Call for Code Deployment Graduation

developer.ibm.com/callforcode

Call for Code®

A multi-year program created by **David Clark Cause** with Founding Partner **IBM**, and in partnership with **United Nations Human Rights** and **The Linux Foundation**, Call for Code features an annual competition and is the largest tech-for-good challenge of its kind.

IBM
Call for Code
Founding Partner

DCC DAVID CLARK CAUSE
Call for Code
Creator

UNITED NATIONS HUMAN RIGHTS
OFFICE OF THE HIGH COMMISSIONER
Call for Code
Charitable Partner

THE LINUX FOUNDATION
Call for Code
Program Affiliate

Call for Code®

Call for Code inspires developers to create **practical, effective, and high-quality applications** that can have an immediate and lasting impact on humanitarian issues **as sustainable open source projects**.

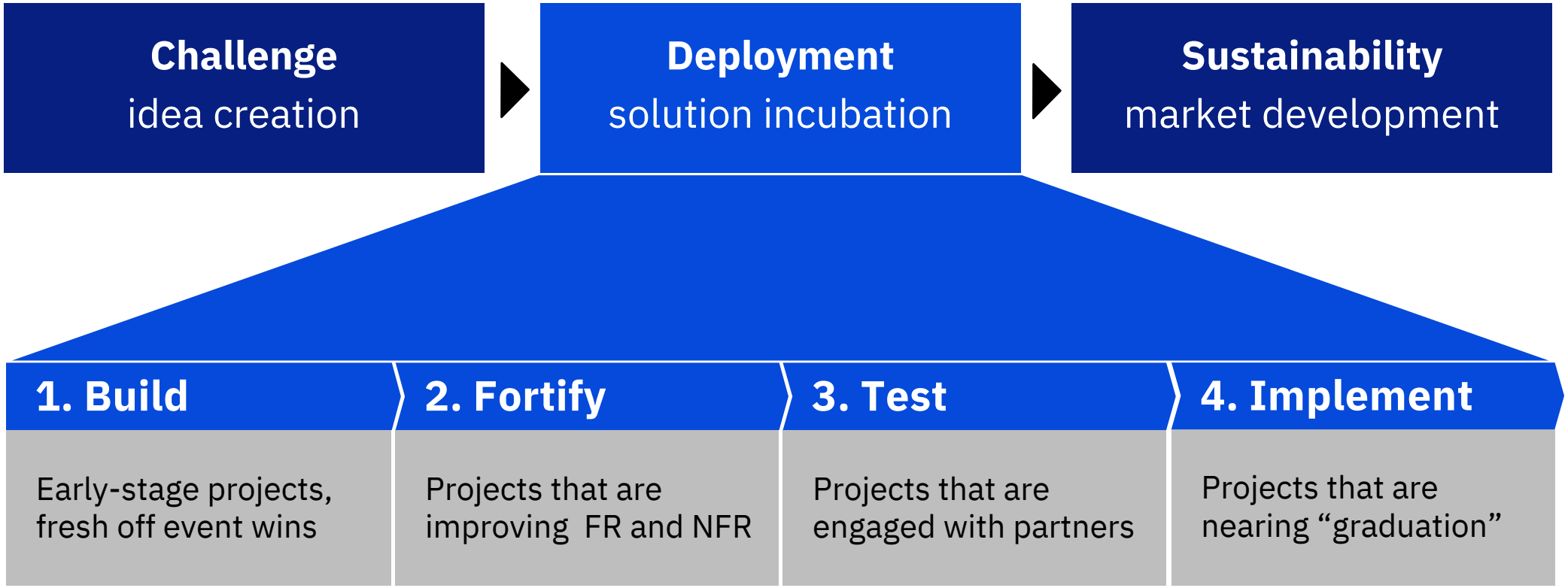
IBM
Call for Code
Founding Partner

DCC DAVID CLARK CAUSE
Call for Code
Creator

UNITED NATIONS HUMAN RIGHTS OFFICE OF THE HIGH COMMISSIONER
Call for Code
Charitable Partner

THE LINUX FOUNDATION
Call for Code
Program Affiliate

A Deployment Framework incubates Call for Code projects

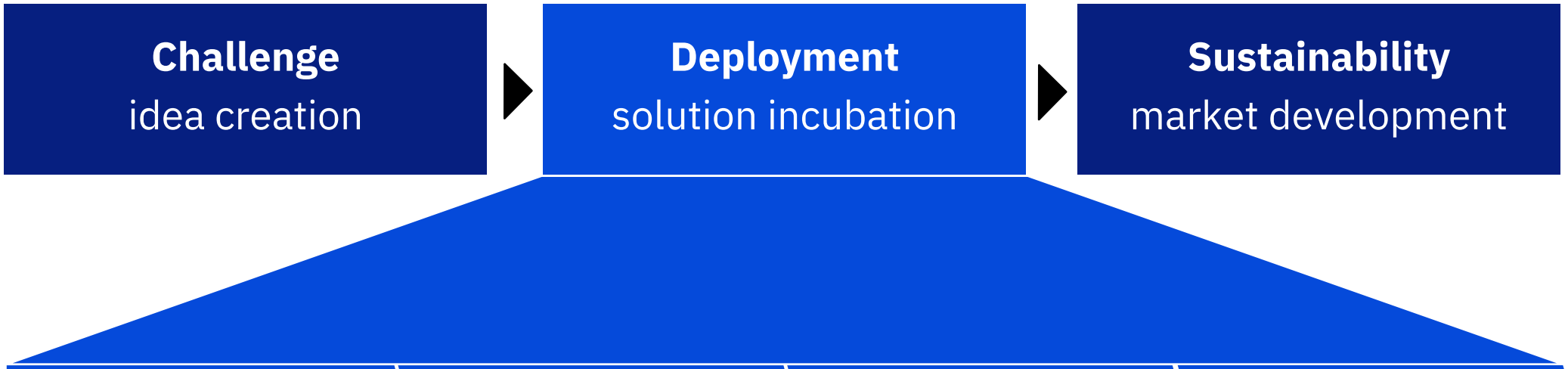


IBM
Call for Code Founding Partner

DCC DAVID CLARK CAUSE
Call for Code Creator

UNITED NATIONS HUMAN RIGHTS OFFICE OF THE HIGH COMMISSIONER
Call for Code Charitable Partner

THE LINUX FOUNDATION
Call for Code Program Affiliate



1. Build

Early-stage projects, fresh off event wins

- Establish competition frameworks that inspire developers and students to develop solutions to the world's greatest challenges
- Provide open source code, community, and culture to ensure that participants have success with our products and services

- **Example exit criteria:**

- Idea has won a competition such as Call for Code or a CGIU event and scores fairly well against the 4 judging criteria.
- The solution may vary in its degree of completeness but should be feasible in the short term.

2. Fortify

Projects that are improving FR and NFR

- Provide hardware, software, branding, user experience expertise to ensure that technology solutions are robust and powerful
- Connect developers with end users, such as public and private beneficiaries and subject matter experts to ensure their innovation matters

- **Example exit criteria:**

- Key functional and non-functional requirements are met.
- A test plan for how it can be deployed is created.
- Funding is officially provided in order to carry out late-stage tests.
- It's common to iterate between the Fortify and Test phases as the solution is hardened and ready to leave the Test phase.

3. Test

Projects that are engaged with partners

- Validate the solutions in areas of need to ensure that the technology solution meets the requirements of a communities of greatest need
- Provide input to guide the improvement of the solutions based on our cloud, data, AI, blockchain and additional expertise

- **Example exit criteria:**

- Solution has been validated by a CFC partner or independent third-party to be effective.
- Or, there can be a business agreement signed with a beneficiary organization.

4. Implement

Projects that are nearing "graduation"

- Drive solutions to their first commercial success through connection with mentorship, funding, and sustainable sources of revenue
- Establish meaningful milestones, ensure that intellectual property is created and protected, and that IBM, its employees, and customers benefit

- **Example exit criteria:**

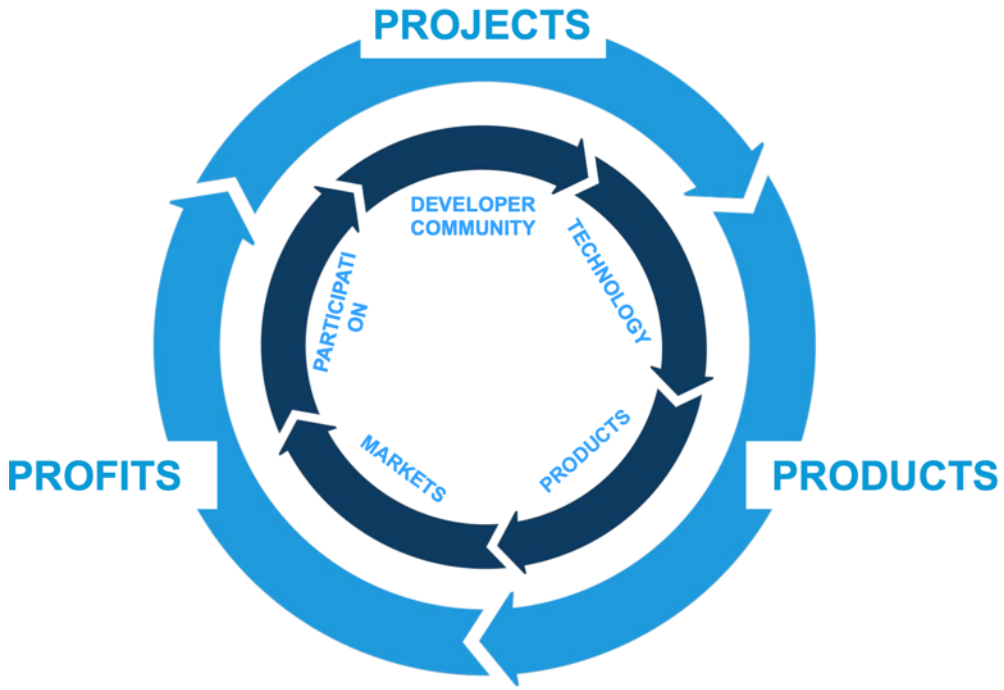
- The solution has been put to work in a real situation for the beneficiary organization.
- A second validation executed or customer is acquired.
- Joint patent applications are filed.
- IBM funding is no longer required. Either a business is self-sufficient or other investments are sustaining the project.

Why host your project at the Linux Foundation

The LF provides a neutral home for the open source code that can power many business models on top

The project legal structure provides a way for many participants to improve the underlying technology

The LF has many avenues to access developers, funding, mentorship, events, and marketing



Successful projects depend on an ecosystem to develop solutions that the market will adopt. Adoption drives sustainability.

IBM
Call for Code
Founding Partner

DCC DAVID CLARK CAUSE
Call for Code
Creator

UNITED NATIONS HUMAN RIGHTS OFFICE OF THE HIGH COMMISSIONER
Call for Code
Charitable Partner

THE LINUX FOUNDATION
Call for Code
Program Affiliate

The basic steps of launching with the Linux Foundation

Once we have worked out the intended **mission and scope** of the sponsored project, we work to build out a formation group to draft the **governance documents** and finalize the **project structure** as a community.

Towards the end of the formation process we often convene a marketing team, and the LF will prepare the launch **press release** with input and feedback from the marketing team.

In addition to increased efficiency, working in formation teams allows for the **community** to begin collaborating before the project even formally launches.

Preparing to open source a project with the Linux Foundation

Ecosystem and documents

- ❑ Prepare mission and scope statement for the LF Technical Charter
- ❑ Analyze code for license consistency with Apache 2 or other as needed
- ❑ Assess ecosystem of launch partners who would be part of the announcement (Ideally at least 1)
- ❑ Choose new name or hand over existing names and trademarks
- ❑ Identify marketing, content, communications or other needs for the project (including from IBM)
- ❑ Prep content for developers to take part in the project on day one
- ❑ Choose Technical Steering Committee members
- ❑ Prepare Slack channel or workspace for contributors
- ❑ Evaluate LFX options for marketing, community, funding

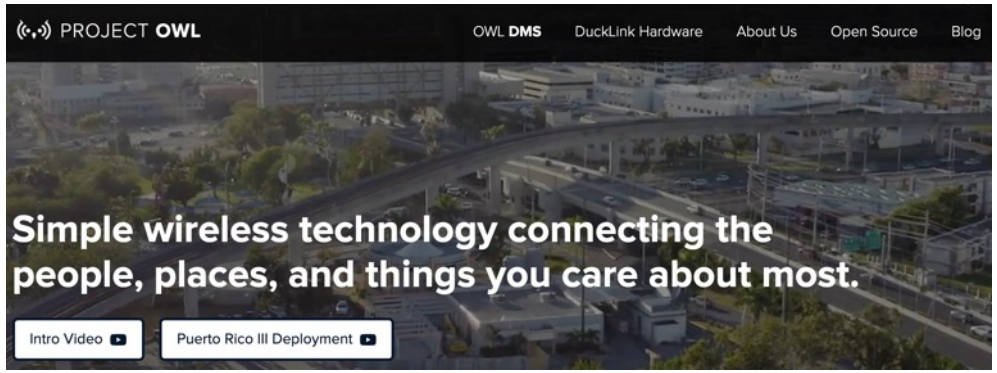
Preparing to open source a project with the Linux Foundation

GitHub and source code updates

- ❑ Add the Apache 2 LICENSE to your GitHub repository and include the specific copyright header to your source code files using the relevant comment syntax for the programming language. Add the license file as is, then use 2018 and list the names of your team members in your individual source files. http://www.apache.org/licenses/LICENSE-2.0#apply
- ❑ Look at improving your documentation so that first time visitors clearly see the problem you are solving and how they can understand the solution in your README.md. Also consider enabling alternative language translations.
- ❑ Look at carefully documenting installation/duplication steps so that new developers can explore the technology on their own systems in order to contribute back fixes. This will also help you identify any potential dependency issues you might have on clean installations.
- ❑ Add a basic CONTRIBUTING.md that describes how others can start to collaborating with you. The Linux Foundation has a few sample files you can use for inspiration. The way you choose to accept fixes is up to you though. Here's a good sample: https://github.com/nodejs/node/blob/master/CONTRIBUTING.md
- ❑ Strongly consider adding the Developer Certificate of Origin bot to your repositories. This provides assurance that anyone who contributes to your repository has the rights to the code they want to provide: https://github.com/apps/dco
- ❑ Consider configuring GitHub pages to create a basic description and FAQ site, like https://gql.foundation/

Project OWL

- ✓ Won CFC in October 2018
- ✓ Service Corps, incubation, and field tests in 2019
- ✓ Created **ClusterDuck Protocol** at LF in March 2020



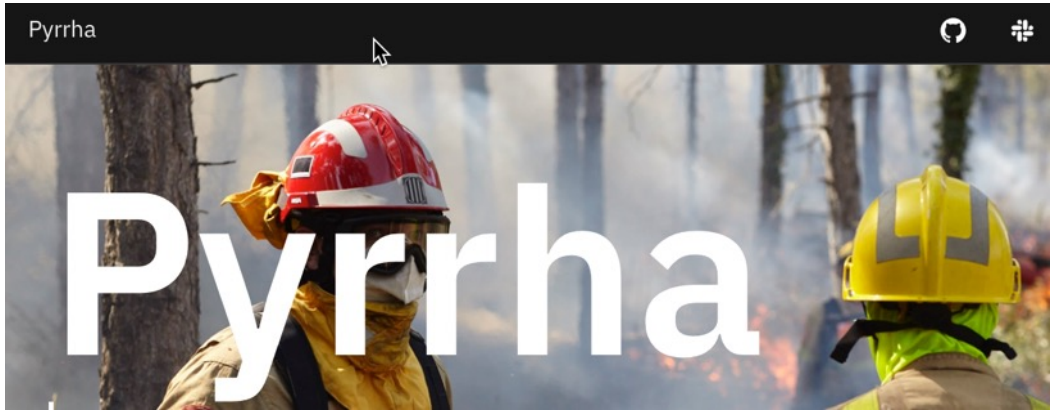
project-owl.com

clusterduckprotocol.org

Prometeo

- ✓ Won CFC in October 2019
- ✓ Service Corps, incubation, field tests in 2020
- ✓ Created **Pyrrha** at LF in July 2021

prometeoplatform.com



pyrrha-platform.org