

JETBRAINS MPS COMMUNITY MEETUP

---

**PUBLISHING MPS CODE**

## SERGEJ KOŠČEJEV

- ▶ Consultant working with MPS since 2015
- ▶ [sergej@koscejev.cz](mailto:sergej@koscejev.cz)
- ▶ MPS Slack: @Sergej Koščejev
- ▶ Twitter: @s\_koscejev
- ▶ Blog: <https://specificlanguages.com>
- ▶ Beyond Parsing, a language engineering podcast with Federico Tomassetti (<http://beyondparsing.com>)

**SOME MOTIVATION**

# THE LEFT-PAD STORY – MARCH 2016

- ▶ Azer Koçulu unpublished >250 modules from NPM
- ▶ One of the unpublished modules was left-pad
- ▶ Indirectly used by many JavaScript projects (including Node.js and Babel)
- ▶ "One programmer almost broke the internet by deleting 11 lines of code" – Business Insider

```
module.exports = leftpad;

function leftpad (str, len) {
  var i = -1;
  len = len - str.length;

  while (++i < len) {
    str = ' ' + str;
  }

  return str;
}
```

# WHAT WOULD IT TAKE TO PUBLISH LEFT-PAD FOR MPS?

1. Is there something to copy & paste? Not really :-(
2. Create the MPS build script (otherwise you don't have a package)
3. **Choose a repository: JetBrains Plugins Repository? Maven Central? Bintray?**
4. **What format to publish?**
5. Sign up at a repository
6. Choose a build system: Ivy? Maven? Gradle?
7. Learn the build system
  - ▶ If it's easy to learn, it probably can't do what you need
8. **Write the publishing script (see point 1)**
9. **Oh, dependencies! How do I specify dependencies?**
10. Forget it, I'll just write a README.md and put it out on GitHub!

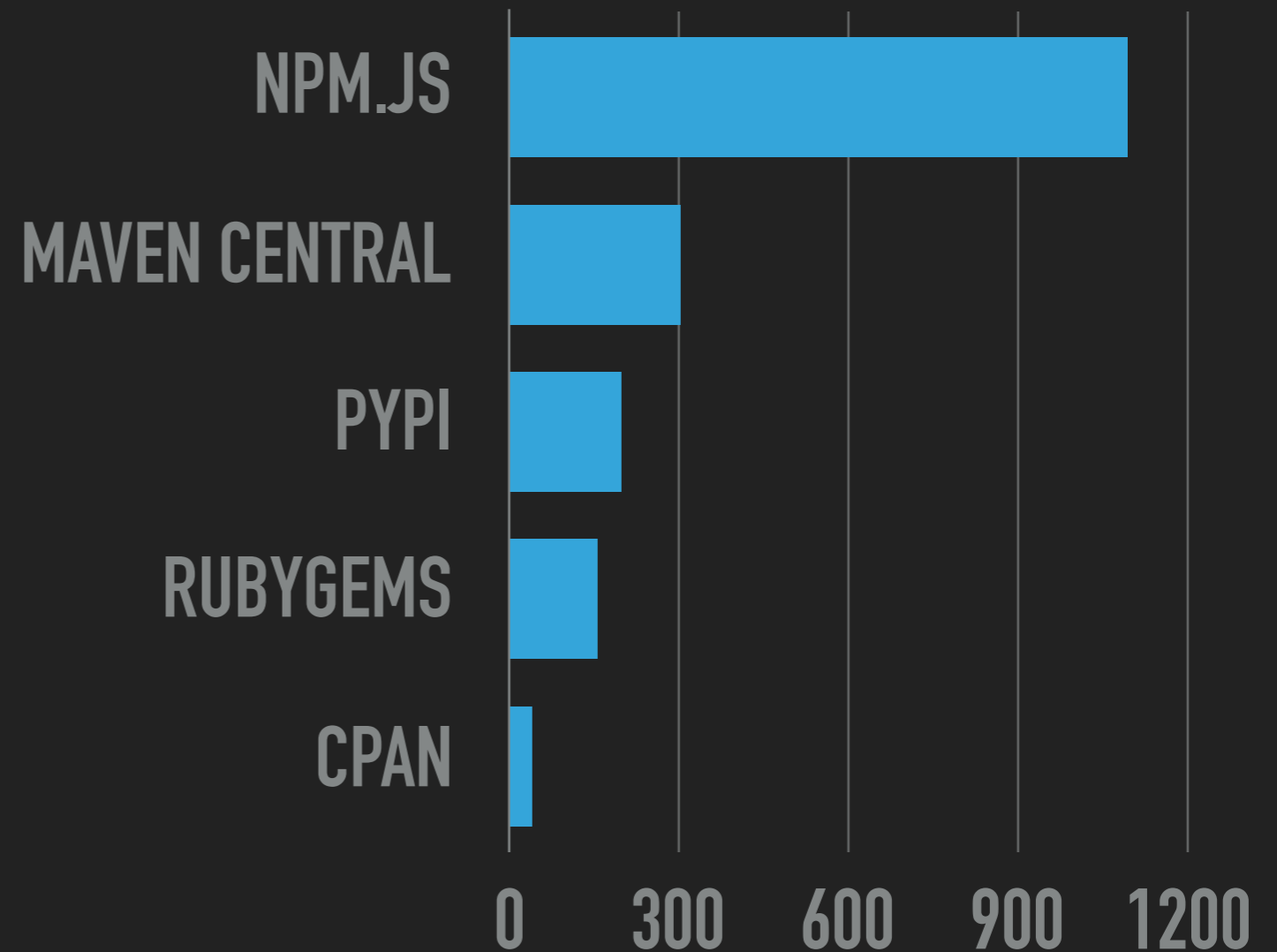
# WHAT WOULD IT TAKE TO CONSUME LEFT-PAD FOR MPS?

1. Hope that it's available somewhere as a package
  - ▶ ...and not just on GitHub with a README.md
  - ▶ ...and not just in the JetBrains Plugin Repository
2. Is there something to copy & paste? Well, not really...
3. Learn a build system
  - ▶ Figure out how to download and unpack an artifact
  - ▶ Figure out how to retrieve dependencies
4. Set up project variables
5. Write the build script
6. Forget it, I'll just write it myself
  - ▶ or copy from GitHub (and forget about updates)
  - ▶ or do without it

**SO WHAT?**

## MODULE COUNTS

- ▶ NPM: 1092K
- ▶ Maven Central: 304K
- ▶ PyPI: 198K
- ▶ RubyGems: 155K
- ▶ CPAN: 40K
- ▶ MPS: ~10?



Source: [modulecounts.com](https://modulecounts.com)



## SOME SECOND-ORDER EFFECTS

- ▶ Only large bundles of packages get published
- ▶ Large bundles are more complicated to maintain
  - ▶ Especially upgrading to each new MPS version
  - ▶ Splitting is difficult
- ▶ Large bundles are more complicated to depend on
  - ▶ Increased RCP sizes
  - ▶ I want just the grammar cells but I get: shadow models, interpreter, trace explorer, etc. etc.

**SOLUTION**

# DECISIONS

- ▶ Repository: Maven (e.g. [bintray.com](https://bintray.com), free for open-source projects)
- ▶ Packaging: Maven (POM + ZIP)
- ▶ Tool: Gradle with a custom plugin
- ▶ Scope: simple plugins that are collections of modules
  - ▶ NOT for building RCPs (yet)

**DEMO**

# THE PLUGIN

# BUILDING THE BUILD

- ▶ The build model has to be created using MPS, slightly adjusted
- ▶ Convention: the build model should be called \*.build.mps (or simply build.mps). It will be built automatically.
  - ▶ No need to check in generated code
  - ▶ No need for an allScripts build solution

# DEPENDENCIES

- ▶ Makes use of Gradle's dependency resolution mechanism
  - ▶ Supports "project dependencies" (multi-module projects)
- ▶ Transitive dependencies supported automatically
- ▶ No distinction between *uses*, *depends*, *extends*, *exports*, *imports*, etc. – that's up to MPS.


**THE FUTURE**



## IMPROVEMENTS

- ▶ Record version of MPS used to produce an artifact
- ▶ Support other major versions of MPS
- ▶ Eliminate duplicate MPS installs on multiproject builds
- ▶ Build RCPs
- ▶ Support JAR dependencies (for stubs)
- ▶ *Generate build models from Gradle dependencies*

## CREDITS

- ▶ I  Gradle
- ▶ mps-gradle-plugin

---

## WHERE DO I GET IT?

- ▶ Gradle Plugin Repository: <https://plugins.gradle.org/plugin/com.specificlanguages.mps>
- ▶ Source: <https://github.com/specificlanguages/mps-gradle-plugin>

**QUESTIONS?**