

# مروری بر مفاهیم اولیه در DevOps



DevOps که از ترکیب دو واژه انگلیسی Development به معنای توسعه و Operations به معنای عملیات و اجرا ایجاد شده، حرکت و فرهنگی است که بر همکاری و ارتباط عمیق توسعه‌دهندگان نرم‌افزار و سایر متخصصان حوزه فناوری اطلاعات در فرایند ارائه محصول نرم‌افزاری تأکید می‌کند. (این واژه از ۲۰۰۹ به بعد رایج شده است)

این حرکت تلاش دارد تا محیطی را فراهم کند که در آن امکان ساخت، آزمایش و ارائه نرم‌افزار با سرعت بیشتر و با پایداری بالاتر تحقق یابد.

DevOps روشی برای تولید نرم‌افزار است که بر ارتباطات، مشارکت و ادغام بین تولیدکنندگان نرم‌افزار و متخصصین IT تأکید می‌کند. در واقع، واکنشی به وابستگی متقابل تولید نرم‌افزار و عملیات IT است. امروز که تکنولوژی رایانش ابری به سازمان‌ها اجازه می‌دهد منابع را به جای چند ماه در چند دقیقه فراهم کنند، وقت آن رسیده که چرخه عمر اپلیکیشن‌ها و هم به تناسب تغییر کند. واضح است که تکنولوژی رایانش ابری تغییرات بزرگی را در توسعه و استفاده اپلیکیشن‌ها وارد و ممکن ساخته است. به طور خاص، تکنولوژی ابری نیازمند ریتم به شدت تندتری از مدیریت اپلیکیشن است.

DevOps روشی است برای تحویل سریع ارزش به مشتری و از بین بردن هر نوع مسئله‌ای که باعث کندی و ناکارآمدی در (فرآیند) تحویل و بکارگیری ارزش شود.

(یک محصول نیست، بلکه روش و فرهنگ همکاری است)



## Culture

یک مفهوم فرهنگی برای کار است. تیم ها و آدم ها به هم نزدیک و تعامل بین آنها محکم باید گردد. اهداف متضاد باید همگرا شوند. تسریع در تغییرات و به خصوص برای عملیات و توسعه بهینه و مشتری مداری در فضای جدید تکنولوژی.

## Automation

مفاهیم CI و Continuous Delivery مطرح میگردند و باید فرآیند های اتوماتیک برای ان تدبیر کرد. امکان خطای انسانی به خصوص در مرحله عملیات باید به حداقل برسد. فرایند های تحویل سرویس و تحویل محصول به مشتری اتوماتیک می گردند.

## Lean

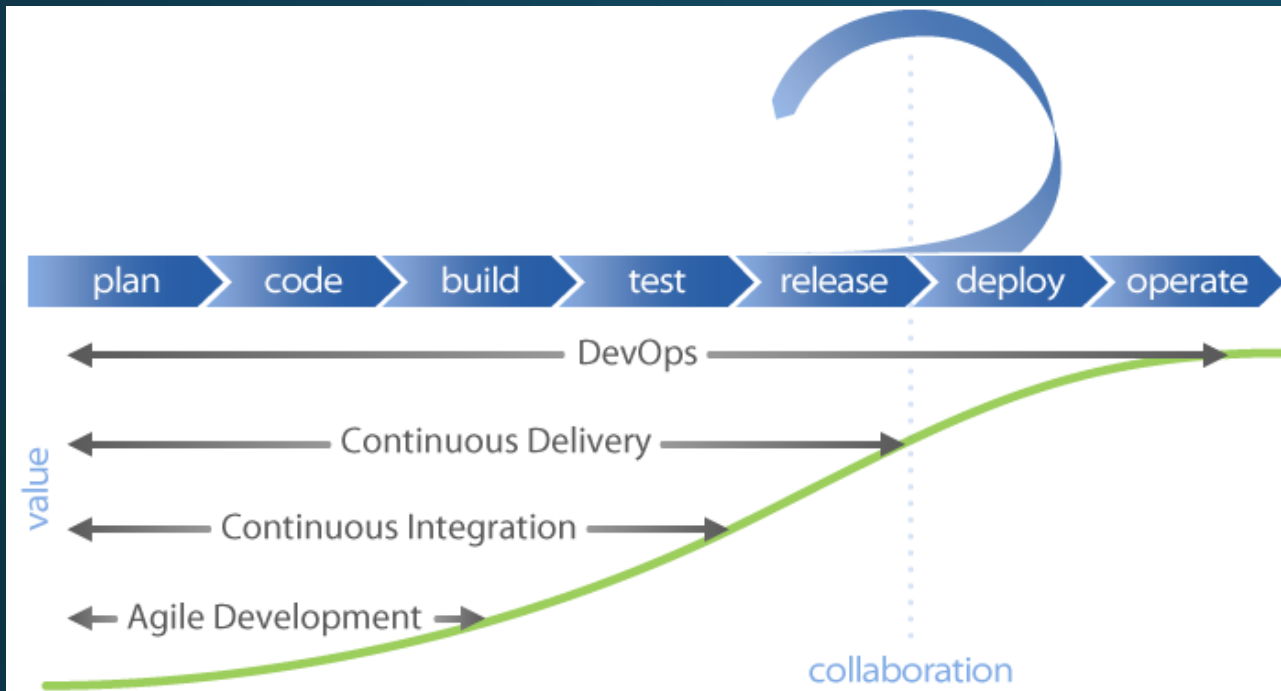
تکیه بر اصول اصلی تولید چابک و ناب نرم افزار که در اینجا، از بین بردن تمامی فرآیندها و کارهای زاید است. یعنی حذف هر ویژگی، فرآیند، فعالیتی که تولید ارزش نمی کنند. کوچک بودن تیم های توسعه، توسعه ویژگی هایی مدرن و مشتری نیاز، کم کردن دوباره کاری

## Measurement

در کلیه سطوح برای ایجاد شفافیت اندازه ها و تصمیم گیری بهتر، اندازه گیری نیاز داریم تا بتوانیم وضعیت موجود را ارزیابی کنیم. معمولاً در هر سطح نرم افزار برای نوع سرویسی در مرحله توسعه و عملیات به چنین مانیتورینگ هایی نیاز داریم و باید ایجاد کردند

## Sharing

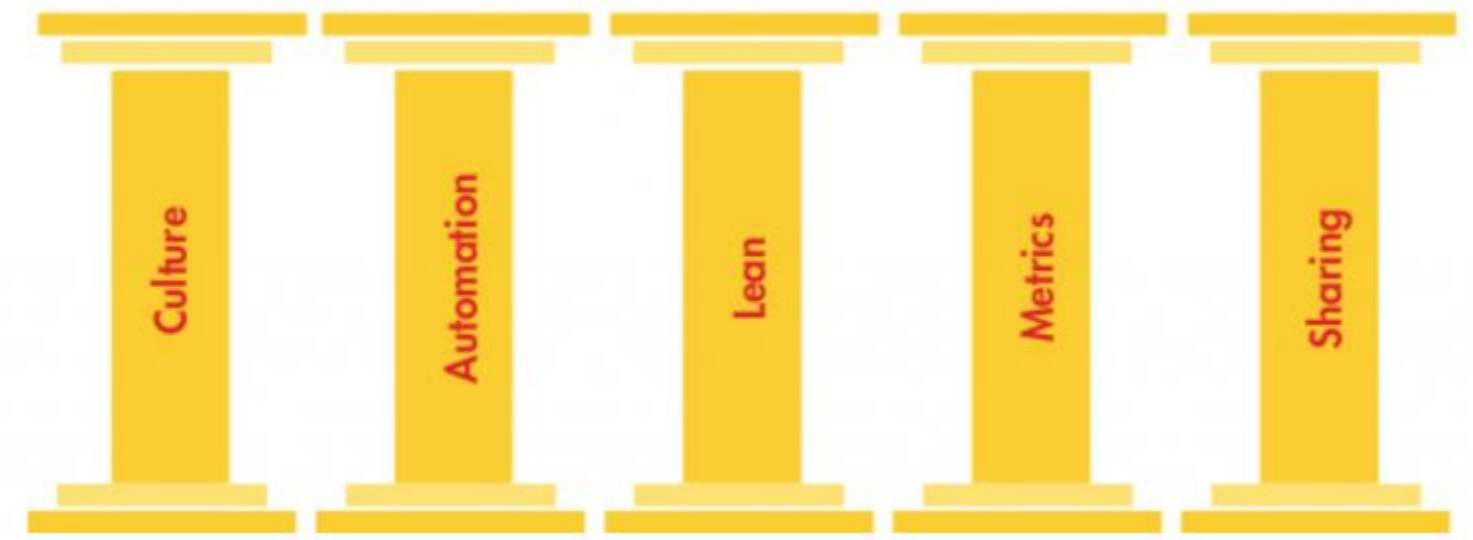
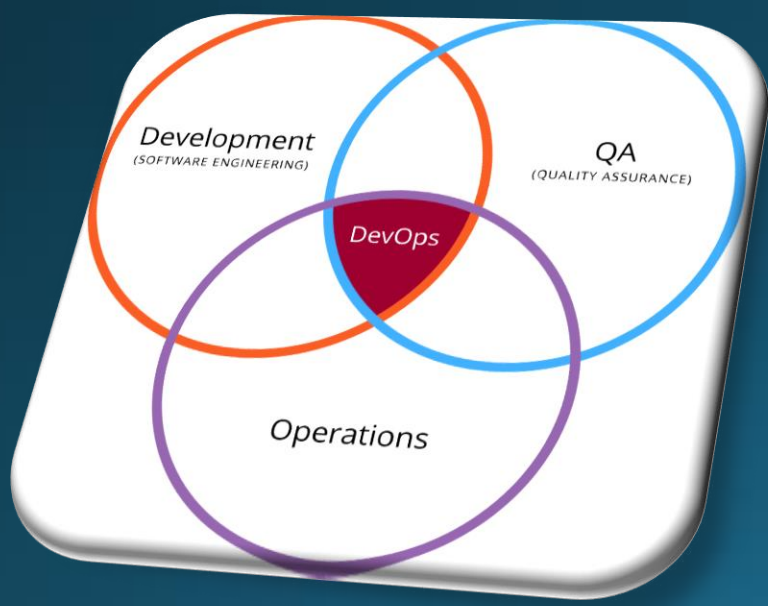
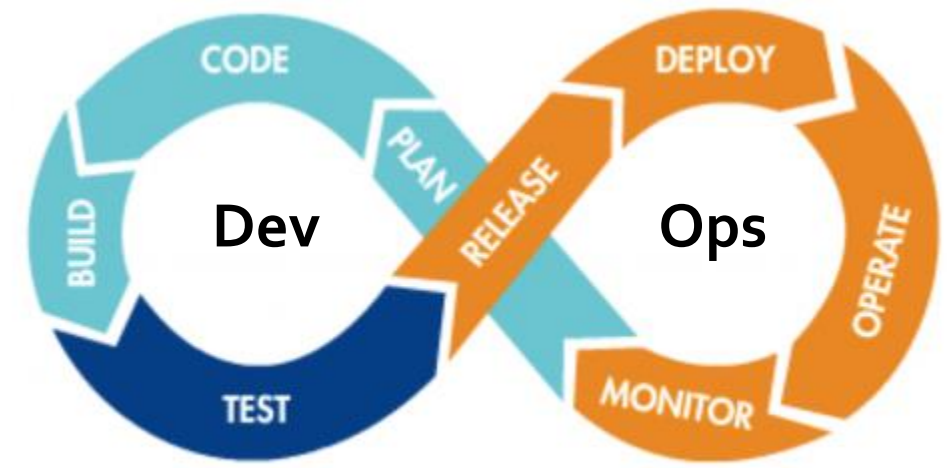
اشتراک گذاری منابع و درس های یادگرفته شده است. از اندازه گیری ها و مانیتورینگ چه درسی گرفتیم؟ پیشتر وجود دیوار مابین اعضای تیم باعث می شد این درس ها بین اعضای تیم پخش نشوند. بین سایر تیم ها نیز به صورت مفهوم هایی فضا های باز این اتفاق می افتد.



- Faster time to market; **Cost**
- Lower failure rate of new releases; **Reliability**
- Shortened lead time between fixes; **Agility, Quality**
- Faster mean time to recovery; **Quality, Agility**

مزایا

Speed  
Scale  
Delivery



## Agile

**Embrace** constant change

**Embed Customer** in team to internalize expertise on requirements and domain

## DevOps

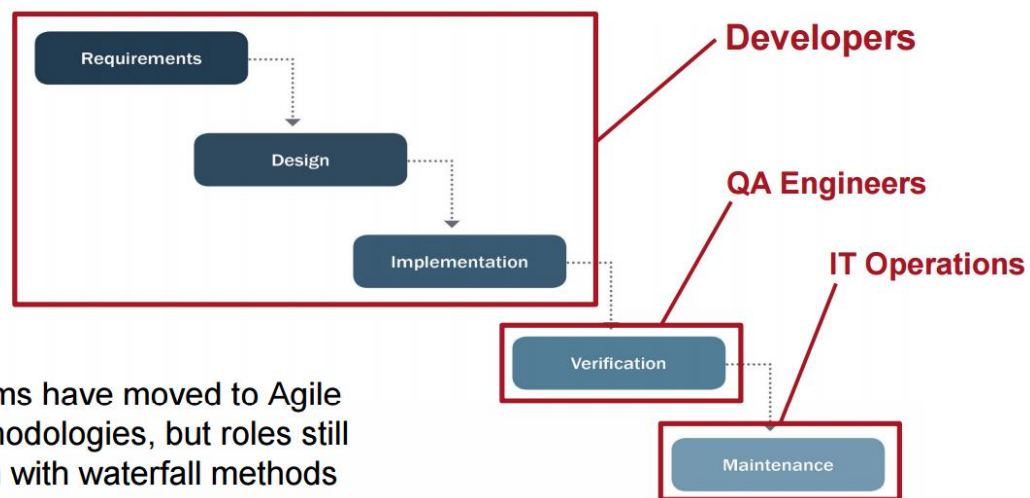
**Embrace** constant testing, delivery

**Embed Operations** in team to internalize expertise on deployment and maintenance



Many **tools** can help you achieve your **DevOps goals**

- Source Control
- Documentation System
- Build System
- Issue Tracking System
- Communication System
- Monitoring system
- Code Review System
- Integration System
- ...

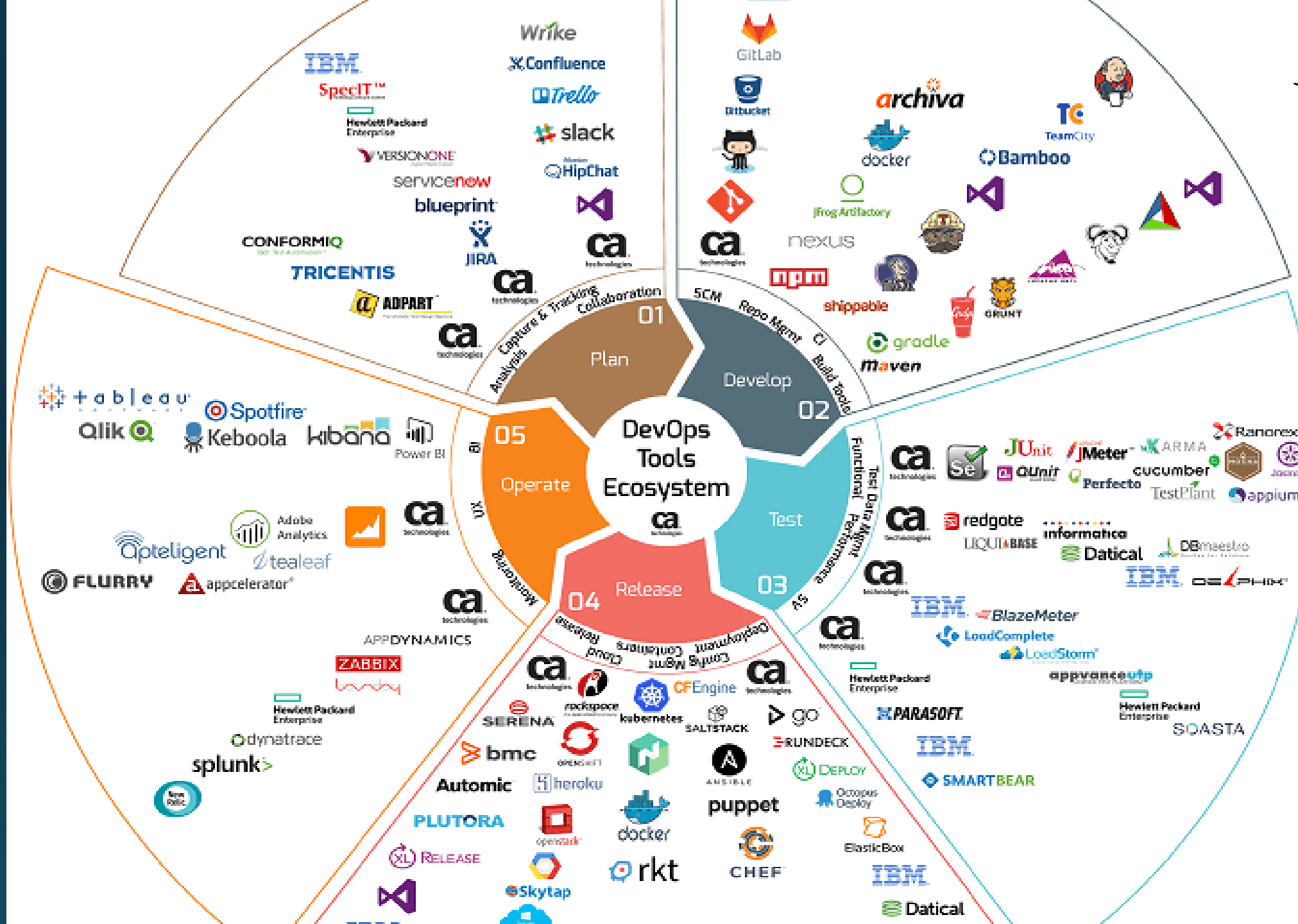


Teams have moved to Agile methodologies, but roles still align with waterfall methods



Category	Traditional Adapted from Source: (Nerur et al., 2005, p.75)	Agile Adapted from Source: (Nerur et al., 2005, p. 75)	DevOps within this research
<b>Fundamental Assumptions</b>	<i>Systems are fully specifiable, predictable, and can be built through meticulous and extensive planning</i>	<i>High-quality, adaptive software can be developed by small teams using the principles of continuous design improvement and testing based on rapid feedback and change</i>	<b>High-quality, adaptive services can be delivered per multidisciplinary end-to-end teams using the principles of continuous improvement, integration, delivery and possible deployment</b>
<b>Role Assignment</b>	<i>Individual – favors specialization</i>	<i>Self-organizing teams – encourages role interchangeability</i>	<b>Multidisciplinary teams with end-to-end responsibility</b>
<b>Communication</b>	<i>Formal</i>	<i>Informal</i>	<b>Informal</b>
<b>Customer's Role</b>	<i>Important</i>	<i>Critical</i>	<b>Critical</b>
<b>Project Cycle</b>	<i>Guided by tasks or activities</i>	<i>Guided by product features</i>	<b>Guided by delivering business value from end-to-end</b>
<b>Development Model</b>	<i>Life cycle model (waterfall, Spiral, or some variation)</i>	<i>The evolutionary-delivery model</i>	<b>Continuous Delivery Model</b>
<b>Desired Organizational Form/Structure</b>	<i>Mechanistic (bureaucratic with high formalization)</i>	<i>Organic (flexible and participative encouraging cooperative social action)</i>	<b>Learning (Interactive, flexible and collaborative people and teams)</b>
<b>Technology</b>	<i>No restriction</i>	<i>Favors object-oriented technology</i>	<b>Automation (Cloud, XaaS)</b>

## DevOps و ا





trends in:

security, (DevSecOps model)

security,(Distributed Security System)

container usage

container orchestration

business models toward release development,

IT business price auto optimization

risk avoidance,

cloud orchestration,

detects the root cause of modern software issues

visibility

microservice auto integration/improvement/transformation

serverless on the fly containers

.....