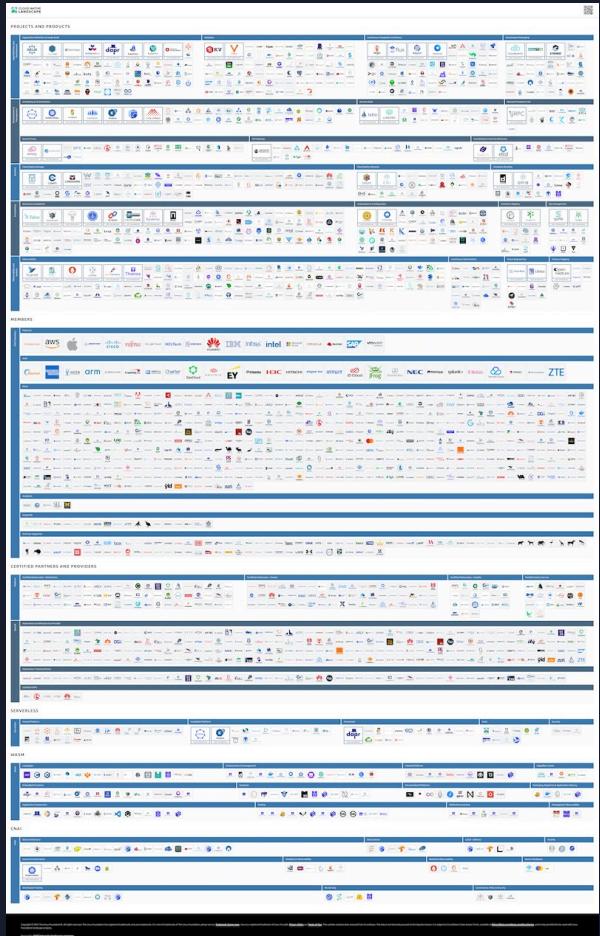


Streamlining DevEx

The Power of CI/CD Standardization and Interoperability



Jeremy Meiss

DevEx / DevRel Consultant
DevOpsDays Kansas City Organizer



A working definition of DevEx

"...the journey of developers and practitioners as they learn and deploy technology, which if successful, focuses on eliminating obstacles that hinder them from achieving success in their endeavors."

-Jessica West, *Co-Founder, DevEx Institute*



DevEx

A Good Developer Experience







Cornell University

arXiv > cs > arXiv:1312.1452

Computer Science > Software Engineering

[Submitted on 5 Dec 2013]

Developer Experience: Concept and Definition

Fabian Fagerholm, Jürgen Münch

New ways of working such as globally distributed development or the integration of self-motivated external developers into software ecosystems will require a better and more comprehensive understanding of developers' feelings, perceptions, motivations and identification with their tasks in their respective project environments. User experience is a concept that captures how persons feel about products, systems and services. It evolved from disciplines such as interaction design and usability to a much richer scope that includes feelings, motivations, and satisfaction. Similarly, developer experience could be defined as a means for capturing how developers think and feel about their activities within their working environments, with the assumption that an improvement of the developer experience has positive impacts on characteristics such as sustained team and project performance. This article motivates the importance of developer experience, sketches related approaches from other domains, proposes a definition of developer experience that is derived from similar concepts in other domains, describes an ongoing empirical study to better understand developer experience, and finally gives an outlook on planned future research activities.

Comments: 5 pages. The final publication is available at [this http URL](#)

Subjects: Software Engineering (cs.SE)

Cite as: arXiv:1312.1452 [cs.SE]

(or [arXiv:1312.1452v1 \[cs.SE\]](#) for this version)

<https://doi.org/10.48550/arXiv.1312.1452>

Journal reference: Proceedings of the International Conference on Software and System Process (ICSSP 2012), pages 73–77,

DevEx isn't new

REF: F. Fagerholm and J. Münch, "Developer experience: Concept and definition," 2012 International Conference on Software and System Process (ICSSP), Zurich, Switzerland, 2012.



[Submitted on 5 Dec 2013]

Developer Experience: Concept and Definition

Fabian Fagerholm, Jürgen Münch

New ways of working such as globally distributed development or the integration of self-motivated external developers into software ecosystems will require a better and more comprehensive understanding of developers' feelings, perceptions, motivations and identification with their tasks in their respective project environments. User experience is a concept that captures how persons feel about products, systems and services. It evolved from disciplines such as interaction design and usability to a much richer scope that includes feelings, motivations, and satisfaction. Similarly, developer experience could be defined as a means for capturing how developers think and feel about their activities within their working environments, with the assumption that an improvement of the developer experience has positive impacts on characteristics such as sustained team and project performance. This article motivates the importance of developer experience, sketches related approaches from other domains, proposes a definition of developer experience that is derived from similar concepts in other domains, describes an ongoing empirical study to better understand developer experience, and finally gives an outlook on planned future research activities.

Comments: 5 pages. The final publication is available at [this http URL](#)

Subjects: Software Engineering (cs.SE)

Cite as: arXiv:1312.1452 [cs.SE]

(or arXiv:1312.1452v1 [cs.SE] for this version)

<https://doi.org/10.48550/arXiv.1312.1452>

Journal reference: Proceedings of the International Conference on Software and System Process (ICSSP 2012), pages 73–77,

DevEx isn't new

"New ways of working such as globally distributed development or the integration of self-motivated external developers into software ecosystems will require a better and more comprehensive understanding of developers' feelings, perceptions, motivations and identification with their tasks in their respective project environments.



[Submitted on 5 Dec 2013]

Developer Experience: Concept and Definition

Fabian Fagerholm, Jürgen Münch

New ways of working such as globally distributed development or the integration of self-motivated external developers into software ecosystems will require a better and more comprehensive understanding of developers' feelings, perceptions, motivations and identification with their tasks in their respective project environments. User experience is a concept that captures how persons feel about products, systems and services. It evolved from disciplines such as interaction design and usability to a much richer scope that includes feelings, motivations, and satisfaction. Similarly, developer experience could be defined as a means for capturing how developers think and feel about their activities within their working environments, with the assumption that an improvement of the developer experience has positive impacts on characteristics such as sustained team and project performance. This article motivates the importance of developer experience, sketches related approaches from other domains, proposes a definition of developer experience that is derived from similar concepts in other domains, describes an ongoing empirical study to better understand developer experience, and finally gives an outlook on planned future research activities.

Comments: 5 pages. The final publication is available at [this http URL](#)

Subjects: Software Engineering (cs.SE)

Cite as: arXiv:1312.1452 [cs.SE]

(or [arXiv:1312.1452v1 \[cs.SE\]](#) for this version)

<https://doi.org/10.48550/arXiv.1312.1452>

Journal reference: Proceedings of the International Conference on Software and System Process (ICSSP 2012), pages 73–77,

DevEx isn't new

"...developer experience could be defined as a means for capturing how developers think and feel about their activities within their working environments, with the assumption that an improvement of the developer experience has positive impacts on characteristics such as sustained team and project performance."

Point of clarification

- "DevEx" by default focuses on "developer"
- View "DevEx" as a whole of the lifecycle





CI/CD

- Standardization
- Interoperability

CI/CD Standardization

- consistency to development pipelines
- reduces friction
- enhances collaboration

Implementing CI/CD Standardization

Assessment and Analysis

- Thoroughly assess your current CI/CD pipelines
- Identify pain points and bottlenecks
- Analyze specific requirements and constraints

Implementing CI/CD Standardization

Define Standardization Goals

- Define goals and objectives, align with strategy and objectives
- Determine success, like reduced deployment times / error rates

Implementing CI/CD Standardization

Select Tools and Practices

- Choose tools & practices aligned with organization needs, goals
- Establish standard templates and configurations for pipelines
- Enforce coding standards for consistency and readability

Implementing CI/CD Standardization

Utilize Version Control

- Store pipeline configs as code in version control systems
- Implement branching and pull request strategies

Implementing CI/CD Standardization

Automated Testing and Validation

- Integrate automated testing and validation into templates
- Implement code reviews and peer validation early in dev process

Implementing CI/CD Standardization

Documentation and Training

- Create comprehensive docs for processes, configs, best practices
- Provide training to ensure understanding and effective use



Optimizing CI/CD Standardization



Optimizing CI/CD Standardization

Continuous Monitoring & Improvement

- Detect pipeline issues and bottlenecks in real-time
- Establish culture of regular reviews and updating pipelines

Optimizing CI/CD Standardization

Governance & Compliance

- Implement governance policies to enforce pipeline standards
- Validate compliance with industry regulations / internal standards
- Regularly audit and assess adherence to standardized practices



Optimizing CI/CD Standardization

Scaling & Adaptation

- Ensure standardized templates can scale and adapt
- Maintain flexibility to accommodate unique project requirements



Optimizing CI/CD Standardization

Feedback Loop & Collaboration

- Foster collaborative environments where feedback & contributions encouraged
- Continuously communicate benefits of standardized pipelines & celebrate successes



CI/CD Interoperability

- seamless integration across diverse toolsets
- fosters flexibility in development environments



Interoperability in CI/CD Systems

Streamlined workflows

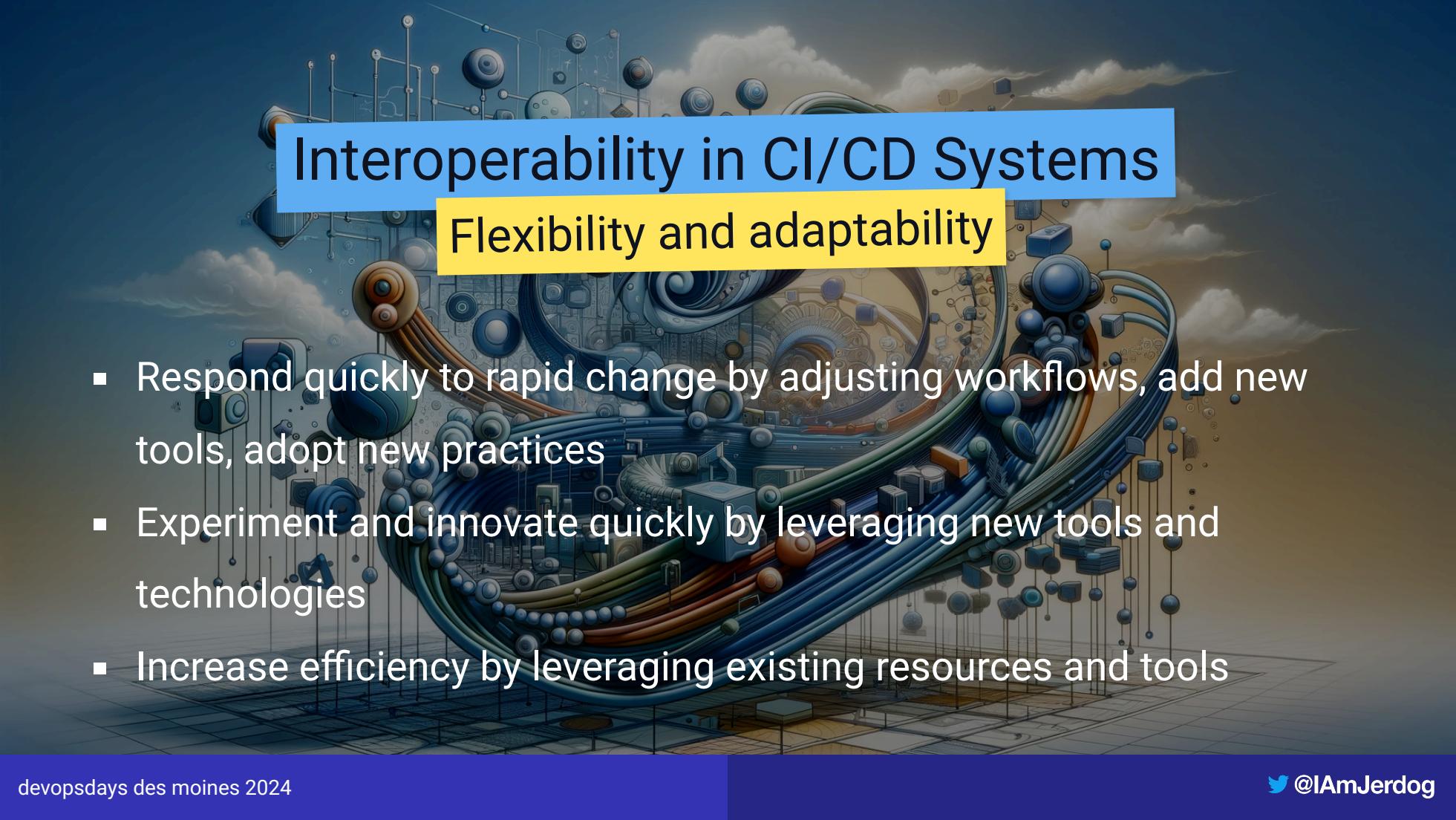
- Reduce manual intervention, increase automation with templates + reusable config
- Eliminate waste and improve efficiency
- Deliver faster with higher quality + better developer experience



Interoperability in CI/CD Systems

Cross-functional collaboration

- Have shared goals and break down silos
- Improve resource utilization and efficiency
- Leverage strengths and expertise of each team



Interoperability in CI/CD Systems

Flexibility and adaptability

- Respond quickly to rapid change by adjusting workflows, add new tools, adopt new practices
- Experiment and innovate quickly by leveraging new tools and technologies
- Increase efficiency by leveraging existing resources and tools

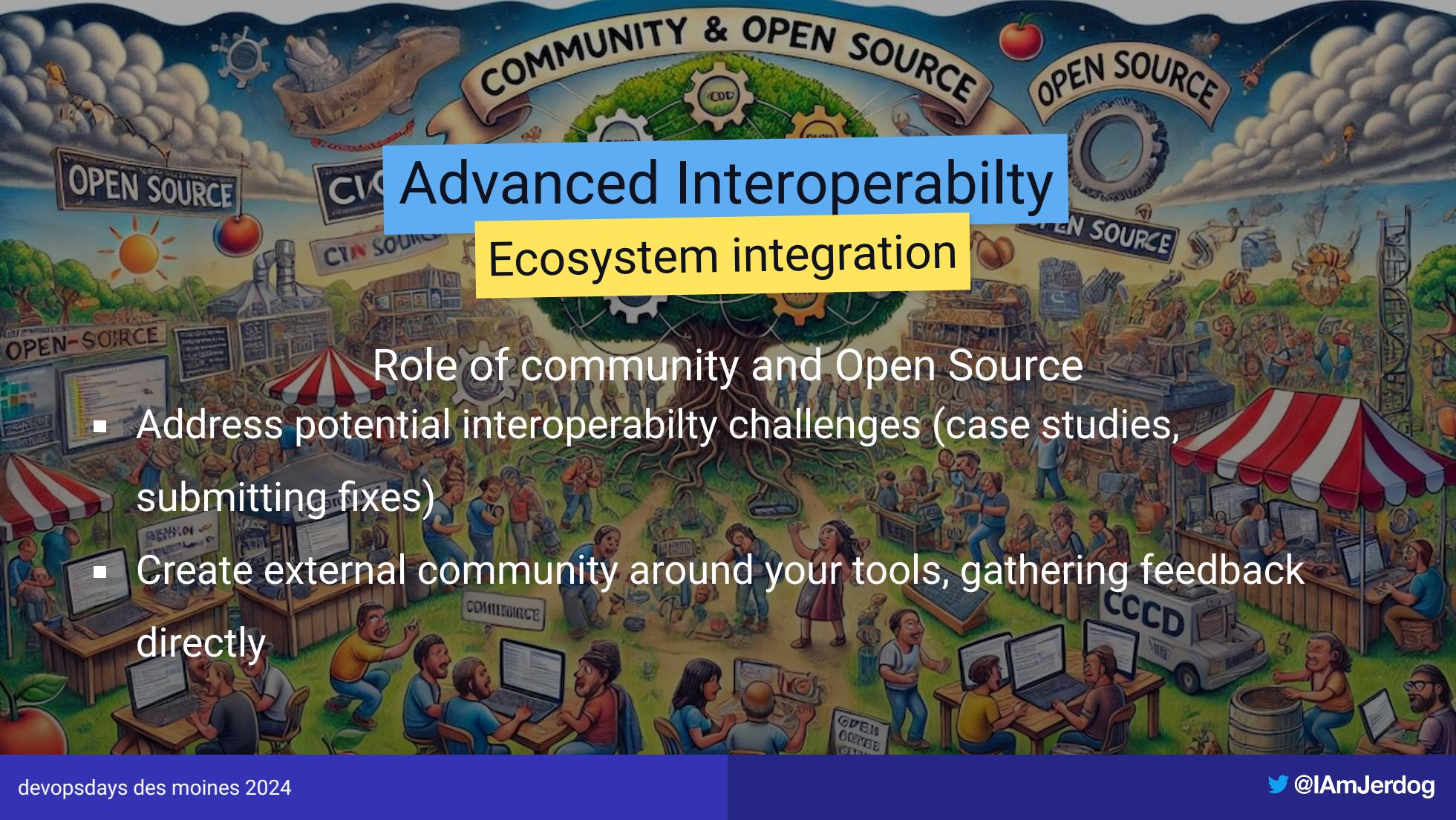




Advanced Interoperability

Ecosystem integration

- Assemble customized toolchain tailored to your requirements
- Minimize manual intervention via End-to-end automation
- Greater visibility + traceability via aggregation of info

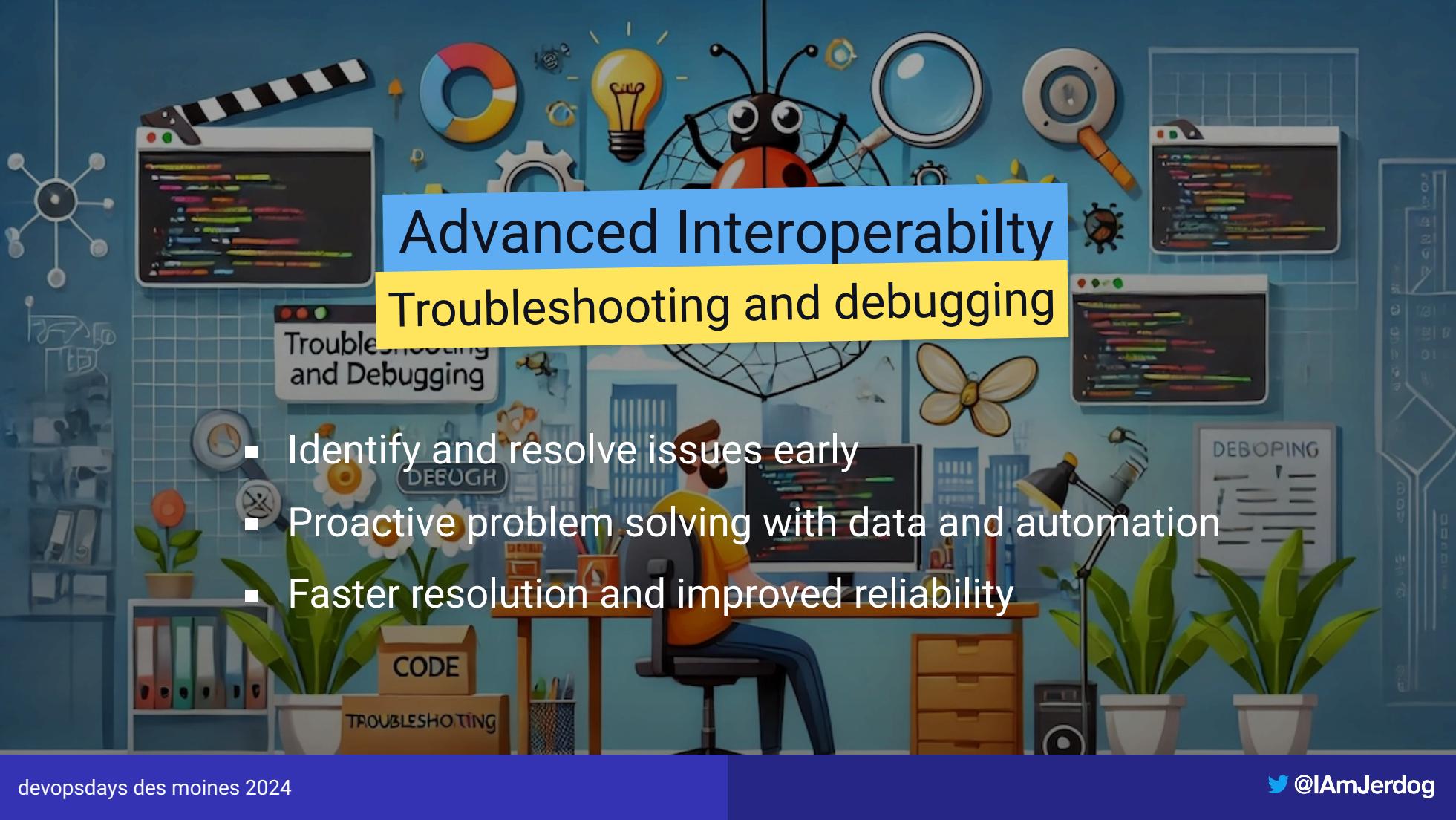


Advanced Interoperability

Ecosystem integration

Role of community and Open Source

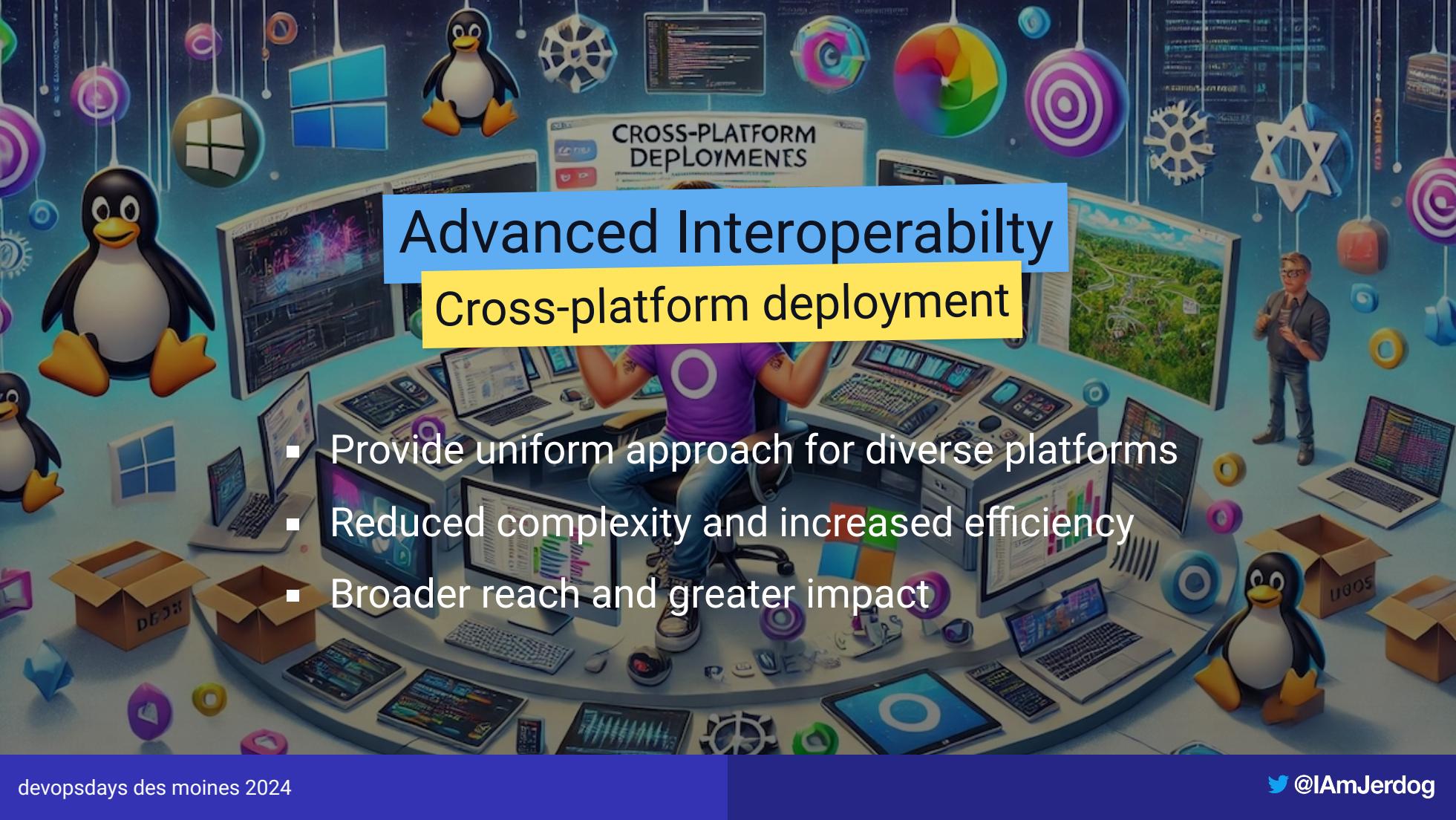
- Address potential interoperability challenges (case studies, submitting fixes)
- Create external community around your tools, gathering feedback directly



Advanced Interoperability

Troubleshooting and debugging

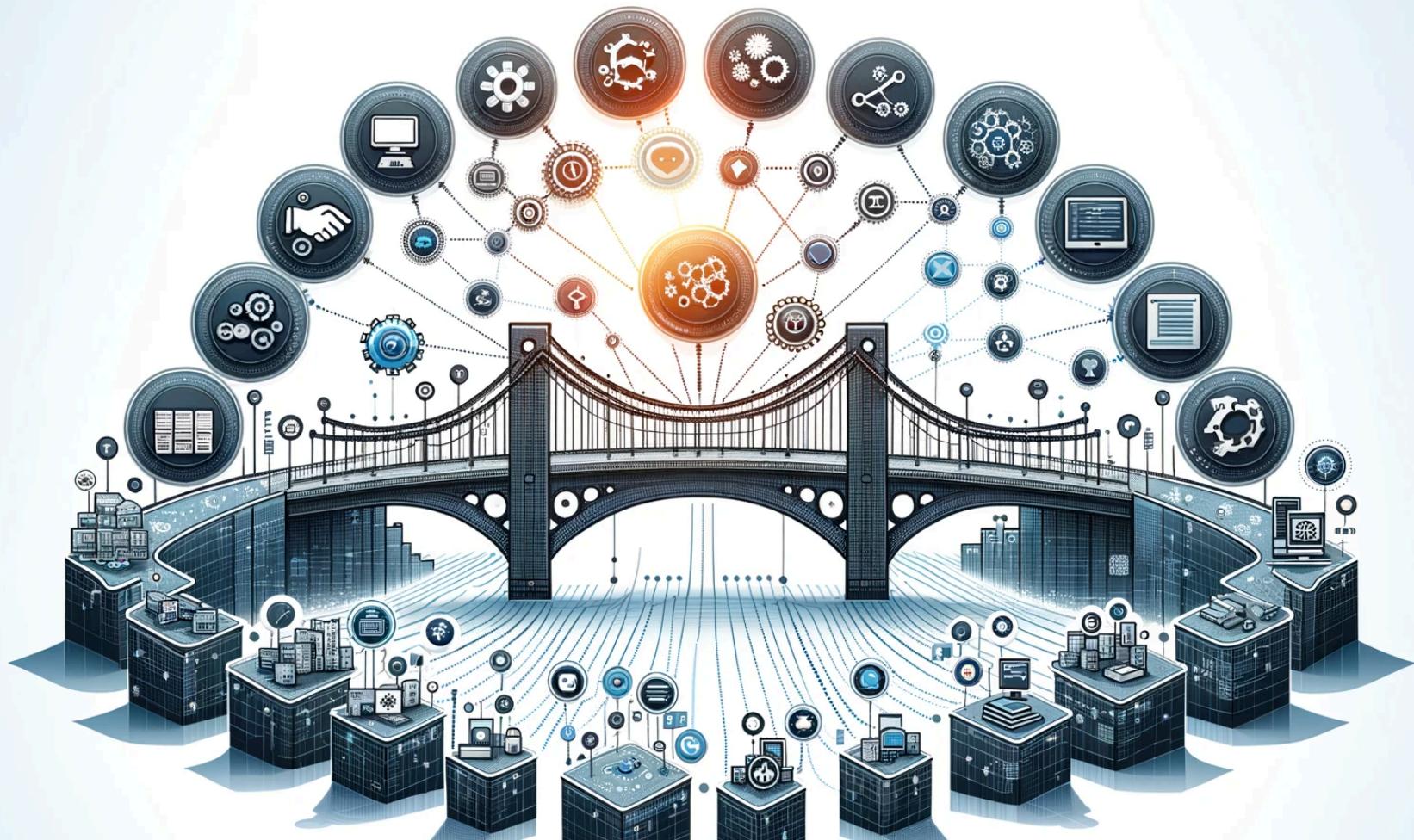
- Identify and resolve issues early
- Proactive problem solving with data and automation
- Faster resolution and improved reliability



Advanced Interoperability

Cross-platform deployment

- Provide uniform approach for diverse platforms
- Reduced complexity and increased efficiency
- Broader reach and greater impact



CI/CD Interoperability Challenges...

...and Remedies

Challenges	Remedy
Diverse Toolsets & Ecosystems	<i>Prioritize critical dependencies, objectives</i>
Data formats and schema differences	<i>Implement unified data formats</i>
Authentication and Authorization	<i>Standardize methods, integrate governance</i>
Versioning and compatibility testing	<i>Clear versioning policies, regular compatibility testing</i>
Lack of documentation	<i>Prioritize efforts + allocate resources, implement standards + process</i>

DevEx reflects an organization's values



Jeremy (#DevRel & #DevEx)

@IAmJerdog · [Follow](#)



If your company does not already have a process for gathering feedback (internal & external) on your product and/or the tools you use, you will not have a good Developer Experience, and I seriously question your commitment to it.

10:57 AM · Jan 25, 2024



26



[Reply](#)



[Copy link](#)

[Read more on X](#)

DevEx is...

...ruthlessly eliminating the barriers (and blockers) that keep your developers (and practitioners) from being successful

-Me

Thank You.



/in/jeremyemeiss



@IAmJerdog



@jerdog



@jerdog@hachyderm.io

END