33rd Annual INCOSE international symposium

hybrid event

Honolulu HI USA

Design Considerations and a Case Study

Scalable, Flexible Implementation of MBSE and DevOps in VSEs

About the Speaker

Cailin is pursuing her PhD in systems engineering at Colorado State University while also managing a cross-functional team delivering mission critical software in industry. Even though she is now working on a large, well-established software systems engineering team, her career has been previously centered on implementing scalable model-based systems engineering and devops practices within small software development teams.

In addition to her technical career, she is passionate about equity education in her personal, professional, and academic aspects of her life. She took on the task of co-chairing the first student-led Diversity, Equity, and Inclusion (DEI) committee in her department three years ago and has been committed to furthering education in the DEI space by fostering connections in her student committee.

Background

How can we incrementally implement processes and tooling in small businesses to deliver improved software products?

VSEs and their Challenges

Very Small Entity (VSE)

An organization that has less than 25 employees[1].

Software VSE Challenges

- Potential micro project team sizes (1-3)
- Industry standards are set by larger businesses
- Competition for business

1. Laporte, Claude Y., et al. "5.3. 2 The Development of International Standards for Very Small Enterprises." INCOSE International Symposium. Vol. 18. No. 1. 2008.

VSEs and Government Contracting

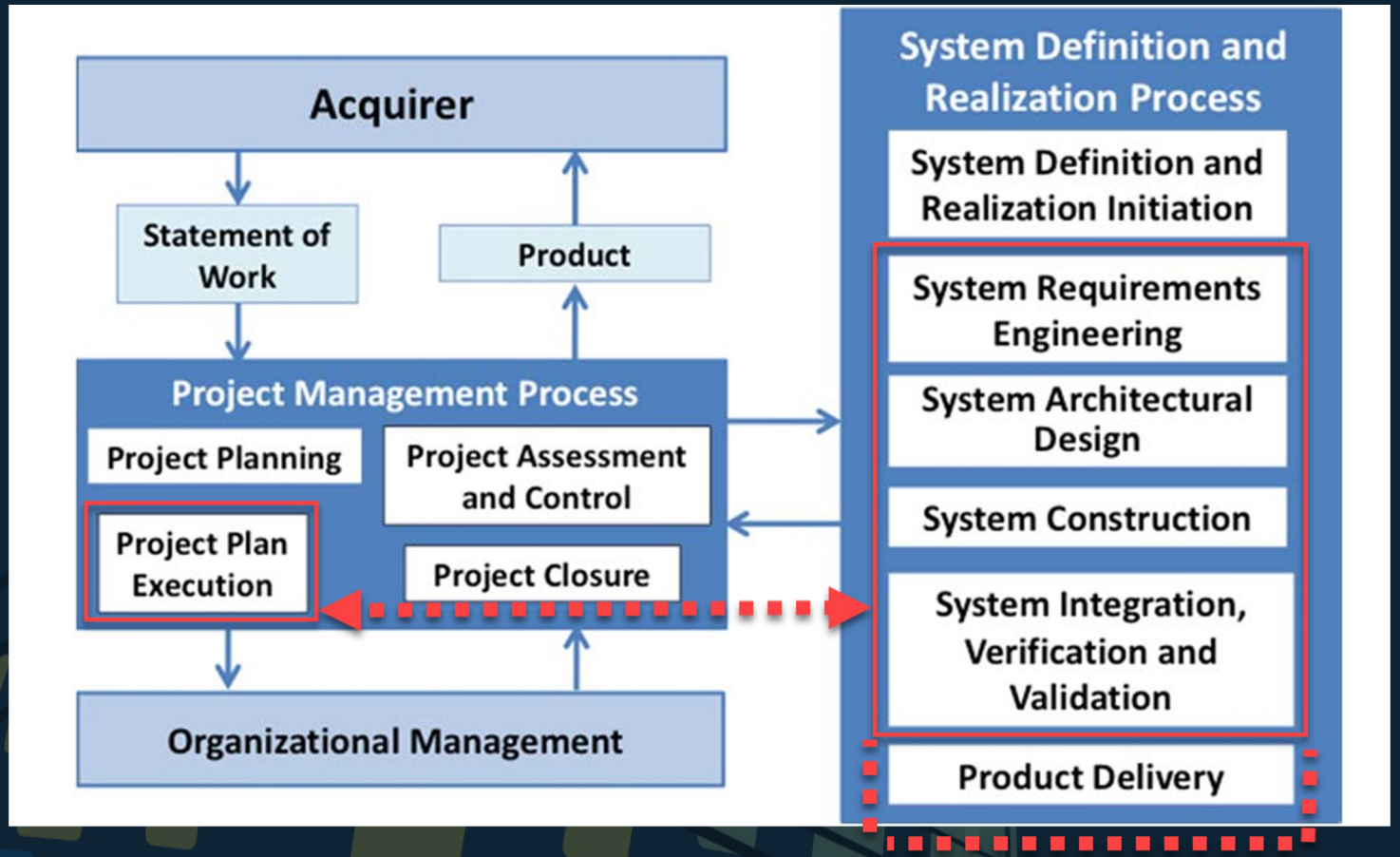
The DoD's *Digital Engineering Workforce Plan* states[1]:

As DoD transitions to digital engineering, there is a need to develop and maintain an acquisition workforce and culture that understands:

- *Model-based engineering*
- *Modern software development practices*
- *Digital engineering processes, methods, and tools*
- *Digital artifacts across the acquisition life cycle*

1. Under Secretary of Defense for Research and Engineering 2022, Digital Engineering Workforce Plan, Washington, DC..

VSEs and Project Plan Execution Strategies



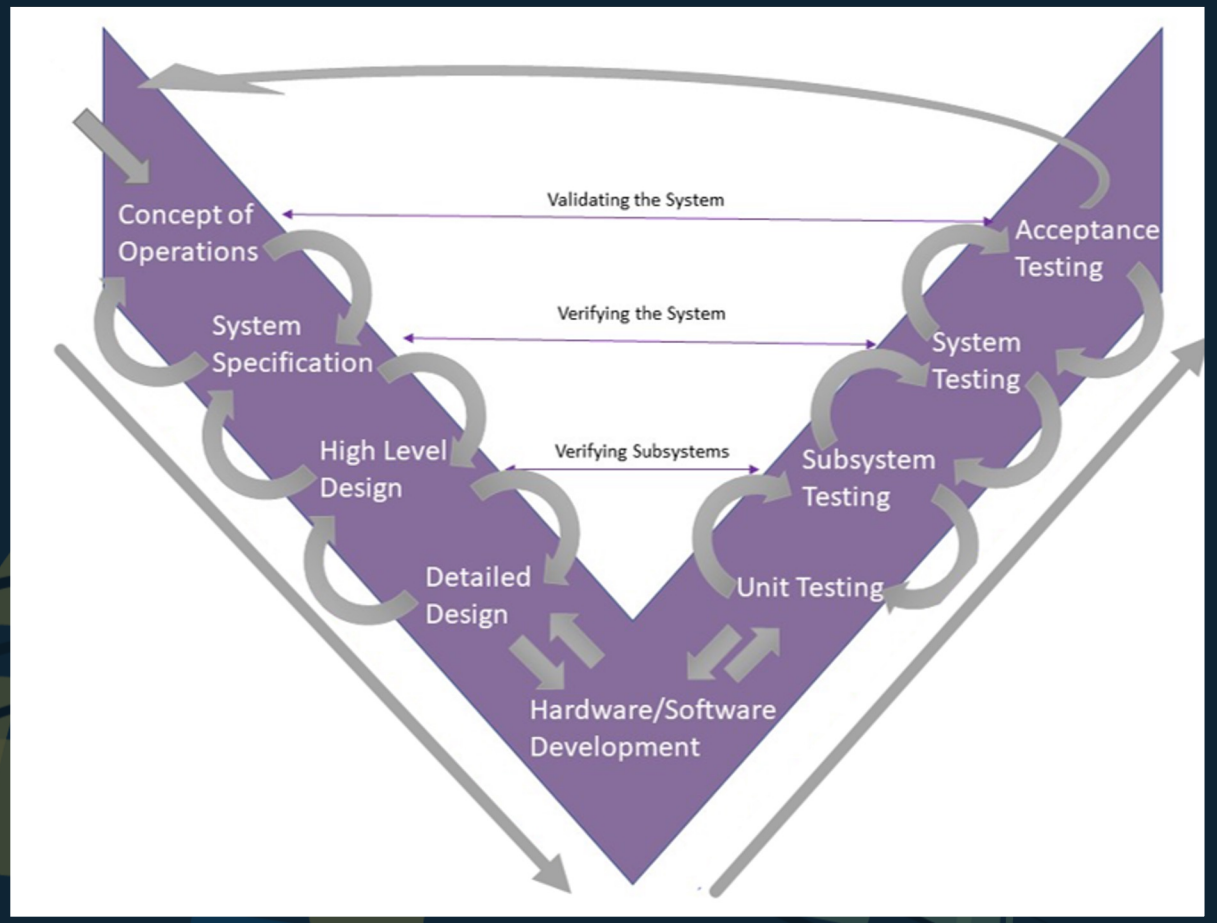
Development Operations (DevOps)

Model-based System Engineering (MBSE)

1. ISO/IEC TR 29110-5-6-2:2014. "Systems Engineering – Life cycle Profiles for Very Small Entities (VSEs) – Part 5-6-2: Systems engineering." Management and engineering guide. Generic profile group: Basic profile, International Organization for Standardization/International Electrotechnical Commission. Geneva, Switzerland. Available at no cost from ISO at: http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

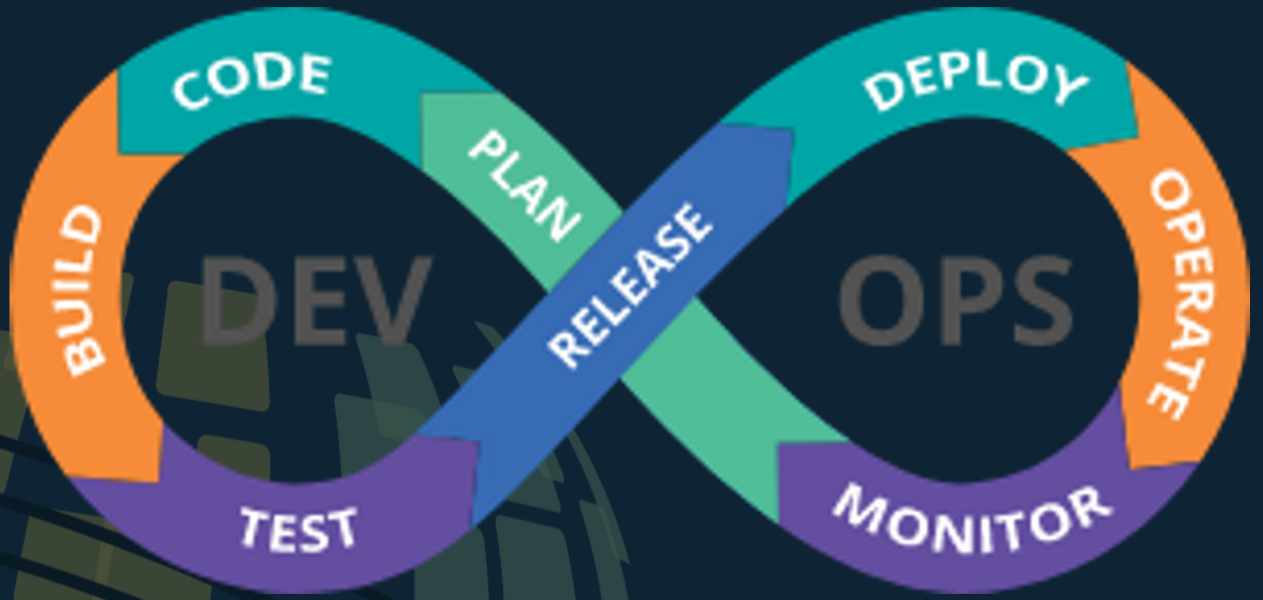
Existing Models and their Benefits/Challenges

Relevant Existing Models



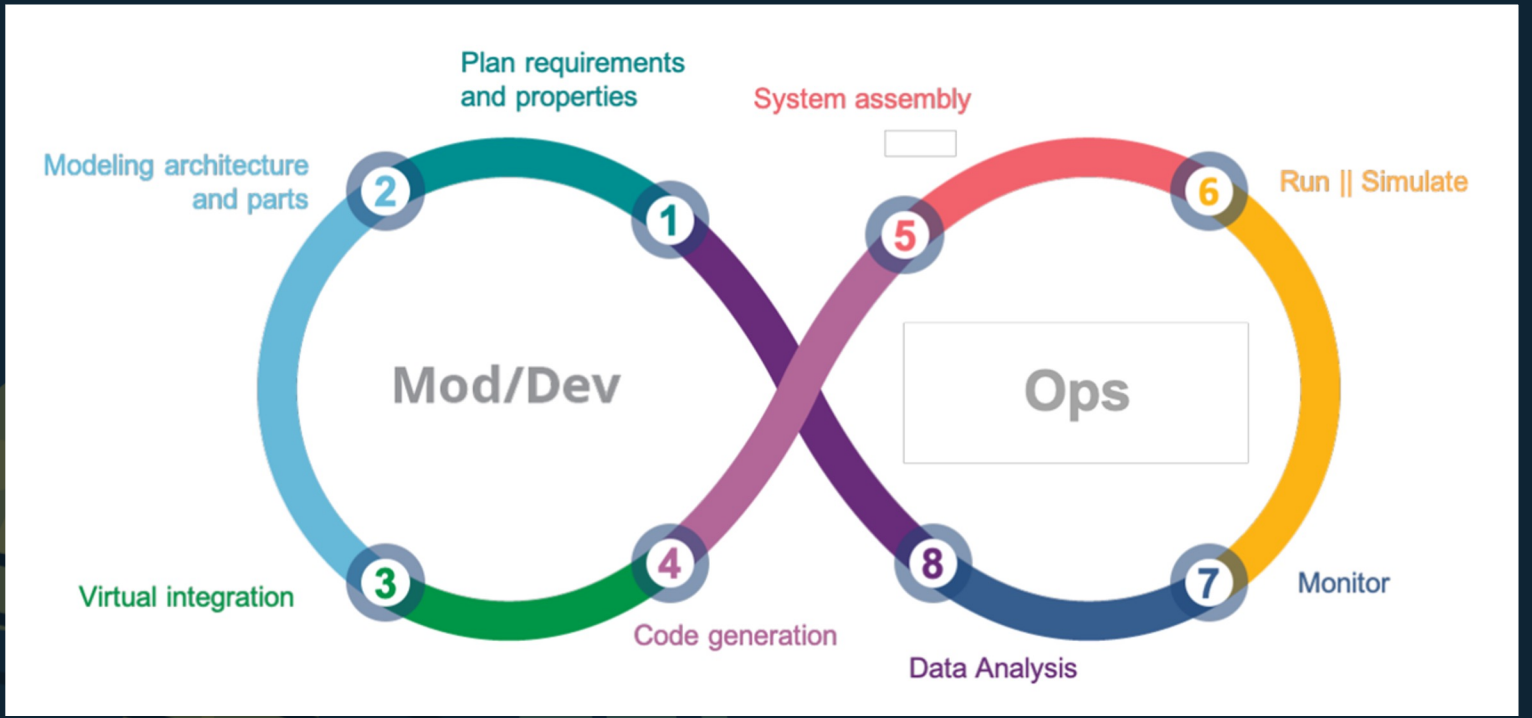
Model-based System Engineering

Relevant Existing Models



Development
Operations

Relevant Existing Models



ModDevOps

A systems/software co-engineering culture and practice that aims at unifying systems engineering (Mod), software development (Dev), and software operation (Ops). The main characteristic of ModDevOps is to strongly advocate abstraction, automation, and monitoring at all steps of system construction, from integration, testing, releasing to deployment and infrastructure management[1].

1. Hugues, J. and Yankel, J. , 2021: From Model-Based Systems and Software Engineering to ModDevOps. Carnegie Mellon University's Software Engineering Institute Blog,. http://insights.sei.cmu.edu/blog/from-model-based-systems-and-software-engineering-to-moddevops/ (Accessed December 9, 2022)

Project Execution Strategy Challenges

MBSE[1]

- Upfront investment
- Adoption strategy
- Purpose & scope definition
- Awareness and change resistance
- Executive leadership sponsorship
- Method definition and extension
- Modularity and reusability
- Complexity management
- Tool dependency and integration
- Large model visualization

DevOps[2]

- Lack of awareness
- Lack of support
- Implementing DevOps technology
- Adapting organizational processes to DevOps

1. Chami, Mohammad, et al. "Towards solving MBSE adoption challenges: the D3 MBSE adoption toolbox." INCOSE International Symposium. Vol. 28. No. 1. 2018.
2. Hamunen, Joonas. "Challenges in adopting a Devops approach to software development and operations." (2016).

Project Execution Strategy Benefits

MBSE[1,2]:

- Finds inconsistencies in design earlier in system development
- Reduces system development cost and risk
- Provides better understanding of projects due to design representation

DevOps[3]

- Promotes teamwork and eliminates silos
- Allows rapid [illegible] of uncertainty
- Improves performance and quality of applications

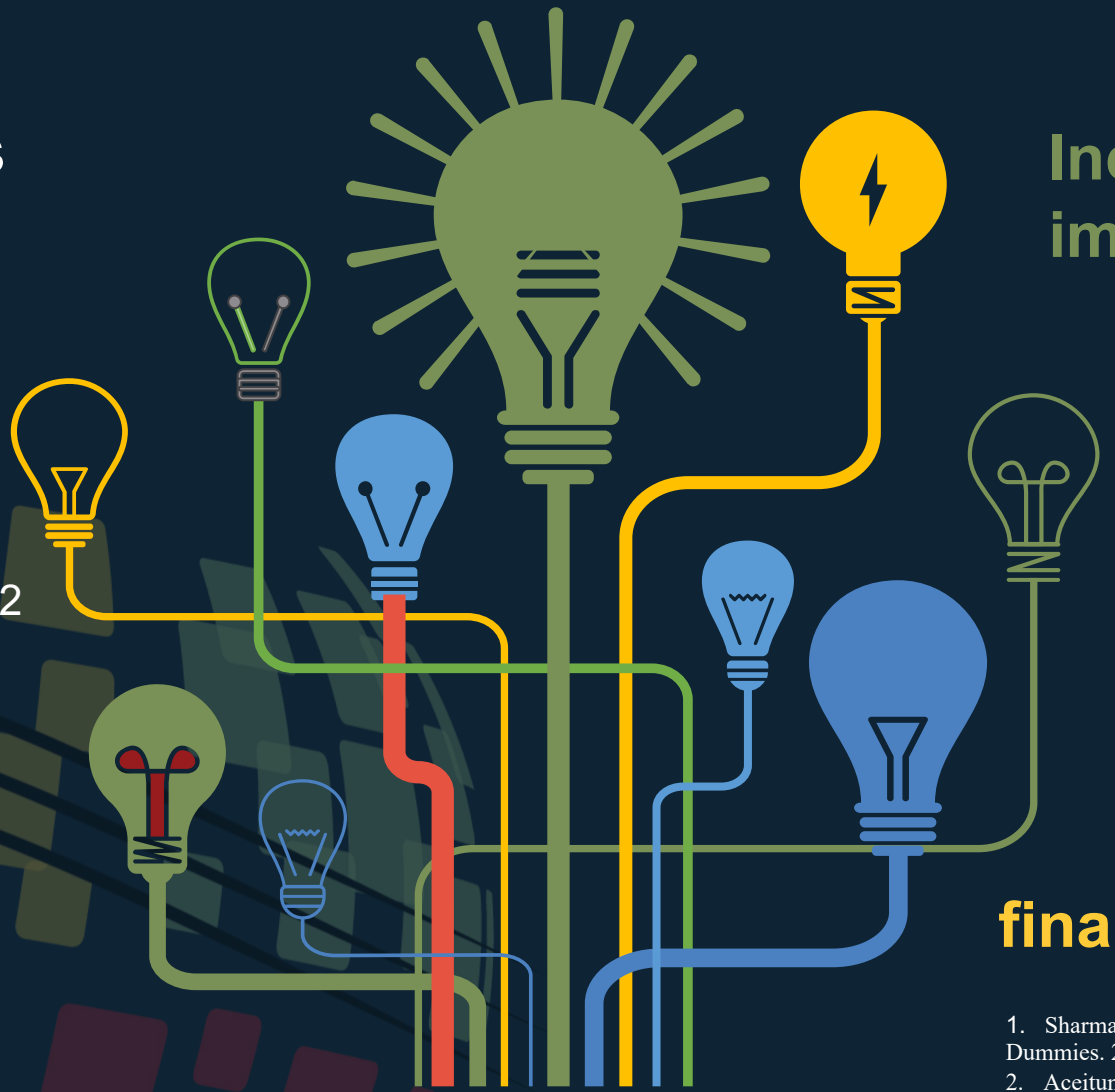
1. McDermott, TA, Hutchison, N, Clifford, M, Aken, E Van, Salado, A & Henderson, K 2020, 'Benchmarking the Benefits and Current Maturity of Model-Based Systems Engineering across the Enterprise - Results of the MBSE Maturity Survey', System Engineering Research Center (SERC), Stevens Institute of Technology, p. 124.
2. Ploeg, Cacia, Kimberly Lai, and Alison Olechowski. "Prioritization of Best Practices in the Implementation of Model-Based Systems Engineering." INCOSE International Symposium. Vol. 32. No. 1. 2022.
3. Anon n.d., "DevOps," Office of the Chief Software Officer, U.S Air Force, viewed 11 December, 2022, <https://software.af.mil/training/devops/>

How do we simplify these Project Execution Plan strategies and paradigms for implementation in VSEs?

The FlexOps Method

FlexOps Vision

Prioritize the core of DevOps with continuous improvement and delivery[1] with MBSE's core to instantiate and reference requirements and model throughout the engineering process[2] while also considering the VSE's main goal: **<u>delivering a system to remain profitable.</u>**



Incremental implementation

Flexibility for VSEs

Proportional personnel and financial resources

1. Sharma, Sanjeev, and Bernie Coyne. "DevOps for Dummies. 2nd IBM Limited Edition." (2015).
2. Aceituna, Daniel, et al. "Model-based requirements verification method: Conclusions from two controlled experiments." Information and Software Technology 56.3 (2014): 321-334.

The FlexOps Method: Steps

Planning requirements and any other project management paradigm planning stages (Scrum, Kanban, etc.)

Modeling the architecture of the system with respect to the requirements defined in the planning step.

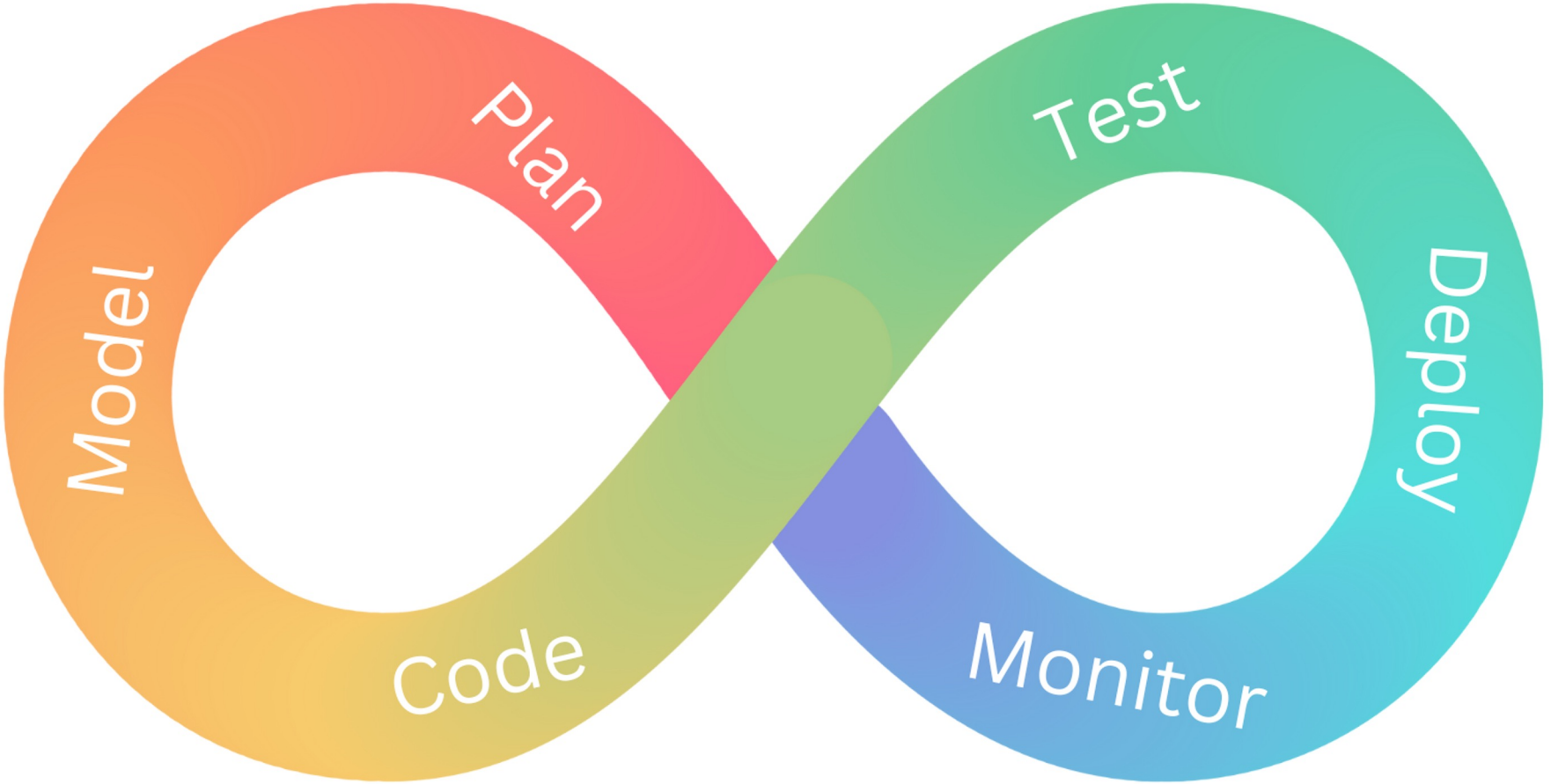
Writing the code for the system is informed by the requirements, model, and plan for execution.

Testing system functionality and verification against the defined requirements and model.

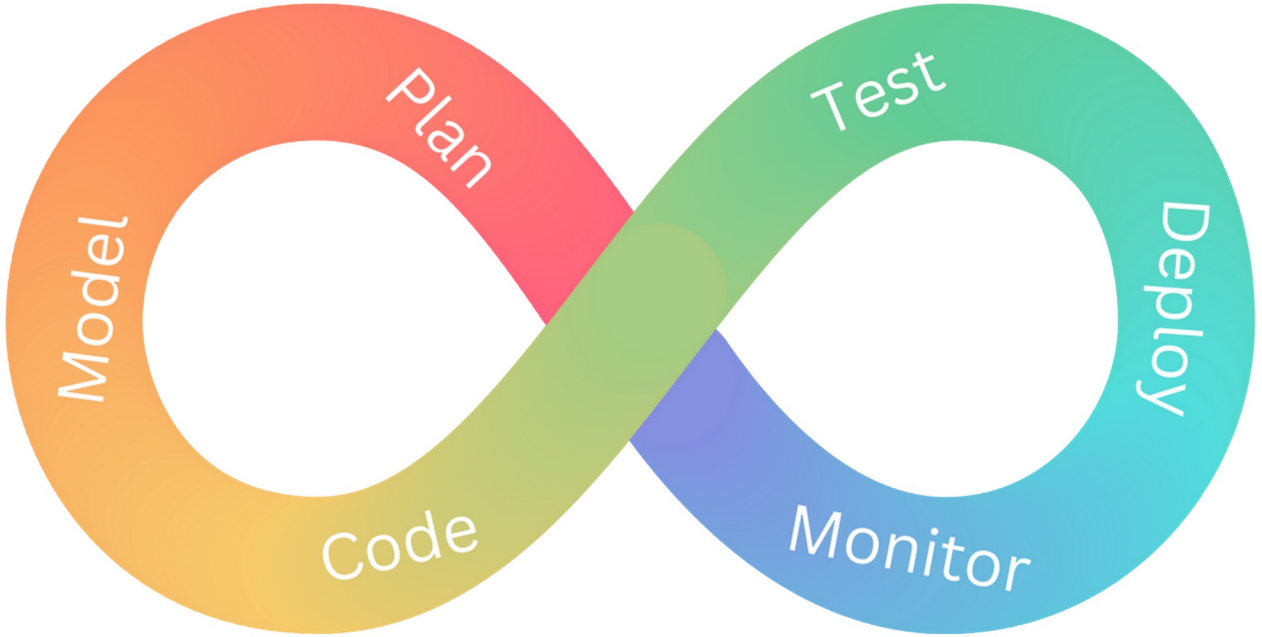
Deploy the system. Deployment practices are up to the VSE and dependent on project size and scope.

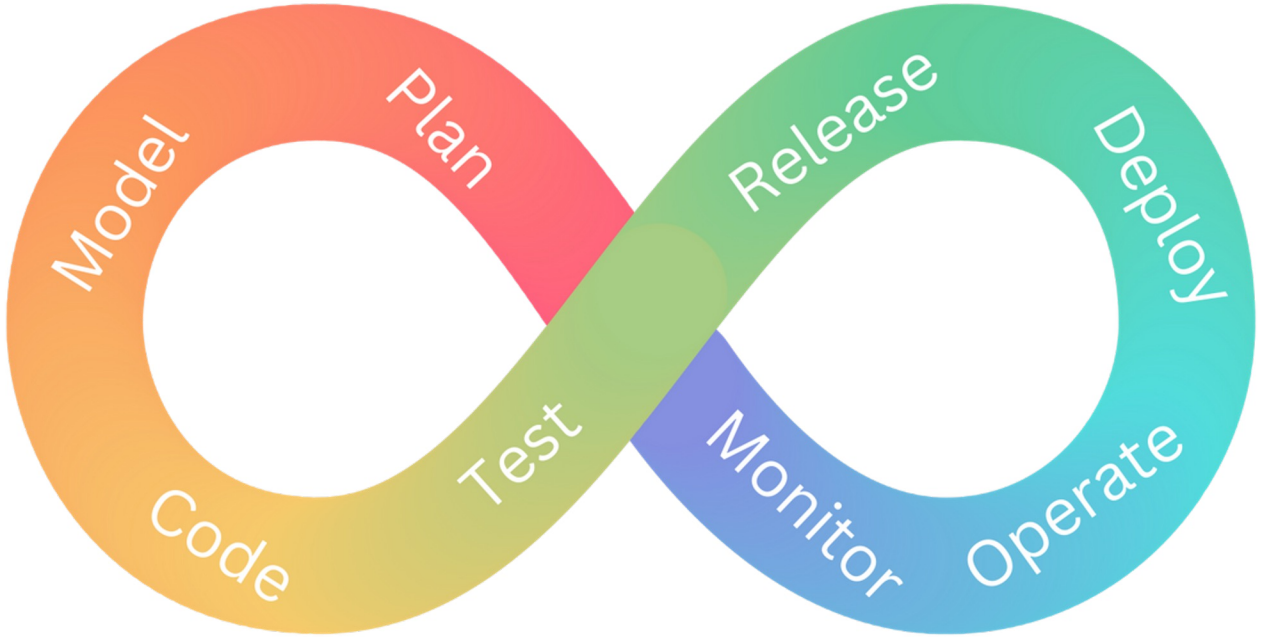
Monitoring the system after deployment is key to assess requirement compliance and customer input.

FlexOps



FlexDevOps FlexOps





Release

Operate

FlexOps Benefits and Challenges

Benefits

- Flexible, incremental approach
- Steps can increase in complexity with VSE maturity

Challenges

- Not applicable for mission critical software/hardware
- Not a “certifiable” implementation of either DevOps or MBSE.

Experimental Application of the Method

Application of the FlexOps Method

Company X

- The FlexOps method was both applied and informed by developing a standard operating procedure at Company X.
- The founder began the company with a consulting agreement developing software as both a service and system for a large company, Company Z.
- The customer needed software to support rapid evaluation of R&D technology efforts (with rapidly evolving requirements).

With limited resources, Company X needed to deliver a software system with continuous improvements as requirements evolved.

Company X FlexOps Implementation - Phase I

FlexOps Step	Supporting Tool(s)	Implementation Description
Plan	PowerPoint, Word, YouTrack	Planning efforts within Company X included requirements definition (PowerPoint & Word), requirements management (PowerPoint & Word), and Scrum planning processes: sprint plannings, backlog grooming, etc (YouTrack).
Model	PowerPoint, YouTrack	Modeling the architecture (PowerPoint) of the system based on the requirements; and reassessing the model with the results of the monitoring and planning steps (YouTrack).
Code	PyCharm, Git, YouTrack	Writing code in the PyCharm Integrated Development Environment (IDE), tracking tasks/user stories within YouTrack, and utilizing version control (Git). Ensuring there is logging functionality within code to provide data for the monitoring step.
Test	PyCharm	Unit testing written within Company X's IDE of choice: PyCharm.
Deploy	Git, Operating System	Pulling the most recent stable commit from Git and installing any dependencies needed for the host operating system.
Monitor	Data logs, Jupyter Notebook, Excel	Analyzing data logs to test and measure requirements compliance.

Company X FlexOps Implementation - Phase II

FlexDevOps Step	Supporting Tool(s)	Implementation Description
Plan	PowerPoint, Word, YouTrack	Planning efforts within Company X included requirements definition (PowerPoint & Word), requirements management (PowerPoint & Word), and Scrum planning processes: sprint plannings, backlog grooming, etc (YouTrack).
Model	PowerPoint, YouTrack	Modeling the architecture (PowerPoint) of the system based on the requirements; and reassessing the model with the results of the monitoring and planning steps (YouTrack).
Code	PyCharm, Git/GitHub, YouTrack	Writing code in the PyCharm Integrated Development Environment (IDE), tracking tasks/user stories within YouTrack, and utilizing version control (Git). Ensuring there is logging functionality within code to provide data for the monitoring step.
Test	PyCharm	Unit testing written within Company X's IDE of choice: PyCharm.
Release	**Git/GitHub**	**Release tags implemented in GitHub, marking commits that were included in a release.**
Deploy	Git/GitHub, Operating System	Pulling the most recent release from Git and installing any dependencies needed for the host operating system.
Operate	**PowerPoint, Word, YouTrack**	**Accepting feedback from the customer and users on the software and data outputs after deployment. Feedback is logged as user stories in YouTrack and any requirements added/changed are captured (PowerPoint & Word).**
Monitor	Data logs, Jupyter Notebook, Excel	Analyzing data logs to test and measure requirements compliance.

What did the tooling cost?

Company X FlexOps Implementation Cost - Year 1

Tool Name	Tool Cost	Employee #	Total Cost
Microsoft 365 Business Premium	$276 per user/year	2	$552
YouTrack	Free	2	$0
PyCharm Professional	$265 per user/year	2	$530
GitHub	$252 per user/year	2	$504
Jupyter Notebook	Free	2	$0
		Total	**$1586**

Company X FlexOps Implementation Cost - Year 2

Tool Name	Tool Cost	Employee #	Total Cost
Microsoft 365 Business Premium	$276 per user/year	4	$1104
YouTrack	Free	4	$0
PyCharm Professional	$265 per user/year	4	$1060
GitHub	$252 per user/year	4	$1008
Jupyter Notebook	Free	4	$0
		Total	**$3172**

Company X FlexOps Implementation Cost - Year 3

Tool Name	Tool Cost	Employee #	Total Cost
Microsoft 365 Business Premium	$276 per user/year	5	$1380
YouTrack	Free	5	$0
Lucid Chart	Free	5	$0
PyCharm Professional	$265 per user/year	5	$1325
GitHub	$252 per user/year	5	$1260
Jupyter Notebook	Free	5	$0
		Total	**$3965**

Company X's FlexOps Total Cost

Tool Cost
Implemented over three years

$8,723

$28,723

Labor + Tools
Implemented over three years

Submarine Warfare Federated Tactical Systems (SWFTS) MBSE Implementation Cost[1]

Implementation Cost
Including 26,111 Personnel Hours over two years

$3.28M

79%

ROI
Valued by the program

1. Rogers III, Edward B., and Steven W. Mitchell. "MBSE delivers significant return on investment in evolutionary development of complex SoS." Systems Engineering 24.6 (2021): 385-408.

FlexOps Benefits and Challenges

Benefits

- Flexible, incremental approach
- Steps can increase in complexity with VSE maturity

Challenges

- Not applicable for mission critical software/hardware
- Not a “certifiable” implementation of either DevOps or MBSE.

Summary & Looking Forward

- FlexOps and FlexDevOps was developed to provide VSEs a model to implement MBSE and DevOps practices within their organizations.
- The methodology was experimentally implemented in Company X, a startup that began as one employee and increased their ability to deliver a software system by applying FlexOps and then FlexDevOps.
- This model helped a VSE increase their footprint within the contract and increase credibility with their customer.

The authors intend to expand this work to other VSEs including small development groups in larger companies that may not have resources within their teams and accurately measure the ROI of FlexOps implementations.

Questions?

33rd Annual INCOSE international symposium

hybrid event

Honolulu HI USA

www.incose.org/symp2023

#INCOSEIS

References

Aceituna, Daniel, et al. "Model-based requirements verification method: Conclusions from two controlled experiments." Information and Software Technology 56.3 (2014): 321-334.

Anon n.d., “DevOps,” Office of the Chief Software Officer, U.S Air Force, viewed 11 December, 2022, <https://software.af.mil/training/devops/>

Bucena, Ineta, and Marite Kirikova. "Simplifying the DevOps Adoption Process." BIR Workshops. 2017.

Chami, Mohammad, et al. "Towards solving MBSE adoption challenges: the D3 MBSE adoption toolbox." INCOSE International Symposium. Vol. 28. No. 1. 2018.

Christensen, Henrik Bærbak. 2016. Teaching DevOps and Cloud Computing Using a Cognitive Apprenticeship and Story-Telling Approach. In Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE ’16). ACM, 174–179. Code: A47.

Hamunen, Joonas. "Challenges in adopting a Devops approach to software development and operations." (2016).

Henderson, Kaitlin, and Alejandro Salado. "Value and benefits of model-based systems engineering (MBSE): Evidence from the literature." Systems Engineering 24.1 (2021): 51-66.

Hugues, J. and Yankel, J. , 2021: From Model-Based Systems and Software Engineering to ModDevOps. Carnegie Mellon University's Software Engineering Institute Blog,. http://insights.sei.cmu.edu/blog/from-model-based-systems-and-software-engineering-to-moddevops/ (Accessed December 9, 2022)

Laporte, C.Y., April, A. and Renault, A., 2006. Applying ISO/IEC software engineering standards in small settings: historical perspectives and initial achievements.

Laporte, Claude Y., et al. "5.3. 2 The Development of International Standards for Very Small Enterprises." INCOSE International Symposium. Vol. 18. No. 1. 2008.

McDermott, TA, Hutchison, N, Clifford, M, Aken, E Van, Salado, A & Henderson, K 2020, ‘Benchmarking the Benefits and Current Maturity of Model-Based Systems Engineering across the Enterprise - Results of the MBSE Maturity Survey’, System Engneering Research Center (SERC), Stevens Institute of Technology, p. 124.

Ploeg, Cacia, Kimberly Lai, and Alison Olechowski. "Prioritization of Best Practices in the Implementation of Model-Based Systems Engineering." INCOSE International Symposium. Vol. 32. No. 1. 2022.

Ravichandran, Aruna, Kieran Taylor, and Peter Waterhouse. "DevOps and real world ROI." DevOps for Digital Leaders. Apress, Berkeley, CA, 2016. 139-150.

Rising, Linda, and Norman S. Janoff. "The Scrum software development process for small teams." IEEE software 17.4 (2000): 26-32.

Rogers III, Edward B., and Steven W. Mitchell. "MBSE delivers significant return on investment in evolutionary development of complex SoS." Systems Engineering 24.6 (2021): 385-408.

Sharma, Sanjeev, and Bernie Coyne. "DevOps for Dummies. 2nd IBM Limited Edition." (2015).

_____, Under Secretary of Defense for Research and Engineering 2022, Digital Engineering Workforce Plan, Washington, DC.

Van Loon, Han. Process Assessment and ISO/IEC 15504: a reference book. Vol. 775. Springer Science & Business Media, 2004.

Wymore, A. Wayne. Model-based systems engineering. CRC press, 2018.