**Infosys**

**DevOps-Redefining IT Strategy**

**December, 2014**

**INFOSYS LIMITED**

**Bangalore**

|  |  |  |  |
| --- | --- | --- | --- |
| Document No. |  | Ver. Rev. : |  |
| Authorized by: |  | Signature/:  Date: |  |

**COPYRIGHT NOTICE**

©2014 Infosys Limited, Bangalore, India. All rights reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this document nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/or any named intellectual property rights holders under this document.

Infosys Limited

Hosur Road

Electronic City, 3rd Cross

Bangalore 560 100

India.

Telephone: (91) (80)28520 261-270

Fax: (91) (80) 8520 362

Website: <http://www.infosys.com>

**Author(s): B Thanigaivelan**

**Date written (MM/DD/YY):17/12/2014**

**Project Involved: BOACIMS1**

**Target readers: Developers, Operations Management, Architects, Consultants**

**Keywords: DevOps, Operations Management**

**Purpose of the document: Understanding Concept Of DevOps**

Contents

[Introduction 4](#_Toc406489051)

[What Is DevOps? 4](#_Toc406489052)

[DevOps in Detail 5](#_Toc406489053)

[Why DevOps? 5](#_Toc406489054)

[How Does DevOps “Work”? 6](#_Toc406489055)

[DevOps Skills 8](#_Toc406489056)

[Benefit from DevOps 9](#_Toc406489057)

[Use case of DevOps 9](#_Toc406489058)

[Conclusion 11](#_Toc406489059)

[Reference 11](#_Toc406489060)

## Introduction

The word DevOps was coined in 2009 by Patrick Debois. Term was formed by combining “development” and “operations,” which provides a starting point for understanding exactly what people mean when they say “DevOps.” Notably, DevOps isn’t a process or a technology or a standard.

## What Is DevOps?

“DevOps (blend of Development and Operations) is a development method that increases the value of communication, collaboration and integration between software development team and system operations team”

DevOps is from handshake of two major trends.

1. Agile Operations

2. Communication and Collaboration between development and Operations team throughout software life cycle.

“Ops” means the collective term to represent System Engineers, Operations Team, Release Managers, Database Administrators, WebSphere Admins and other sub disciplines

“Dev” is short form for Development Team but in reality it is wider with teams involved in developing the software product like Quality Analyst, Product engineers and other partner teams.

DevOps can be more practically defined as

**DevOps is the best practice of operations and development experts participating together in entire software product lifecycle from design phase through the development process to production support. All experts have at least basic understanding of other business subjects.**

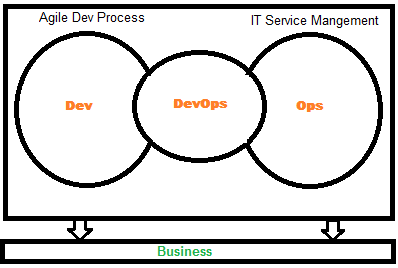
## 

## Why DevOps?

In delivering quality software to valuable customers, Dev and Ops are in conflict with each other.

Dev expects need for change as development works on new features, bug fixes etc.., and they want the changes rolled out quickly to production.

Once the Software is delivered, the Operation Department avoids changes to ensure stability.



DevOps provides solution to the Dev/Op Department conflicts. It bridges the gap between the Agile software development process and Operations.

## How Does DevOps “Work”?

DevOps has different capabilities to improve the product delivery. Following are common traits of it

* Collaboration
* Automation
* Continuous Integration
* Continuous delivery
* Continuous testing
* Continuous monitoring
* Rapid remediation

**Collaboration**

The need for collaboration is more significant while involved in delivery of software product. Lack of Communication and Collaboration will lead to increase in defect ratio. Collaboration between partner teams avoids confusion and helps to solve the issue quicker.

**Automation**

DevOps relies heavily on automation—and that means you need tools. Automation can be done for Build, Deployment, Testing, Monitoring, System Rollouts, System Configuration.

**Continuous Integration**

You usually find continuous integration in DevOps cultures because DevOps emerged from agile culture.

Continuous integration (CI) is a software engineering practice in which changes are immediately tested and report is generated for rapid feedback. Based on the feedback the code is actively triaged and corrected as quick as possible

**Continuous Testing**

Continuous Testing is practice from agile methodology. When testing is done periodically after every deployment, issue will not be extended. After the deployment of fresh code in environment, testing is done to make it easy for partner teams to work on error-free code. This process improves the speed of code movement to production environment.

**Continuous Delivery**

Continuous delivery is extension of Continuous Integration. DevOps draws the release pipeline like Independent and Integrated releases. The increase in number of continuous short term releases improves the delivery model.

**Continuous Monitoring**

As there is increase in release numbers, there’s no way to implement the kind of rigorous pre-release testing that characterizes waterfall development. Therefore, in a DevOps environment, failures must be found and fixed in real time.

In fact, some monitoring experts say that continuous monitoring must be the part of service delivery. For the purpose monitoring the tools used in production environment can be also used in development to find the performance issue before they get to production.

Two kinds of monitoring are required for DevOps:

* Server monitoring
* Application performance monitoring.

## DevOps Skills

* DevOps Tools – Ability to admin and customize.
* Scripting skills- Scripting skills to IT operation.
* Coding skills- Developer skills for automation.
* Re-engineering skills- Idea and Process to improve

## Benefit from DevOps

**Technical Benefits:**

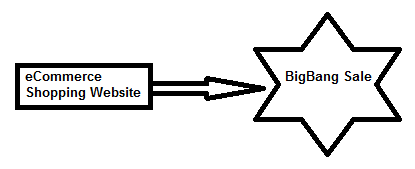
* Consistent Operation Delivery
* Continuous Software Delivery
* Infrastructure Stability
* App Deployment Speed
* Faster mean time to recovery
* Faster solution to problem

**Business Benefits:**

* Operation Cost reduces
* Shortened lead time to add new features
* Improved client service
* Time is available to add value
* Metrics Capture and improvement

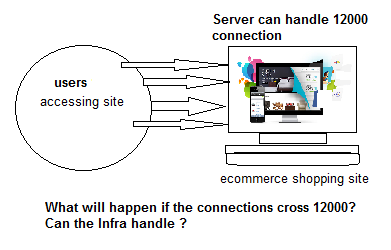
## Use case of DevOps

Let’s take example of ecommerce shopping website which is conducting a “Big Bang Sale”.



**Problem:**

Ecommerce site can handle only 12000 connections what if it exceeds the limit? How to Overcome?

****

**Solution: DevOps**

Continuous Monitoring of the system and providing the automation script for routing the users to additional server when system reaching the exceed limit. This helps to serve the customers better without making them wait in pipeline. **DevOps improves business as customer delight increases.**

Think if **no DevOps in the place** then the system would crumble as there is NO collaboration between teams, Tools to handle, Automation. Trust of business is affected which in turn slacks future business as well.

## Conclusion

From above topics discussed, it’s clearer that the use of DevOps will extend the Operation model and quality of service provided to the client. DevOps definitely improves the speed of deployment, reduces the defect ratio and save lot more time for organization which makes better delivery

## Reference

1. <http://en.wikipedia.org/wiki/DevOps>