Open Project Management

from an “open” perspective

UNIT 5

Instructor: Dr. Bradly Alicea

http://bradly-alicea.weebly.com

Lecture 22

Open Project Management

Welcome Back!

Automation II and Related Topics

What was once automated is now taken for granted

Automobiles: automation of the horse-drawn vehicles (automation of horsepower).

- transmissions became automatic starting in the 1940s.
- information technology became predominant in the 1980s
    - automated mechanical automation and human work.

Word Processors: gradual automation of the physical aspects of writing and editing.

Up to the 1970s: streamlining workflow for typists (mechanical, electronic).

1980s onward: automation of the whole editing cycle (software).

Automation In DevOps: Why and How To Automate DevOps Practices

https://www.bmc.com/blogs/automation-in-devops/

DevOps: Agile-based relationship between project development and IT operations (advocating better communication and collaboration between these two business units).

Automation drive by competitive pressures.

Automation In DevOps: Why and How To Automate DevOps Practices (con’t)

https://www.bmc.com/blogs/automation-in-devops/

DevOps: Agile-based relationship between project development and IT operations (advocating better communication and collaboration between these two business units).

Automation drive by competitive pressures.

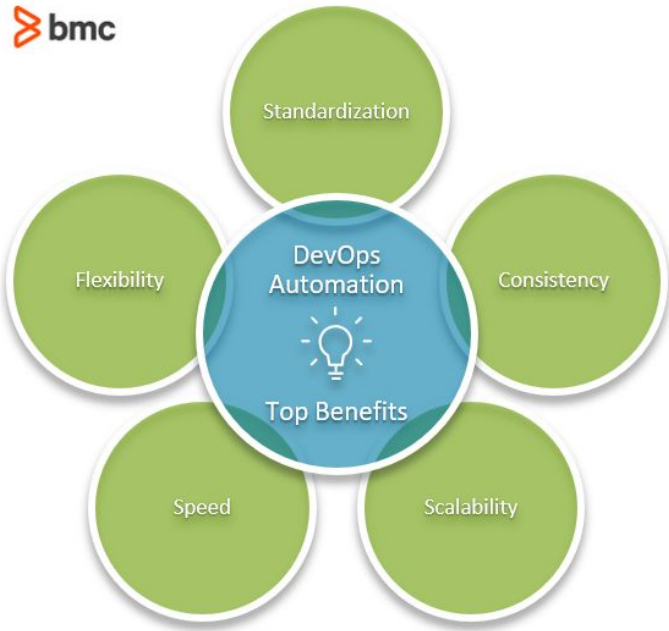
- increase cross-team collaborations, and eliminates the need for large teams.
- automate manual and repetitive tasks.
- improvements to design, software deployment, and monitoring software function.
- enables “fast” movement of development, reduced human error.

Automation In DevOps: Why and How To Automate DevOps Practices (con't)

https://www.bmc.com/blogs/automation-in-devops/

Automation enables standardization. A tension between standardization and adaptability:

- standardization should easily adapt to both new requirements and technological changes.
- automation depends on external factors such as organizational needs and technological feasibility.

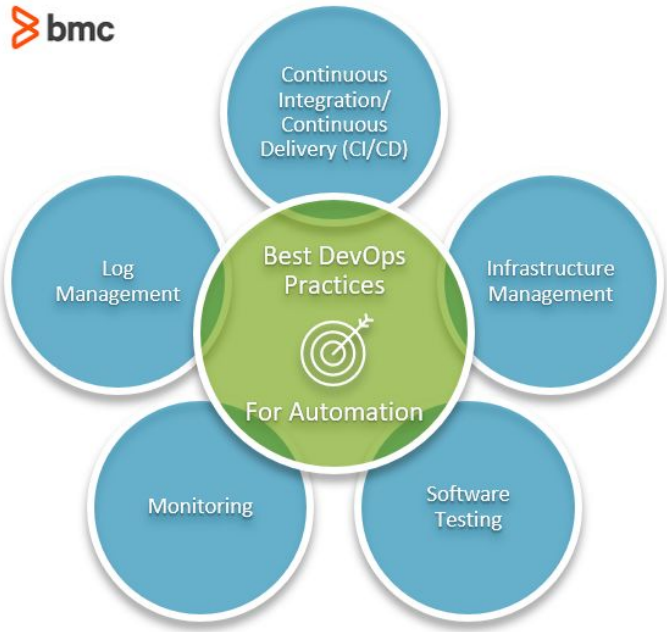


Automation In DevOps: Why and How To Automate DevOps Practices (con't)

https://www.bmc.com/blogs/automation-in-devops/

Which things in your organization are most amenable to automation?

- Continuous Integration/Delivery (CI/CD).
- managing logs and data collection.
- monitoring and testing.
- infrastructure management.



How the open source development community can build more accessible software

https://about.gitlab.com/blog/2021/04/07/how-the-open-source-community-can-build-more-accessible-products/

Open-source comes from values: should be fully-inclusive.

"*The more inclusive and diverse, the better the end product*"

Inclusive design:

- intentionally consider and meet needs of different abilities and workflow preferences.
- anyone can contribute, and should be useful to everyone.

How the open source development community can build more accessible software

https://about.gitlab.com/blog/2021/04/07/how-the-open-source-community-can-build-more-accessible-products/

6 reasons people with disabilities should use Linux: https://opensource.com/life/15/4/6-reasons-people-disabilities-should-use-linux

- reliable, durable, and can deal with obsolescence.
- full control and ownership.
- assistance from a large international community.

The Four Faces of Mass Customization (Gilmore and Pine)

https://hbr.org/1997/01/the-four-faces-of-mass-customization

Mass customization using four approaches (not mutually exclusive)

Collaborative: maintain a dialogue with users.

Adaptive: users can alter the product themselves (not open-source *per se*).

The Four Faces of Mass Customization (Gilmore and Pine)

https://hbr.org/1997/01/the-four-faces-of-mass-customization

Mass customization using four approaches (not mutually exclusive)

Collaborative: maintain a dialogue with users.

Adaptive: users can alter the product themselves (not open-source *per se*).

Cosmetic: different UI/UX for different users.

Transparent: customization is not visible to the user.

Unit 5 Recap

Project Sustainability

Lifecycle and Sustainability

- hype cycles and the long term
    - project work in context
- maintenance vs. planning

Open-source Project Maintenance

- seasonal updates and maintenance
- best practices and dealing with conflict

Project Sustainability (con’t)

Projects as a Series of Goals

- mentorship and involvement.
- developmental cycles and open-source.

Cycles of Working Open

- contribution models and working open for discovery.
- release and support cycles.
    - schedule models, issues and milestones.
- feature cycle management (from completion to death).

Project Sustainability (con’t)

Software Lifecycles

- six formal software life cycle models
    - release lifecycle (Ubuntu example).
    - qualitative assessment of Demo or Die.
    - modernizing an old or aging project.

Evaluatory Methods

- Security and sustainability.
- open-source maturity models.
- User path analysis.

Project Sustainability (con’t)

Automation I

- pros and cons of automation.
- origins of automation.
    - project Cybersyn (large-scale project automation).
    - seeing like a State (drawbacks of large-scale project automation).

Automation for Project Management (with ChatGPT)

- transformations by 2030.
- opportunities (what can automation do?)
- other benefits and robotic process automation.

Project Sustainability (con’t)

Automation II and Related Topics

- automation taken for granted.
- automation in DevOps practice.
    - feasibility and practicality of automation.
- open-source and accessibility.
- four examples of mass customization.