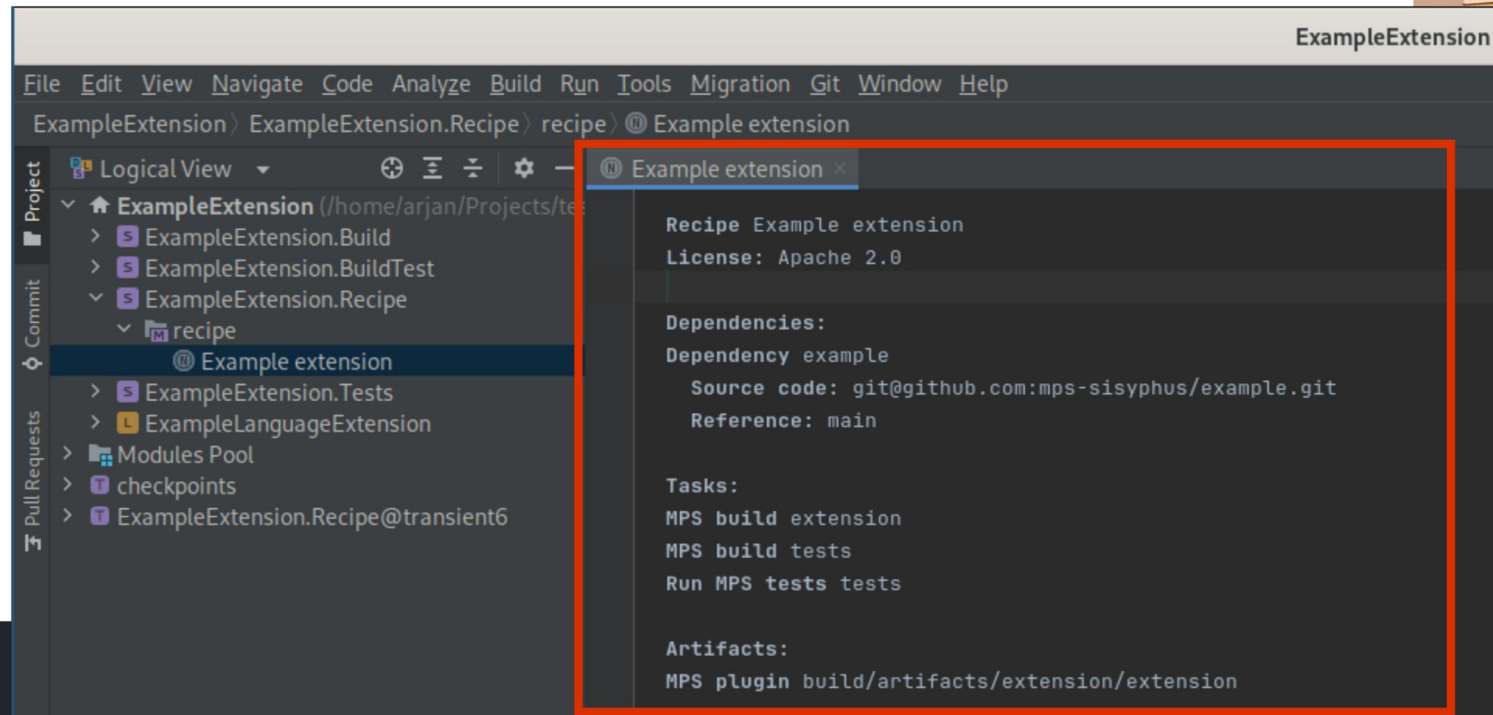
# The goals of this project

1. Have an uniform way of building MPS projects
2. Allow modular MPS projects
3. Try to use default MPS components as much as possible
4. Keep the MPS workflow
5. Specify at one place
6. Inspire people (and JetBrains 😜) to fix this
7. Work together as a community



# It all starts with a recipe

- Open a MPS project
- Create a solution and model for the build recipe
- Add the Sisyphos.recipe language
- Define the recipe
  - Specify dependencies
  - Link your MPS build scripts
- Remove build\*.xml files
  - e.g. .gitignore them



```
Recipe Example extension
License: Apache 2.0

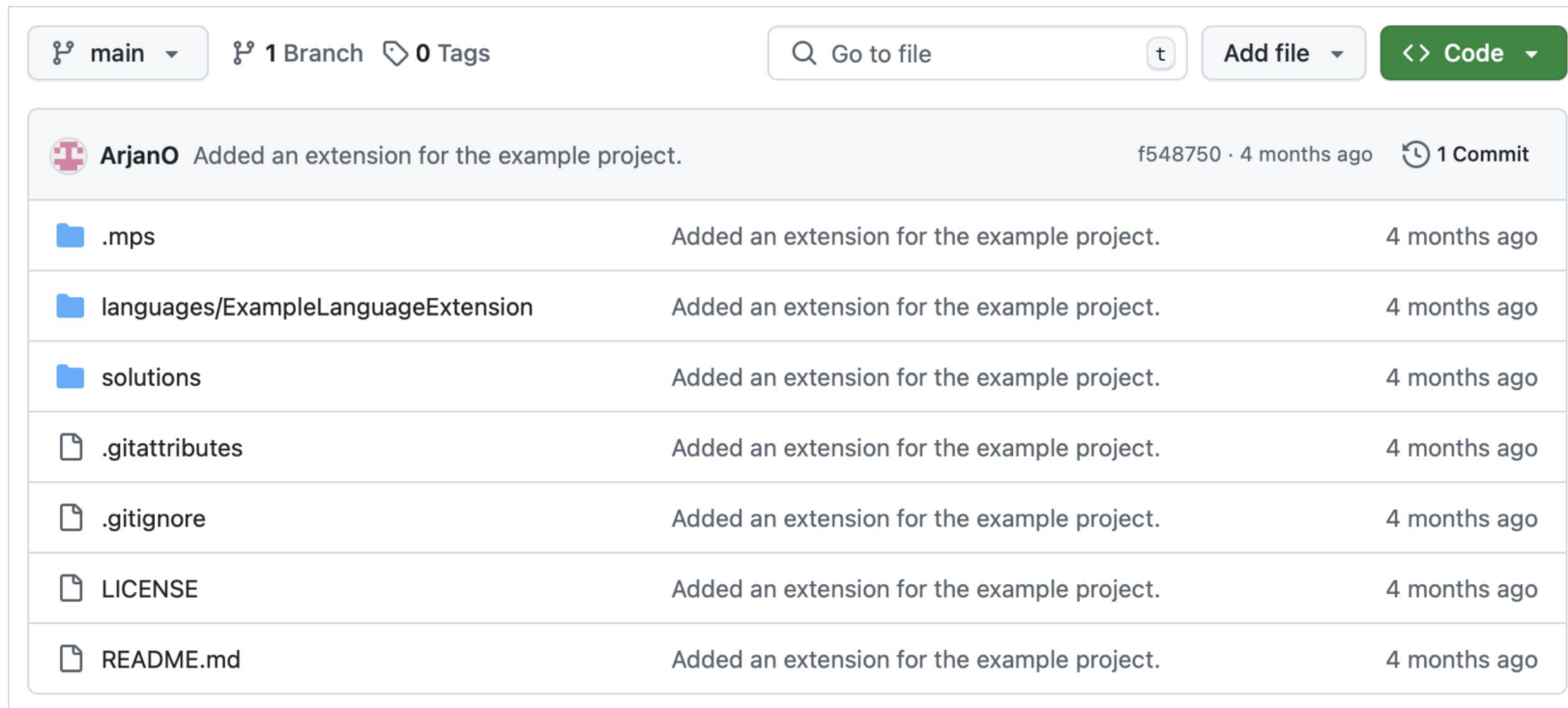
Dependencies:
Dependency example
  Source code: git@github.com:mps-sisyphus/example.git
  Reference: main

Tasks:
MPS build extension
MPS build tests
Run MPS tests tests

Artifacts:
MPS plugin build/artifacts/extension/extension
```

# Clean MPS projects 🥰

We don't have **build.xml** file(s) anymore



main 1 Branch 0 Tags

Go to file t Add file <> Code

ArjanO Added an extension for the example project. f548750 · 4 months ago 1 Commit

📁 .mps	Added an extension for the example project.	4 months ago
📁 languages/ExampleLanguageExtension	Added an extension for the example project.	4 months ago
📁 solutions	Added an extension for the example project.	4 months ago
📄 .gitattributes	Added an extension for the example project.	4 months ago
📄 .gitignore	Added an extension for the example project.	4 months ago
📄 LICENSE	Added an extension for the example project.	4 months ago
📄 README.md	Added an extension for the example project.	4 months ago



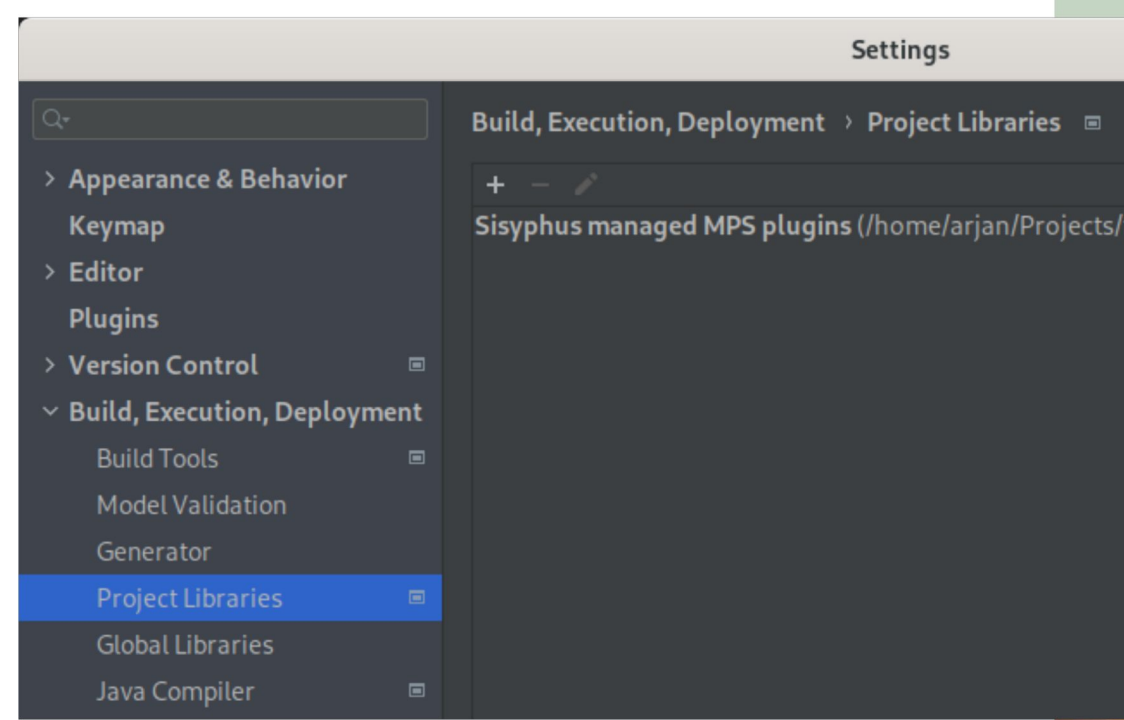
# Let Sisyphos show its cooking strength

```
#!/bin/sh
export MPS_HOME=/apps/MPS-2024.1 # Path to your MPS installation.
sh "$MPS_HOME/plugins/Sisyphus/bin/sisyphus.sh"

# You are also allowed to have Sisyphus outside your MPS installation.
# For example when developing use:
sh "~/projects/mps-sisyphus/build/Sisyphus/Sisyphus/bin/sisyphus.sh"
```

# What will it do?

1. Generate sisypkos.recipe file
2. Create a mps-plugin repo and install this in your project.
3. Fetch the dependencies repo's
  - Run this process for the repository
4. Copy the dependencies to the project plugins repo
5. Generate ANT build files
6. Run tests
7. Create a SBOM (Software Bill Of Material)



# Software Bill Of Material (SBOM)

1. Part of the Digital Markets Act (DMA) EU law
2. An ingredient list of your software

The screenshot displays a software management interface for a project named 'example-extension 1.0.0'. The dashboard includes a navigation sidebar on the left, a top navigation bar with the breadcrumb 'Home / Projects / example-extension 1.0.0', and a main content area. The main content area features a header with the project name and version, a 'View Details >' link, and a row of five circular status indicators (red, orange, yellow, green, grey) each containing the number '0'. Below this is a horizontal menu with tabs for 'Overview', 'Components 3', 'Services 0', 'Dependency Graph 0', 'Audit Vulnerabilities 0 0', 'Exploit Predictions 0', and 'Policy Violations 0 0 0 0'. The 'Components' tab is active, showing a table of components. The table has columns for Component, Version, Group, Internal, License, Risk Score, and Vulnerabilities. Three components are listed: JetBrains MPS (version 2024.1, Apache-2.0 license), example (version main, Apache-2.0 license), and mps-sisyphus (version 1.0.0, Apache-2.0 license). Each component has a risk score of 0 and a vulnerability bar showing 0 vulnerabilities. The interface also includes buttons for '+ Add Component', '- Remove Component', 'Upload BOM', and 'Download BOM', along with filters for 'Outdated only' and 'Direct only'. A search bar and a refresh button are also present.

Component	Version	Group	Internal	License	Risk Score	Vulnerabilities
JetBrains MPS	2024.1			Apache-2.0	0	0
example	main				0	0
mps-sisyphus	1.0.0			Apache-2.0	0	0

Showing 1 to 3 of 3 rows

# Under the hood

- Uses MPS runner to do things inside the MPS world
  - **jetbrains.mps.build.mps.runner** language
- Patched by **Sisyphus.build.mps.runner** language to:
  - Allows to run code from a plugin
  - Add support for passing information to the MPS runner
- Patched by **Sisyphus.build.mps.patch** to:
  - Remove unwanted tasks like generating
- We hope that JetBrains will make this possible out of the box 🙏

# Booting the build process

- Exists out of two runs using the MPS runner
- First run, just enough to:
  - Extract the recipe information
- Then we can fetch the dependencies
- Second run: generate **build.xml**
  - Run the generator for build solutions



# Next steps

- Detect conflicting dependencies
- Add local caching
  - Building always everything is not ideal
- Add support for artifact repo's
  - A Maven central
- Pull Java dependencies from Maven central
- Pull from JetBrains MPS plugin repository (marketplace)





# We are looking forward for your contribution

<https://github.com/mps-sisyphus/mps-sisyphus>



3X10



Seville 17-19 October, 2024

<https://langdevcon.org>