

# C++ Dependency Management: Package Consumption vs Development

Boris Kolpackov

Code Synthesis

v1.0, June 2018

***CODE  
SYNTHESIS***

## Packages

Node	(npm)	600,000
Python	(PyPI)	140,000
Rust	(cargo)	16,000
C++	(?)	?

# Versioning

## *continuous versioning*

1.2.3	release
1.2.3-a.1	pre-release
1.2.3-b.2	pre-release
1.2.3-a.0.20180531072540.e4d2d772de35	snapshot

# Package Repositories

- *version control*-based
- *archive*-based

# Version Control-Based Repository

## Convenient for Package Developers

- Easy to setup (GitHub)
- Project Repository == Package Repository

# Archive-Based Repository

## Convenient for Package Consumers

- Searchable, packages are easy to discover
- Reliable, packages won't disappear
- Secure, packages are signed
- Fast

## Key Points

- Easy creation, testing, and delivery of packages
- Frictionless dependency upgrades/downgrades
- Integrated toolchain with auto-synchronization
- Automated continuous versioning
- Archive and version control-based repositories

[build2.org](http://build2.org)