

ECP Community BOF Days

March 30 – April 1, 2021

Hosts: Ashley Barker (ORNL) and Osni Marques (LBNL)

Tiered Testing

Tim Haines (UW-Madison)

Jim Willenbring (Sandia)

Ryan Adamson (ORNL)

slides: github.com/hainest/TieredTesting

11AM EST



EXASCALE COMPUTING PROJECT

General Categories of Testing

Development Tests

- Tests run to protect stability while making changes to the code

Post-installation “smoke tests”

- A sanity test suite users can run after installing your software

Continuous Integration Testing

- High frequency, automated testing aimed at preventing changes from breaking key branches

Scalability Testing

- Running tests on very large systems (e.g., thousands of nodes)
- Not currently part of ST testing efforts

Continuous Deployment

- Combines testing and publishing software
- Not discussed here, see *Benefiting from ECP CI* on April 14th

Development Tests

These are the tests run prior to releasing a new version of your software

Examples:

- Unit
- Functional
- System
- Memory
- Acceptance

See these resources for more details:

<https://bssw.io/items/what-is-cse-software-testing>

<https://ideas-productivity.org/wordpress/wp-content/uploads/2016/04/IDEAS-TestingWhatAreSoftwareTestingPractices-V0.2.pdf>

The ST community policies only require that each project have some kind of testing.

Post-installation “Smoke Tests”

These tests are executed immediately after a successful build or installation to verify that your software can run in the current environment

Can be a subset of *development tests*

Should run in only a few minutes

- It’s just a **sanity check**!

Used by E4S teams to verify your software in their environments

- Making test results available through CDash is being worked on

Post-installation “Smoke Tests”

If you have smoke tests, please add them to your spack recipe so they will be run with `spack test`.

See **Checking an Installation** in the Spack docs for details

https://spack.readthedocs.io/en/latest/packaging_guide.html#checking-an-installation

If you have created validation tests in ECP-project/testsuite, migrate them to your spack package.

Post-installation “Smoke Tests”

Also update the ST Product Readiness page in Confluence

[Dashboard](#) / [2.3 Software Technology](#)

 [Edit](#)

 [Save for later](#)

 [Watching](#)

 [Share](#)

...

[/ ECP ST Exascale Readiness Portal](#) 

ECP ST Product Readiness

Created by Mike Heroux, last modified by Christian Trott on 2021-03-08

This page provides a single portal to information about product team efforts in preparation for the Aurora and Frontier platforms. Most efforts for Frontier can be leveraged for El Capitan as well.

NOTE: This page does not ask for NDA information. Note that the definition of each checkbox only asks if efforts are underway, but does not ask for the status of the efforts.

<https://confluence.exascaleproject.org/display/1ST/ECP+ST+Product+Readiness>

Post-installation “Smoke Tests”

And the SI Build and Test Dashboard

Dashboard / 2.4.4 Software Deployment at Facilities

 Edit

 Save for later

 Watching

 9+

 Share

...



SI Build and Test Dashboard

Created by Ryan Winter, last modified by Tammy Dahlgren on 2021-03-22

This page provides a snapshot of the status of ST Build and Test efforts based on manual updates. Summary information can be found at:

- Automated CI (non-NDA) DOE E4S builds and PR testing (OLCF Ascent, NERSC Cori, and Gitlab infrastructures): <https://cdash.spack.io>
- E4S (and xSDK-only): <https://docs.google.com/presentation/d/1BvKqfWgkuUMIWR1ABn-eMv7oMKqZBXar2J3KiCy1oxw/edit?usp=sharing>

<https://confluence.exascaleproject.org/display/HISD/SI+Build+and+Test+Dashboard>

Continuous Integration Testing

CI is the recommended method of running your tests often and automatically

The ECP ecosystem makes heavy use of CI

There are no hard-and-fast rules about what CI should be.

Each L4 team can develop their own practices.

Some information for getting started:

<https://bssw.io/items/what-is-continuous-integration-testing>

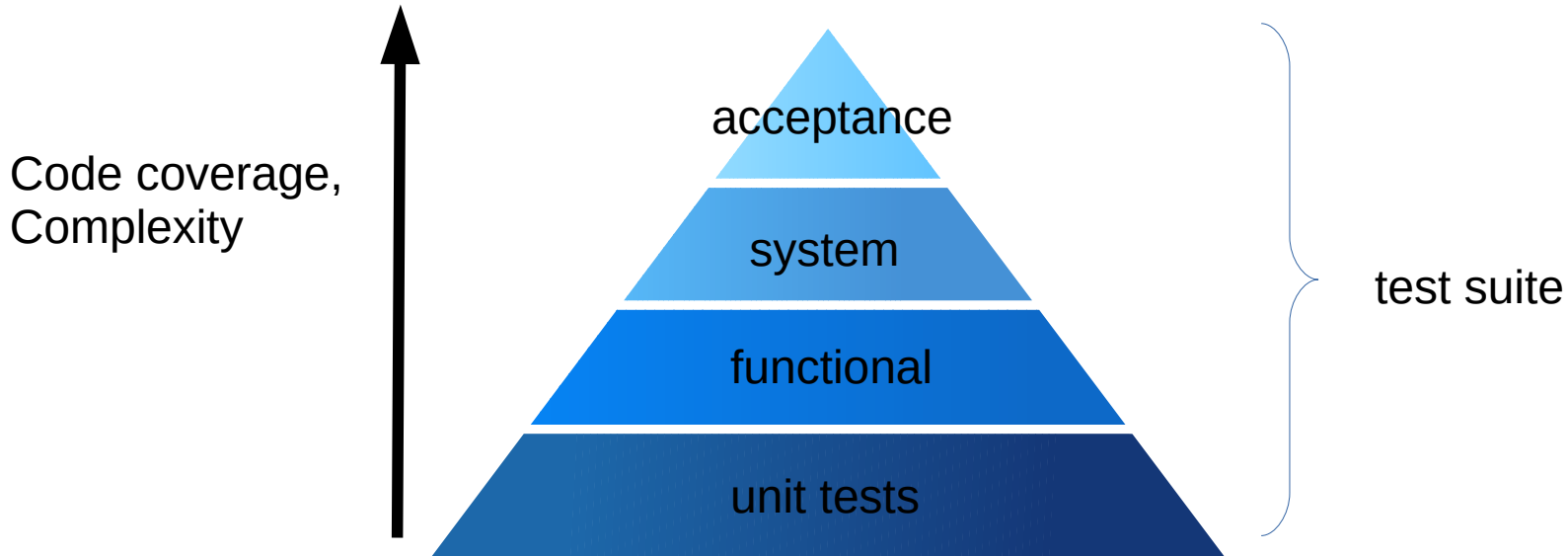
Most products have a CI Coordinator. Talk to him/her about CI for your project.

- CICs: If you would like more information about ECP's role in CI, please contact me

See also Benefiting from ECP CI on April 14th for a deep-dive

What is Tiered Testing?

1. Running different types of tests before creating a release of your software (aka your *test suite*)



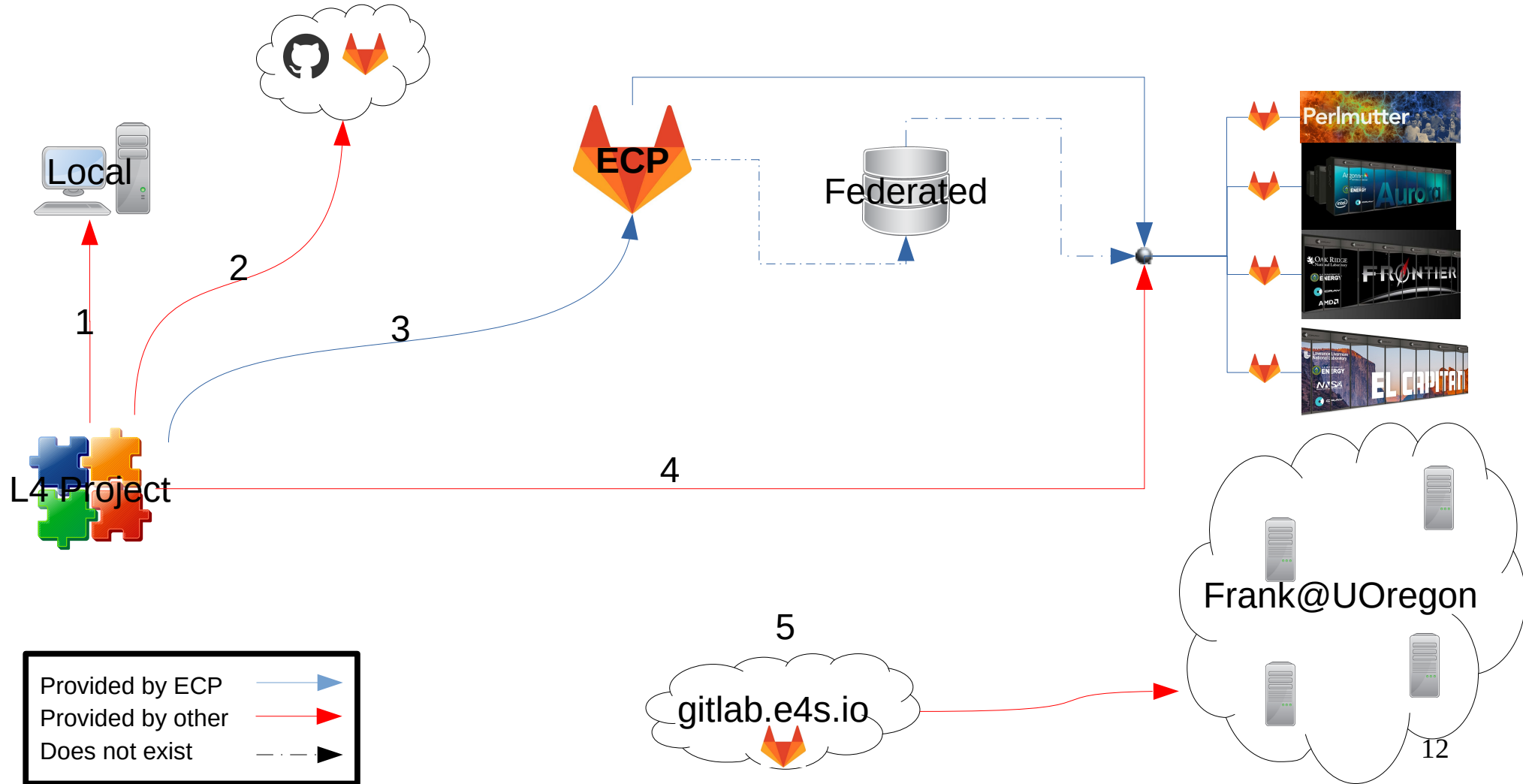
What is Tiered Testing?

1. Running different types of tests before creating a release of your software (aka your *test suite*)
2. Mapping tests to available hardware platforms

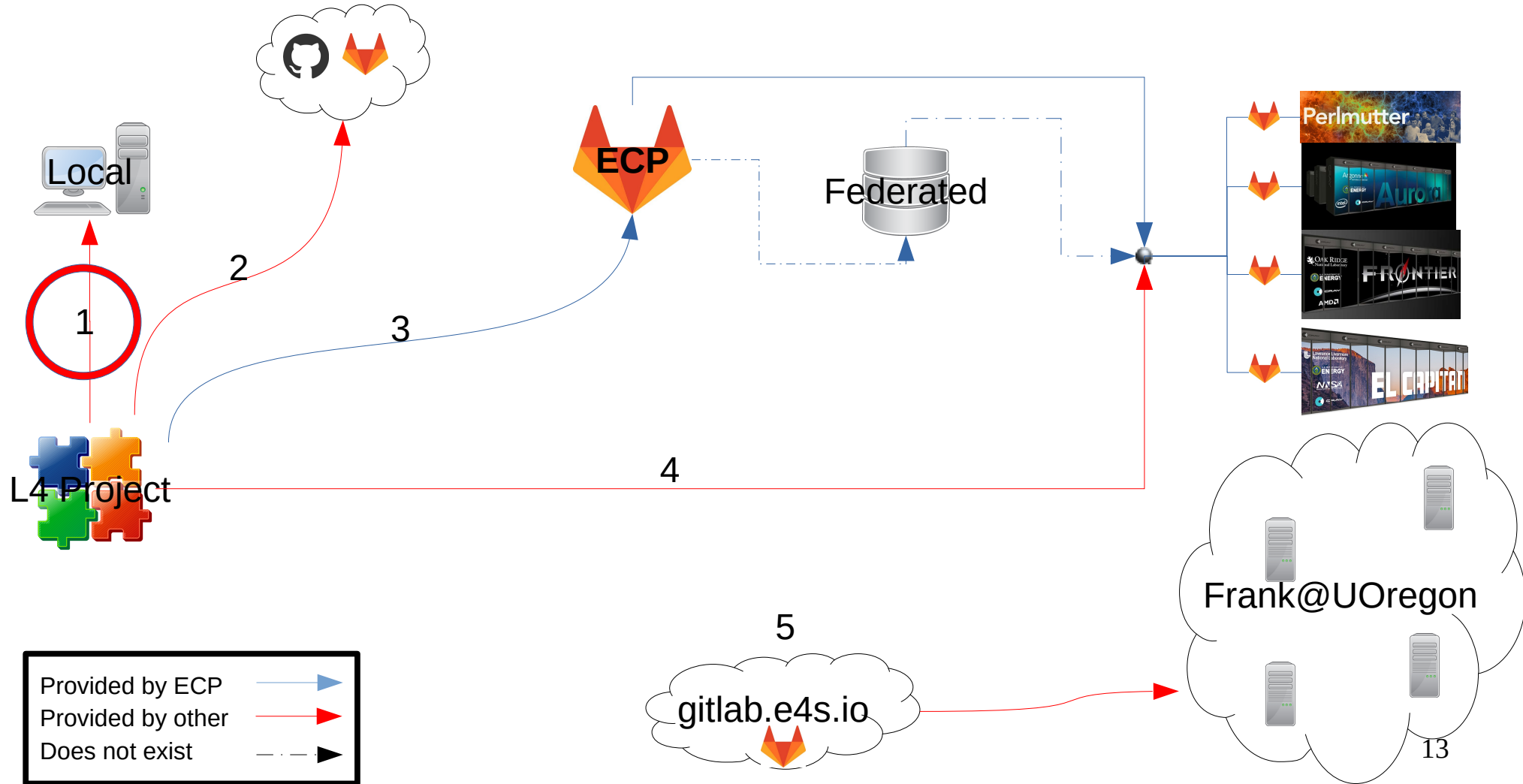
What is Tiered Testing?

Where can you run your test suite?

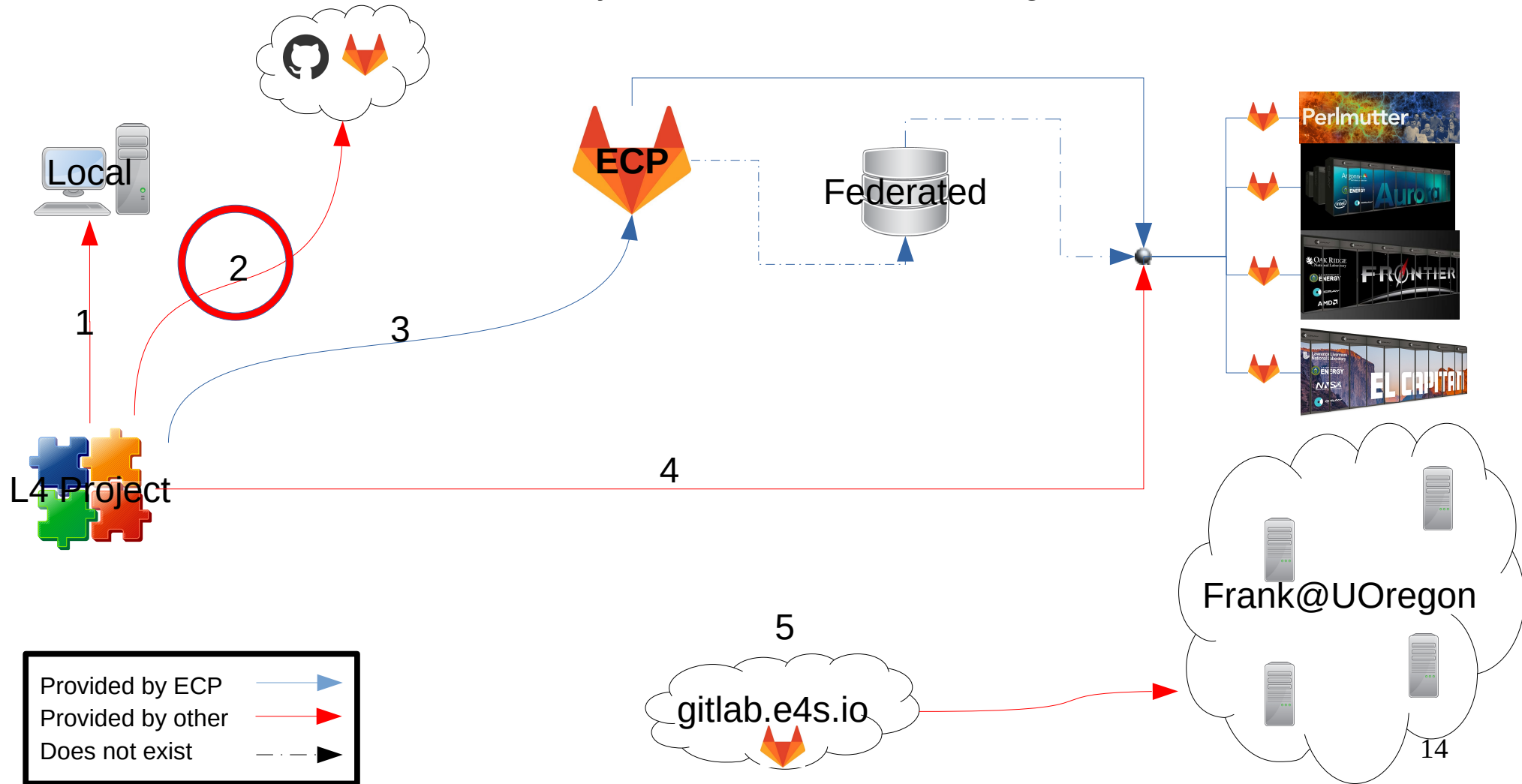
System Access for L4 Testing



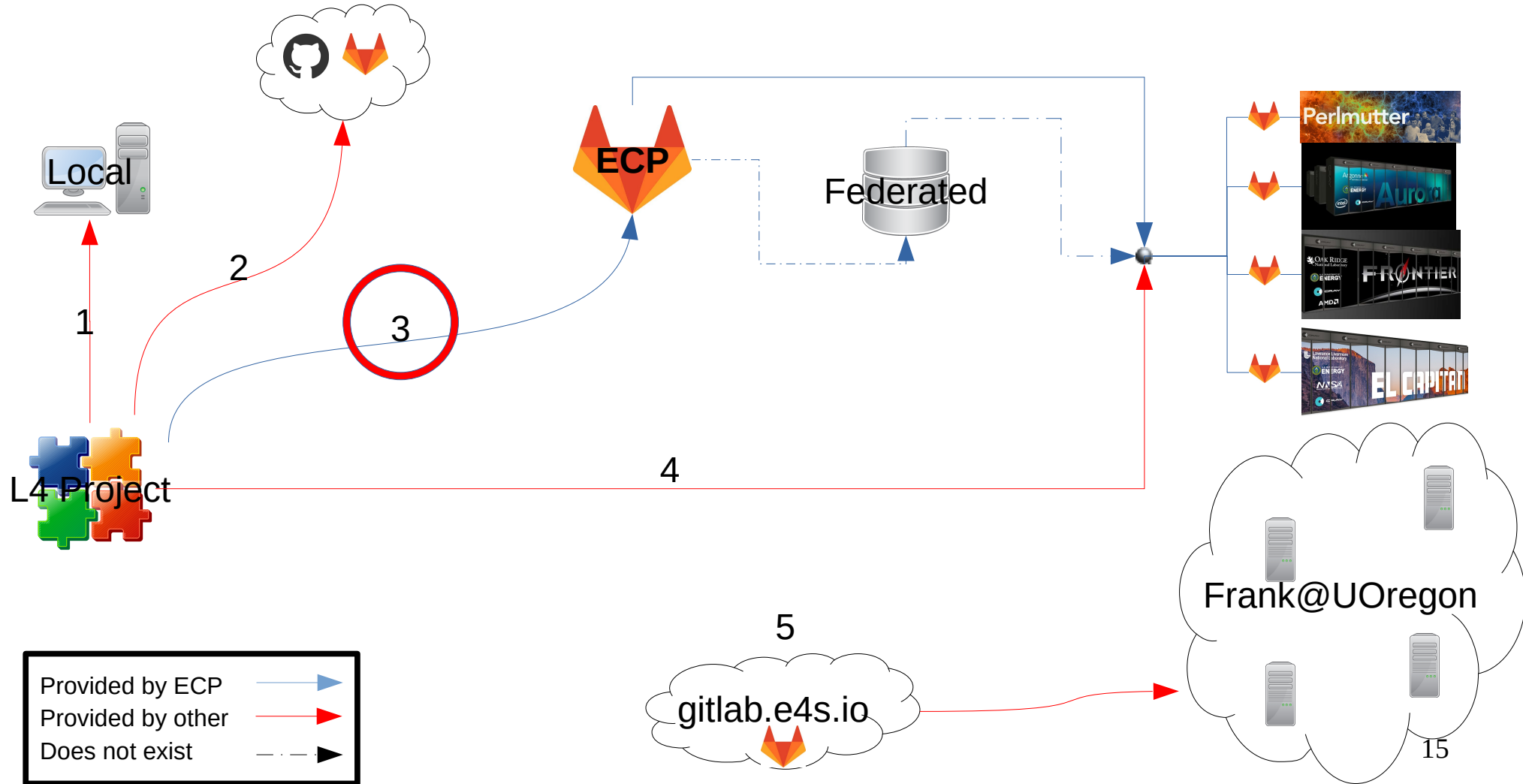
System Access for L4 Testing



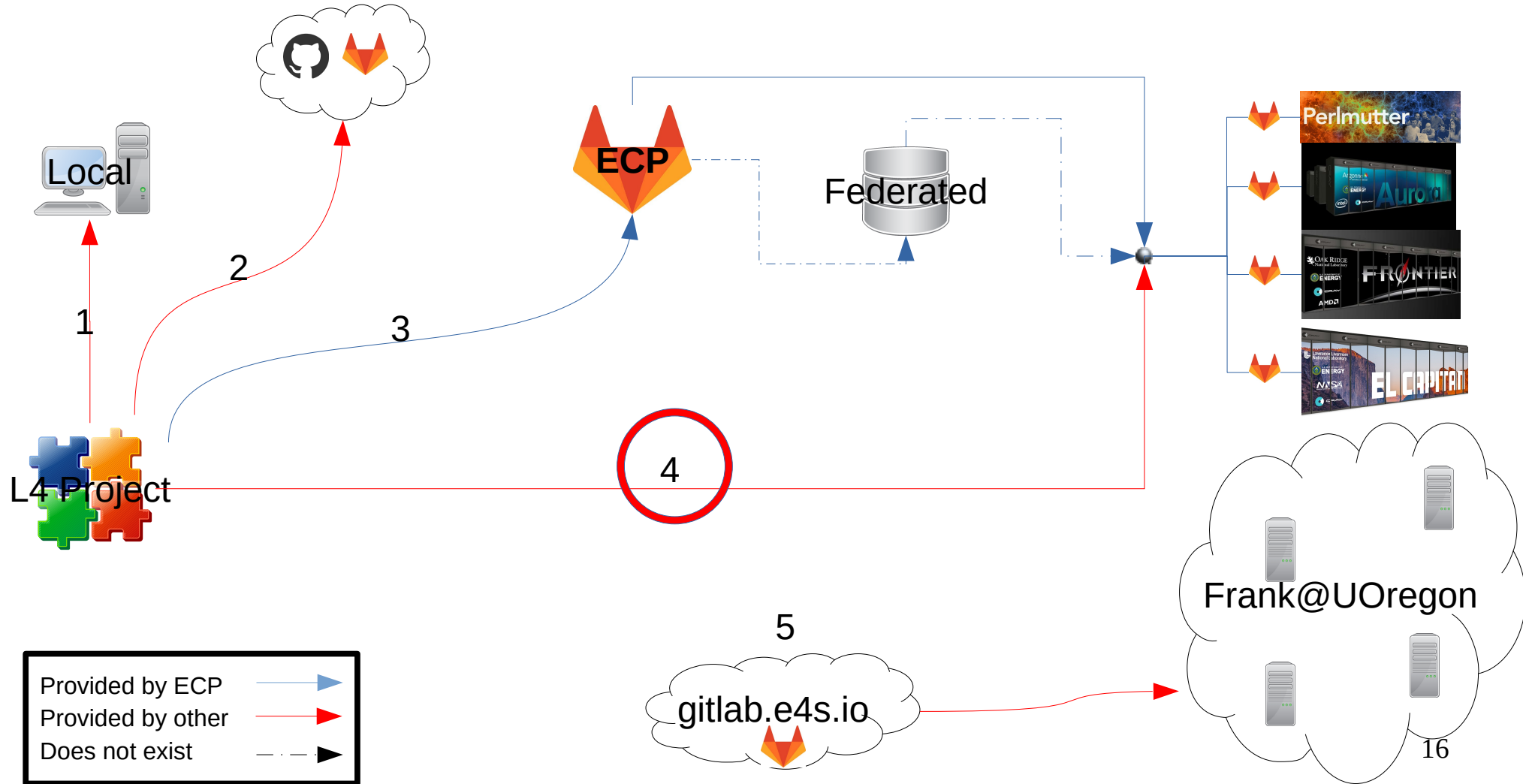
System Access for L4 Testing



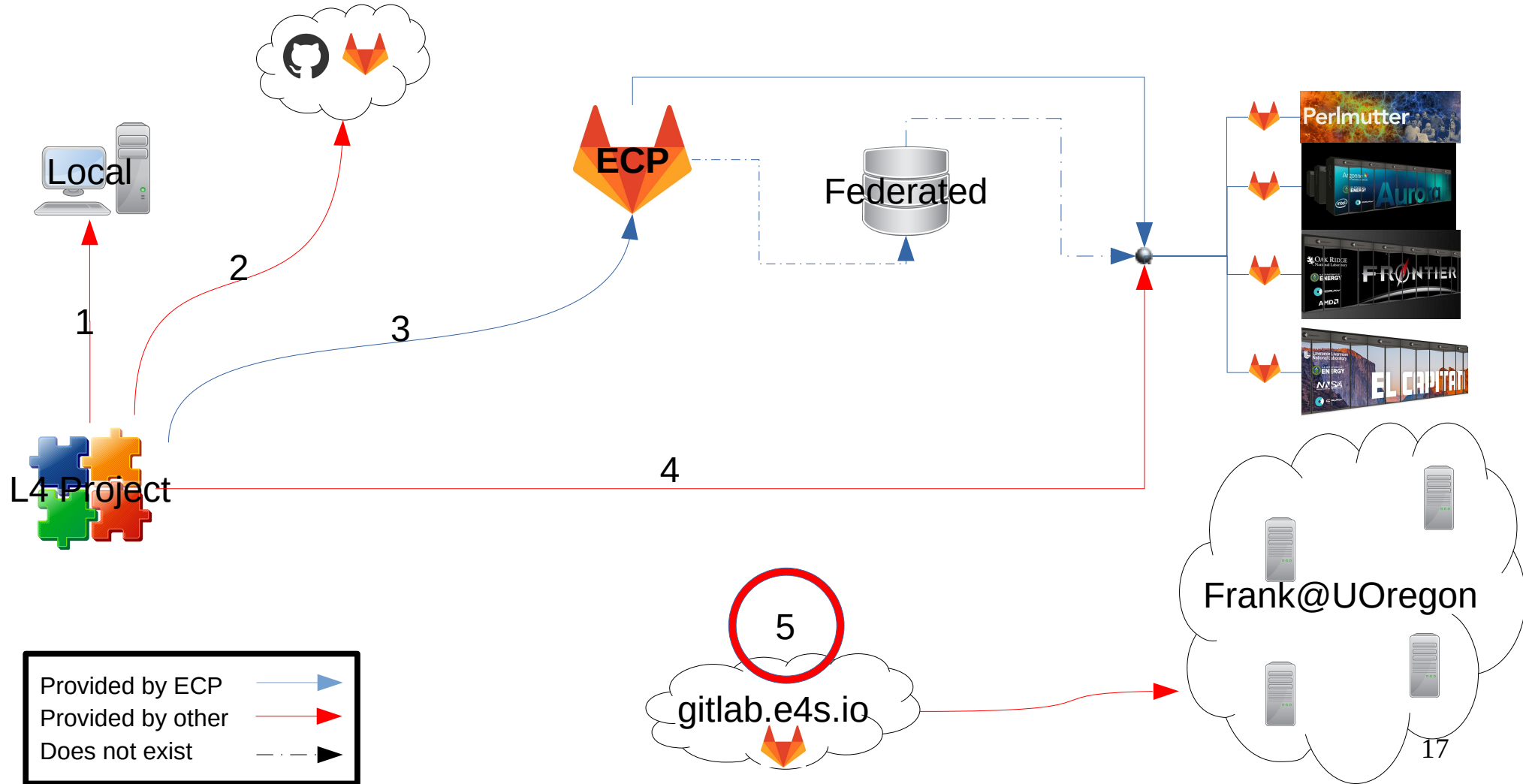
System Access for L4 Testing



System Access for L4 Testing



System Access for L4 Testing



What is Tiered Testing?

1. Running different types of tests before creating a release of your software (aka your *test suite*)
2. Mapping tests to available hardware platforms

General Categories of Testing

Development Tests

- Tests run to protect stability while making changes to the code

Post-installation “smoke tests”

- A sanity test suite users can run after installation

Continuous Integration Testing

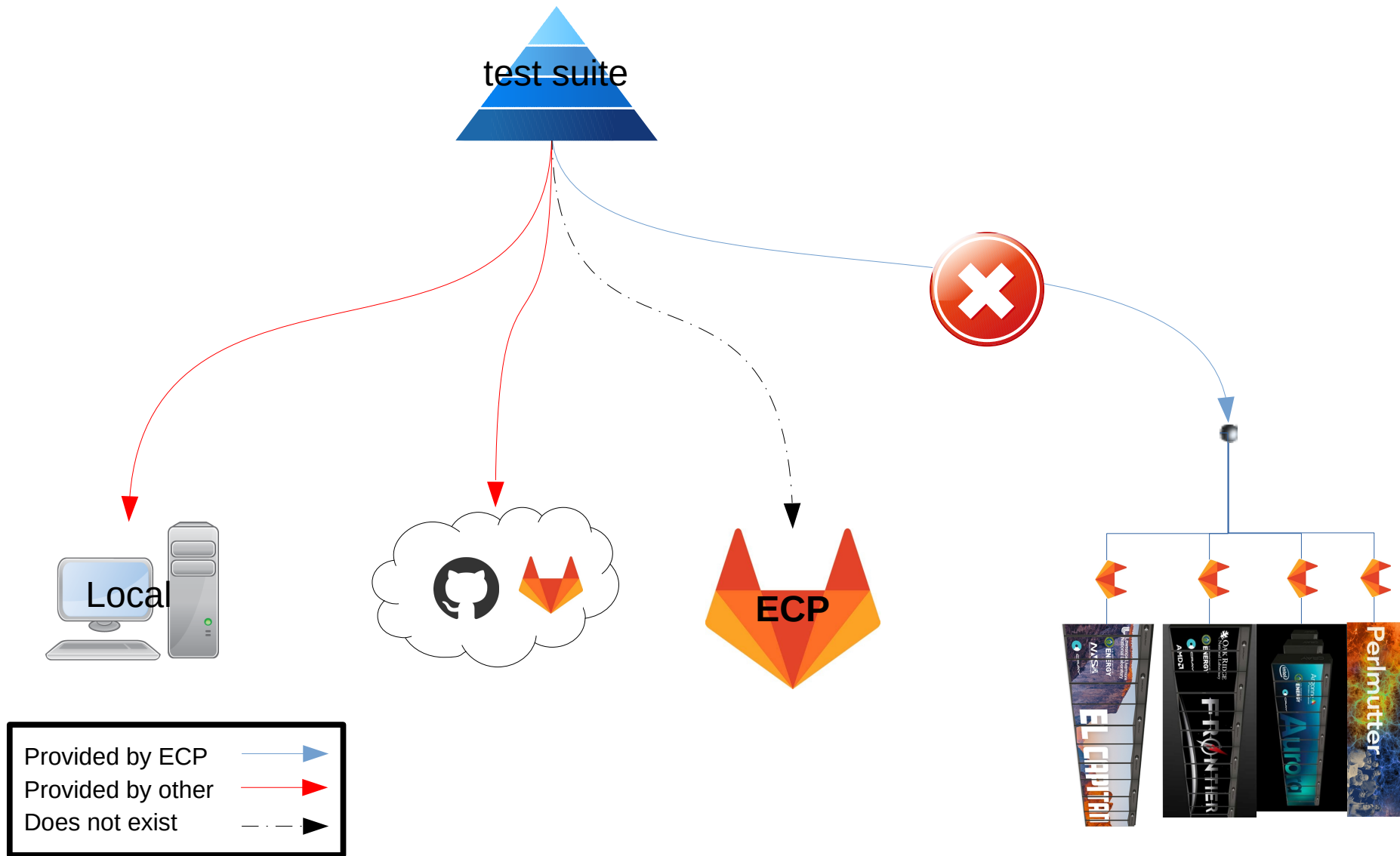
- High frequency, automated testing aimed at preventing changes from breaking key branches

Scalability Testing

- Running tests on very large systems (e.g., thousands of nodes)
- Not currently part of ST testing efforts

Continuous Deployment

- Combines testing and publishing software
- Not discussed here, see *Benefiting from ECP CI* on April 14th





LCF Resource Sub-Allocation

The ECP, ALCF, and OLCF are instituting baseline awards of time, e.g., set amount of time, that will be granted to all subprojects that currently are running on the LCF resources through an ECP User Program award. NO PROPOSAL IS NEEDED if you are eligible.

Eligibility: Each subproject that has a current award through the ECP User Program will receive the baseline award for the upcoming quarter and does not need to complete the form below. However, the project WILL need to complete the proposal form if either of the following conditions are true:

- * If their project does not have current allocation and/or
- * If the project wants to request **more time** than will be given to them in the baseline award.

Subprojects will receive the following baseline awards of time.

- * Theta: 1000 node-hours **per quarter** for Theta for AD and ST projects
- * Summit: 3000 node-hours **per quarter** for AD projects, 1500 node-hours **per quarter** for ST projects

General Categories of Testing

Development Tests

- Tests run to protect stability while making changes to the code

Post-installation “smoke tests”

- A sanity test suite users can run after installation

Continuous Integration Testing

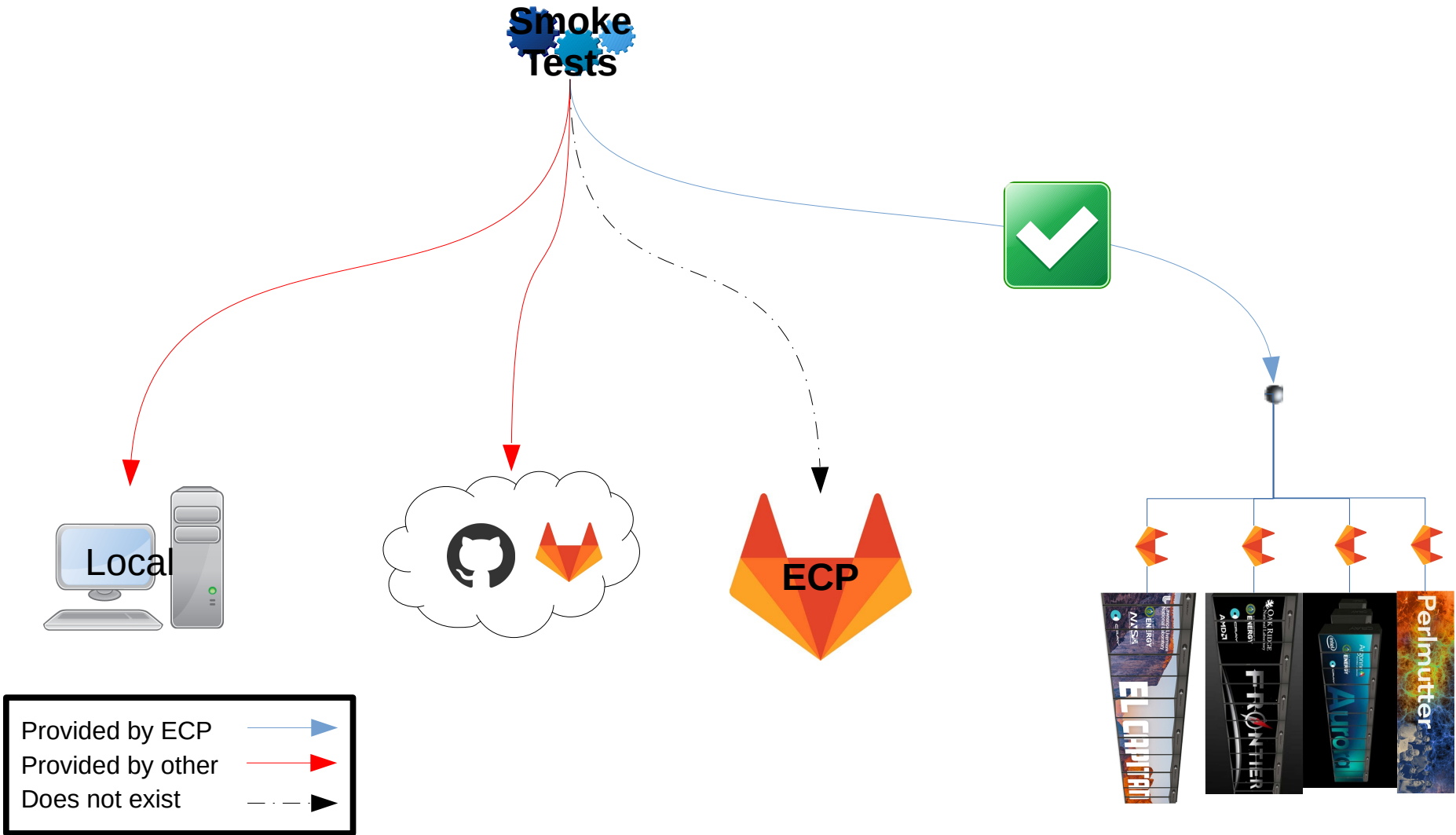
- High frequency, automated testing aimed at preventing changes from breaking key branches

Scalability Testing

- Running tests on very large systems (e.g., thousands of nodes)
- Not currently part of ST testing efforts

Continuous Deployment

- Combines testing and publishing software
- Not discussed here, see *Benefiting from ECP CI* on April 14th



General Categories of Testing

Development Tests

- Tests run to protect stability while making changes to the code

Post-installation “smoke tests”

- A sanity test suite users can run after installation

Continuous Integration Testing

- High frequency, automated testing aimed at preventing changes from breaking key branches

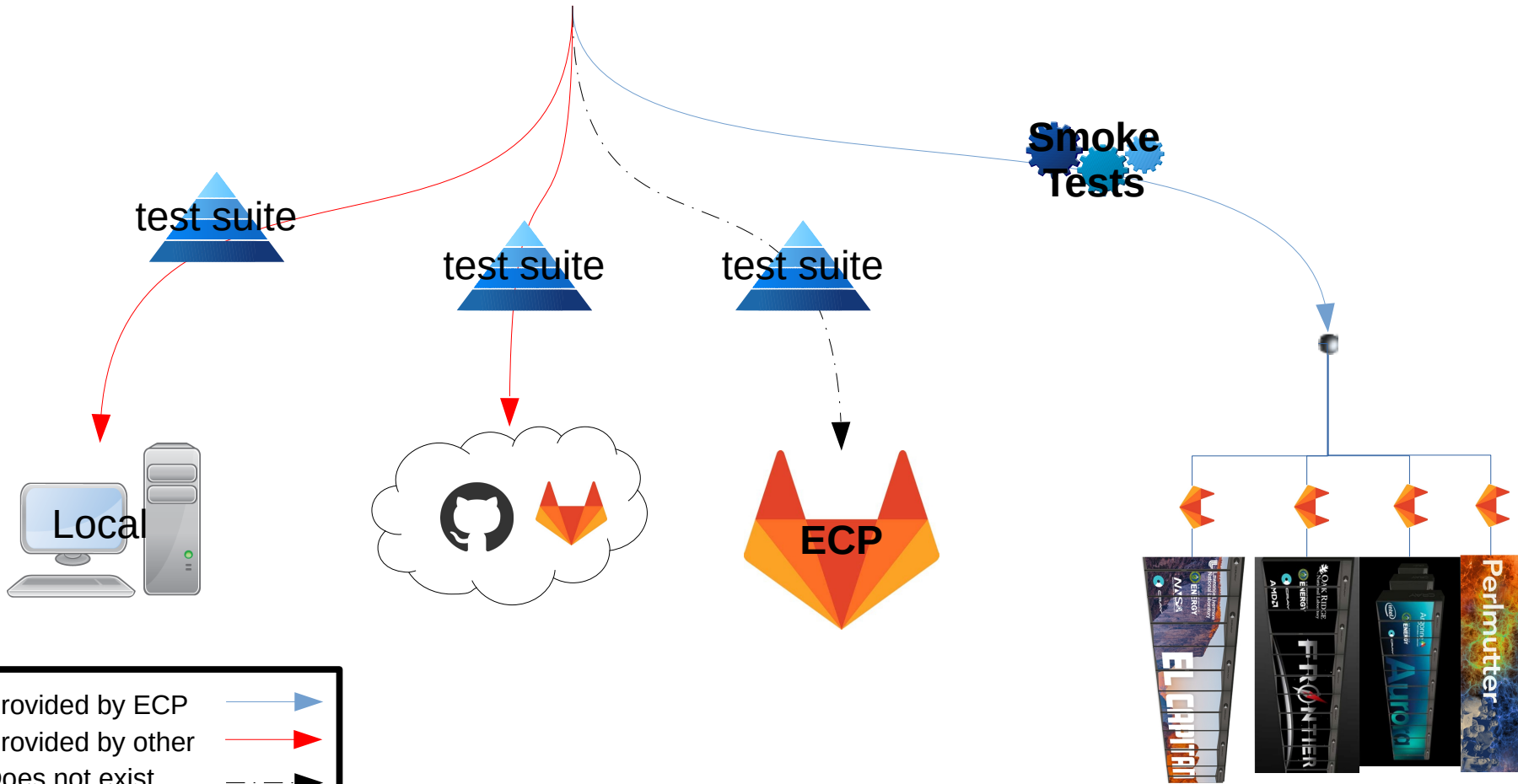
Scalability Testing

- Running tests on very large systems (e.g., thousands of nodes)
- Not currently part of ST testing efforts

Continuous Deployment

- Combines testing and publishing software
- Not discussed here, see *Benefiting from ECP CI* on April 14th

Continuous Integration



Provided by ECP

Provided by other

Does not exist

What is Tiered Testing?

1. Running different types of tests before creating a release of your software (aka your *test suite*)
2. Mapping tests to available hardware platforms
3. Running tests at different points in the deployment process

Software Deployment in ECP



Software Deployment in ECP



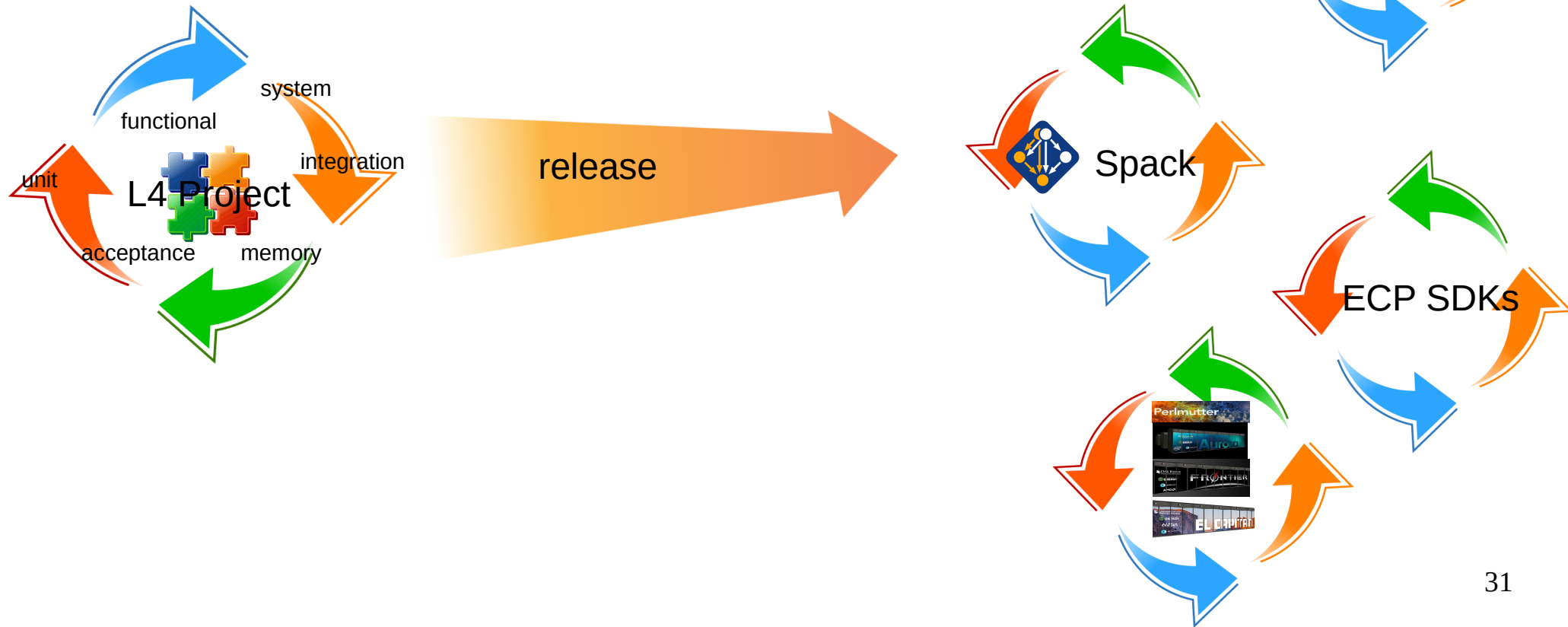
Software Deployment in ECP



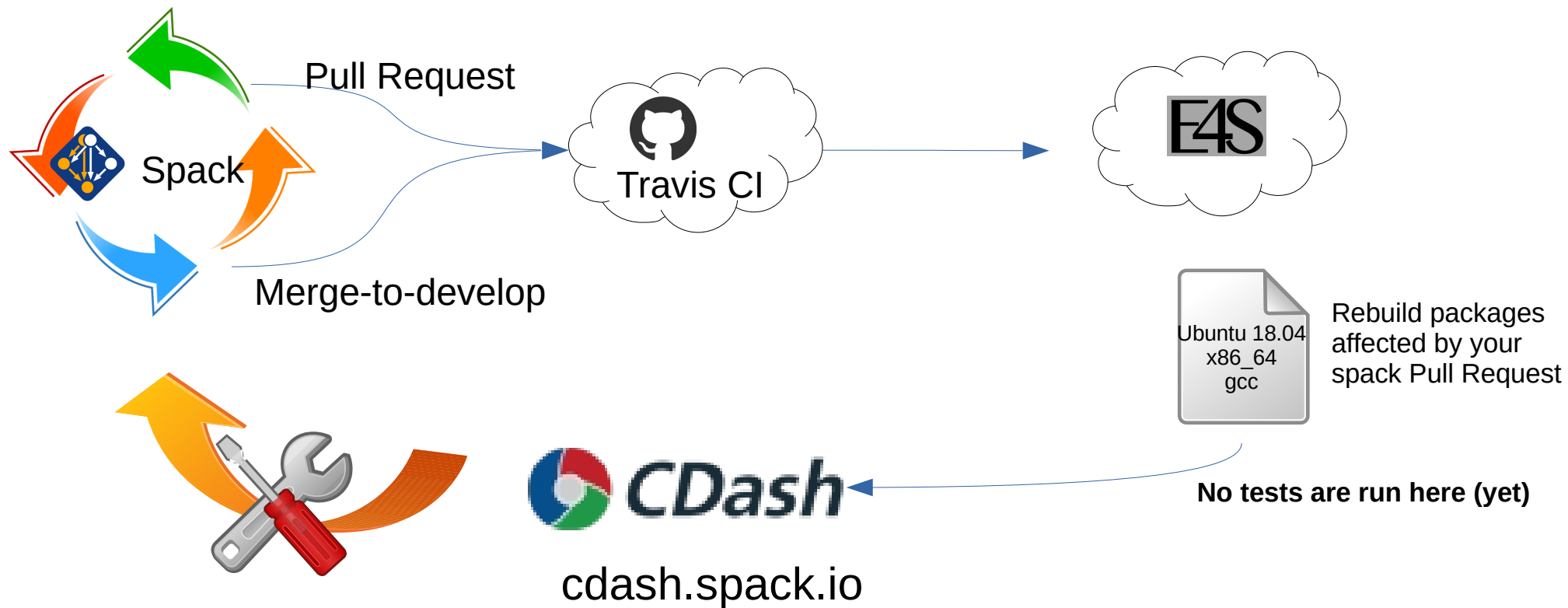
Software Deployment in ECP



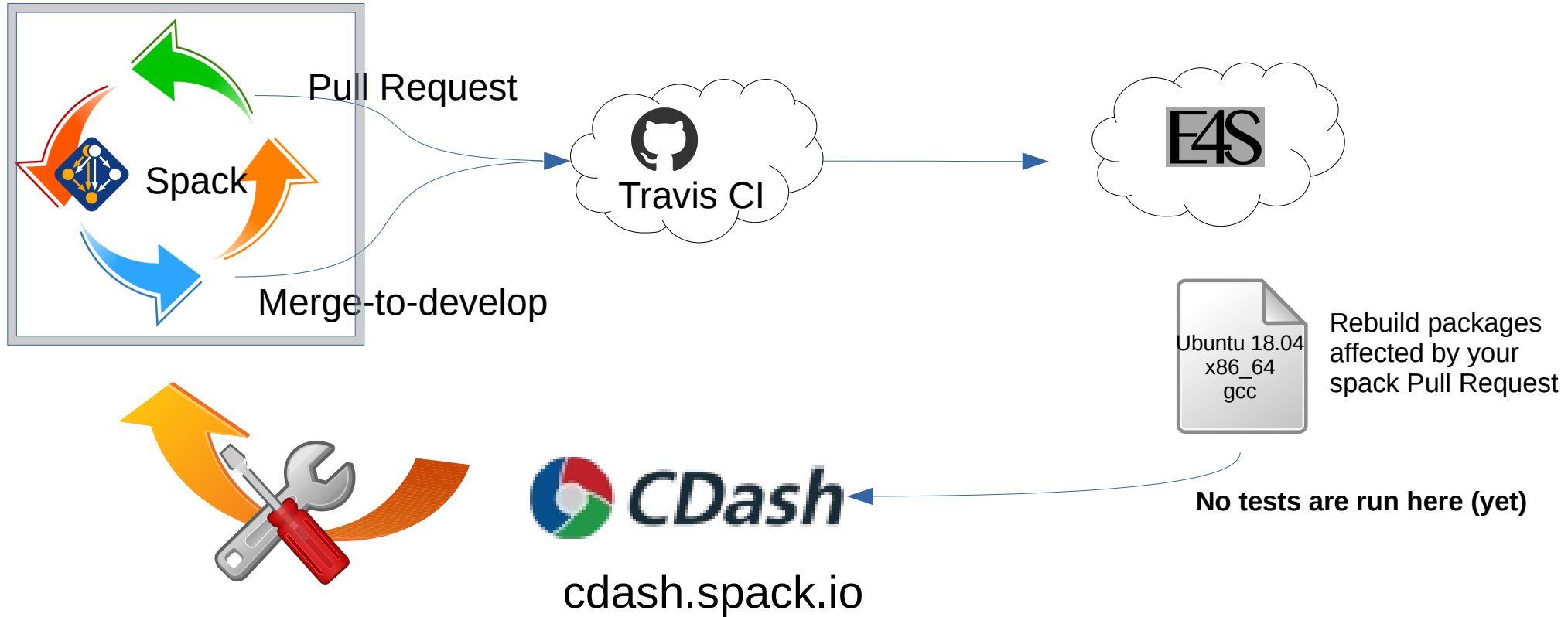
Software Deployment in ECP



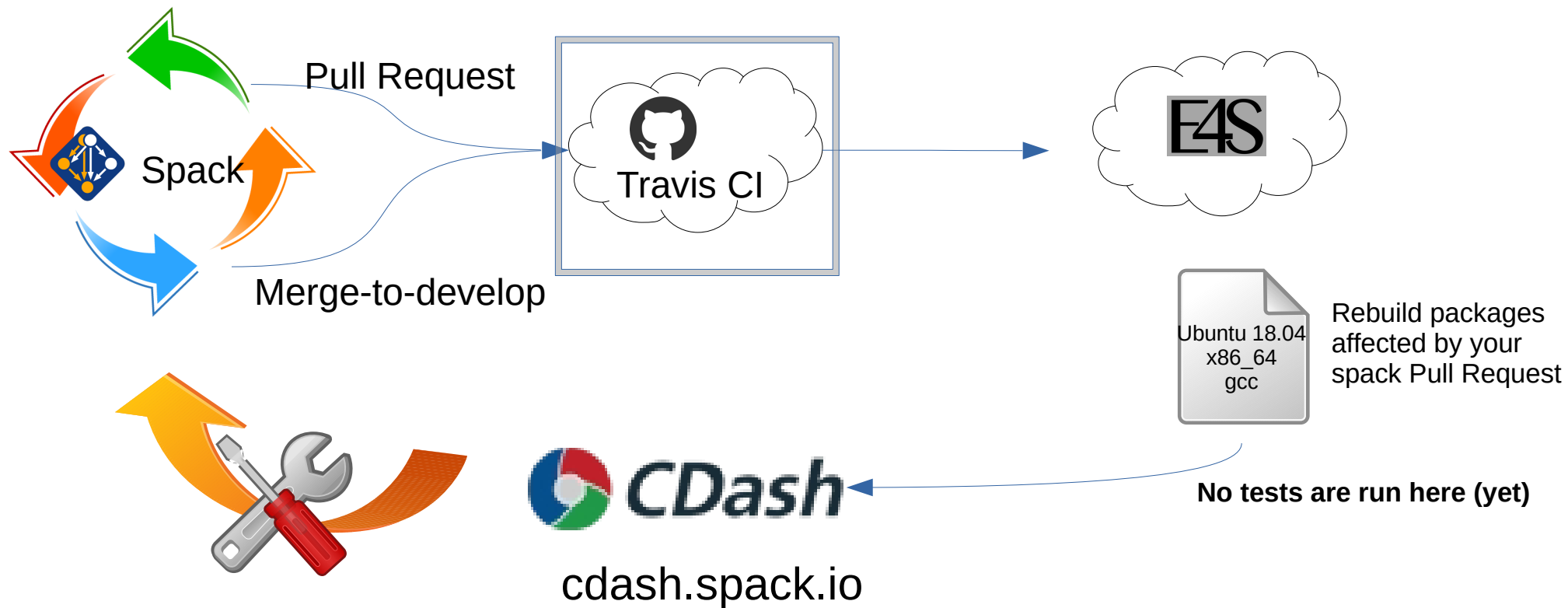
Simplified Spack CI workflow



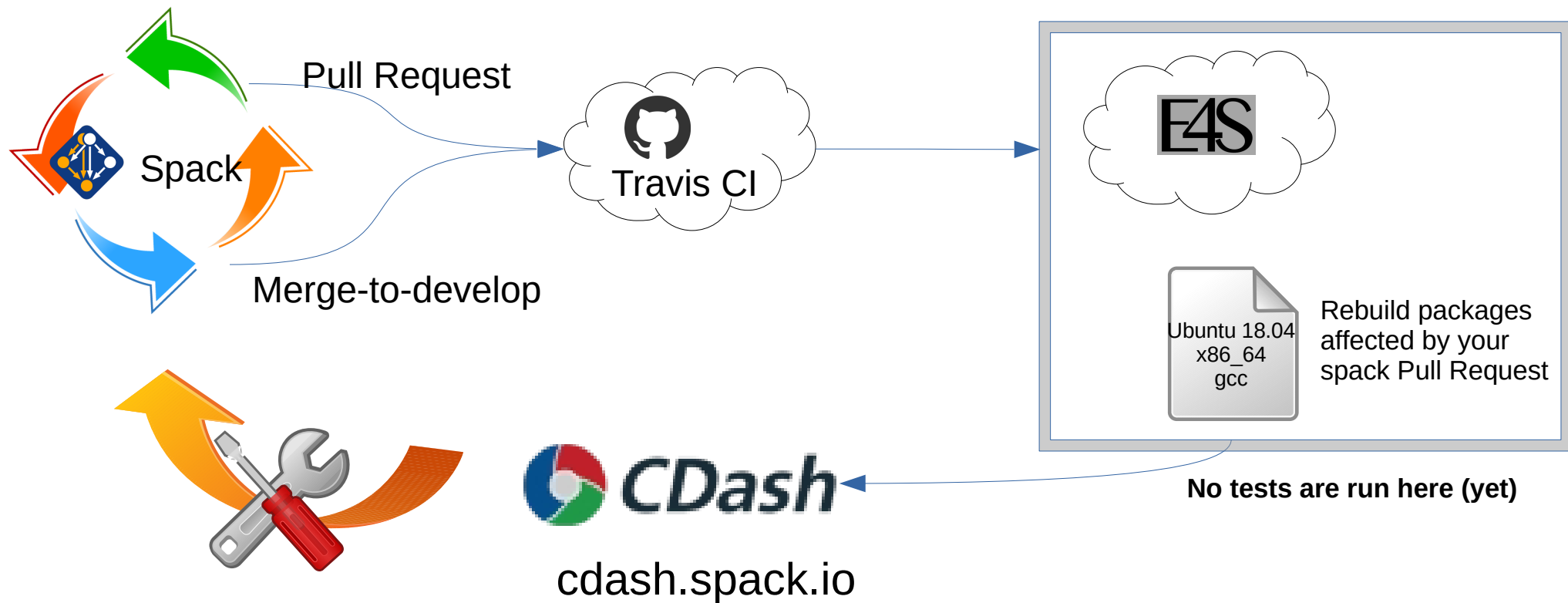
Simplified Spack CI workflow



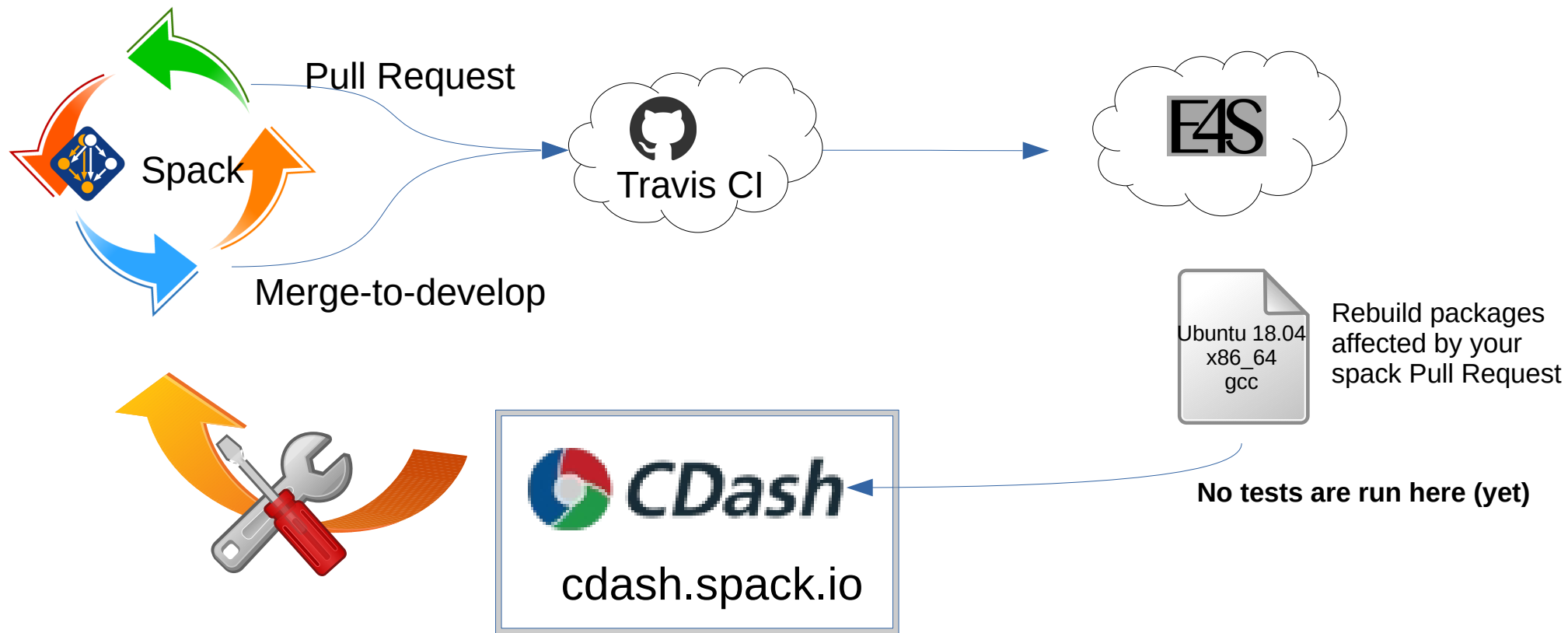
Simplified Spack CI workflow



Simplified Spack CI workflow



Simplified Spack CI workflow



cdash.spack.io



Spack

< PREV

CURRENT

Dashboard

Calendar

Project

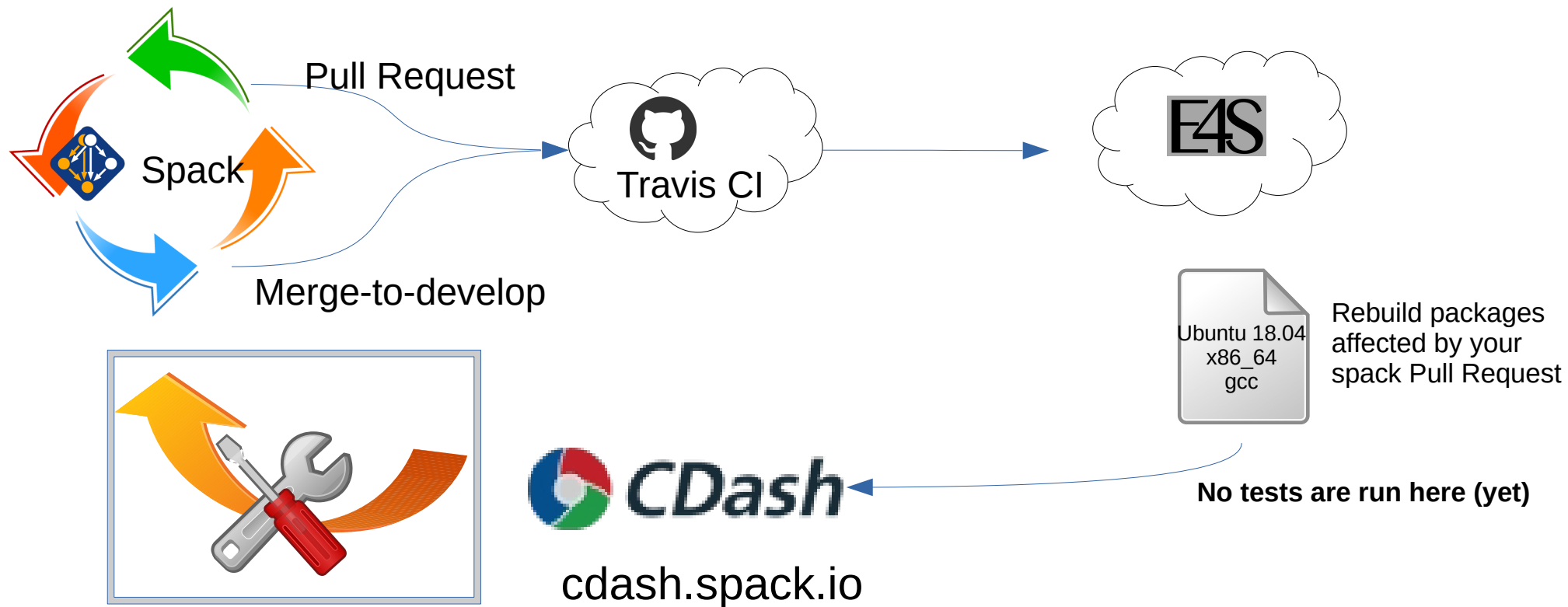


Latest DOE nightly E4S builds *28 builds*

[view timeline]

Site	Build Name	Update	Configure		Build		Start Time ▼
		Revision	Error ▼	Warn ▼	Error ▼	Warn ▼	
NERSC - Cori E4S 21.02	libcircle@0.3.0%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	1	0			Mar 26, 2021 - 13:15 UTC
NERSC - Cori E4S 21.02	perl@5.32.1%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	1	1	50 ⁺⁹ ₋₁₁	1	Mar 26, 2021 - 13:10 UTC
NERSC - Cori E4S 21.02	opari2@2.0.5%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	1	0	0	0	2 hours ago
NERSC - Cori E4S 21.02	raja@0.13.0%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	0	91 ⁺⁹⁰	16 ⁺¹⁶	28 ⁺²⁸ ₋₃	Mar 26, 2021 - 18:00 UTC
NERSC - Cori E4S 21.02	conduit@develop%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	0	0	8	0	Mar 13, 2021 - 03:28 UTC
NERSC - Cori E4S 21.02	libcap@2.25%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	0	0	4	0	2 hours ago
NERSC - Cori E4S 21.02	phist@1.9.3%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	0	4	2	0	Mar 26, 2021 - 15:38 UTC
NERSC - Cori E4S 21.02	scorep@6.0%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	0	17	0	25	1 hour ago
NERSC - Cori E4S 21.02	cubew@4.5%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	0	3	0	0	1 hour ago
NERSC - Cori E4S 21.02	netcdf-c@4.7.4%intel@19.1.2.254 arch=cray-cnl7-haswell (DOE nightly E4S builds)	b56d65	0	1	0	1	Mar 26, 2021 - 04:17 UTC

Simplified Spack CI workflow



E4S

E4S is a curated binary distribution of more than 80 ECP applications on multiple architectures, operating systems, and compilers

Visit e4s.io for details

ECP Annual Meeting

Tutorial: Mon, April 12 10AM

BoF: Tue, April 13 1pm

Breakout: Wed, April 14 4pm

Breakout: Thur, April 15 2:30pm

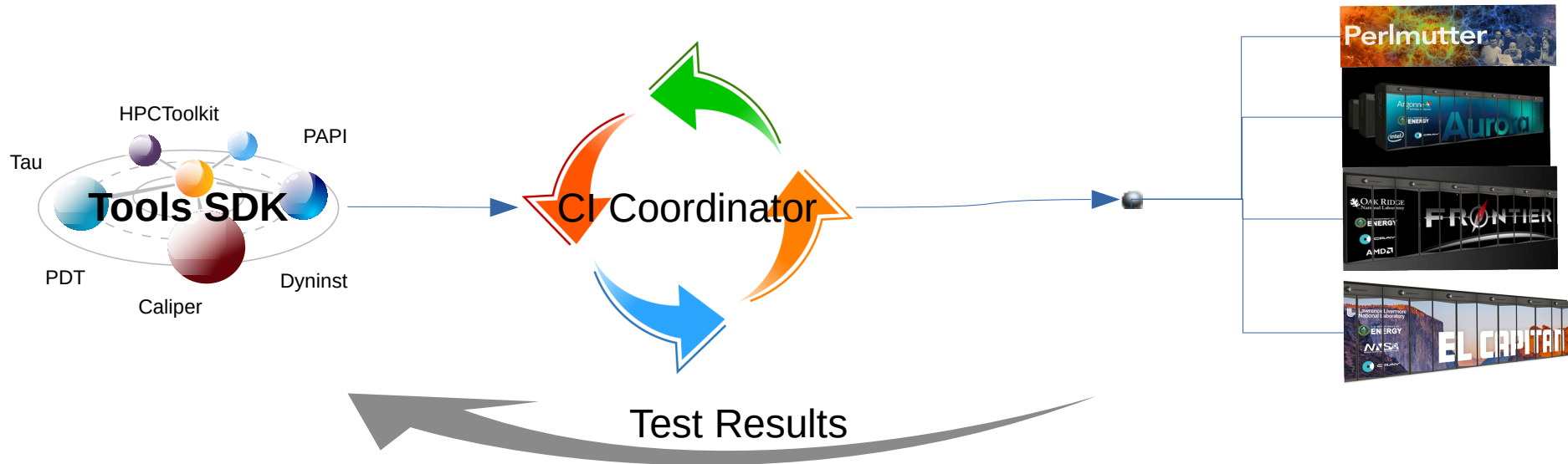


SDK Testing

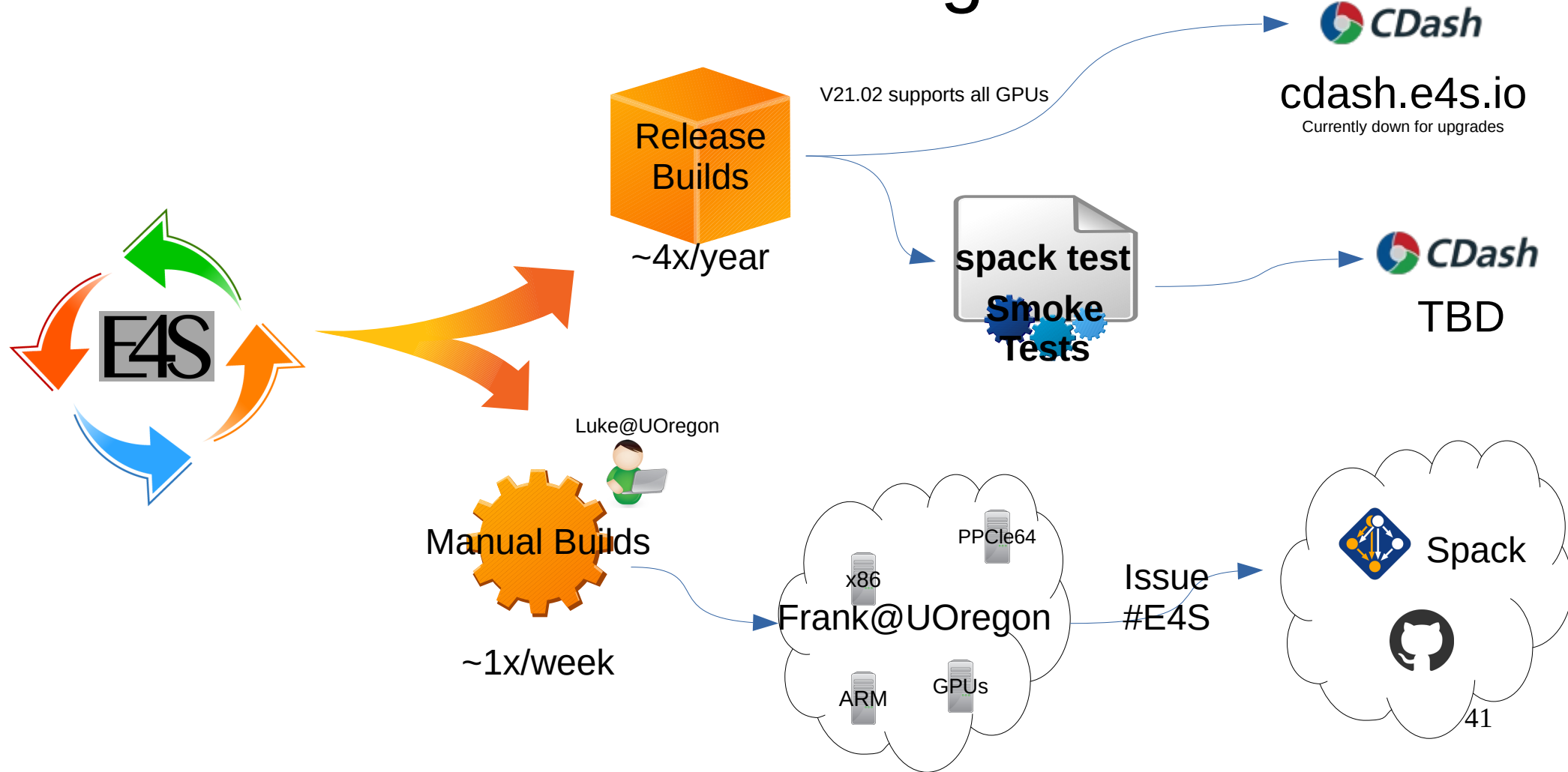
The ECP Software Development Kits (SDKs) are intended to be a cohesive set of closely-related applications that receive more intensive testing and validation.

There is currently no standard for testing in the SDKs.

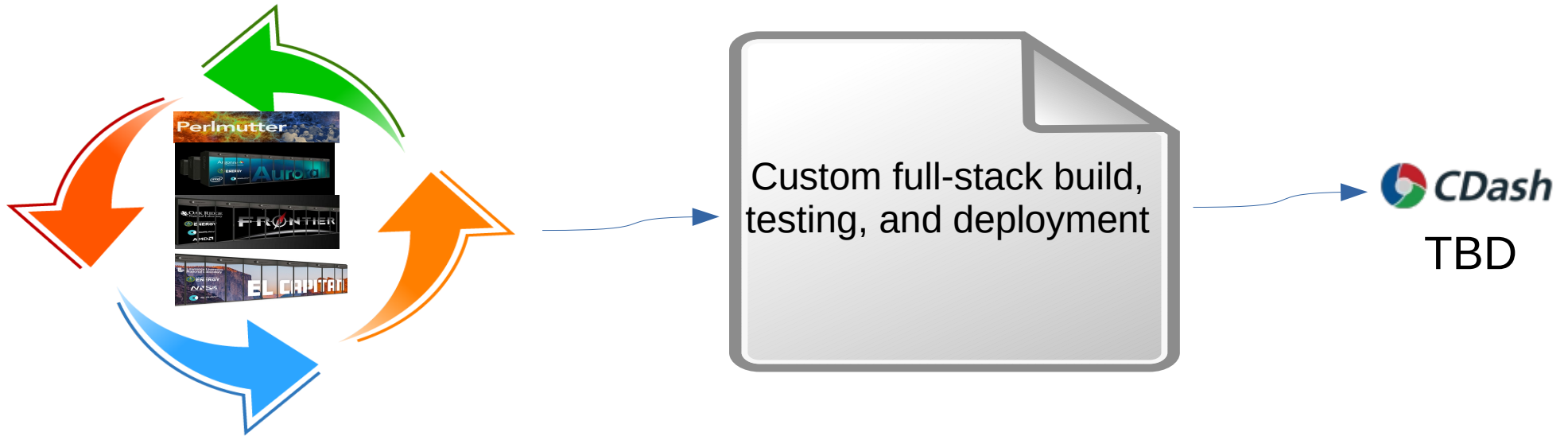
Example: Tools SDK



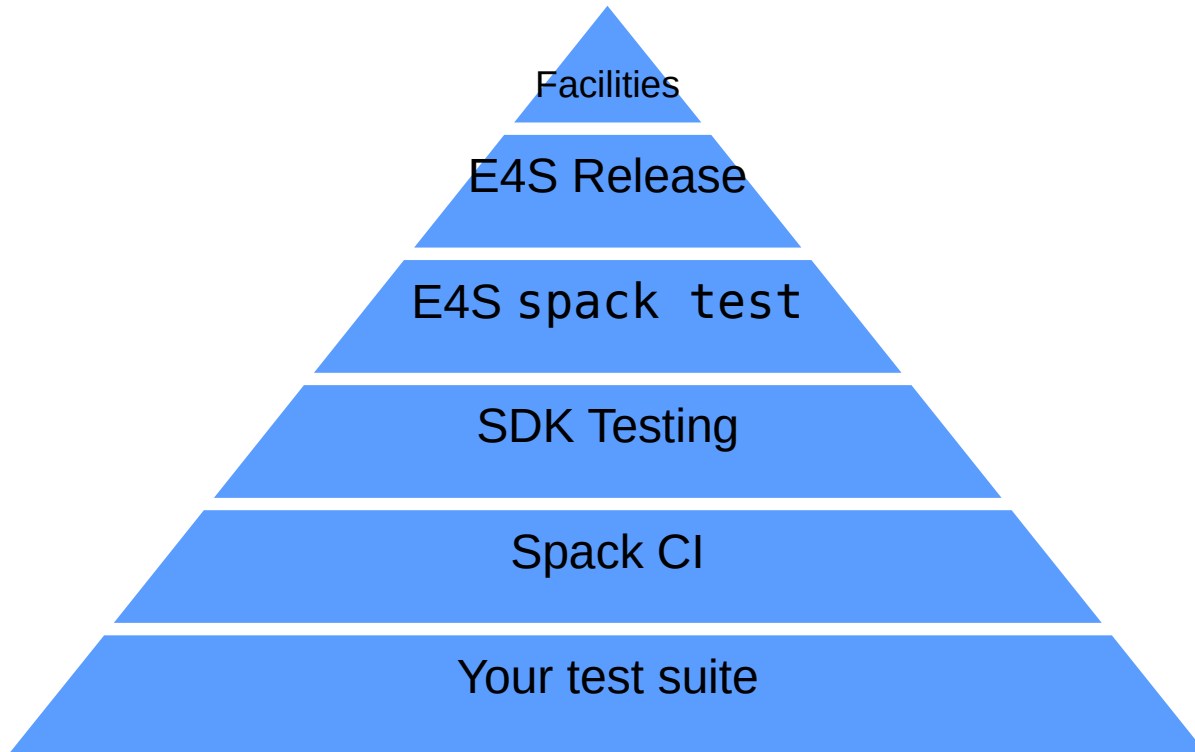
E4S Testing



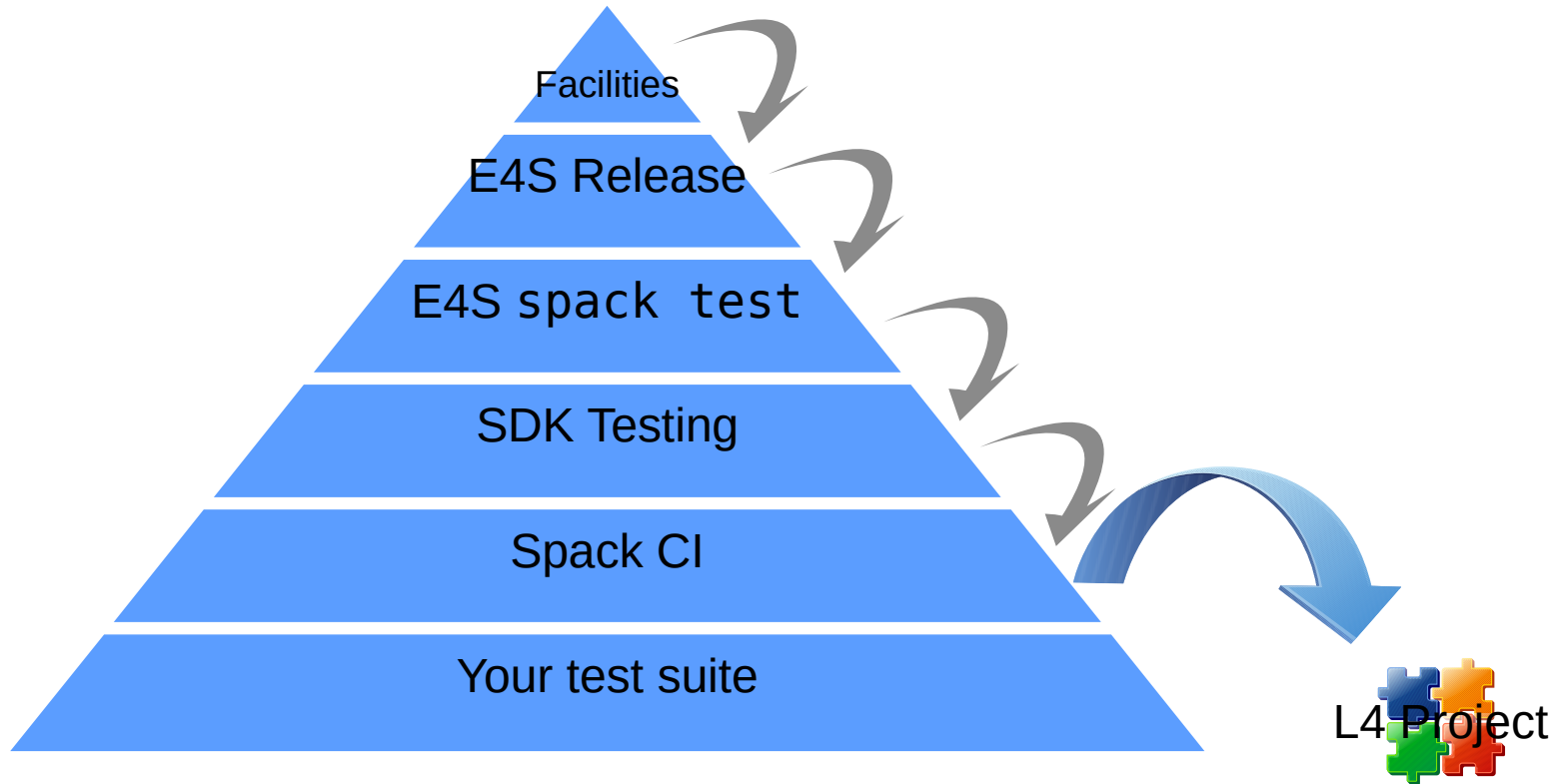
Facilities' Testing



What is Tiered Testing?



What is Tiered Testing?



Thank You

A massive thanks to everyone who helped bring this BoF to life!

- Jim Willenbring
- Ryan Adamson
- Sameer Shende
- The ECP SDK Pis
- The OACISS team at Uoregon

slides: github.com/hainest/TieredTesting

A special thanks to

- Luke Peyralans
- Wyatt Spear
- Tammy Dahlgren