Polarion Software®

WHITE PAPER

ARE YOU A DEVOPS HERO?

BY STEFANO RIZZO

Europe, Middle-East, Africa: Polarion Software GmbH
Hedelfinger Straße 60 — 70327 Stuttgart, GERMANY
Tel +49 711 489 9969 - 0
Fax +49 711 489 9969 - 20
www.polarion.com - info@polarion.com

Americas & Asia-Pacific: Polarion Software, Inc.
100 Pine Street, Suite 1250, San Francisco, CA 94111, USA
Tel +1 877 572 4005 (Toll free)
Fax +1 510 814 9983
www.polarion.com - info@polarion.com

Abstract

DevOps is neither a “social trend” nor a buzzword. It is a genuine business concern arising more and more from the daily pressures between your Operations and IT departments.

This whitepaper introduces three different maturity levels in addressing the DevOps need, then provides a self-assessment questionnaire to let companies indentify the maturity level of their existing DevOps approach.

As a key takeaway the reader will receive recommendations on how to reach a higher maturity level in merging together tools and methods between Development and Operations departments.

Is DevOps a Real Need?

It is fairly easy and often correct to ask whether a new marketing trend is just “new”, or if it originates from a genuine need. DevOps (the rising wave of tools and methods to put together Development and Operations departments) are based on solid arguments and not just marketing hysteria.

Some of the arguments are:

- Agile methods have increased the release frequency and pace and as a result have enormously decreased the release preparation time
- Virtualization, cloud, mobile, mixed environments have also increased target system complexity and;
- SOA and the integration of 3rd party services adds another degree of complexity into the mix

So, while Agile seemed to solve most of the historical issues faced by the development labs in companies under time-to-market demands, continuous innovation, and fast competition pressures, this “solution” actually moved the problem into Operations. So the problem for the company remains unchanged.

What is DevOps?

DevOps has been defined as a movement, a set of principles, practices, concepts, or methods. It is in fact an umbrella concept that refers to the software development methodology that improves the integration between silo activities through collaboration, communication and automation.

DevOps is the way to integrate Development and Operations, to create a more collaborative and productive relationship between these teams whose operational tasks are interdependent.

In a traditional organization with separate departments for Dev, IT operations and QA, development methodologies (such as Agile software development) usually do not cross-departmental integration with IT support or QA.

The numerous steps involved, which cut across multiple domains such as development, QA, network engineering, and operations makes the deployment of complex web-based and milti-tier applications highly error prone, considering that many such steps involve manual intervention and routine tasks.

The biggest pressure-point for Operations is the increased release frequency that is one of the most popular and appreciated practices common to every modern Agile development method. So while development seems to be able to survive the crazy time-to-market needs driving innovation, Operations continue to struggle.

On the other hand, the adoption of Agile methods within development and their direct impact on Operations with practices like Continuous Delivery, have given some relief to Operations: more and more Agile development teams manage the deployment.

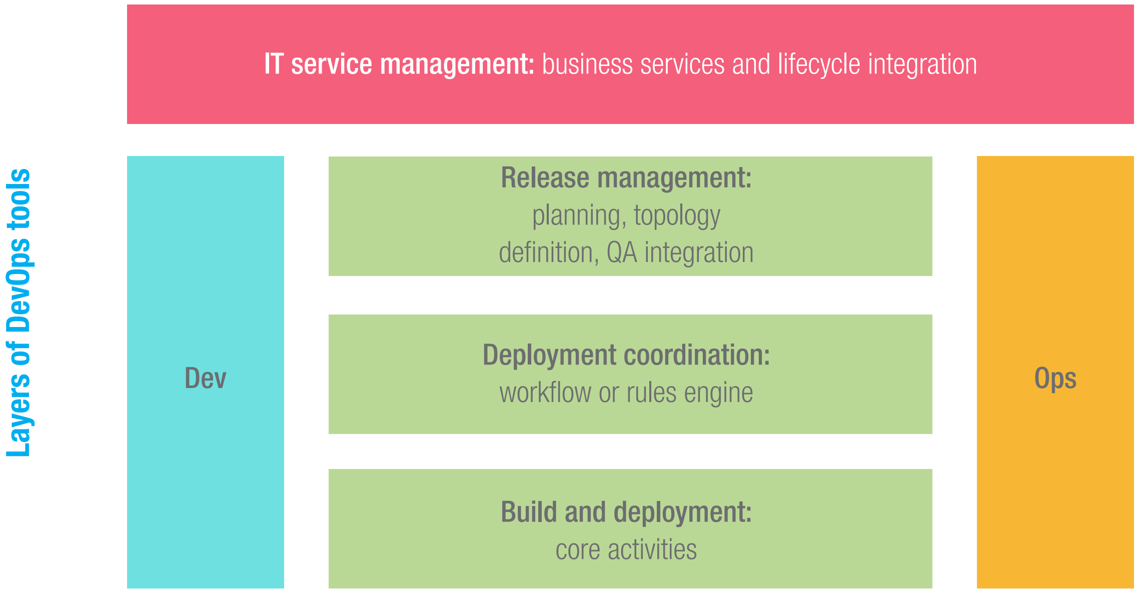
To add to the complexity of Operations, the use of public and hybrid cloud services is increasing, leading operators to manage application instances deployed to services instead of managing servers, backups and hardware faults.

So the landscape of Operations is changing even more rapidly than in the past and some Ops professionals have already been merged into Agile development teams. But in most cases, to increase Operations' efficiency and reduce the production risk it is critically necessity to improve overall collaboration and process automation between Development and Operations. This is, in a nutshell, the goal of DevOps.

We can say that DevOps is a way to put Agile practices into Operations by adopting Agile values and practices. In a similar way that Agile has included Testers as equals in the development process, Build and Ops are now no longer hidden in the basement corner. However, the newest members continue to use legacy tools which cannot keep up with their Agile teammates.

According to the well respected analyst group Ovum "DevOps is evolving to embrace businessfacing IT services such as helpdesk and enterprise architecture through closer integration":

> *"DevOps offers the opportunity to tackle these challenges and bring efficiencies into operations; improve collaboration between all stakeholders, adopting Agile values in working styles; and benefit from new automated solutions that target continuous delivery, release management, and deployment automation."*
>
> Michael Azoff, Principal Analyst at Ovum



IT service management: business services and lifecycle integration
Layers of DevOps tools
Dev
Release management: planning, topology definition, QA integration
Deployment coordination: workflow or rules engine
Build and deployment: core activities
Ops

[Author: Michael Azoff, "DevOps: Agile Operations and Continuous Delivery" (IT017003561) 30 Sep 2011 Reference Code: IT017003561 Publication Date: 30 Sep 2011]

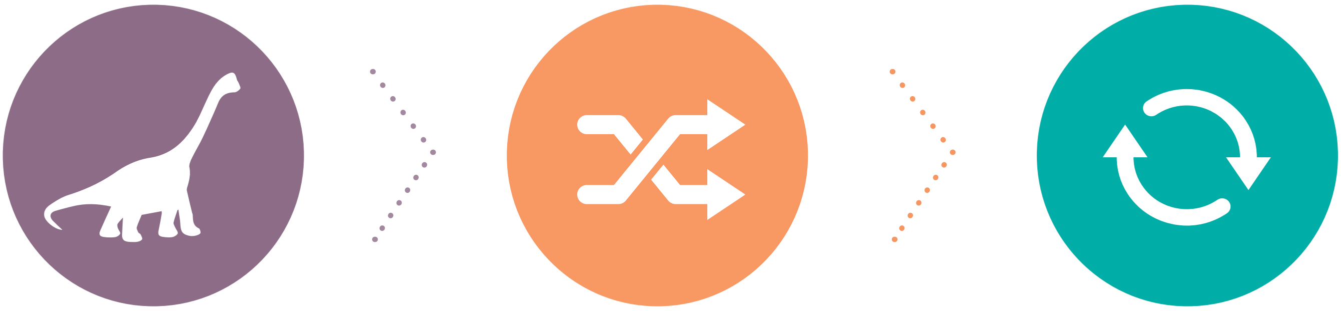
Key Take-away

The paradigm is changing. Agile development and the increased use of public and hybrid cloud services are reshaping the Operations center by removing inefficient and unsuitable work processes.
The adoption of Agile methodologies in development areas is resulting in higher-frequency deployments into production and therefore demands on Operations are increasing.

DevOps is designed to improve the effectiveness of an Operations department, whose pace must be aligned with Development's frequent agile releases.
Release management and automation software solutions help organizations produce software products and services and address the urgency in getting the product to market on time.

DevOps Approaches

DevOps is becoming a priority in most organizations today and there are several approaches to how you can execute a DevOps strategy. We have defined three different DevOps approaches:



Dinosaur approach (DinoOps)

Dev and Ops are separated, with a basic communication channel.

Dev

- Provide Ops with releases

Ops

- Provide Dev with bugs and production failures

DinoOps best practice: define a Release Management Process, independent from Development process. The process must be easy, repeatable with predictable timing and results.

Mixed approach (MixOps)

Dev and Ops run separate processes with common collaboration layer (bi-directional communication).

Dev

- Be involved in selecting the application stack
- Configure and deploy virtual or cloud servers (potentially)
- Deploy their applications
- Monitor application and system health
- Respond to applications problems as they arise.

Ops

- Manage the hardware infrastructure
- Configure and monitor networking
- Enforce policies around backup, DR, security, compliance, change control, etc.
- Assist in monitoring the systems
- Manage active directory
- Asset tracking
- Other non-production, application-related tasks

Agile approach (AgileOps)

Dev and Ops are the same team, sharing a single process (with Ops backlog) orchestrated by a single product/platform.

Some consequences:

- User stories cover release and production needs
- There is a backlog owner for Ops
- Every iteration covers deployments at least into test environment

DevOps Backbone

In order to support MixOps, Development and Operations it must share a common toolset, called here DevOps backbone.
A DevOps backbone is the place where Development and Operations meet. It is the orchestra director of your DevOps, the shared platform to manage common processes and to ensure that Development and Operations are aligned. To some extent, it provides agility to the Operations department and consciousness of operational needs into the Development team.

The common processes managed by the DevOps backbone are:

- Collaboration (task and issue management, comments, approvals, message tracking...)
- Version and change management: versioning, change requests, change management, collaborative coding, bug fixing and traceability management on scripts and code
- Requirements management (requirements must cover also Ops needs)
- QA (test cases, derived from requirements, must include also production tests; risk management encompasses production)

DevOps Self-Assessment

If you work in a company with separated Development and Operations/IT departments you can assess the degree of maturity of your DevOps setup by answering a few questions pertaining to your specific role.

Management questions

YES / NO

1. Do your developers have visibility over Operations progress? □
2. Does your IT have a consistent visibility over the actual Development release schedule, including delays and cancellations? □
3. Do you involve Operations in the early development stages so they can be aware of the future needs and start planning IT setup and deployment options? □
4. When something goes wrong is it clear who is in charge of it? □

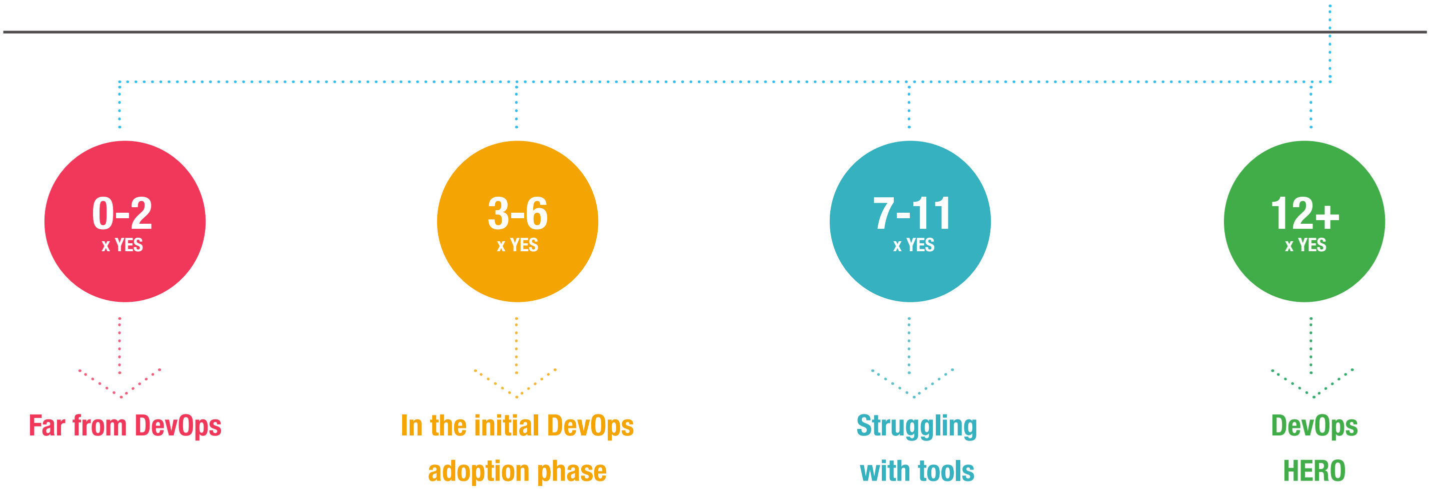
Process questions

1. Do your business or user requirements include requirements about the production environment? □
2. Do you define test cases and acceptance criteria in the early phases of development that include production needs (examples: response time, number of concurrent users, uptime...)? □
3. Do you plan development and deployment activities in the same chart? □

Toolset questions

1. Do your tools facilitate the communication between Dev and Ops (vs. create a physical barrier between them, i.e. “silos”)? □
2. Do your tools support your process (vs. mandate their process)? □
3. Do your Dev and Ops tools share the same workflow? □
4. Can the same workflow engine orchestrate your Dev and Ops tools? □
5. Are deployment scripts under version control? □
6. Can you track and manage changes to deployment scripts? □
7. Do you have traceability between development and deployment assets? □
8. Do you maintain all development and operational assets in a single asset DB? □

TOTAL SCORE: 0 **x YES**


0-2 x YES — Far from DevOps

3-6 x YES — In the initial DevOps adoption phase

7-11 x YES — Struggling with tools

12+ x YES — DevOps HERO

Far from DevOps

If you answered more than 12 “nos”, then you are in the preliminary stages of DevOps maturity. You are not yet fully aware of the benefits DevOps can provide. If this describes your organization you will recognize that your Operations are affected by the pressures described earlier in this paper.
Recommendation: consider moving your Development towards an open and extensible ALM platform. The recipe is to start with a MixOps approach then merge teams into AgileOps after 6-12 months.

In the initial DevOps adoption phase

If you answered between 9 and 12 “nos”, you are probably mid-way to creating a common processes and your Operations are getting closer to your Development department.
Recommendation: Do not waste any more time and move straight to an AgileOps approach. In order to do that, start evaluating a unified platform for your Dev and Ops. If it’s not realistic for you, then you might follow the instructions below for those “struggling with tools”.

Struggling with tools

If you collected a significant number of “yes” in the Management and Process questions, but you have more than 4 “no” in the Toolset questions, this may mean that you are already struggling with Dev and Ops tools misalignment.
Recommendation: Start to align Dev and Ops into an extensible unified platform for Dev and Ops. In case this is not possible, at least in the short term, you should adopt a MixOps approach. A DevOps backbone, as previously described, integrated or at least coupled with your Development and Operation tools, could be the right technical solution for you.

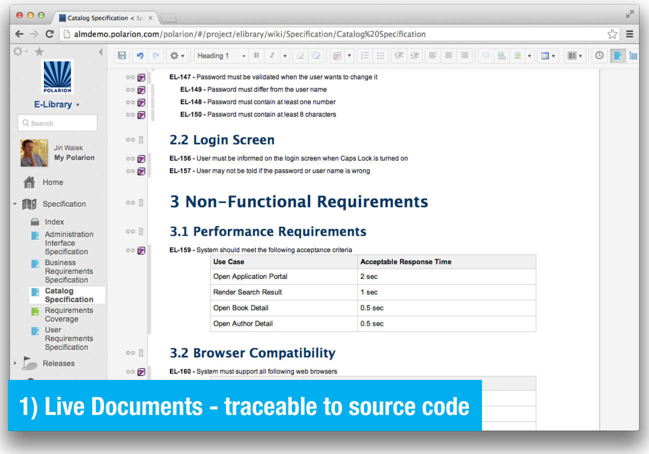
DevOps HERO

If you answered more than 12 “yes”, please give me a call: I’m eager to listen how much you benefit from your DevOps approach and setup.

Your Solution: Polarion ALM

Polarion ALM is typically considered as a powerful tool for Development teams. An increasing number of our customers are using it more and more to manage their DevOps needs.
The following features explain why Polarion customers are more successful with DevOps:

- Integration (actually Polarion ALM is a single tool, so there is nothing to integrate...) Fig. 1
- Information/asset uniqueness (single source, no replicas)
- Ability to store and version software, configuration files and deployment scripts Fig. 2
- Manage variants of software, configuration files and deployment scripts Fig. 3
- Define and reuse deployment workflows
- Provide content driven workflow and process knowledge to tackle complex deployments
- Embedded QA features Fig. 4
- Govern (plan-execute-track) the change of deployed assets Fig. 5
- Ability to define and maintain patterns and templates in managing releases
- Embedded tools like Maven and Ant for many build and release activities Fig. 6
- Strong integrations with popular tools like Hudson and Jenkins
- Platform openness to any 3rd party release management or build tool



almdemo.polarion.com/polarion/#/project/elibrary/wiki/Specification/Catalog%20Specification

EL-147 - Password must be validated when the user wants to change it

EL-149 - Password must differ from the user name

EL-148 - Password must contain at least one number

EL-150 - Password must contain at least 8 characters

2.2 Login Screen

EL-156 - User must be informed on the login screen when Caps Lock is turned on

EL-157 - User may not be told if the password or user name is wrong

3 Non-Functional Requirements

3.1 Performance Requirements

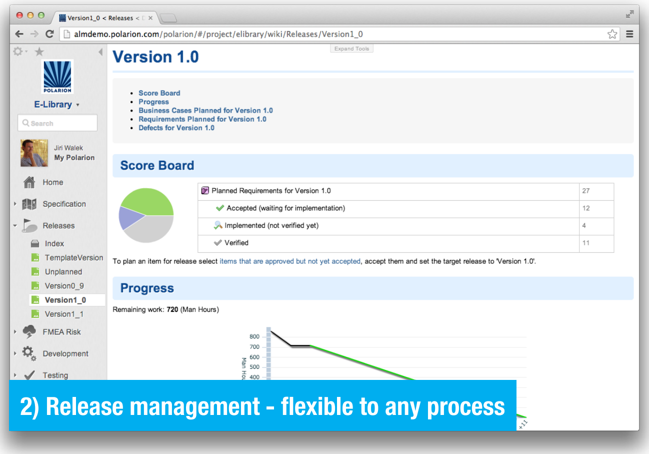
EL-159 - System should meet the following acceptance criteria

Use Case	Acceptable Response Time
Open Application Portal	2 sec
Render Search Result	1 sec
Open Book Detail	0.5 sec
Open Author Detail	0.5 sec

3.2 Browser Compatibility

EL-160 - System must support all following web browsers

1) Live Documents - traceable to source code



almdemo.polarion.com/polarion/#/project/elibrary/wiki/Releases/Version1_0

Version 1.0

- Score Board
- Progress
- Business Cases Planned for Version 1.0
- Requirements Planned for Version 1.0
- Defects for Version 1.0

Score Board

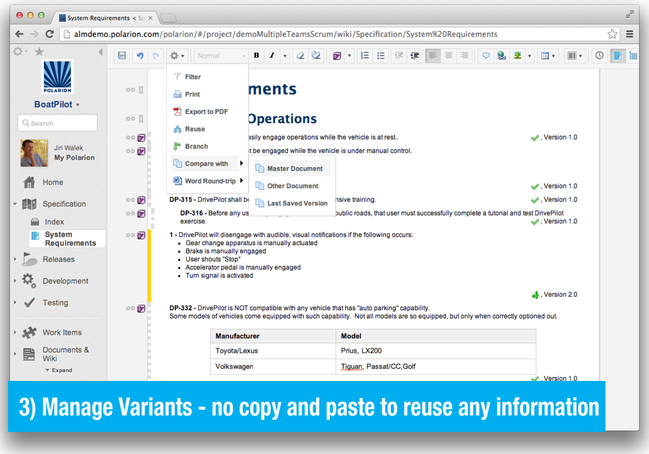
Planned Requirements for Version 1.0	27
✓ Accepted (waiting for implementation)	12
Implemented (not verified yet)	4
✓ Verified	11

To plan an item for release select items that are approved but not yet accepted, accept them and set the target release to 'Version 1.0'.

Progress

Remaining work: **720** (Man Hours)

2) Release management - flexible to any process



almdemo.polarion.com/polarion/#/project/demoMultipleTeamsScrum/wiki/Specification/System%20Requirements

ments

Operations

asily engage operations while the vehicle is at rest..

t be engaged while the vehicle is under manual control.

DP-315 - DrivePilot shall b nsive training.

DP-316 - Before any us public roads, that user must successfully complete a tutorial and test DrivePilot exercise.

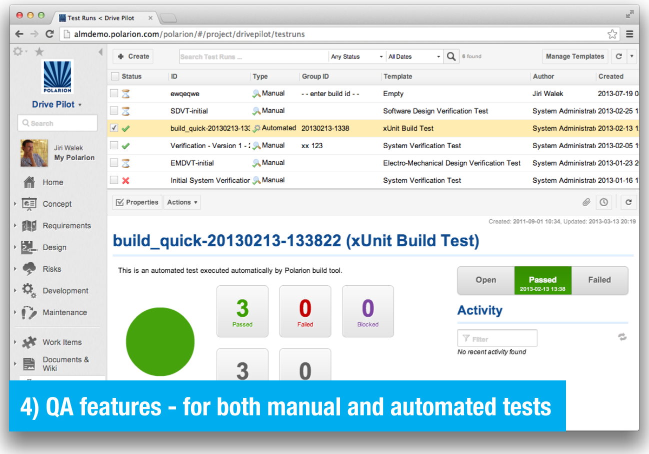
1 - DrivePilot will disengage with audible, visual notifications if the following occurs:

- Gear change apparatus is manually actuated
- Brake is manually engaged
- User shouts "Stop"
- Accelerator pedal is manually engaged
- Turn signal is activated

DP-332 - DrivePilot is NOT compatible with any vehicle that has "auto parking" capability. Some models of vehicles come equipped with such capability. Not all models are so equipped, but only when correctly optioned out.

Manufacturer	Model
Toyota/Lexus	Prius, LX200
Volkswagen	Tiguan, Passat/CC,Golf

3) Manage Variants - no copy and paste to reuse any information



almdemo.polarion.com/polarion/#/project/drivepilot/testruns

Status	ID	Type	Group ID	Template	Author	Created
	ewqeqwe	Manual	- - enter build id - -	Empty	Jiri Walek	2013-07-19 0
	SDVT-initial	Manual		Software Design Verification Test	System Administrat	2013-02-25 1
✓	build_quick-20130213-13:	Automated	20130213-1338	xUnit Build Test	System Administrat	2013-02-13 1
✓	Verification - Version 1 - :	Manual	xx 123	System Verification Test	System Administrat	2013-02-05 1
	EMDVT-initial	Manual		Electro-Mechanical Design Verification Test	System Administrat	2013-01-23 2
✗	Initial System Verificatior	Manual		System Verification Test	System Administrat	2013-01-16 1

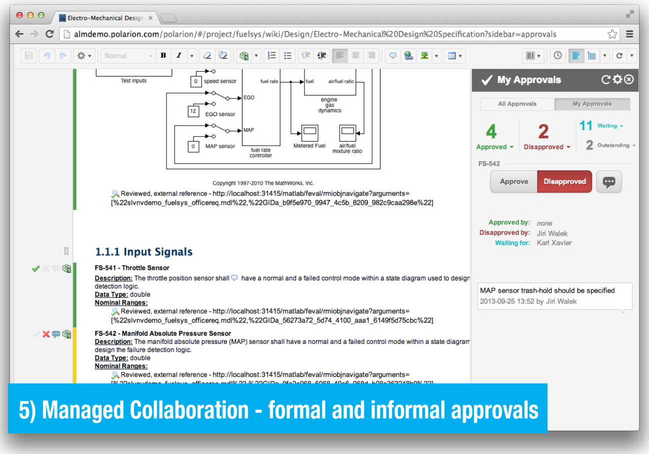
Created: 2011-09-01 10:34, Updated: 2013-03-13 20:19

build_quick-20130213-133822 (xUnit Build Test)

This is an automated test executed automatically by Polarion build tool.

No recent activity found

4) QA features - for both manual and automated tests



almdemo.polarion.com/polarion/#/project/fuelsys/wiki/Design/Electro-Mechanical%20Design%20Specification?sidebar=approvals

Copyright 1997-2010 The MathWorks, Inc.

Reviewed, external reference - http://localhost:31415/matlab/feval/rmiobjnavigate?arguments=[%22slvnvdemo_fuelsys_officereq.mdl%22,%22GIDa_b9f5e970_9947_4c5b_8209_982c9caa298e%22]

1.1.1 Input Signals

FS-541 - Throttle Sensor

Description: The throttle position sensor shall have a normal and a failed control mode within a state diagram used to design detection logic.
Data Type: double
Nominal Ranges:

Reviewed, external reference - http://localhost:31415/matlab/feval/rmiobjnavigate?arguments=[%22slvnvdemo_fuelsys_officereq.mdl%22,%22GIDa_56273a72_5d74_4100_aaa1_6149f5d75cbc%22]

FS-542 - Manifold Absolute Pressure Sensor

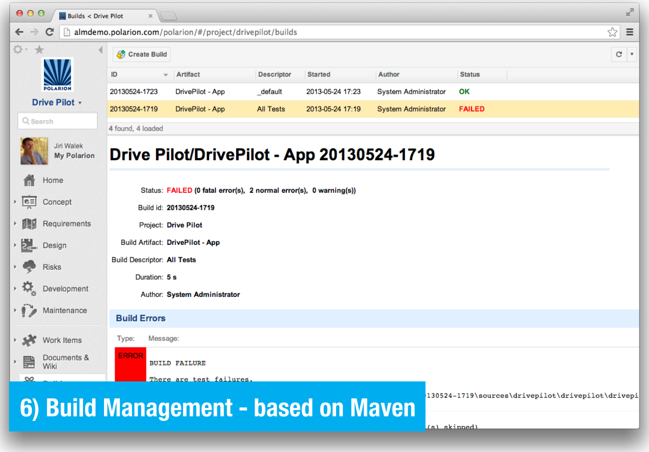
Description: The manifold absolute pressure (MAP) sensor shall have a normal and a failed control mode within a state diagram design the failure detection logic.
Data Type: double
Nominal Ranges:

Reviewed, external reference - http://localhost:31415/matlab/feval/rmiobjnavigate?arguments=

Approved by: *none*
Disapproved by: Jiri Walek
Waiting for: Karl Xavier

MAP sensor trash-hold should be specified
2013-09-25 13:52 by Jiri Walek

5) Managed Collaboration - formal and informal approvals



almdemo.polarion.com/polarion/#/project/drivepilot/builds

ID	Artifact	Descriptor	Started	Author	Status
20130524-1723	DrivePilot - App	_default	2013-05-24 17:23	System Administrator	OK
20130524-1719	DrivePilot - App	All Tests	2013-05-24 17:19	System Administrator	FAILED

4 found, 4 loaded

Drive Pilot/DrivePilot - App 20130524-1719

Status: FAILED (0 fatal error(s), 2 normal error(s), 0 warning(s))
Build id: 20130524-1719
Project: Drive Pilot
Build Artifact: DrivePilot - App
Build Descriptor: All Tests
Duration: 5 s
Author: System Administrator

Build Errors

Type: Message:

```
ERROR BUILD FAILURE
      There are test failures.
130524-1719\sources\drivepilot\drivepilot\drivepi
(s) skipped)
```

6) Build Management - based on Maven

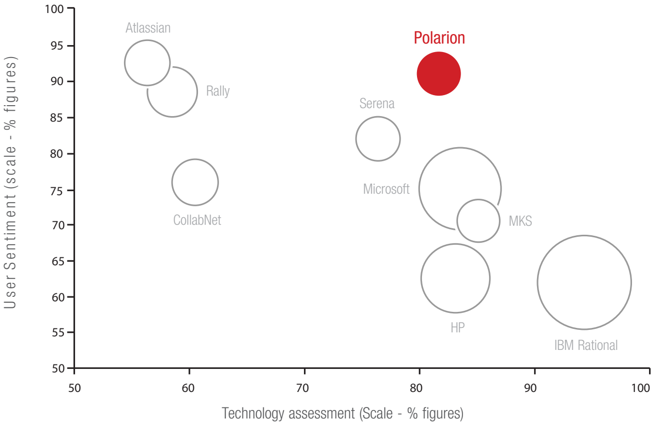


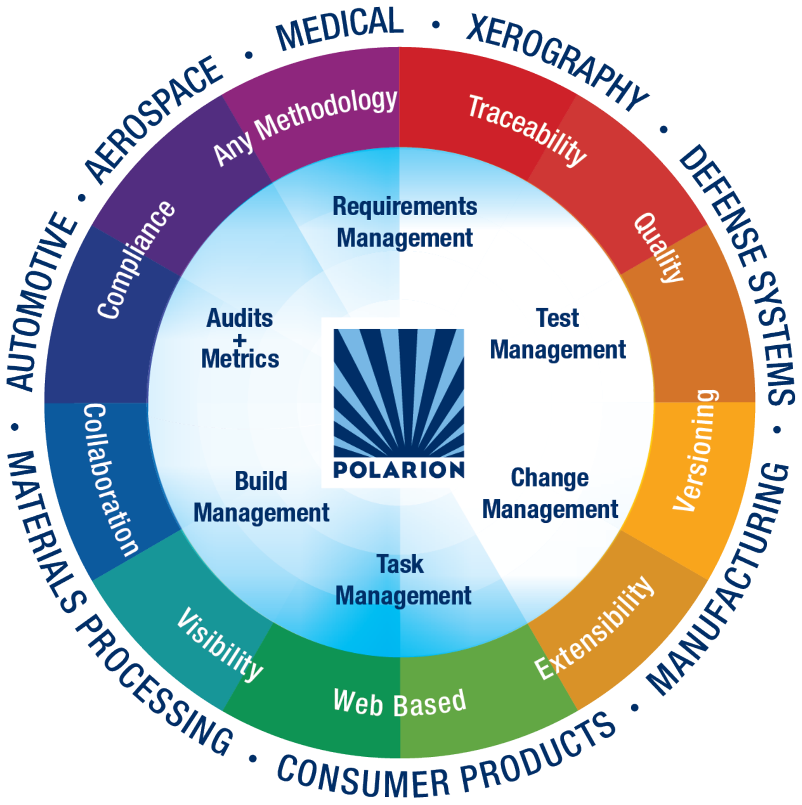
Figure 14: ALM Decision Matrix

Source: Ovum / Polarion

So is Polarion ALM the right choice for you to work with for DevOps? Are you in the initial DevOps phase or struggling with tools and whether or not you should consider to move to a MixedOps or AgileOps approach. Polarion ALM, already the perfect toolset for Development, is ready right now to become your DevOps backbone or your single and unified AgileOps toolset. It's the right choice for you... unless you are already a Polarion DevOps hero!

About Polarion Software

Polarion Software is the creator of the world's fastest enterprise scale web-based ALM solution. The success of Polarion Software is best described by the hundreds of Global 1000 companies and over 1 Million users who rely daily on Polarion's Requirements Management, Quality Assurance and Application Lifecycle Management solutions in their business processes. Polarion is a thriving international company with offices across Europe and North America, and a wide ecosystem of partners world-wide. For more information, visit www.polarion.com.



Handy Links

- Customer Testimonials
- Polarion Events & Webinars
- Contact Polarion Software