

Session 1414: DevOps and Cloud Service Management and Operations Mini-Bootcamp

**Pam Geiger
Jim Palistrant**

Bluemix Learning & Skills

**InterConnect
2017**



Please note

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.

The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Agenda

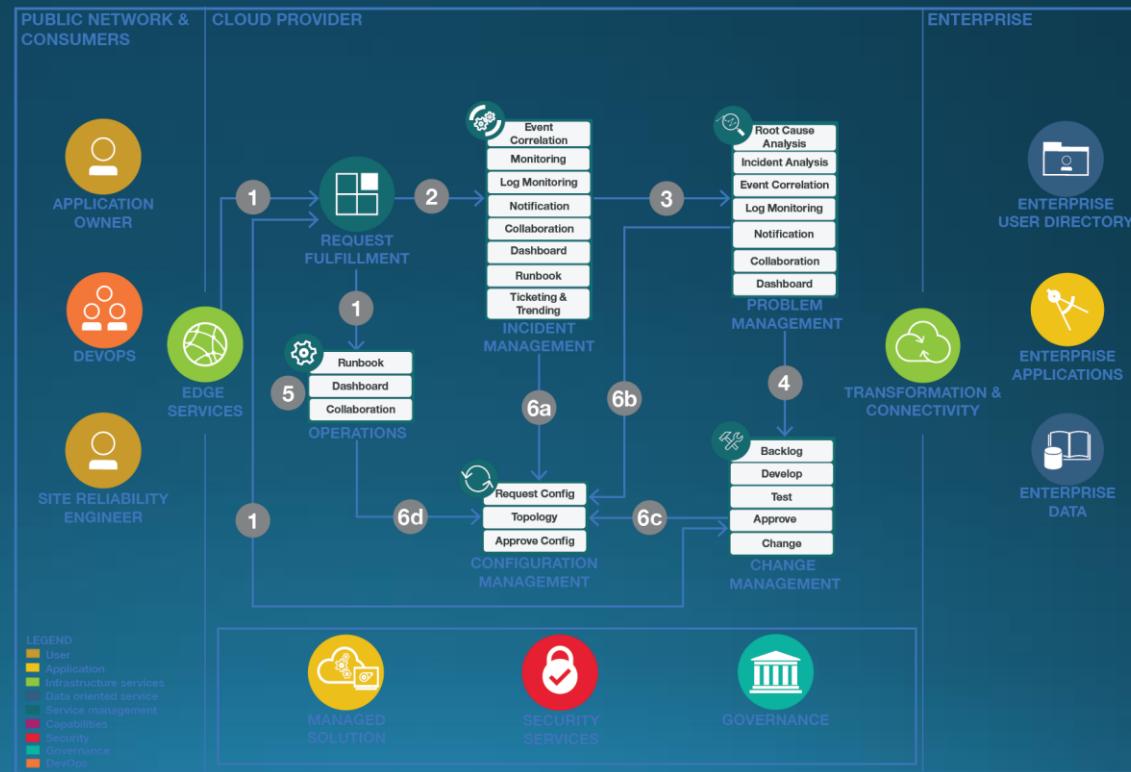
- DevOps and Cloud Service Management and Operations (CSMO) Overview
- CSMO Bluemix Garage Offering
- Build to Manage

DevOps and CSMO Overview

Cloud Service Management and Operation

- [Operations Transformation](#)
- [Incident Management](#)
- [Problem Management](#)
- [Change Management](#)
- [Dashboarding & Reporting](#)
- [Operations](#)
- [Compliance](#)
- [Build to Manage](#)
- [Release Management](#)
- [Configuration Management](#)
- [Service Continuity](#)
- [Performance & Capacity Management](#)
- [Service Level Management](#)
- [Financial Management](#)
- [Business Performance](#)
- [Continued Service Improvement](#)

- Cloud Service Management and Operations refers to the entirety of activities – organized in persona, processes, and tools – that are performed by an organization to plan, design, deliver, operate and control IT and Cloud services offered to customers..
- Assure Availability and Performance of applications running on the IBM Cloud platform, given the target SLA of 99.99% availability for applications
- Reduce mean-time-to-repair to restore service as quickly as possible
- Managing and controlling operational risks and threats



Cloud Service Management – Where to start ?



- **Core**

These are the core practices of Cloud Service Management and Operations that most companies have to follow right from the beginning of the transformation to cloud. These practices assure availability and visibility of services.

An example is Incident that rapidly restores the service, while Problem Management identifies the root-cause and prevents the reoccurrences of these incidents.

- **Scale**

As the cloud adoption increases, these practices need to be followed to scale with the growing demand. They align parallel activities performed by other entities with the needs of Service Management.

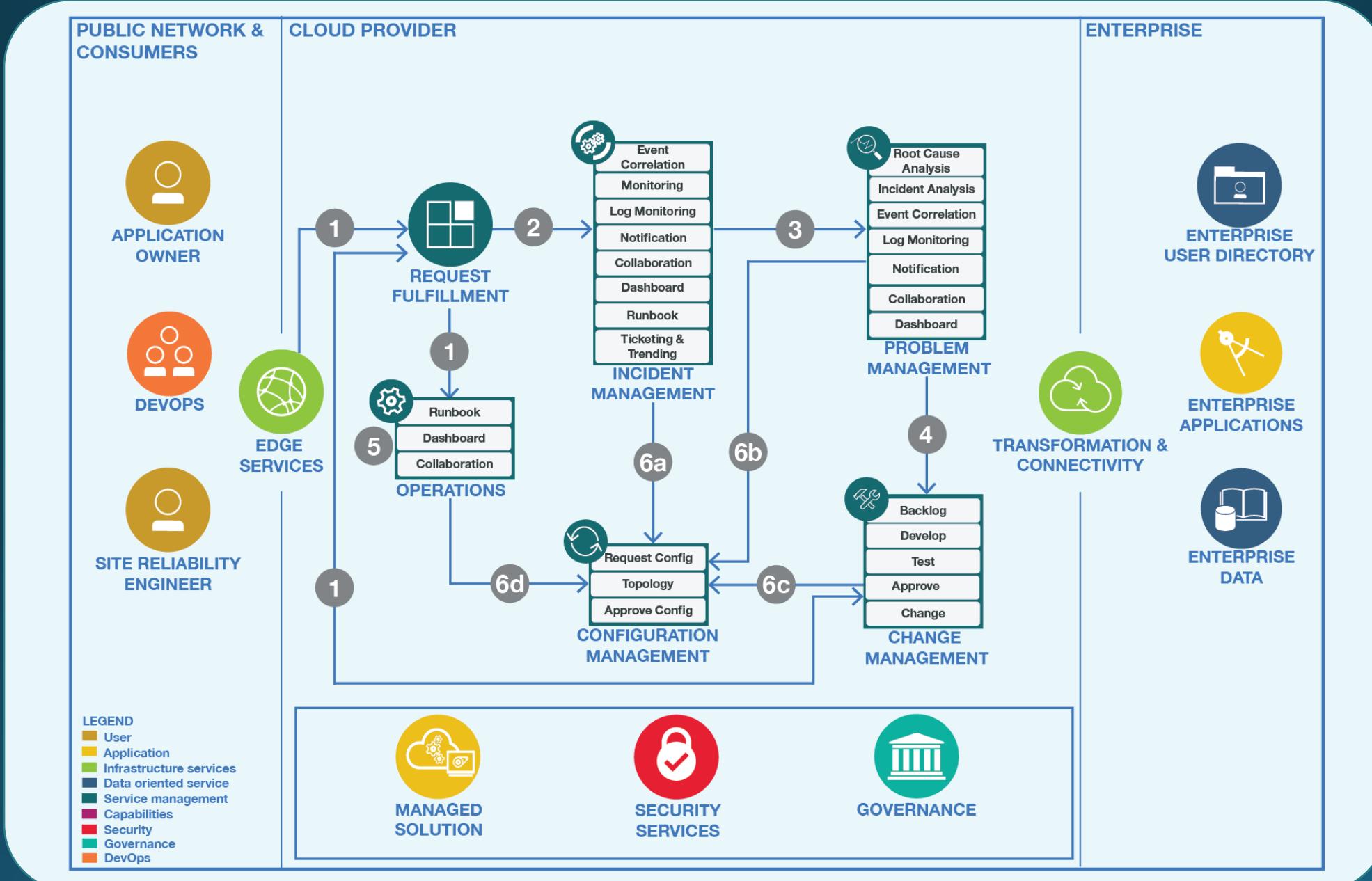
For instance, Developers build manageable micro-services.

- **Service-orientation**

These practices add a service-oriented view and business-focus to Service Management. Activities are now strongly aligned with SLAs and business KPIs.

As an example, business performance information is visualized in dashboards and used to prioritize operational activities.

Cloud Service Management and Operation



Operations Transformation

- Optimize Service Management through DevOps culture and Cloud Computing
- Change in mindset:
 - Flexibility before Efficiency; IT philosophy is „Yes“; Decentralized Governance
 - Accept failures as fact, need for graceful recovery
- New Practices
 - Automation everywhere
 - Highly collaborative operations, “ChatOps”
- Traditional Roles change: DevOps, EnvOps, First Responder, Site Reliability Engineers
- Processes change: from Change & Release Management to Continuous Delivery

Practices

 The Circuit Breaker pattern ★★★★☆ (1 Review)

In a cloud application with many interdependent microservices, use of the Circuit Breaker pattern ensures that a single failure does not take down an entire service.

 Auto-scaling applications ★★★★☆ (1 Review)

Use auto-scaling for your applications to ensure that enough resources are available at peak times and to save costs by reducing the allocated resources during times of low usage.

 Automated monitoring ★★★★☆ (1 Review)

Automated monitoring is the most reliable way to ensure high availability of your applications and reduce downtime.

 Fast recovery ★★★★☆ (0 Ratings)

Continuous availability requires a new approach to how applications are developed and deployed. You can use techniques to manage non-disruptive change.

 Resiliency ★★★★☆ (0 Ratings)

Traditionally, test scenarios are based on typical product usage patterns, but to expose software flaws, you also have to gracefully handle unstable or unpredictable hardware and test the solution's resiliency to failures.

 Chaotic testing ★★★★☆ (0 Ratings)

Continuously improve stability and recovery time of your production environment by injecting ongoing random failures.

 High availability ★★★★★ (3 Reviews)

To maintain 99.999% availability, the IBM® Bluemix® Garage Method team practices development, monitoring, and operations techniques for high availability.

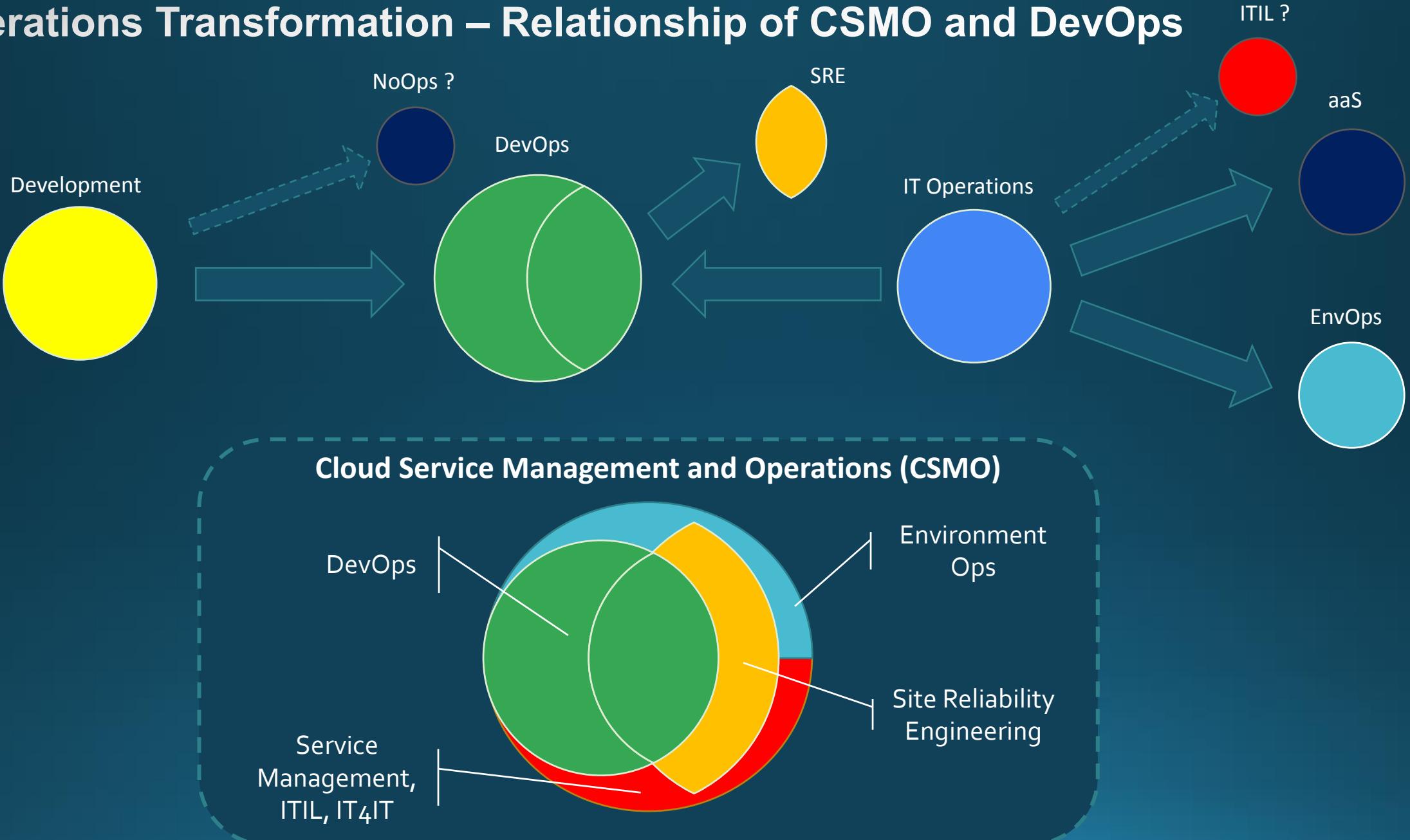
 Operations automation ★★★★☆ (0 Ratings)

Operations is a costly business. Automation enables you to reduce costs and focus your valuable staff on higher value tasks.

Operations Transformation – Organizational Change



Operations Transformation – Relationship of CSMO and DevOps



Operations Transformation – Roles in Service Management

First Responder

- Receives the alerts via collaboration tooling
- Research to determine if the problem real
- Reviews known/existing issues to determine if outage is a known issue
- Tries and resolves the issue via the prescribed scripts (runbooks)
- Gain concurrence from the customer it is resolved
- If not solved, assignment to an incident owner occurs
- Collaborates with Customer if more details are needed for problem resolution

First Responder are not the typical Level-1 support. Their responsibility and skill is rather a Level 2.5 support

Incident Owner

- Receives the incident information
- Collaborates with SMEs to restore service
- Updates key stakeholders with status and expected resolution times
- Interfaces and works with vendors to isolate problems and drive resolution
- Collaborates with 1st responder if more information is needed from customer
- Brings the incident to closure
- Opens Problem ticket for analysis and acceptance as a Problem

SME / Incident Specialist

- Investigates the problem in Monitoring tools for more details.
- Inspects logs
- Tests and verifies issues
- Recommends fixes if missing instruction for 1st Responder / Fixes the problem
- Proposes changes if updates needed, Requests Change Management
- Implements change
- Provides data for SRE review

Site Reliability Engineer (SRE)

- Ensuring a Durable Focus on Engineering
- Full access to and authority to fix, extend, and scale the code to keep it working and harden it against threats.
- Pursuing Maximum Change Velocity Without Violating a Service's SLO
- Focused on reliability, scalability and development of the Cloud infrastructure
- Eliminate or reduce the losses through root cause analysis
- Identifying Trends

Fundamentally, it's what happens when you ask a software engineer to design an operations function.

Availability Manager (AVM)

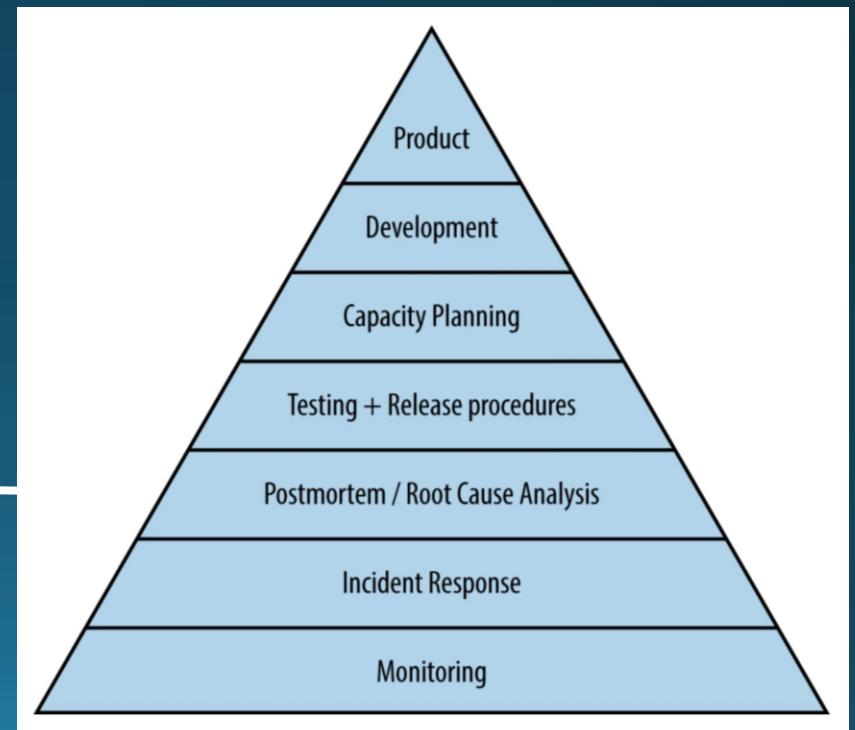
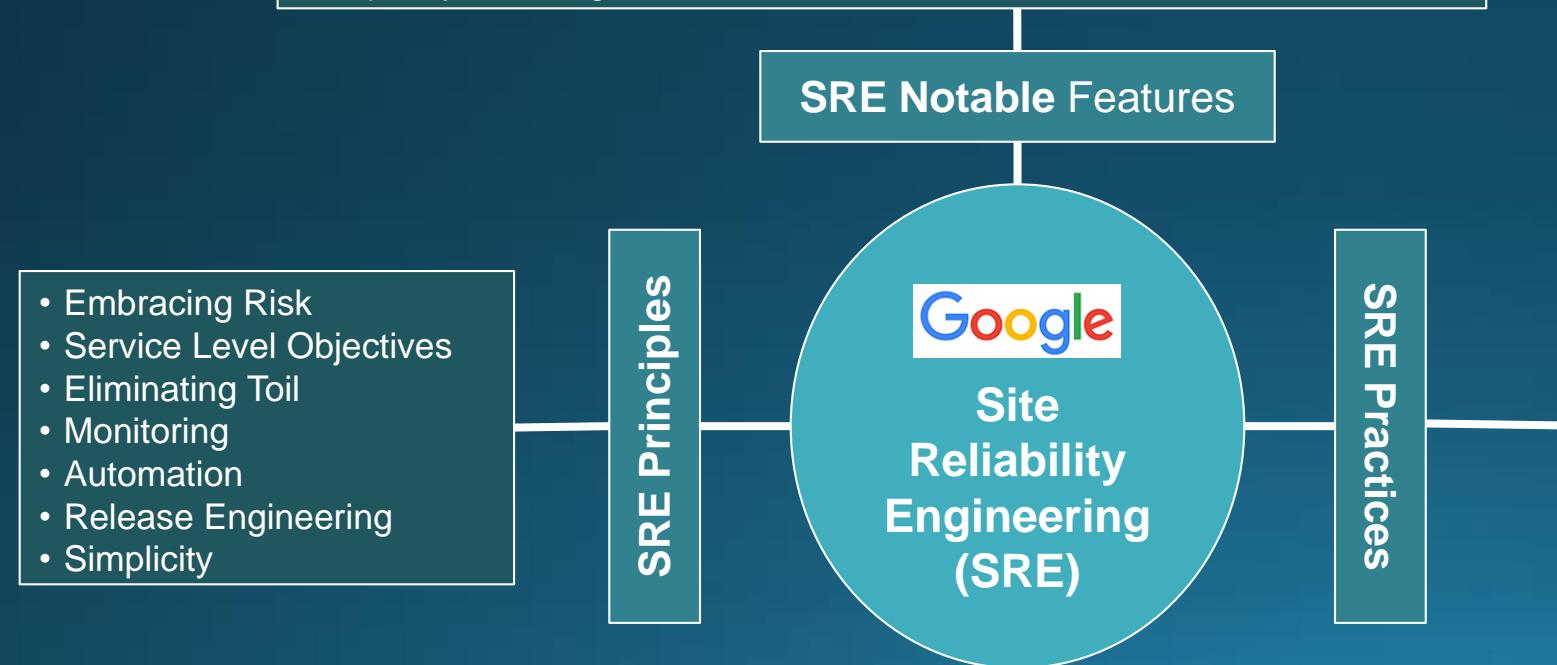
- Responsibility for the management overall incident and restoration of service, including approving changes and obtaining additional resources. In addition, this role is also responsible for driving the Root Cause Analysis thru to completion.

What is Site Reliability Engineering (SRE) ?

“Fundamentally, it’s what happens when you ask a software engineer to design an operations function.”

- Ops to scale with load through Automation
- 50% time spend on toil - 50% on engineering projects (improvements)
- Error budget
- Actionable alerts
- 4 Golden Signals: Latency, Traffic, Errors, Saturation
- Change management:
 - Progressive rollouts
 - Problem detection
 - Change rollbacks
- Capacity Planning

SRE is a highly automated service management approach, designed for supporting products & services running at global scale in a homogeneous cloud environment



Operations Transformation – Bridging from Traditional IT Management to Cloud

- 1 - Re-invent ITIL for cloud native
- 2 - Modernize ITIL for cloud enabled
- 3 - Integrate across to enable a seamless experience that results in confidence

Traditional Operations Process Driven (e.g ITIL)

Service Transition

- Change
- Release Service Operations
- Monitoring / Event
- Incident
- Problem
- Service Request
- Security and Compliance

Service Strategy

Service Design

Service Improvement

DevOps Tool Chain



For Example

Build monitoring scripts

Deploy scripts

Monitor



Still need to accomplish the same goals – but the method is different.

Embedded and automated

Shared Responsibility

What used to be done purely in Operations, is now accomplished across the DevOps lifecycle

Information needs to be integrated across both old and new

Cloud Service Management and Operations offering in the Bluemix Garage

The Bluemix Garage methodology drives transformational change using design thinking.

- **Deliver Operations for New Bluemix Development Culture**
- **Harden Applications for Production Availability**
- **Increase Service Management Efficiency**

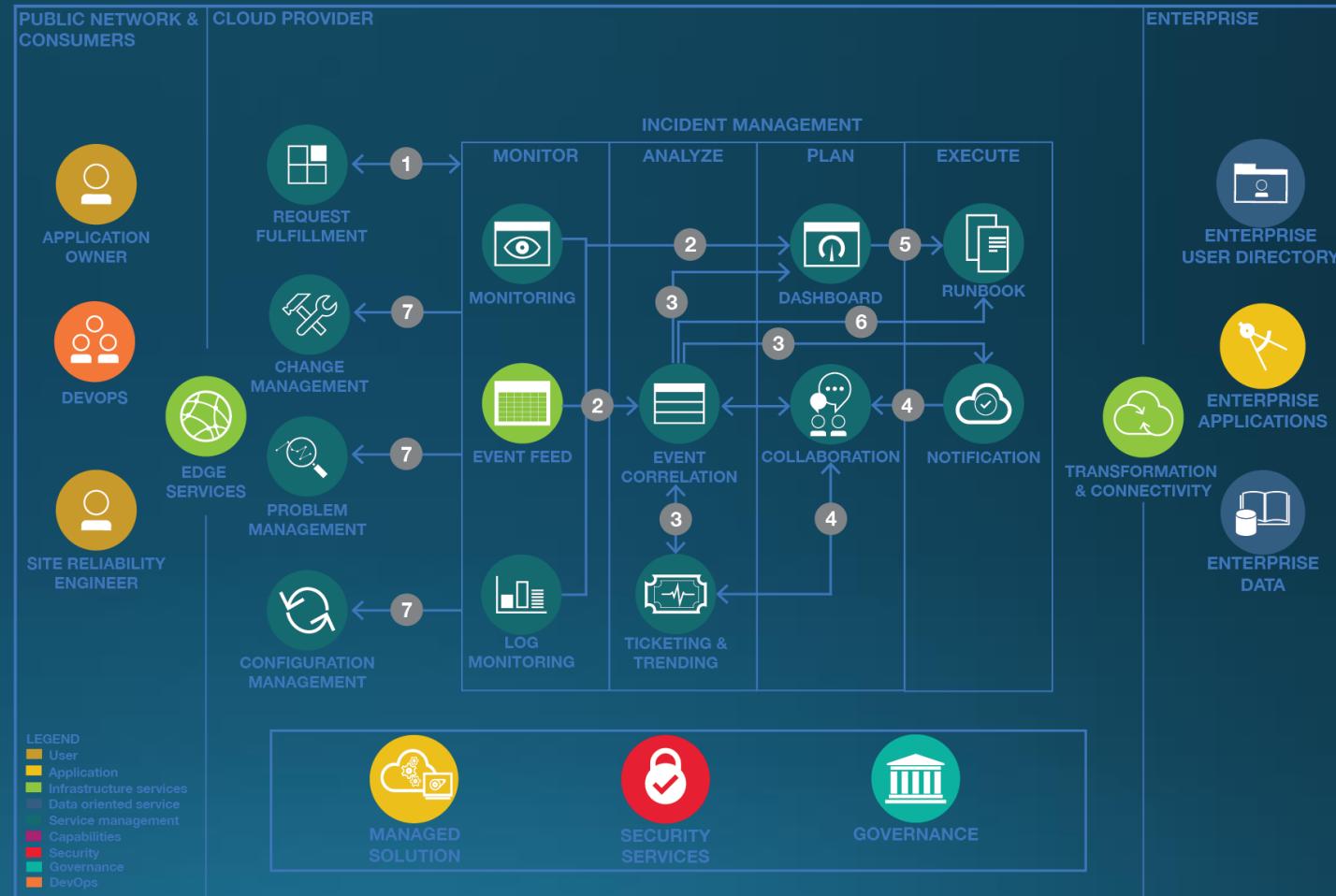
The CSMO Offering addresses the **specific needs of the Operational side**, just like the Bluemix Garage does for the Development side. Plus, we will enable cross-fertilization and integration between these offerings.

- **Visit** : Overview, live showcases, dialog and success stories
- **Workshop** : Uses design thinking to define personas, needs and transformational needs
- **MVP** : Assist and partner to develop a working solution that begins the transformation
- **Transformation** : Provide leadership and guidance in extending the MVP in transforming operational capabilities and processes

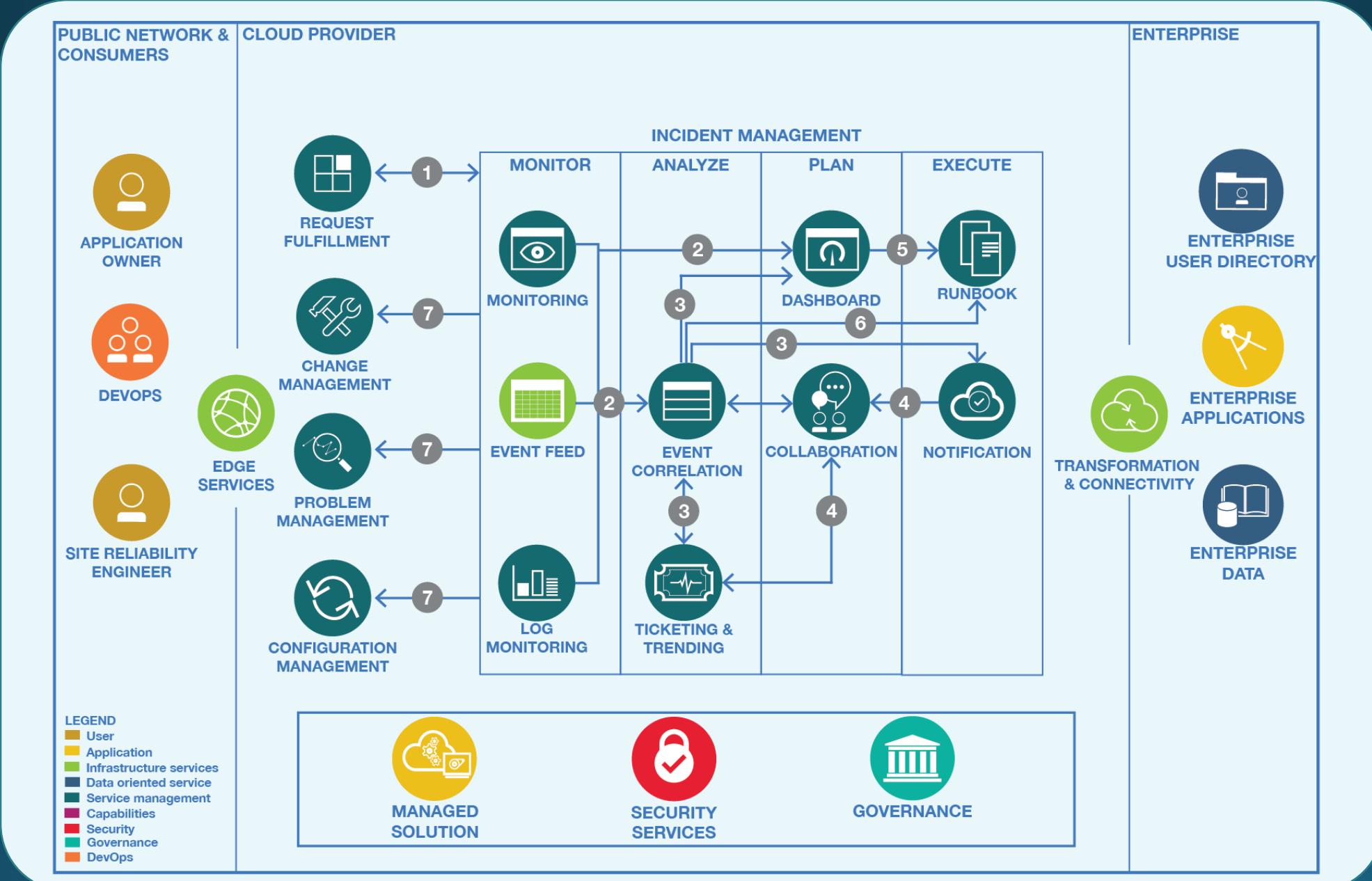


Incident Management

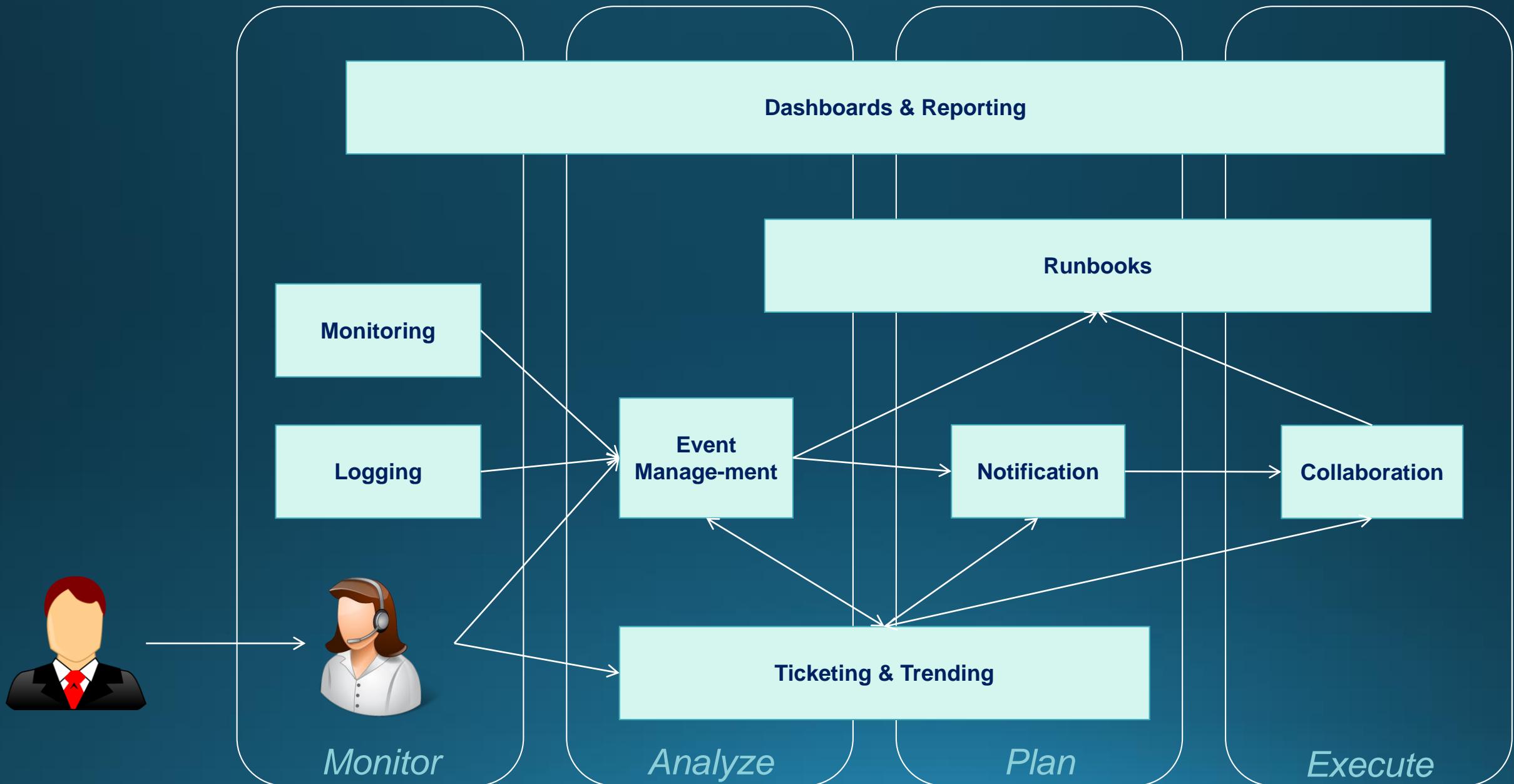
- Assure Availability and Performance of applications running on the IBM Cloud platform, given the target SLA of 99.99% availability for applications
- Reduce mean-time-to-repair to restore service as quickly as possible
- Managing and controlling operational risks and threats



Incident Management



Incident Management Tool Chain



Incident Management ChatOps

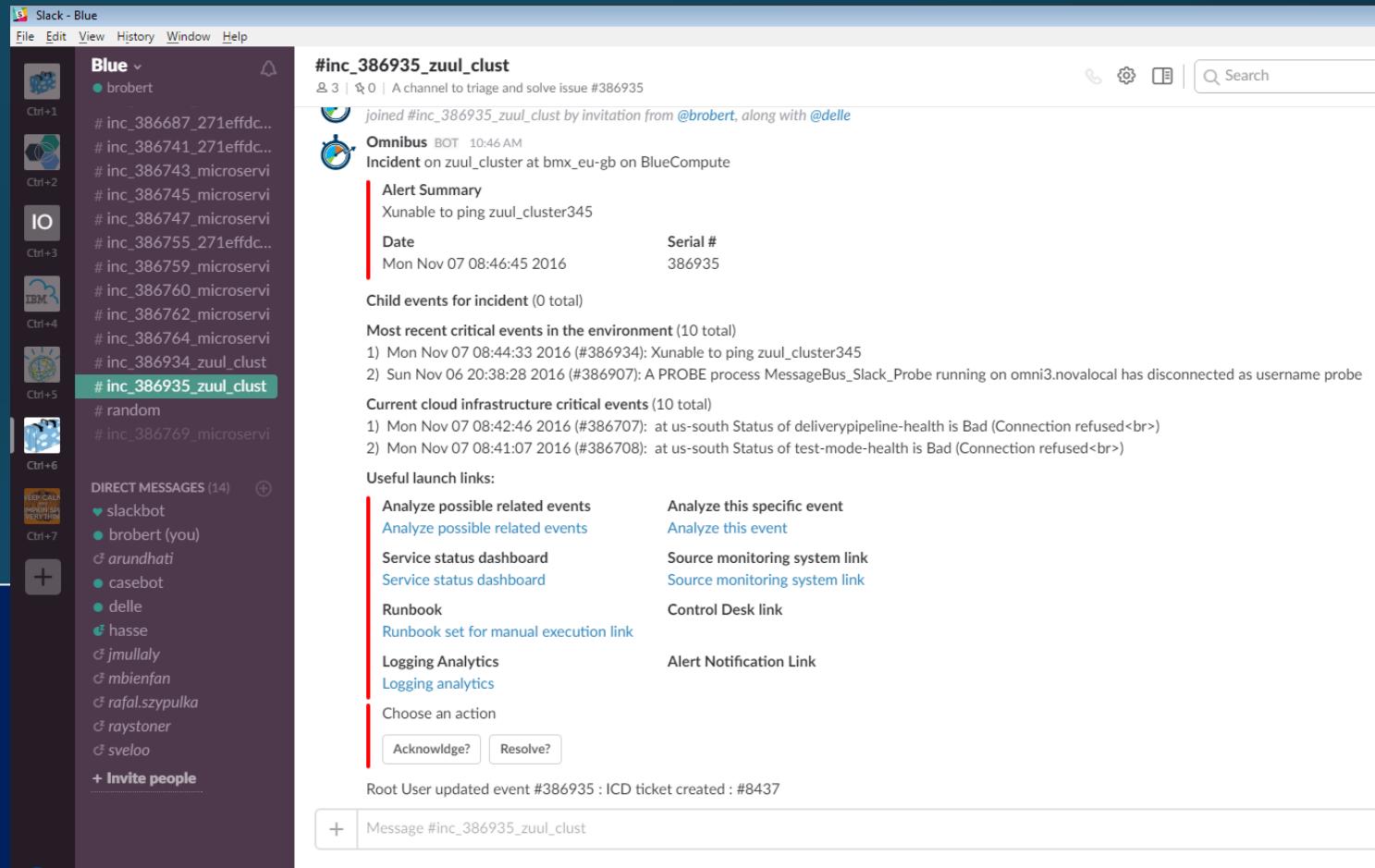
Rapid restoration of service through collaboration

Instant Collaboration between SMEs ...

- Various Operations roles
- Developers
- Vendor / Provider

... and between Humans and Applications (through BOTs)

```
noi pull - retrieve the event from NOI
noi ack - acknowledge the event
noi deack - de-acknowledge the event
noi ticket - open an ICD ticket for the event
noi ans send - open an ANS alarm for the event
noi journal - add a journal entry to the event
noi resolve - resolve event (set severity to 0)
noi critical - re-raise event (set severity to 5)
noi sev # - set the severity of the event to #
noi rba - retrieve the Runbook associated with the event
noi list - show the latest events in the ObjectServer
```



Incident Management

Collaboration with IBM Support through IBM Care

Integrating IBM Cloud Support for an seamless Client Experience
Simplified communication; chat-like “Care sessions”

Client View

The screenshot shows a chat interface for 'IBM Care'. The top navigation bar includes 'IBM Care', a user icon, and a search bar. The main area has tabs for 'Team' and 'Sub-Categories' (set to 'Credentials'). A sidebar on the left lists 'Asset Reuse Manager', 'Bluemix > Data & Analytics > Cloudant', and 'Recommended Assets'. The main pane shows a conversation between 'Vivek Grover' and 'Paul Bullis'.

- Vivek Grover: 'Can you provide the API key and username that you have configured for the Cloudant connection?' (Sep 21, 2016 1:54:48 PM)
- Paul Bullis: 'My API Key is "1234567890111213" and my password is "test1234"' (Sep 21, 2016 1:57:06 PM)
- Vivek Grover: 'Thanks! I see that your API key does not match the key in your account. Your API Key for Cloudant is "1234567890111214".' (Sep 21, 2016 1:58:41 PM)

Annotations:

- A callout points to the first message from Paul Bullis with the text: 'Sessions start with guided support, based on problem description, we use our cognitive tools to return associated assets so clients can begin self-diagnosing'
- A callout points to the second message from Paul Bullis with the text: 'Clients can provide instant feedback to support on asset quality from within a Care Session'
- A callout points to the third message from Vivek Grover with the text: 'Contributions to the session as simple as a text message'

IBM Support View

The screenshot shows the 'IBM Care' interface from the support perspective. It displays a 'Service Request Details' card for a request titled 'Unable to connect to Cloudant from NodeJS'. The card includes a 'Description' section with a purple background, 'Customer' information, 'Problem Summary', 'Resolver', 'Severity', and a 'Watson Tone Analysis' section.

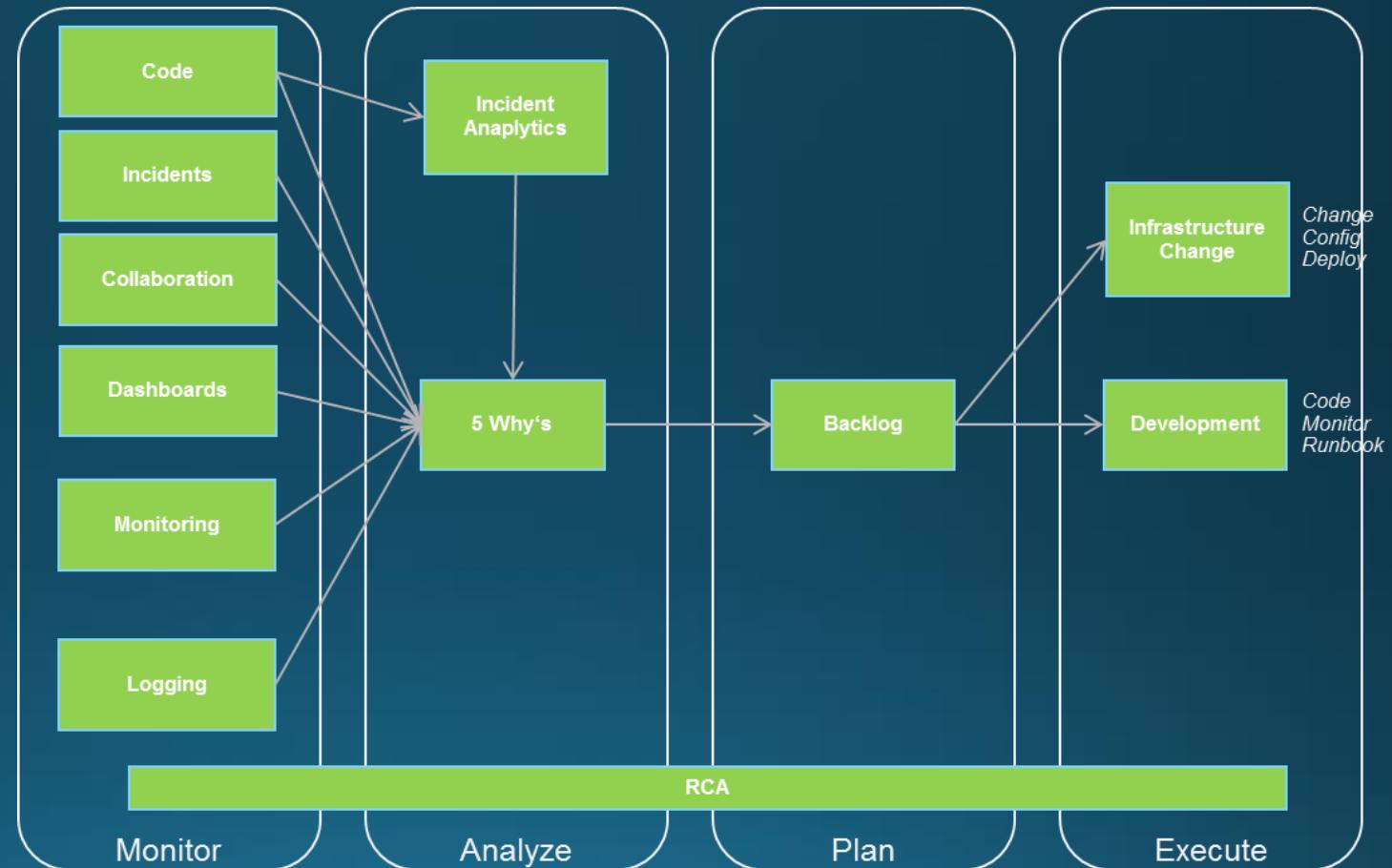
Annotations:

- A callout points to the 'Description' section with the text: 'Agents have a rich set of features available to assist clients and expedite resolution'
- A callout points to the 'Customer' section with the text: 'Agents see the assets that the clients already applied in the session history'
- A callout points to the 'Problem Summary' section with the text: 'Agents review the client details, problem summary and severity'
- A callout points to the 'Watson Tone Analysis' section with the text: 'Agents understand client temperature throughout the engagement'
- A red circle highlights the 'Write your message here...' input field at the bottom right.
- A callout points to the 'internal' comment area with the text: 'Agents can add "internal" only comments to the session at any time'

Emotion Tone	Score	Likelihood
Anger	0.29	UNLIKELY
Disgust	0.28	UNLIKELY
Fear	0.84	LIKELY
Joy	0.04	UNLIKELY
Sadness	0.50	LIKELY

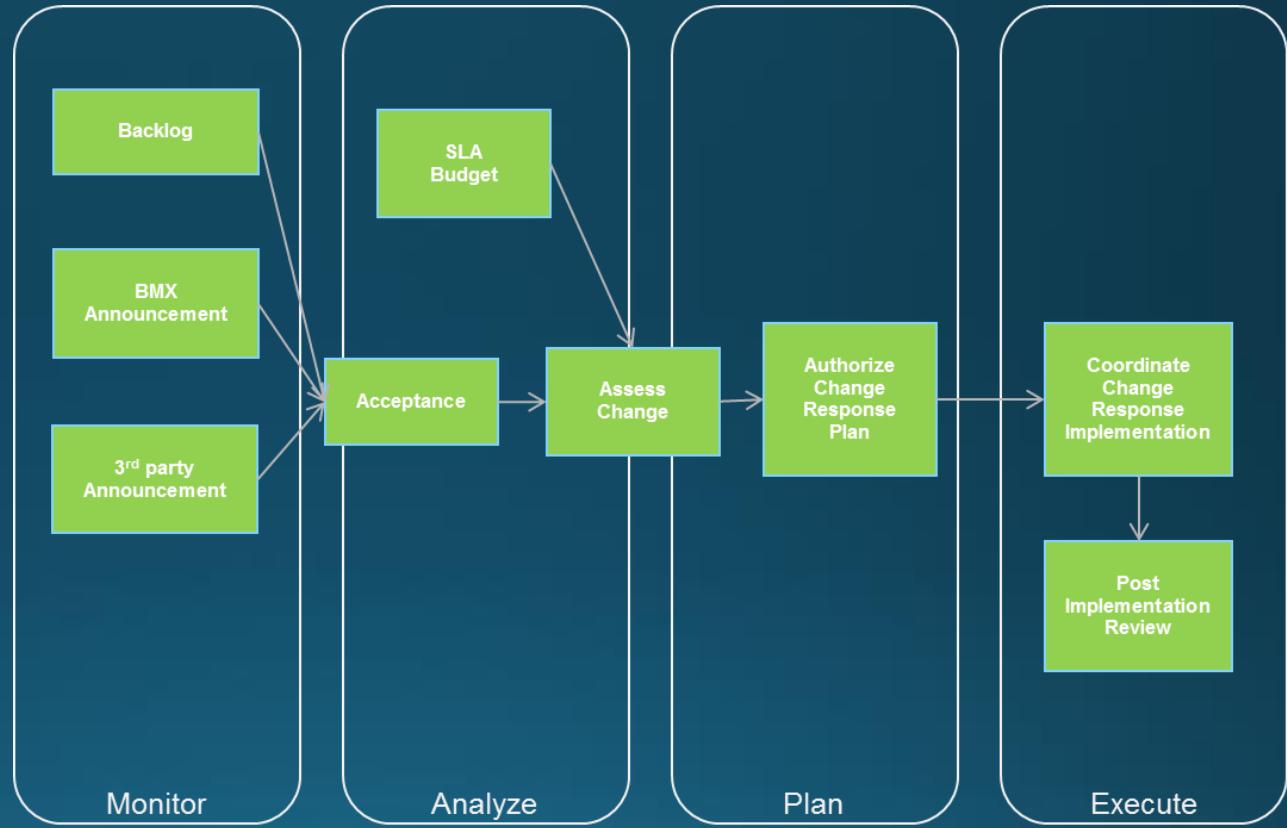
Problem Management

- Identify and resolve the root cause of service disruptions
- Prevent Incidents from happening again.
- Find trends in incidents, group those incidents into problems, identify the root causes of problems, and initiate change requests against those problems.
- Continued Service Improvement



Change Management

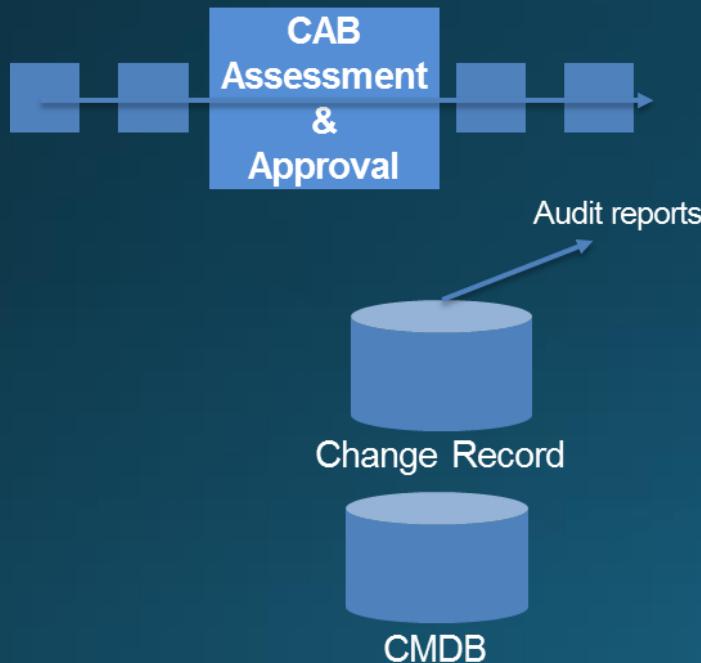
- The purpose of the Change Management process is to achieve the **successful introduction of changes** to an IT system or environment. Success is measured as a balance of the **timeliness and completeness of change implementation**, the cost of implementation, and the **minimization of disruption** caused in the target system or environment.
- Automation is used to
 - Implement progressive rollouts
 - Quickly and accurately detect change-related problems („deployment marker“)
 - Roll back changes safely when problems arise
- Scheduling of Changes affected by current „Error Budget“



Change Management – Processes Change

Traditional IT

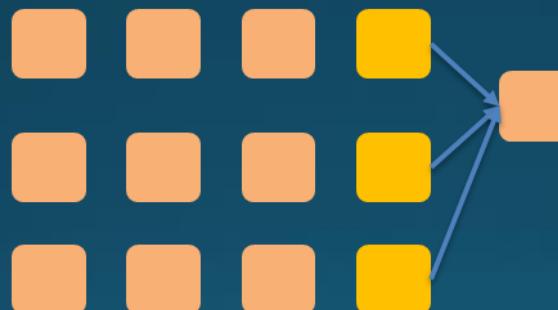
Manual Change & Release
Some automation



Cloud-Enabled IT

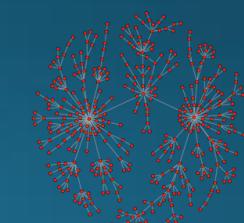
Automated build / deploy
of VMs & Containers

Stage Gates, Co-ordinated Releases



Cloud-Native IT

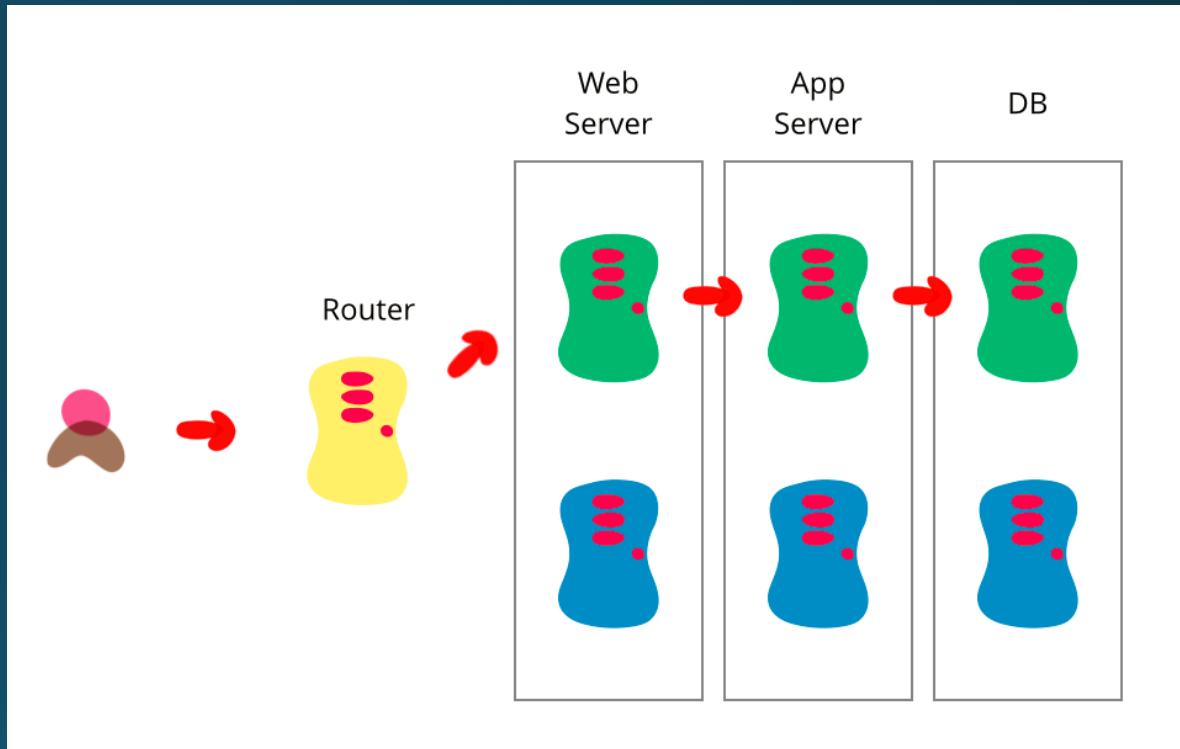
Continuous Integration
Continuous delivery to production
Cloud-native runtimes (node.js)
Pipeline per microservice



Topology / Relationship Graph

Change Management – Blue-Green Deployment

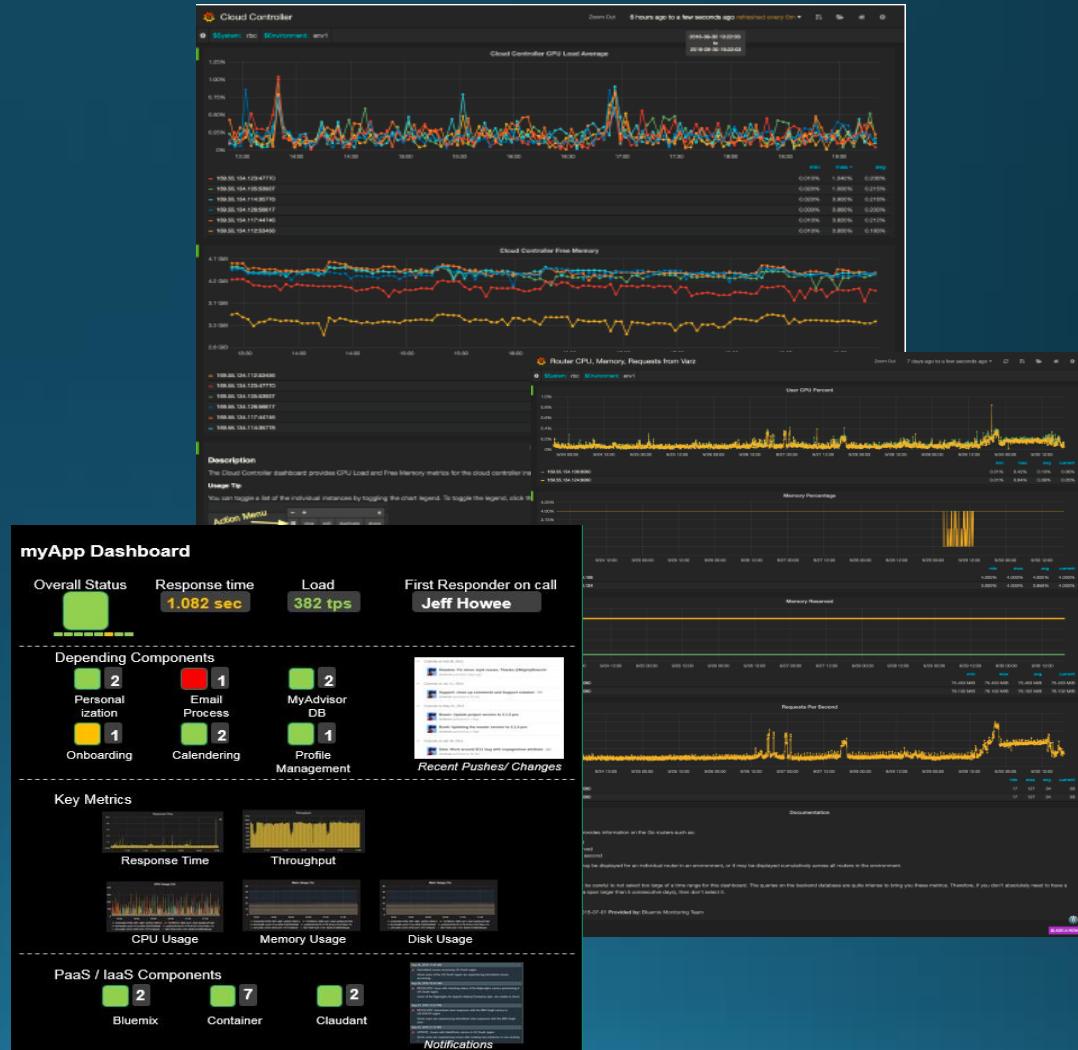
Blue-green deployment also gives you a rapid way to **rollback** - if anything goes wrong you switch the router back to your blue environment.



Note on databases: Separate the deployment of schema changes from application upgrades. So first apply a database refactoring to change the schema to support both the new and old version of the application, deploy that, check everything is working fine so you have a rollback point, then deploy the new version of the application.

Dashboarding & Reporting

- Dashboards provide a summary view of an application's core metric and key performance indicators (KPI). They also show *relevant* information on Platform Health. Reports harden the information of broader consumption.
- Dashboards don't replace dedicated UIs, but rather federate information from multiple places into a single console.
- Dashboards are application-specific and persona-driven (not a one size fits all), information shown should be actionable.
- SRE teams are typically event driven and avoid the situation that requires someone to stare at a screen watching for problems. Still, SRE teams use a number of dashboards to visualize the health and status of their Cloud environments.
- Typical Uses for dashboards
 - Event Aggregation / Correlation
 - Analyzing Long-Term Trends
 - Comparing over time or experiment groups
 - Collecting ad-hoc retrospective analysis
 - Mash-up of disparate sources to find correlations
 - Visualizing recent changes
- 4 Golden Signals: Latency, Traffic, Errors, Saturation
- Deployment Markers



Dashboarding & Reporting – First Responder



Dashboarding & Reporting – SRE

Site Reliability Eng Mock v1

Zoom Out Last 30 minutes Refresh every 10s

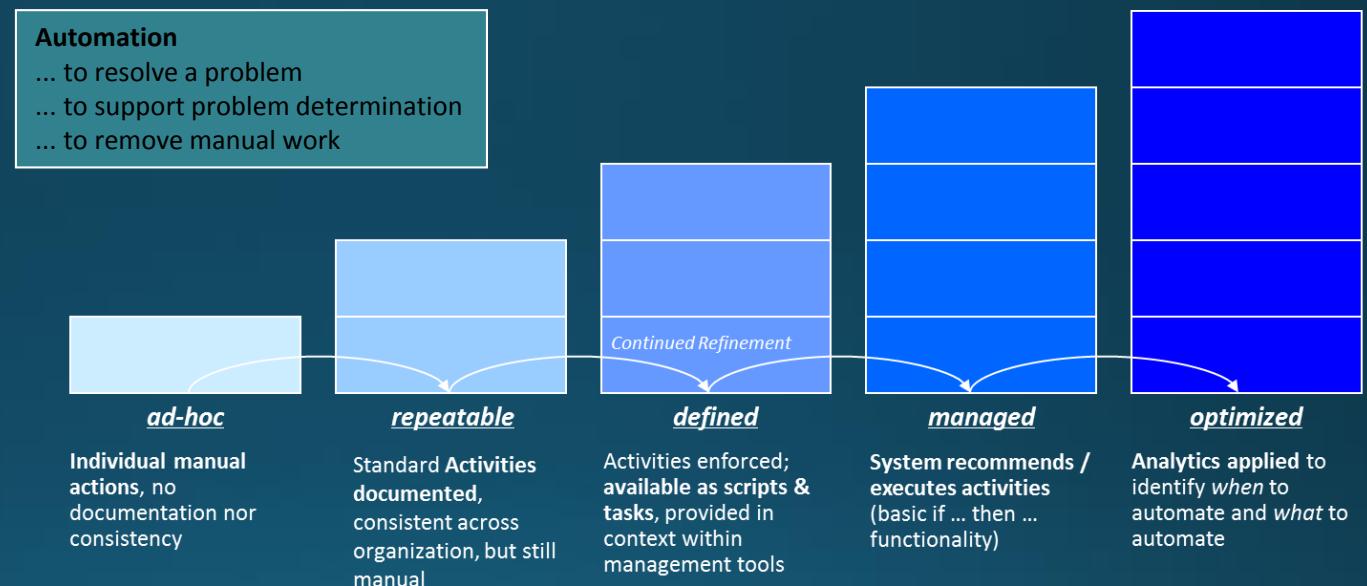
Bluemix Application Summary

AppName	State	Status	AppSvrApdex	AppSvrRspTime	AppSvrThroughput	ErrorRate
voicemail_show_toneanalysis	Started	Red		957		1
php_mysql_jwmsjr_2015	Started	Green		178		1
microservicesordersapi	Started	Green	1	68.6		
microservicesui	Started	Green	0.92	56.1		
monverify	Started	Green	1	28.1		
microservicescatalogapi	Started	Green				
voicemail_show	Started	Green	0.5	525		
microservicesui_ab	Unknown	Green				
microservicescatalogapi_ab	Unknown	Green				
microservicesordersapi_ab	Unknown	Green				
retrieve_and_rank_java_jwmsjr_1058	Stopped	Not Monitored				
microservices_ordersapi_rstoner_1725	Started	Not Monitored				

+ ADD ROW

Operations

- Operations describes the activities performed to deliver the right set of services at the right quality and at competitive costs for customers. IT Operations Management executes day-to-day routine tasks related to the operation of infrastructure components and applications, for instance
 - Application / Systems Availability and Health Checks
 - Compliance Checks
 - Monitoring of Performance and Capacity
 - Backups
- In cloud environments, manual work related to these activities should be minimal in favor of **long-term engineering project work**. „*If a human operator needs to touch your system during normal operations, you have a bug.*“ (*).
- Elimination of manual tasks and **automation** of repetitive tasks are vital.
- Transition to **Self-Service** further reduces the need for Operations

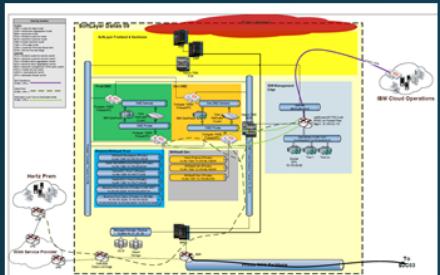


R - responsible
A - accountable
C - consulted
I - informed

Operations – Operational RACI

Offering to develop a responsibility assignment matrix for the Operational aspects of a solution:
RACI matrix and Process Swimlanes for key scenarios for a given solution / organization.

Architecture



Processes & Scenarios



Organization



RACI Matrix

Scenarios

Organizations

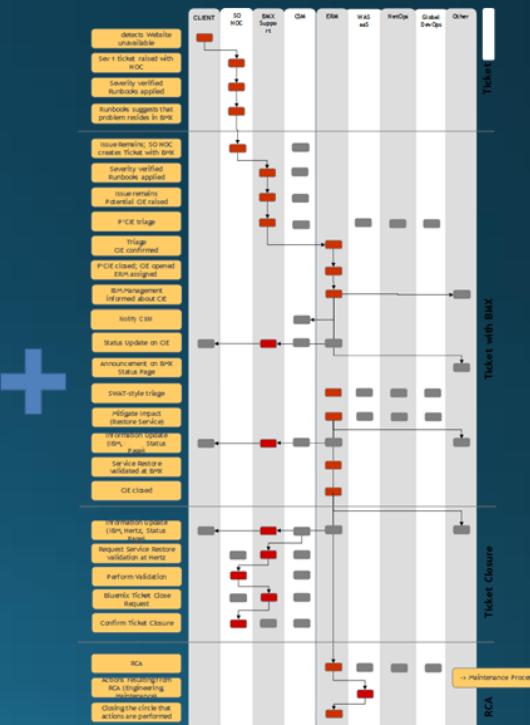
RACI

Technical Towers

Management Disciplines

Domain	Req. No.	Area	Possible Symptom	Possible Root Cause	Procedures	IBM								
						IBM SO	BlueMix Support	SoftLayer	WASaaS	Net Ops (including DMZ Service)	Global DevOps "SRE"	SDS Tools	Cloud SOC	MSS SOC
Incident Management														
Incident	11.1	Application	App unresponsive (internal Monitor)	App unresponsive	Determine to switch center; recycle App or switch data center; may lead to SRE intervention.	I	R/A							
Incident	11.2	Application	App unresponsive (internal Monitor) OR Database Monitor fire (internal Database Monitor fire)	Database issues	Determine to switch center; recycle App or switch data center.	I	R/A							
Incident	11.3	Application	App unresponsive (WebSphere connections stack; Website monitoring)	Legacy (internal systems) issue	No switch	A	R	(I)	(I)					
Incident	11.4	Application	App unresponsive (internal Monitor)	Server issue (Hardware - ESX Server down)	Determine to switch center; restart the node or switch data center.	I	R/I	R/I/C	A	R/I/C				
Incident	11.5	Application	App unresponsive (internal Monitor)	Server issue (OS or plain WAS down)	Determine to switch center; restart the node or switch data center.	I	R/I/A	R/I/C						
Incident	11.6	Application	App unresponsive (internal Monitor)	Server issue (WAS config payload down)	Determine to switch center; restart the node or switch data center.	I	R/I/A	R/I/C		R/I/C				
Incident	11.7	Application	SCCD Ticker requesting the switch	Switch center - unknown	Follow Runbook to switch center.	R	R/A/I							
Incident	11.8	Application	Eric application behavior / App Down.	Security incident on application	Determines RCA of incident and remediate. Fall back to a previous version if needed.	R	R/A	C		I	C		C	C
Incident	11.9	Application	Application becomes unresponsive or security incident is detected.	Security patch on application	Migrate to instance that has not been affected or back out of patch if possible.	R/A	C		I		C	C		
Incident	11.10	Application	Website unreachable	Active denial of service on Security Incident on WASaaS	Perform incident response and RCA analysis on application.	I	R/A		C	I	R/C		C	C

Process Swimlanes



Compliance Management

- Compliance Management (as part of (Governance, Risk Management, and Compliance – GRC) is targeted to address these concerns:
 - Ensuring compliance with evolving and increasingly complex regulations
 - Managing through regulatory change / avoiding fines
 - Coordinating compliance across multiple diverse jurisdictions
 - Building efficient operations
- Compliance is defined by laws & regulations (i.e. HIPPA, EU Data Protection), Certifications (i.e. FIPS), and Frameworks (i.e. FISMA, PCI). To meet more stringent compliance requirements, there may be specific client-specific requirements on top.
- Implementing Compliance Management
 - Defining and Implementing Compliance Controls
 - Steady monitoring / assessment of conformity, alerting if there is a violation of a control object
 - Assessment and Response - ideally automatic – to these violations. This could range from runbook automation to strategies like active deployment (Blue/Green) to assure that the services and SLAs are not disrupted.
 - Reporting and Documentation of Compliance information
 - Validating by internal and/or external auditors via process walkthroughs and evidence evaluation.



Link to Security spine

Icon by DesignContest, <http://www.designcontest.com>

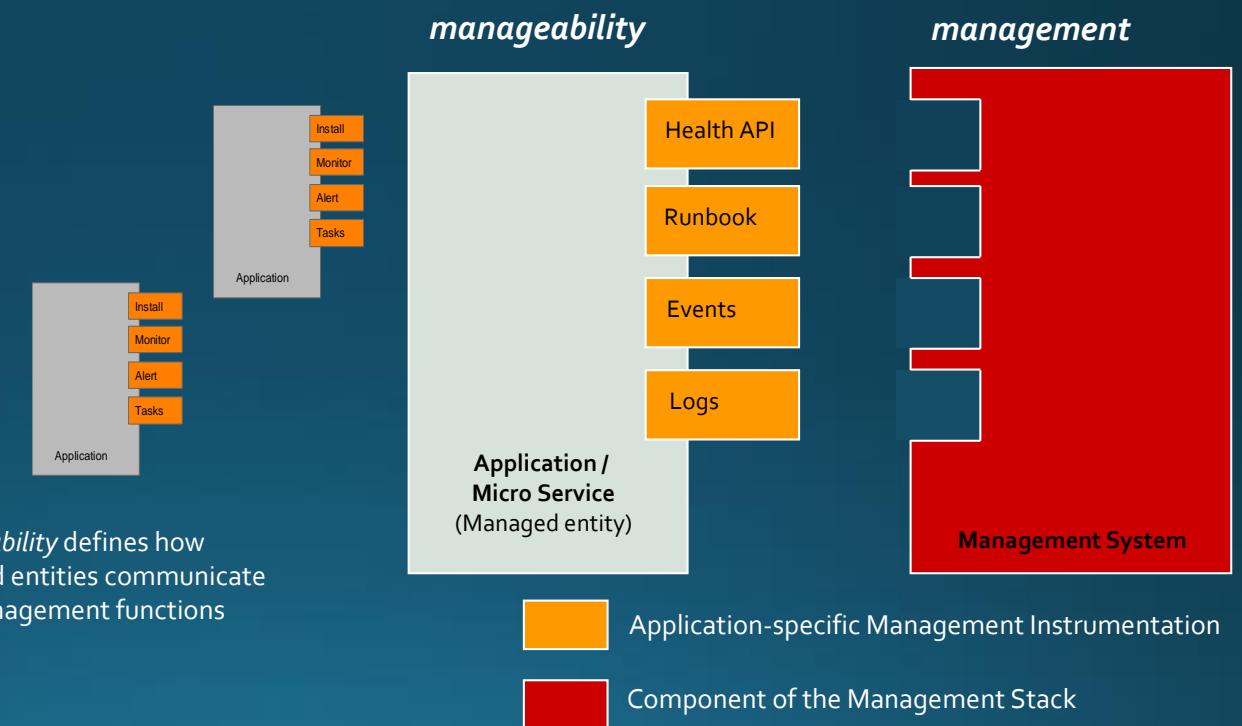
Build To Manage – How to build manageable applications

- As Development and Operations come closer together, new practices arise to better enable operations for cloud-based applications.

- Shift Left: Build manageable applications / micro-services

- Practices Include

- HealthCheck API
- Deployment Correlation
- Topology Information
- Test Cases and Scripts
- Runbooks
- Log Format and Catalog
- Distributed Tracing
- Event Format and Catalog
- Monitoring Configuration
- First Failure Data Capture



Build To Manage

The screenshot shows a web browser window with the URL <https://p1.www.ibm.com/ibmweb/basic/profile/healthcheck>. The page displays a green 'PASS' status, the date and time 'Thu, 28 Jul 2016 12:24:52 GMT', and the version 'VERSION 1.1'. Below this, a table lists four service endpoints with their response times:

PASS	WebIdentityRead	202 ms
PASS	WebIdentityUpdate	1304 ms
PASS	SBS Provision Connect	296 ms
PASS	IWaaS Connect	331 ms



<https://github.com/ibm-cloud-architecture/build-to-manage>

Build To Manage - The Twelve-Factor App

Popular platform-as-a-service provider Heroku maintains a manifesto of sorts called **The Twelve-Factor App**. It outlines a methodology for developers to follow when building modern web-based applications and can be seen as the best practices for creating microservices. The **twelve-factor app** is a methodology for building software-as-a-service apps that:

- Use **declarative** formats for setup automation, to minimize time and cost for new developers joining the project;
- Have a **clean contract** with the underlying operating system, offering **maximum portability** between execution environments;
- Are suitable for **deployment** on modern **cloud platforms**, obviating the need for servers and systems administration;
- **Minimize divergence** between development and production, enabling continuous **deployment** for maximum agility;
- And can **scale up** without significant changes to tooling, architecture, or development practices.

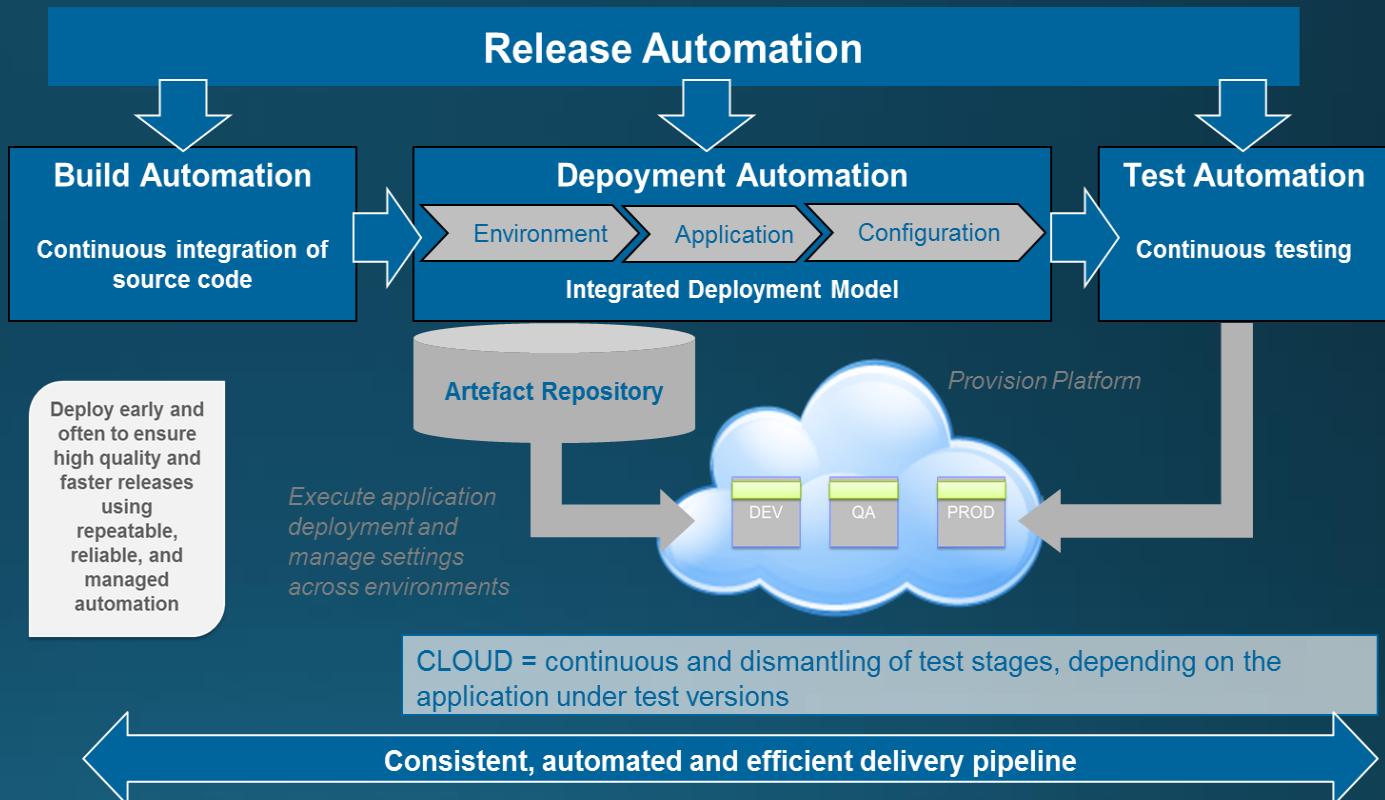
The twelve-factor methodology can be applied to apps written in any programming language, and which use any combination of backing services (database, queue, memory cache, etc).

1. **Codebase** - One codebase tracked in revision control, many deploys
2. **Dependencies** - Explicitly declare and isolate dependencies
3. **Config** - Store config in the environment
4. **Backing services** - Treat backing services as attached resources
5. **Build, release, run** - Strictly separate build and run stages
6. **Processes** - Execute the app as one or more stateless processes
7. **Port binding** - Export services via port binding
8. **Concurrency** - Scale out via the process model
9. **Disposability** - Maximize robustness with fast startup and graceful shutdown
10. **Dev/prod parity** - Keep development, staging, and production as similar as possible
11. **Logs** - Treat logs as event streams
12. **Admin processes** - Run admin/management tasks as one-off processes



Release Management

- DevOps characteristics
 - Deploy frequently using repeatable and reliable processes, fail early
 - Automate, Version, Test everything (including Infrastructure)
 - Continuously validate operational quality characteristics

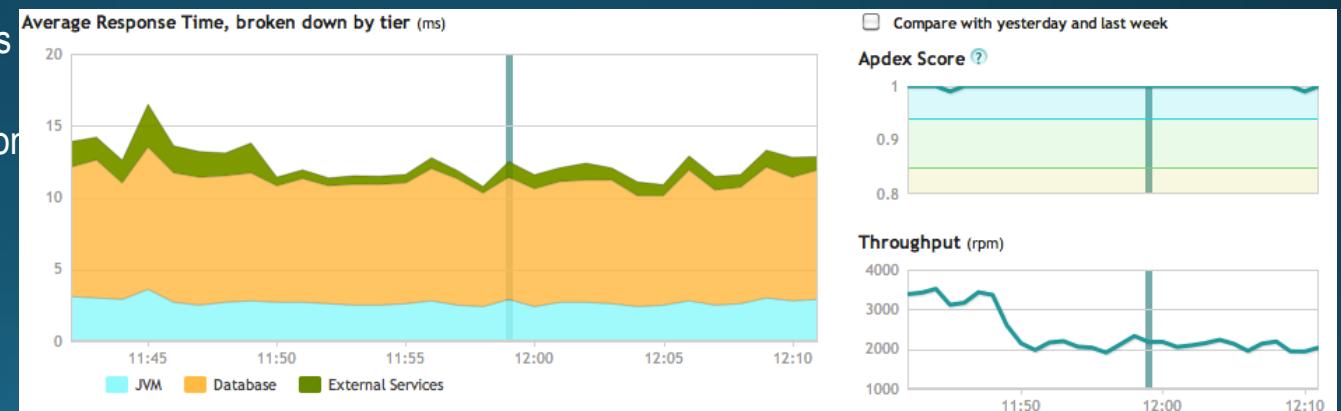
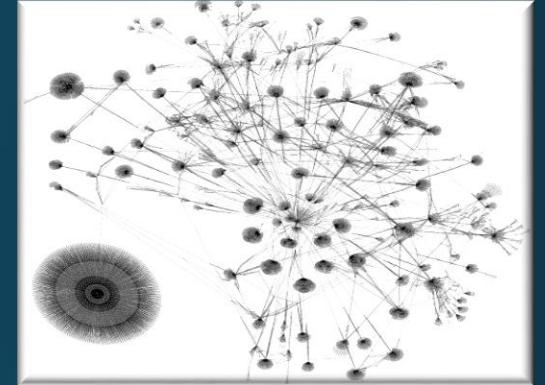


Link to DevOps spine

Icon by DesignContest, <http://www.designcontest.com>

Configuration Management

- **Configuration Management** identifies, controls, and maintains all elements in the IT infrastructure called Configuration Items.
- The purpose of the Configuration Management process is to maintain the **integrity of the configuration item** (CI) employed in, or related to, IT systems and infrastructure, and to provide accurate **information about CIs and their relationships**.
- The role of **Configuration Management Systems** (CMS, CMDB) shifts towards holding meta data such information on SLA, Client, Business Relationships, Topology, etc.. Technical information is dynamically pulled directly from the platform at the time it is needed. New Technologies such as Graph Databases allow scale and rapid model extensions.
- Continuous Integration and Continuous Deployment methodologies enable a rapid flow of features into the production. It becomes increasingly important for operation teams to be able to easily and effectively correlate performance and stability issues with application and configuration changes that has been made. **Deployment Markers** can send notification triggers to management systems, enabling them to track when an issue was introduced due to a change.



Service Continuity

- Service Continuity encompasses IT disaster recovery planning and wider IT resilience planning.
- It spans the entire Infrastructure (i.e. Availability Zones) and Application Architecture (i.e. dealing with stale data)
- Service Continuity contains the following aspects:
 - High Availability
 - Disaster Recovery
 - Backup / Restore
 - Archiving



Link to Resiliency spine

Icon by DesignContest, <http://www.designcontest.com>

Performance and Capacity Management

- The primary goal of Capacity management is to ensure that IT resources are right-sized to meet current and future business requirements in a cost-effective manner.
- It includes monitoring of performance and throughput metric. In a cloud environment, the traditional metrics such as CPU, Memory, Disk are less relevant than other parameters: the 4 Golden Signals: Latency, Traffic, Errors, Saturation
- Performance analysis of measurement data, including analysis of the impact of new releases on capacity.
- Performance tuning of activities to ensure the most efficient use of existing infrastructure

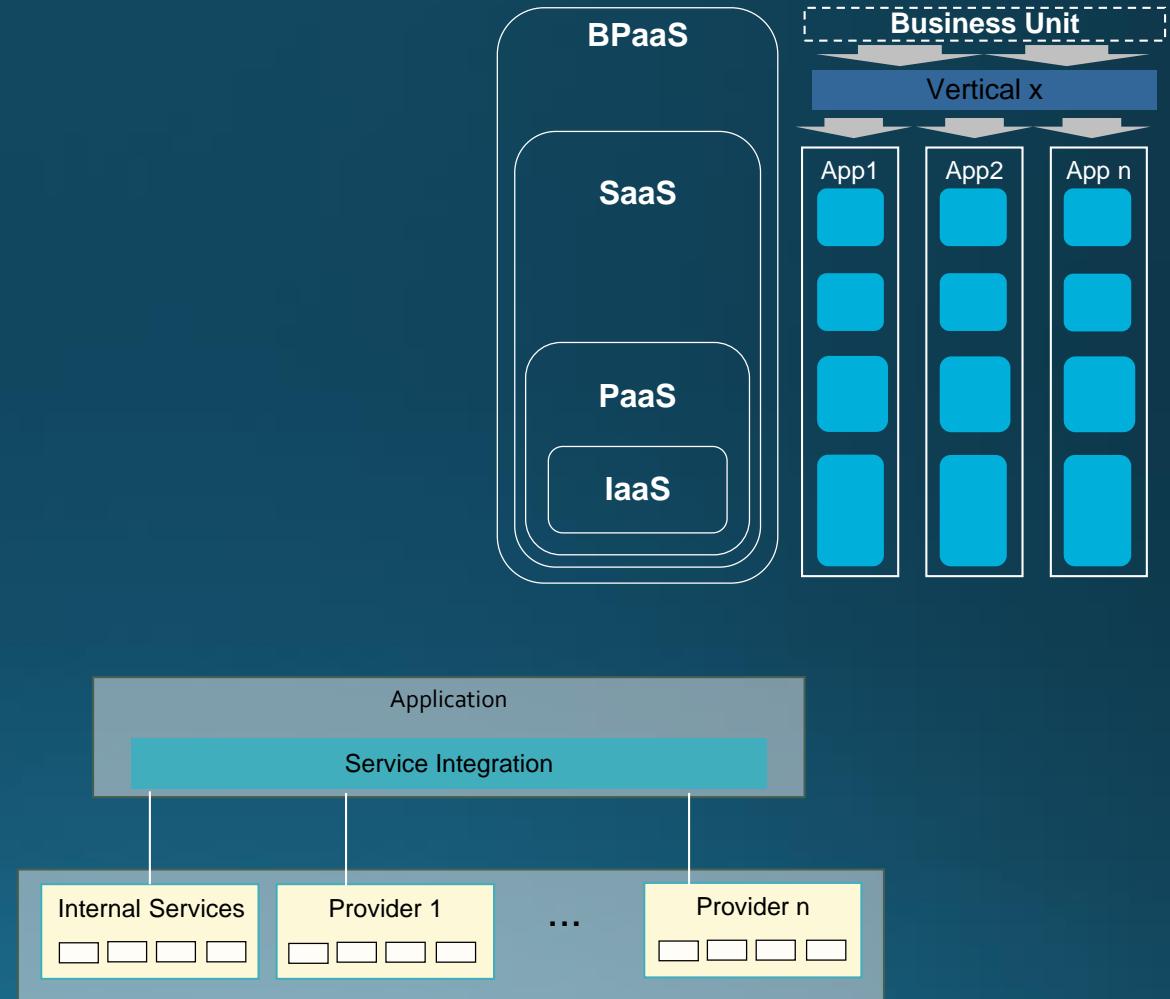


Link to Performance spine

Icon by DesignContest, <http://www.designcontest.com>

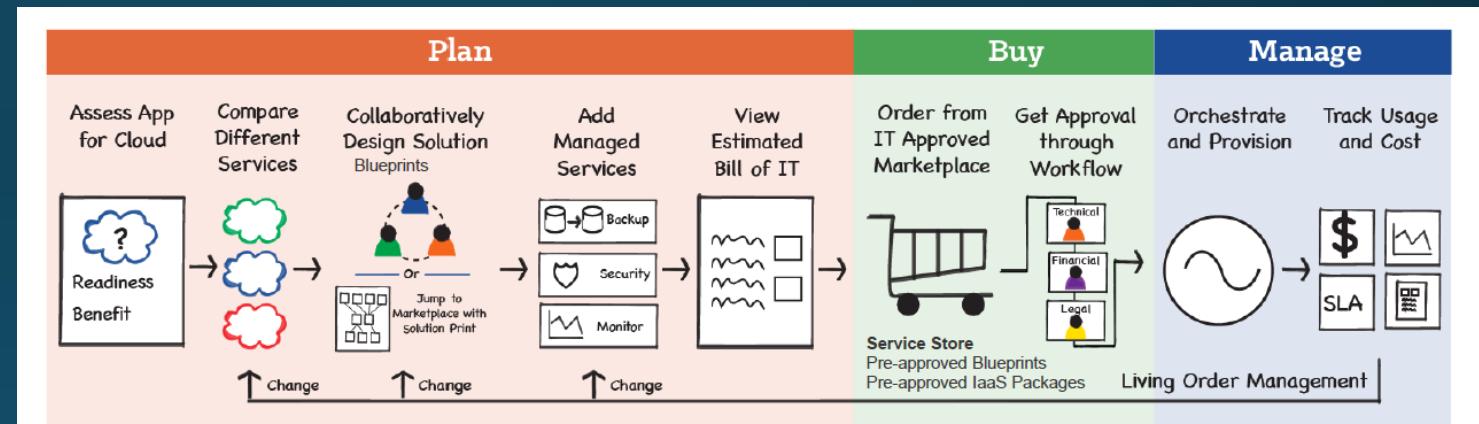
Service Level Management

- A **service-level agreement (SLA)** is defined as an official commitment between a service provider and the consumer. Particular aspects of the service – quality, availability, reliability, responsibilities – are agreed between the service provider and the service user.
- A **service level objective (SLO)** is a key element of a SLA. SLOs are agreed as a means of measuring the performance of the service provider and are outlined as a way of avoiding disputes between the two parties based on misunderstanding.
- **Error Budgets** (maximum allowable threshold for errors and outages) help to trade off reliability against other goals such as introducing new functionality. Error budget is a measure of risk, it is the amount of headroom you have above your SLA.
- SLA information should be presented in the **Dashboards** to help prioritization of work.
- Create a **culture of accountability** across all levels of the IT organization
- **Service Integration** becomes the default as applications will be based on internal services as well as external services across multiple providers. Supplier Management manages the interactions with third party organizations



Financial Management

- The aim of Financial Management is to give accurate and cost effective stewardship of IT assets and resources used in providing IT Services. It is used to plan, control and recover costs expended in providing the IT Services negotiated and agreed to in a service-level agreement (SLA).
- Specific to Cloud, a key activity is to aggregate, compare, estimate, provision, and manage cloud services and associated costs across multiple providers. Value proposition:
 - **Reduce shadow IT** – flexibility of choice for end users within the organization's compliance framework.
 - **Rapid financial decision making** for the CIO by consolidating all cloud service costs on a single dashboard.
 - **Negotiate better T&C's** with Cloud providers based on performance visibility.
 - Enforce organizational **policy compliance** for service selection – based on cost, location, workload or performance requirements.
 - **Integrated service management** across traditional and cloud IT services – enabling the virtual data center.
- Understand Development and Operations Costs, enable appropriate Billing



Business Performance

- Business-oriented operational insight - **Alignment of Business Performance with Service Management and Operations** is critical to ...
 - Understand the impact of business needs on IT services and infrastructure
 - Link key business service components and capabilities to the goals of the business
 - Align response of IT staff and service providers to priorities of the business
- Especially for agile applications in the Cloud, **direct feedback** of end users is desired. Practices such as A/B testing is applied to compare between releases.
 - Effectiveness of recent functionality / code change
 - Customer sentiment analysis
- Business Performance data will be used for instance
 - In dashboards to append application data with business data
 - In incident Management (Monitoring, Correlation, Prioritization) to prioritize activities according to business impact
 - In backlog prioritization to decide what to do next
- IBM has developed Process Maps and **KPIs available for more than 30 industries.**



Proactively
communicate
value



Build
stakeholder
trust



Track
operational
health

Continued Service Improvement

- **Continued Service Improvement (CSI)** continually improves the effectiveness and efficiency of services and processes.
- 7 Steps towards CSI:
 - What should be measured?
 - What can be measured?
 - Gather data
 - Process the data
 - Analyse the data
 - Present and use the information
 - Implement corrective action
- **50% of the SRE time** should be allocated to improvements (automation, improving reliability, performance and utilization)
- Improvements **driven from the end-user perspective**: Perceived Availability and Performance, Sentiment Analysis, etc.
- **Analytics and Cognitive technology** can assist in the analysis.
 - Event Analytics
 - Predictive Monitoring
 - Increasing Tier-1 Resolution with Cognitive Technology
 - Improvements to Incident / Problem / Change





IBM[®] Bluemix Garage[™]

**Cloud Service Management
& Operations Offering (CSMO)**

IBM Bluemix Garage CSMO Offering

Transform your thinking around

Cloud Service Management and Operations (CSMO)

IBM Bluemix Garage's CSMO offering helps enterprise clients adopt the right IT Operations practice for Cloud by identifying the organizational roles, processes, skills and tools necessary for optimal Service Management on Bluemix.

Following the same “lean” approach emphasized in all Garage Offerings, Garage’s CSMO Offering brings your team guided assistance from our subject matter experts and helps transform the thinking around Service Management as a new Cloud-based tools chain and Operations model is created collaboratively. Garage’s CSMO Offering helps traditional IT Operations teams to improve efficiencies and developers to create applications that are “built-to-manage.”

IBM Bluemix Garage CSMO Offering

Definition | What is?

Cloud Service Management and Operations (CSMO)

Refers to the entirety of activities--organized in persona, processes and tools--that are performed by an organization in order to plan, design, deliver, operate and control Cloud IT Service Management principles and processes.

IBM Bluemix Garage CSMO Offering

Definition | What is?

IBM Bluemix Garage Method

Is a well-honed and well-integrated set of tracks and practices leveraging IBM Design Thinking, Lean Startup, Agile Development and DevOps with the power of Cloud. It is targeted at helping you to rapidly bring an innovation idea to life in the form of Minimum Viable Product or MVP – the smallest idea (app, API, technology, etc) that can be developed and tested to discover if it achieves the anticipated business outcome.

IBM Bluemix Garage CSMO Offering

Definition | What is?

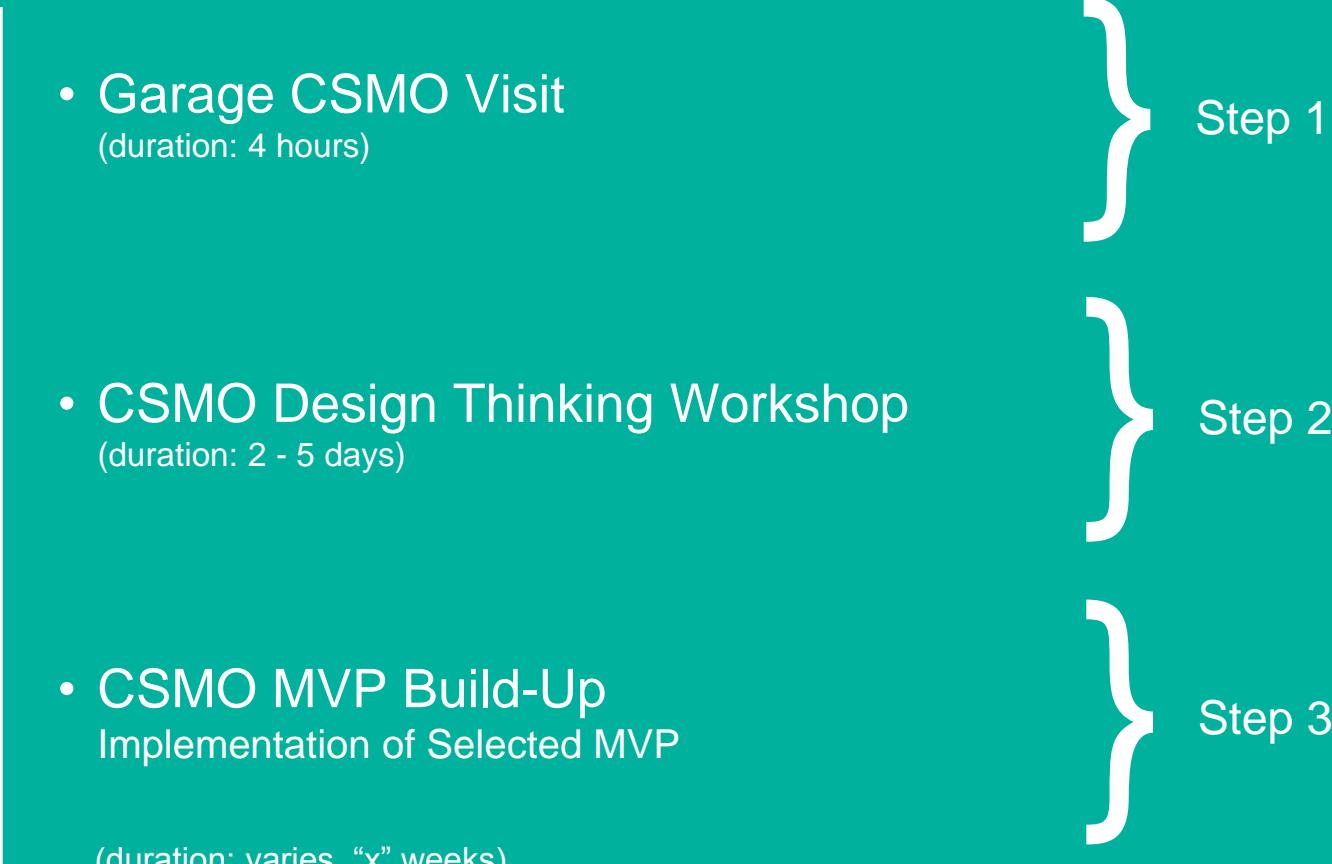
Built-to-Manage

Refers to the techniques and best practices used by application developers to provide operational information; for example, improved logging generating performance statistics, and providing service topology information.

IBM Bluemix Garage CSMO Offering

Offering At a Glance:

- Garage CSMO Visit
(duration: 4 hours)
 - CSMO Design Thinking Workshop
(duration: 2 - 5 days)
 - CSMO MVP Build-Up
Implementation of Selected MVP

(duration: varies, “x” weeks)
- 
- Step 1
- Step 2
- Step 3

Garage CSMO Visit

Garage CSMO Visit

During your half day Garage CSMO Visit, hear IBM's perspective on how Service Management will necessarily change in a hybrid-Cloud and DevOps world.

Learn best practice and perspectives from our subject matter experts to help you navigate on your journey of cloud adoption.

CSMO Visit Goals:

- Understand tools that enable an optimized operations process.
- See firsthand how Cloud operations teams collaborate to quickly resolve issues.
- Understand methods that developers can utilize to harden application code for the Bluemix environment.
- Experience a tour of an operations team providing support for Bluemix-based applications.
- Understand our practices to optimize availability for Bluemix-based applications.
- Provide insight into building a Cloud-based tools chain.
- Understand next steps to engage with us on your candidate project.

CSMO Design Thinking Workshop

CSMO Design Thinking Workshop

Workshop Participants:

CLIENT TEAM (Suggested):

Head of IT Operations (Manager)

Line of Business (Architect)

Methods and Tools (Architect)

Lead Developer (User)

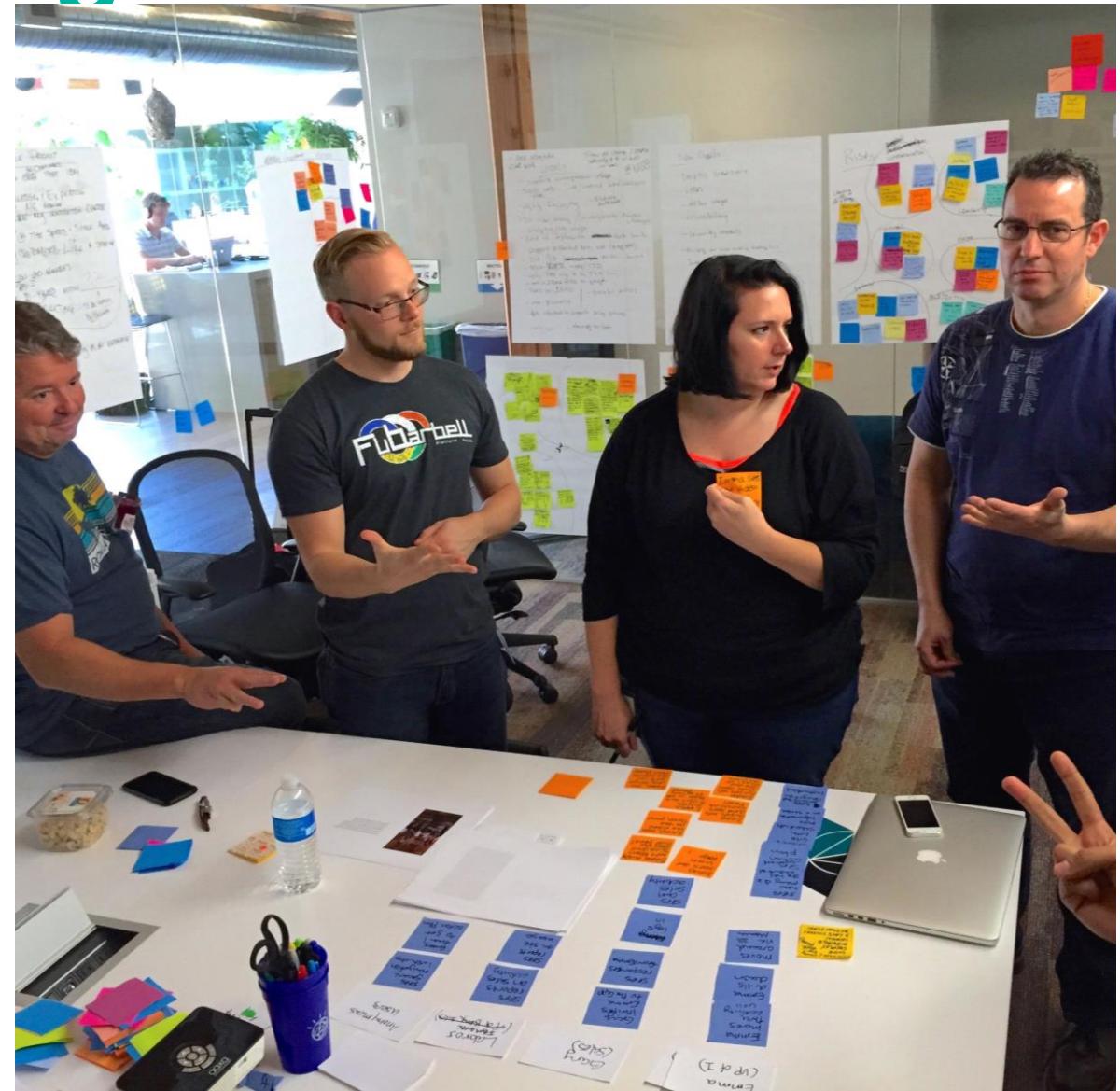
Lead Ops Engineer (User)

L1 Administrator (End User)

IBM BLUEMIX GARAGE TEAM:

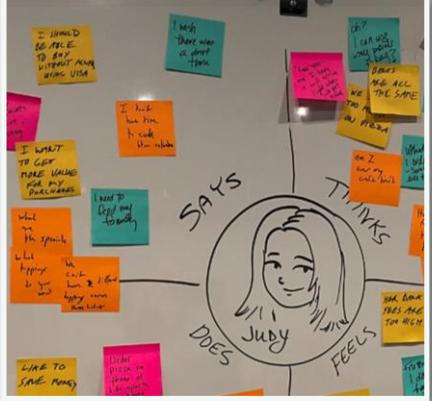
Design Thinking Facilitator(s)

CSMO Practice SME(s)



CSMO Design Thinking

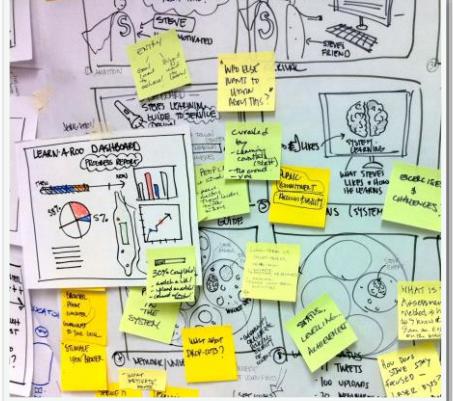
Workshop Activities: Workshop



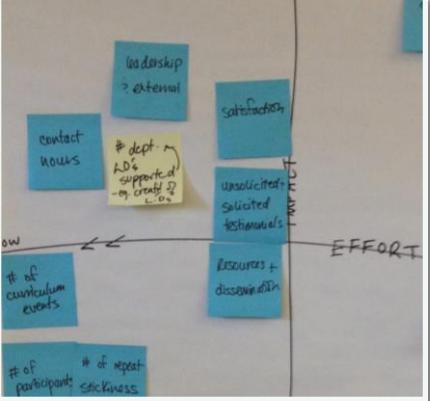
Stakeholder +
Empathy Maps



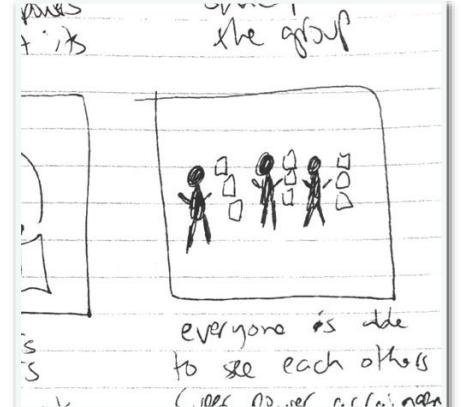
As-Is Scenario



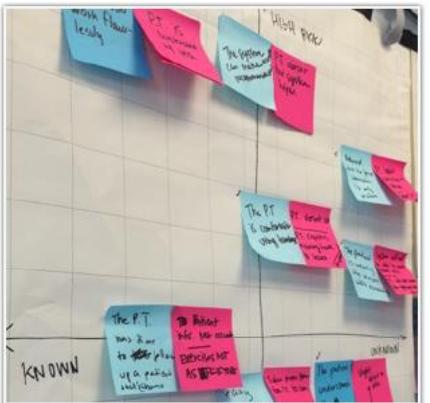
Ideation



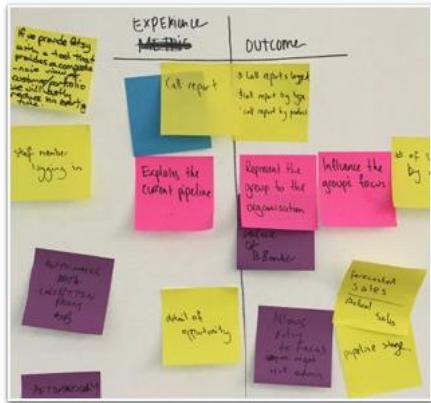
Prioritization



Storyboarding



Assumptions



Hypothesis / Metric

A business user can get access to web applications needed to do this/her job in 1 minute or less without having to create or

MVP



Goals / Non-Goals



Retro

CSMO Design Thinking

Workshop

Workshop Goals:

- Leverage IBM Design Thinking and CSMO Practice Team subject matter expertise to drive transformation of IT.
- Using ‘Personas’ and ‘As-Is Scenario Maps’ gain empathy for and deep understanding of current Services Management pain points.
- Using the tools of ‘Ideation’ and ‘Storyboarding’ define future-state for Services Management alleviating current pain points.
- Decide upon the one or two things most necessary to enhance and enable your transformation to cloud. This is your ‘MVP’ from an operations perspective and likely includes changes in Process, Culture and/or Technology.
- Detail candidate MVP’s Scope/Metrics and Skill/Timeline.

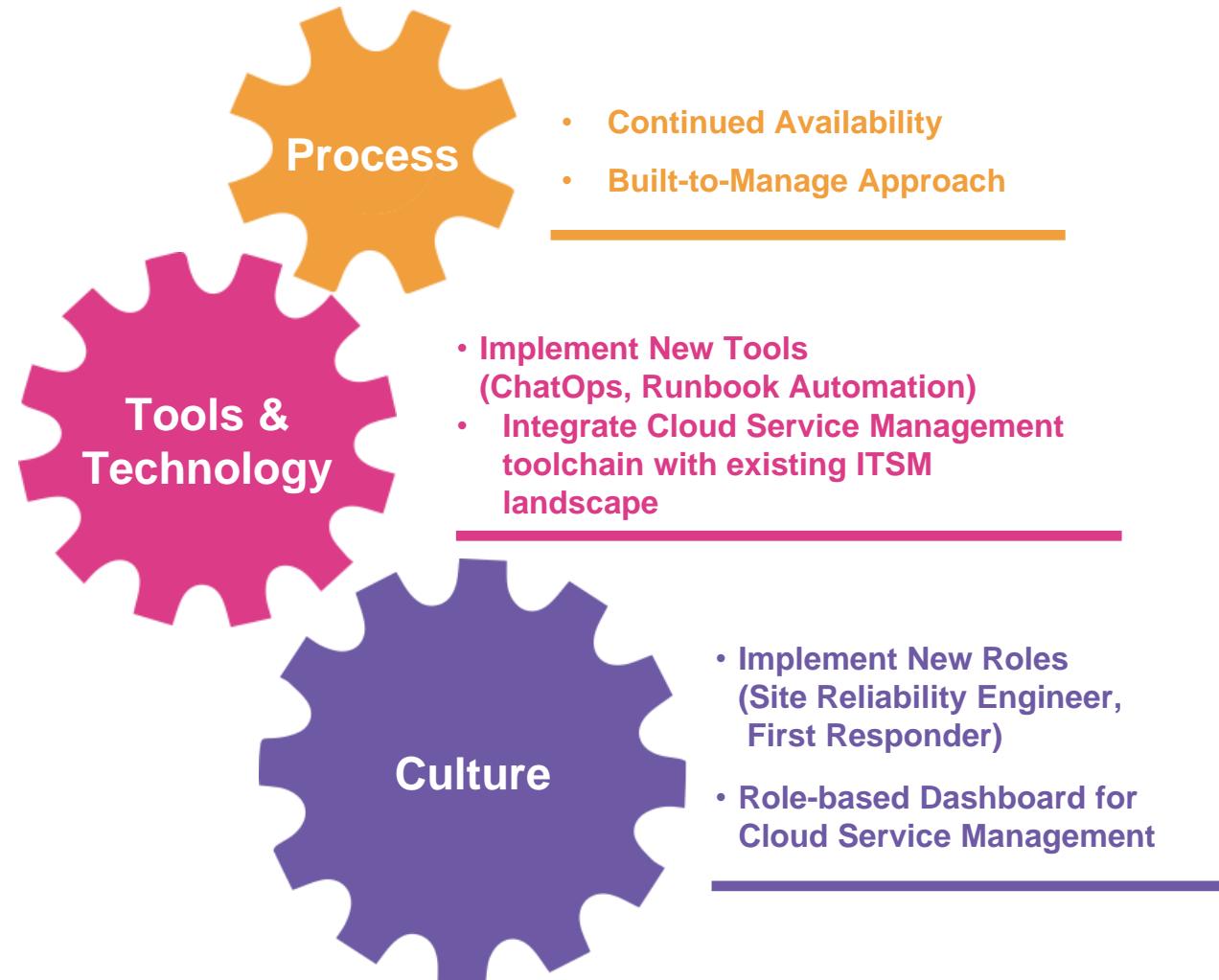
Insights and Learnings captured
in a Workshop Outcomes Deck
delivered to your team.

Implementation of Selected MVP

Implementation of Selected MVP

After appropriately sizing and scoping your MVP, Garage Team team drafts a proposal to execute on your MVP and, in collaboration with your team, deploys and instantiates the MVP over the next several weeks.

Reference Architecture for CSMO



Build to Manage

Operations goals vs development goals

Operations

- Minimize risk
- Maximum availability

Development

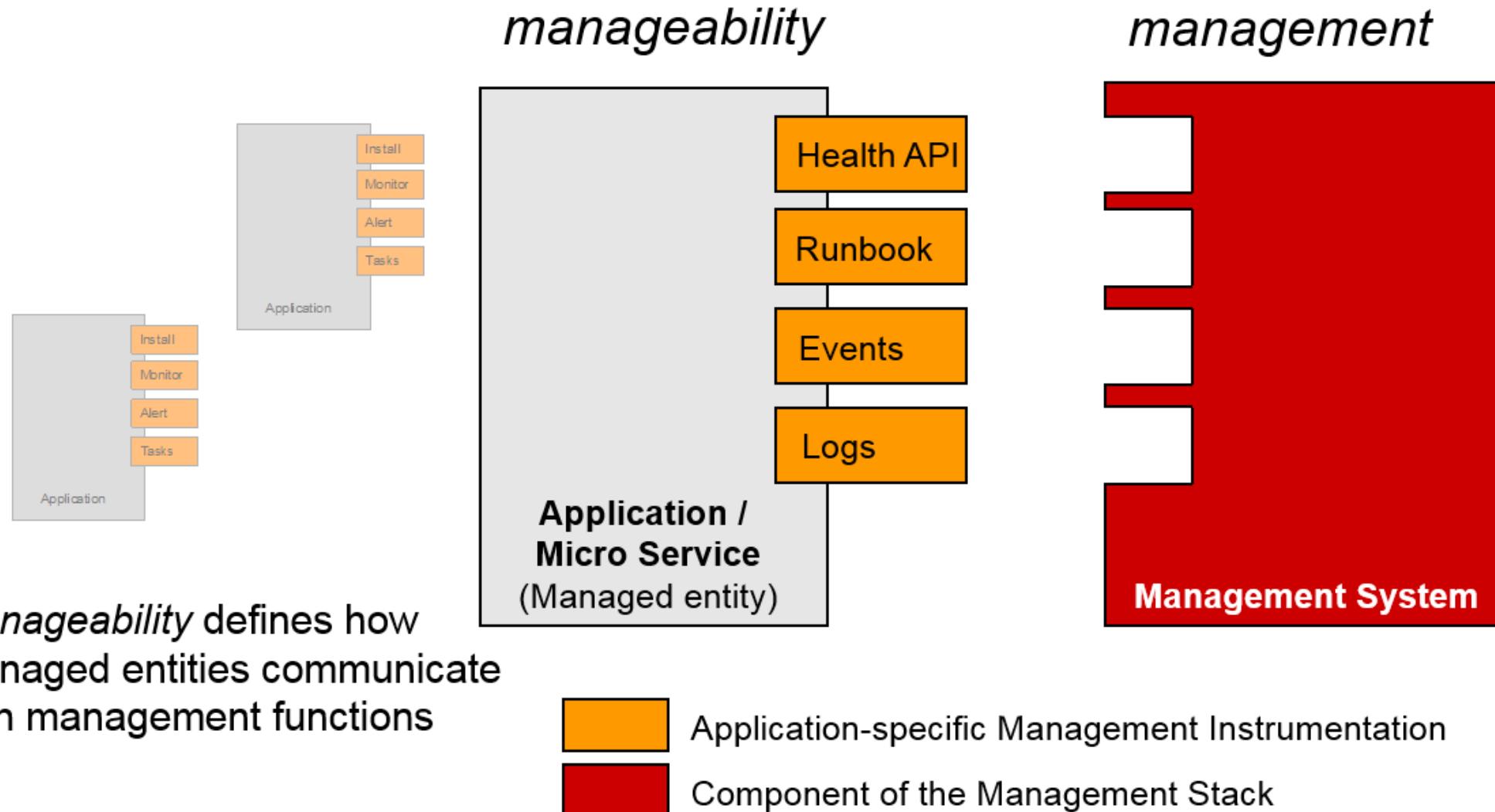
- Rapid incremental change
- Feedback loop



Possible solutions

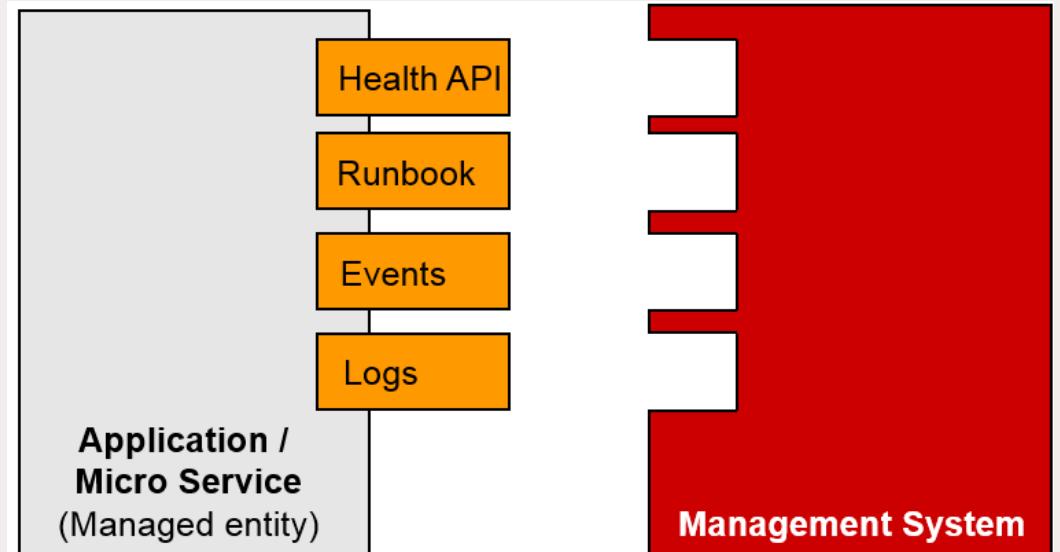
- Developers support their own code
 - Opposing goals are owned by the same person, enabling the individual or team to negotiate a compromise between the two
 - Broad skill sets required.
- DevOps Culture
 - Create teams with broad mix of skills with the same set of goals
- Build to Manage
 - Activities developers can do to instrument applications or provide manageability aspects as part of an application release,

Build to manage approach



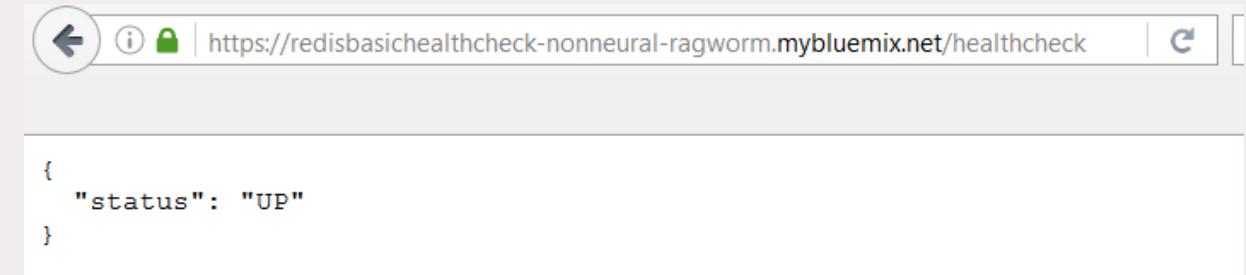
Build to manage aspects

- HealthCheck API
- Log Format and Catalog
- Deployment correlation
- Distributed Tracing
- Topology Information
- Event Format and Catalog
- Test Cases and Scripts
- Runbooks
- First Failure Data Capture



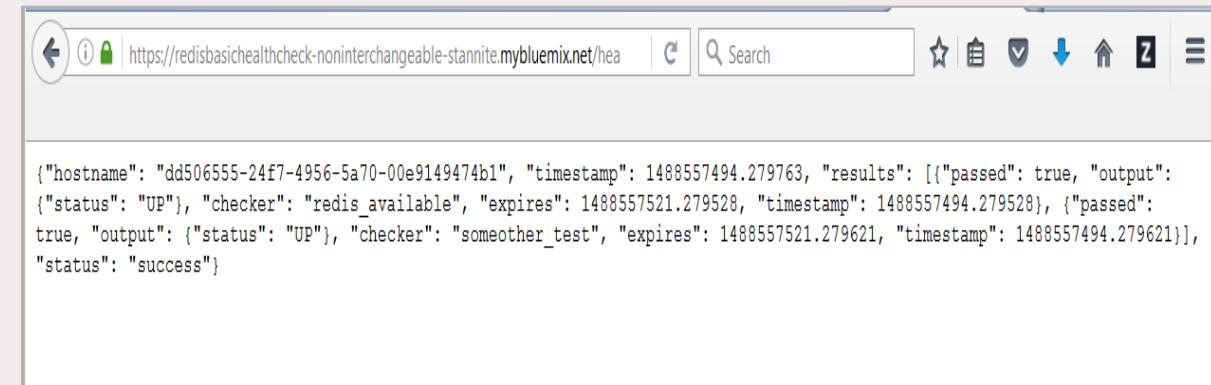
HealthCheck API

- A quick, standardized method to validate the status of a component and it's dependencies



A screenshot of a web browser window. The address bar shows a secure connection to <https://redisbasichealthcheck-nonneural-ragworm.mybluemix.net/healthcheck>. The page content is a JSON object:

```
{  
  "status": "UP"  
}
```



A screenshot of a web browser window. The address bar shows a secure connection to <https://redisbasichealthcheck-noninterchangeable-stannite.mybluemix.net/hea>. The page content is a JSON object containing detailed health check results:

```
{"hostname": "dd506555-24f7-4956-5a70-00e9149474b1", "timestamp": 1488557494.279763, "results": [{"passed": true, "output": {"status": "UP"}, "checker": "redis_available", "expires": 1488557521.279528, "timestamp": 1488557494.279528}, {"passed": true, "output": {"status": "UP"}, "checker": "someother_test", "expires": 1488557521.279621, "timestamp": 1488557494.279621}], "status": "success"}
```

Log Format and Catalog

- Generate a unique identifier(UID) on every log entry type
- Track the UIDs that you have created in a log catalog
- Logging ‘best practices’
 - Don’t log too much
 - Don’t log too little
 - Don’t log everything at the same level
 - Allow dynamically-changeable and remotely-changeable logging
 - Include timestamps
 - Temporary log queues
 - Log performance data
 - Create unique identifiers for transactions
 - Keep multi-line events to a minimum
 - Add the exact method name / line number to the error message



Deployment correlation

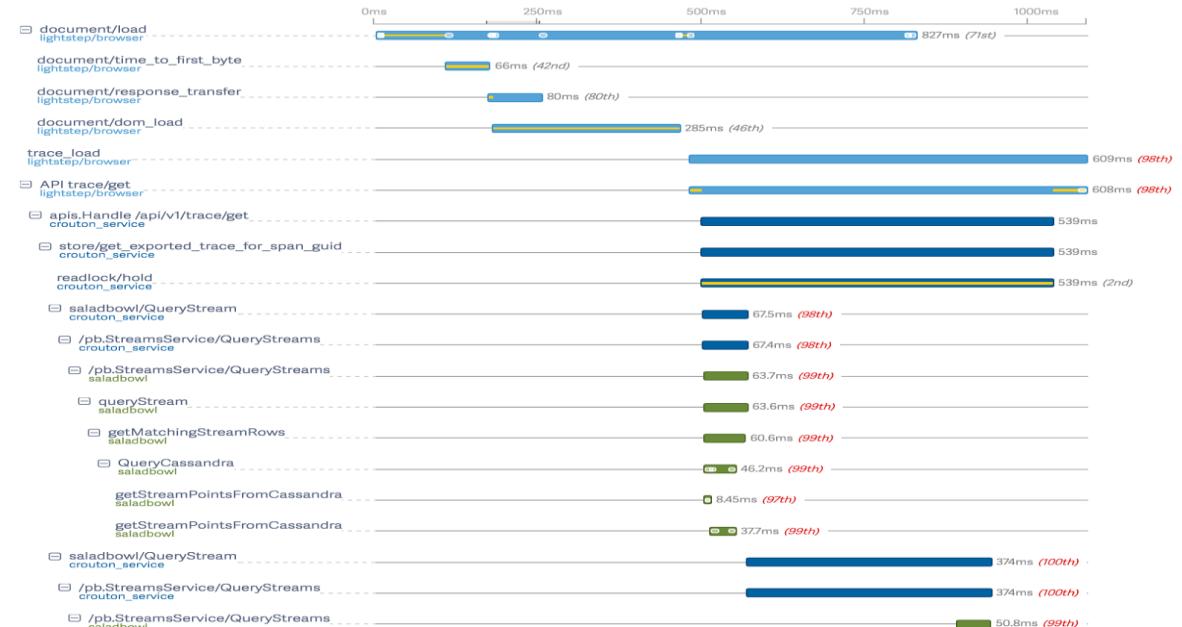
Use Deployment markers to track state of application deployment

- Can be used to send notification triggers to management systems
- Users can leverage information in dashboards
- Marker information can be sent to Event Management system for correlation with other events

Distributed Tracing

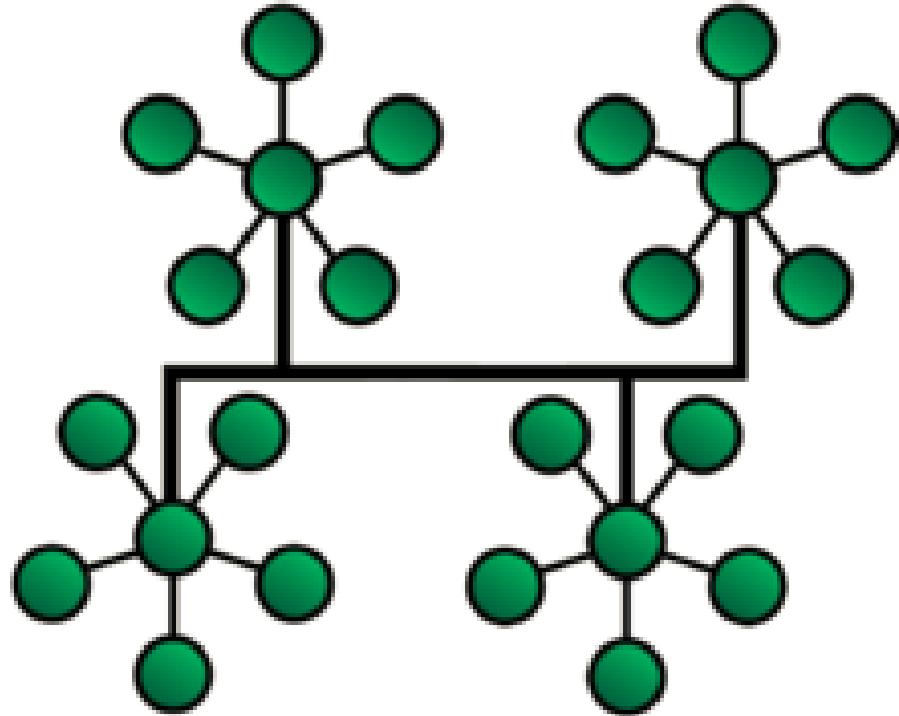
Help to understand how and why complex systems misbehave.

- A unique Transaction ID is passed through the call chain of each transaction in a distributed application topology, such as micro services based topologies
- Pass the id generated on an entry point to each subsystem that is used
- Ideally transparently, via wrapper functions and http headers where possible. .
- Helpful for debugging and gathering analytics



Topology Information

- In traditional IT, CMS or CMDB holds Configuration Item (CI) information and relationships between CIs
- In cloud, shifts from tangible assets to services and APIs
 - Relationships and dependencies are important to understand
 - Recommend developers provide deployment descriptors



Event Format and Catalog

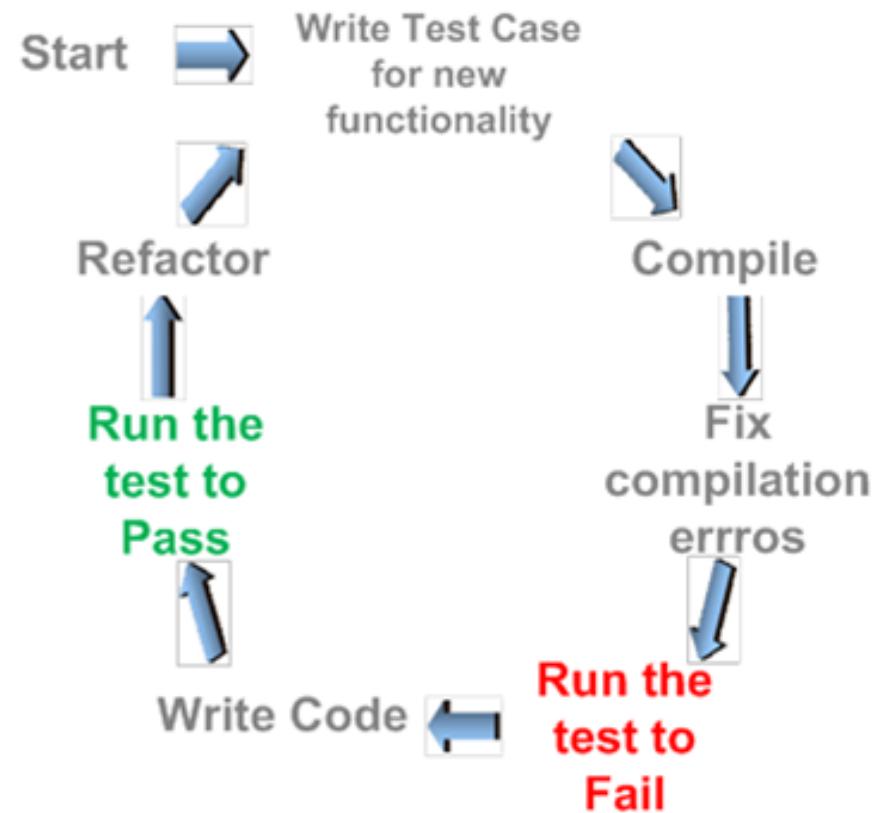
- Events should provide relevant data as part of their payload to allow for correlation of multiple events
 - Application names or Identifiers
 - Transaction identifiers, or information
 - Topology information

Netcool/OMNIBus Event List : Filter="All Events", View="Default"			
File	Edit	View	Alerts
			All Events
			Default
			Top [OFF]
Group	Summary	Last Occurrence(+)	Count
	A sql process running on loureed has disconnected	10/24/03 14:05:08	1
	Diskspace alert	10/24/03 14:44:32	1572
	Diskspace alert	10/24/03 14:44:40	1383
	Diskspace alert	10/24/03 14:45:43	4732
	Diskspace alert	10/24/03 14:46:02	5293
	Machine has gone offline	10/24/03 14:50:35	8222
	Machine has gone online	10/24/03 14:50:58	8222
9C0A823A	Attempt to login as admin_12 from host loureed failed	10/24/03 14:51:00	1
9C0A823A	A e@9C0A8225C@9C0A823A1:0.0 process running on loureed has connected	10/24/03 14:51:04	1
	Link Up on port	10/24/03 14:51:29	201
	Port failure : port reset	10/24/03 14:51:31	1532
6 2 6 2 12 6 All Events			
0 rows inserted, 21 rows updated and 0 rows deleted 10/24/03 14:52:44 root LON_PRIM [PRI]			

Test Cases and Scripts

Test Driven Development

- Requirements turned into test cases
- Software is built to pass the tests
- Developer responsible for test cases and scripts
- Test code treated like production code
- Must work correctly for both positive and negative cases



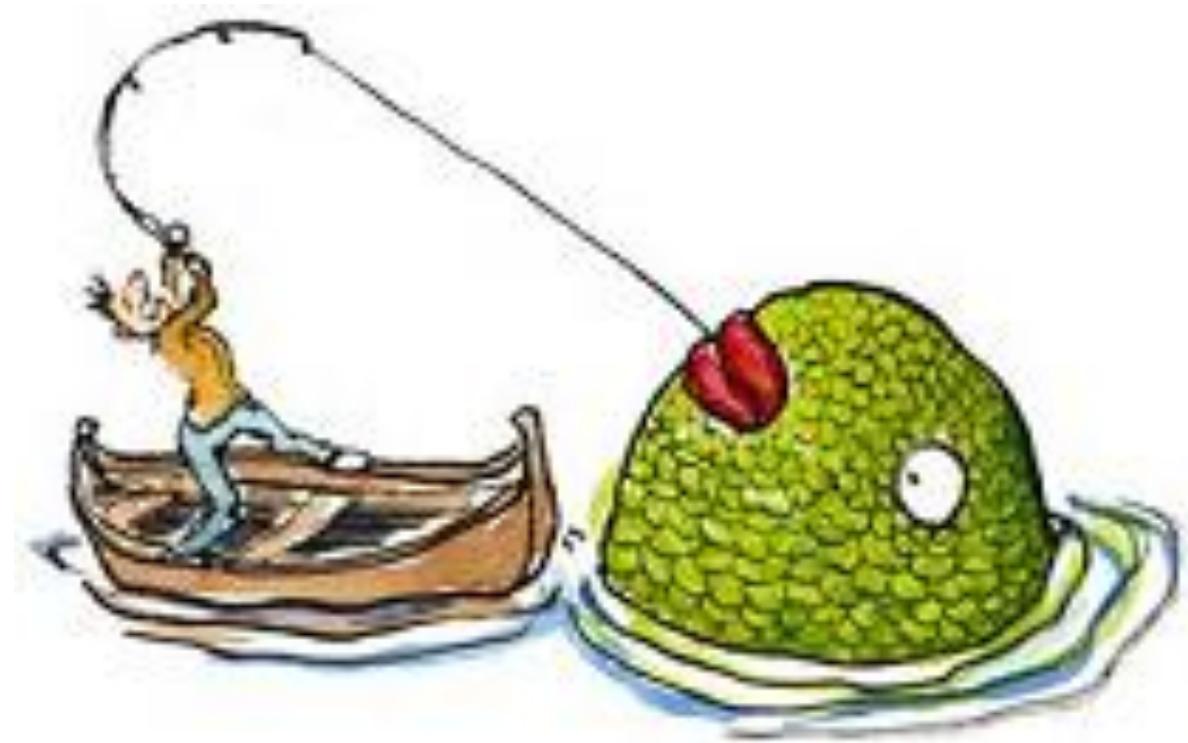
Runbooks

- Compilation of routine procedures and operations that the system administrator or operator carries out
- Electronic or in physical book form
- Typically contains procedures to begin, stop, supervise, and debug the system



First Failure Data Capture

- Instantly capture useful data after a problem has been logged
- Collect enough data so that a problem can be diagnosed without having to reproduce it
 - Can include trace logs, message logs, dumps of in-memory data structures, and so forth.



Where to start – some recommendations

Mature organization, majority of infrastructure in traditional IT

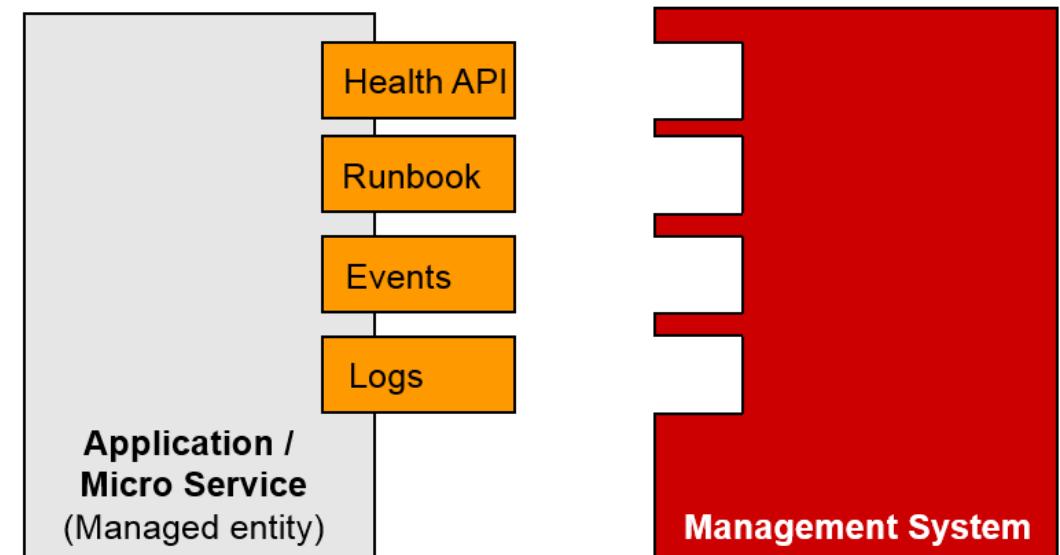
- Runbooks
- Log Format and Catalog
- Event Format and Catalog

Emerging organization, Cloud Foundry/Bluemix based infrastructure

- HealthCheck API
- Log Format and Catalog
- Test cases and scripts

Aggressively adopting microservices

- HealthCheck API
- Distributed Tracing
- Topology Information
- Deployment Correlation



References

- Optimal Logging: <https://testing.googleblog.com/2013/06/optimal-logging.html>
- Open Tracing: <https://www.cncf.io/blog/2016/10/20/opentracing-turning-the-lights-on-for-microservices/>
- Build to Manage: <https://github.com/ibm-cloud-architecture/build-to-manage>
- Bluemix Garage: <https://www.ibm.com/devops/method/category/manage>

Notices and disclaimers

- Copyright © 2017 by International Business Machines Corporation (IBM). No part of this document may be reproduced or transmitted in any form without written permission from IBM.
- **U.S. Government Users Restricted Rights — use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.**
- Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information.
This document is distributed “as is” without any warranty, either express or implied. In no event shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity. IBM products and services are warranted according to the terms and conditions of the agreements under which they are provided.
- IBM products are manufactured from new parts or new and used parts.
In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply.”
- **Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.**
- Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.
- References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.
- Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.
- It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer is in compliance with any law.

Notices and disclaimers continued

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. **IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a particular, purpose.**

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

IBM, the IBM logo, ibm.com, Aspera®, Bluemix, Blueworks Live, CICS, Clearcase, Cognos®, DOORS®, Emptoris®, Enterprise Document Management System™, FASP®, FileNet®, Global Business Services®, Global Technology Services®, IBM ExperienceOne™, IBM SmartCloud®, IBM Social Business®, Information on Demand, ILOG, Maximo®, MQIntegrator®, MQSeries®, Netcool®, OMEGAMON, OpenPower, PureAnalytics™, PureApplication®, pureCluster™, PureCoverage®, PureData®, PureExperience®, PureFlex®, pureQuery®, pureScale®, PureSystems®, QRadar®, Rational®, Rhapsody®, Smarter Commerce®, SoDA, SPSS, Sterling Commerce®, StoredIQ, Tealeaf®, Tivoli® Trusteer®, Unica®, urban{code}®, Watson, WebSphere®, Worklight®, X-Force® and System z® Z/OS, are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml.

InterConnect 2017

