AstraZeneca

Site Presentation: AstraZeneca

Jonathan Jackson, PhD

EasyBuild User Meeting 2021

Wednesday Jan 27th
2021 @ 19:15 UTC

1

Introduction

AstraZeneca

About This Talk

1. Introduction
2. Scientific Computing Platform
3. EasyBuild Implementation
4. Successes and Challenges
5. Future

About me

- Lead engineer for applications on our Scientific Computing Platform
- 2 years with AstraZeneca
- 4 years working with EasyBuild
- 10 years in HPC
- 6 years as a HPC end-user in academia

2

Scientific Computing Platform

Photo by Science in HD on Unsplash

The Scientific Computing Platform

Supporting pre-clinical Research and Development activities

Purpose

- Drug discovery
- Synthesis and manufacture process modeling
- Next generation sequencing
- Machine learning and AI

Applications

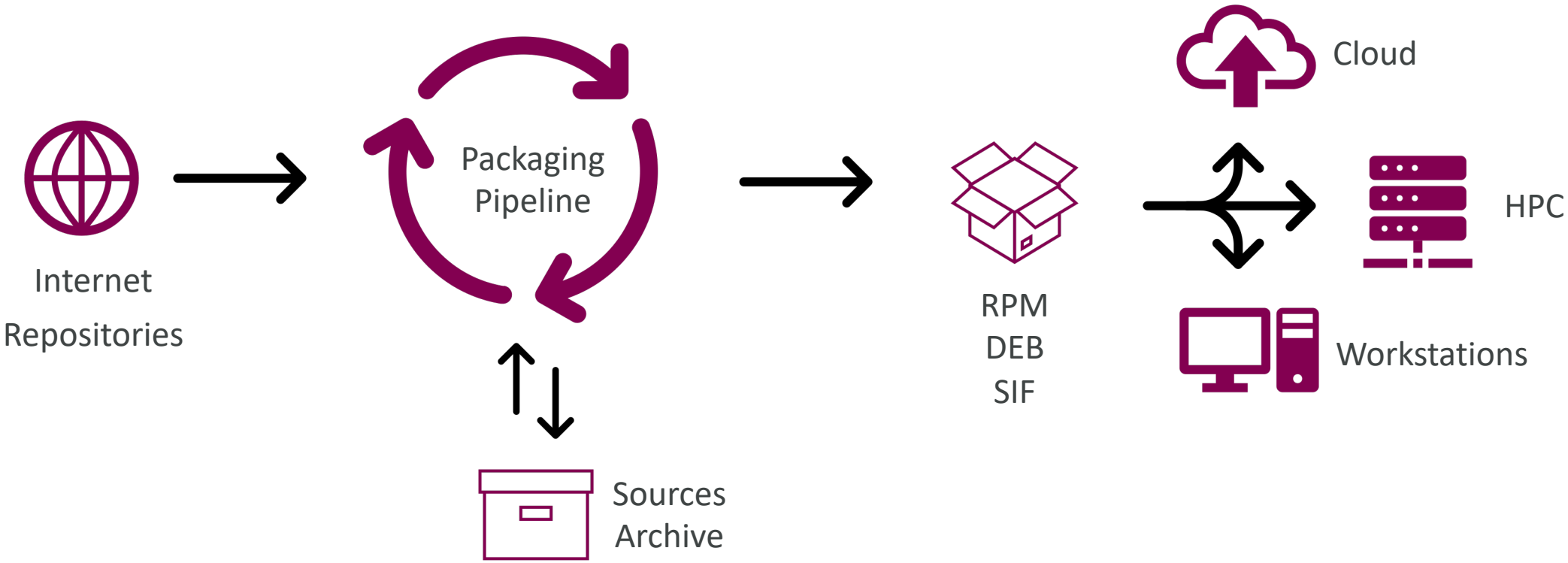
- 800+ EasyBuild applications
- 30+ web-based scientific services
- 2 Linux distributions
- 1600+ user accounts

Infrastructure

- 7800+ CPU cores, 382 GPUs
- 6 Virtual Desktop servers
- 3.4 PB high-performance storage
- Private Cloud

Core Design Criteria

- Reproducible pipeline
- Deployable artifacts
- Offline rebuild



Extra design considerations

- Isolated testing environment
- Optimize for available CPUs
- Regression Testing
- Keep It Simple

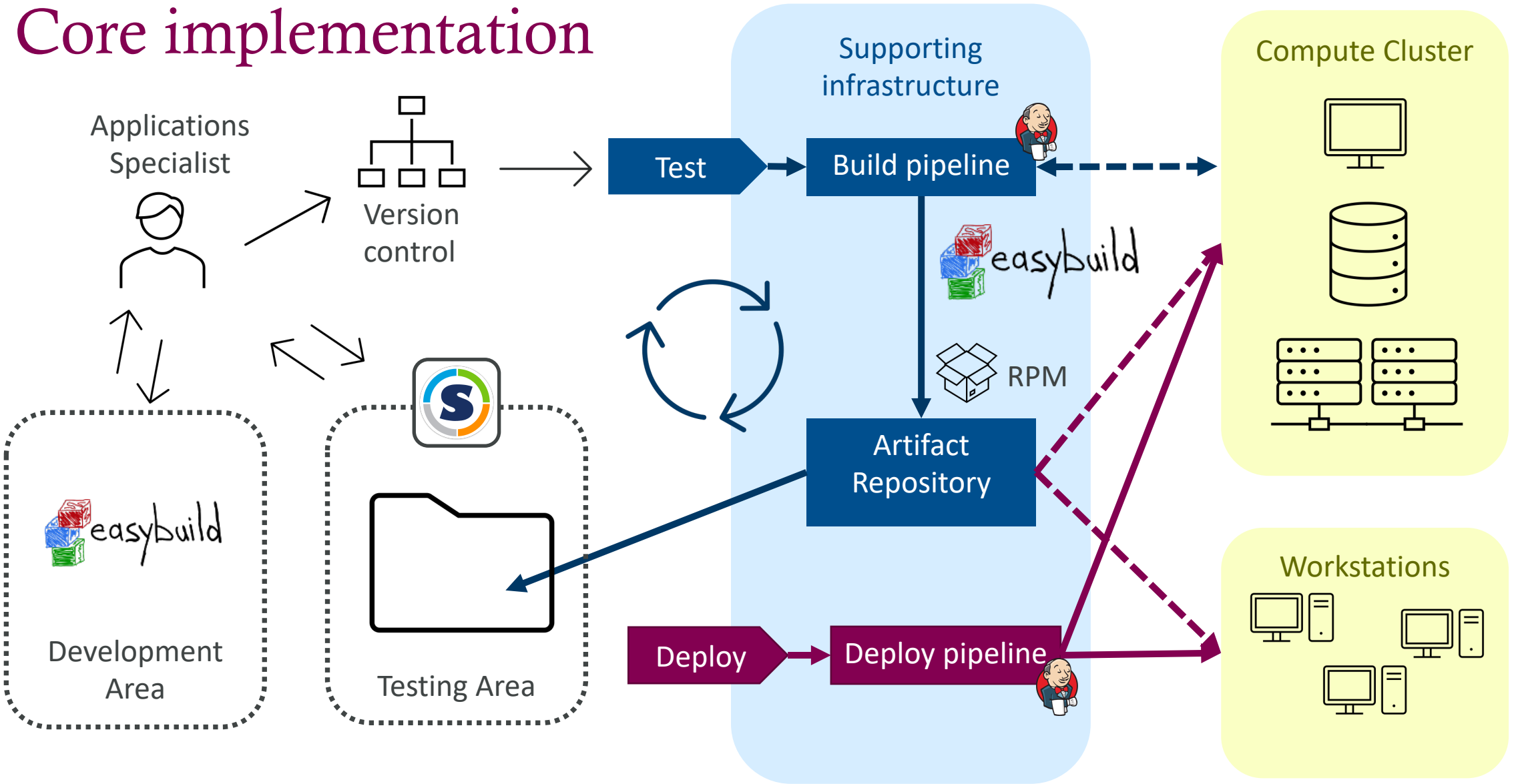
Reasons for choosing EasyBuild

- ✓ Dependency management
- ✓ Deployable artifacts RPM/DEB/SIF
- ✓ Community
- ✓ Build logs
- ✓ Checksums and version pinning
- ✓ Reproducible builds
- ✓ Minimize OS level dependencies

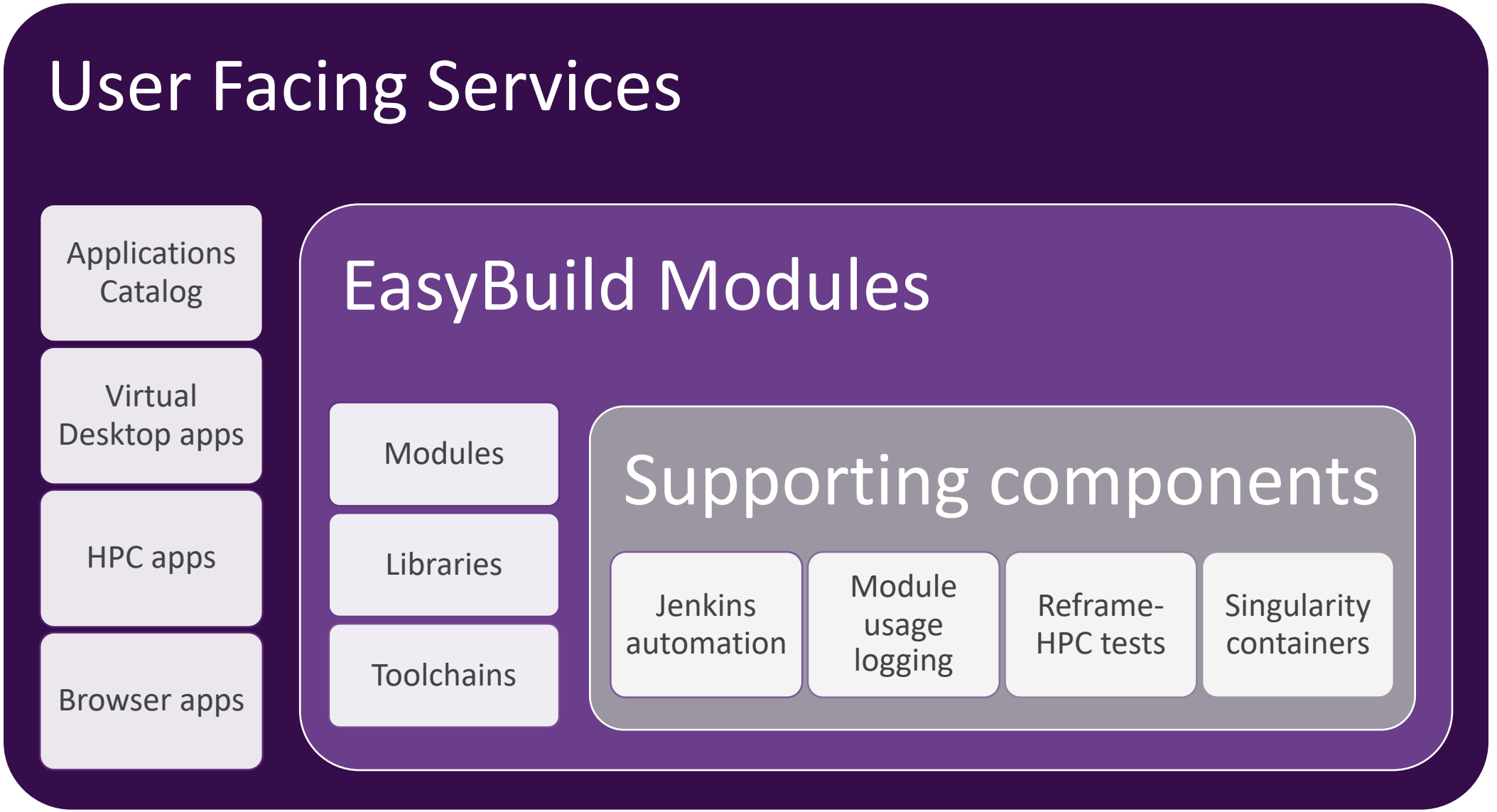
3

Easybuild Implementation

Core implementation



Applications Ecosystem



4

Successes and Challenges

Successes

EasyBuild community standards

- Saves time and improves quality
- Custom variants still possible

Deployable artifacts

- RPM workflow in place
  - Can be limits on RPM size/file count
- DEB / Singularity to follow

Reproducible pipeline

- Parameterized Jenkins job
  - Easyconfig file & Git branch
  - EasyBuild version
  - Build environment: GPU / CPU / memory

Offline Rebuild

- EasyBuild download step executed first
- EasyBuild build step runs without internet access

Reframe-HPC for regression testing

Challenges

Keeping it simple

- Supporting 2017a and 2019a generation toolchains
- Requires backporting easyconfig to our toolchain
- User communication and education
- Application lifecycle management

How to integrate containers with modules?

- Containers generated by EasyBuild
- Containers sourced from external repositories

How to manage conda?

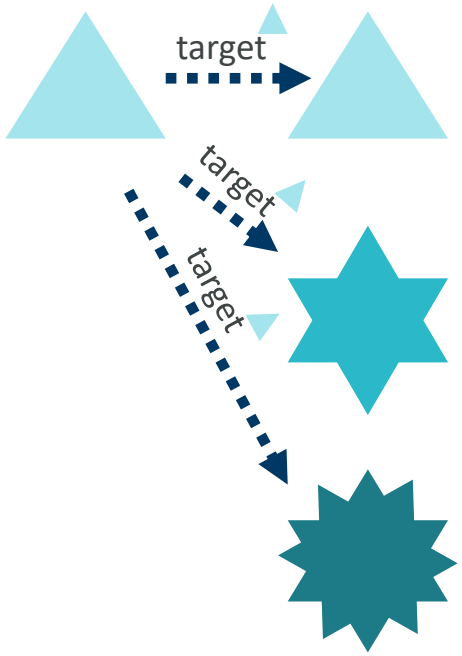
- May conflict with R, Python, CUDA
- Provides its own toolchain

Compiling for target architecture

Current State

Compile on **Deploy** to

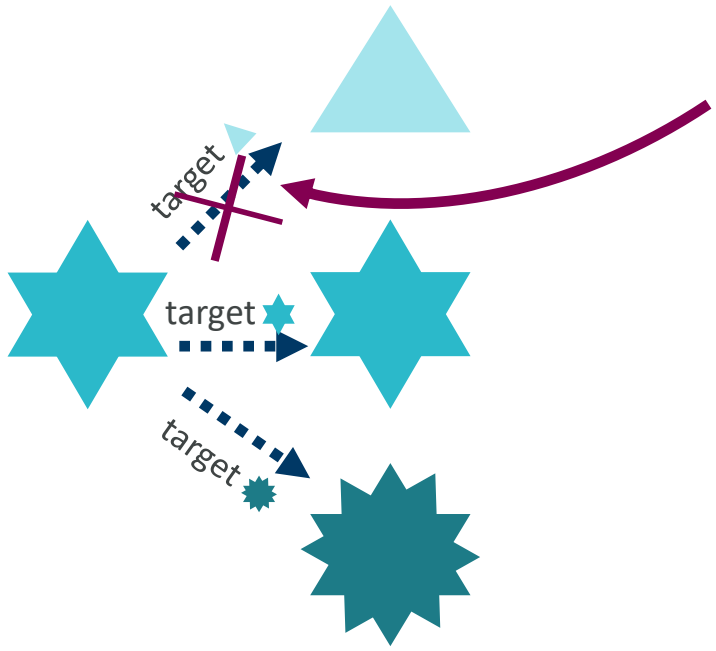
Minimum CPU architecture



Most CPU features

Ideal state

Compile on **Deploy** to



- Some build tools appear to ignore **`march=`** flag

Compile on	Minimum	Any
`march=`	Minimum	Target
Deploy to	All	Target

5

Future

Goals

- Increase our contributions to Easybuilders repositories
- Become proficient in easyblocks

- Reframe-HPC regression testing dashboards
- Continuous integration
- Container deployment integrated with modules
- All compiled applications optimized for target platform

- User Self-service via applications catalog
- Power users able to leverage EasyBuild for custom modules

Thanks to the EasyBuild community and the SCP Team