



Enterprise Deployment experiences from Application Release & Deployment

Alexandre Abi Khaled, MSc.
DevOps Solutions Architect, Europe
Alexandre.abikhalel@nl.ibm.com





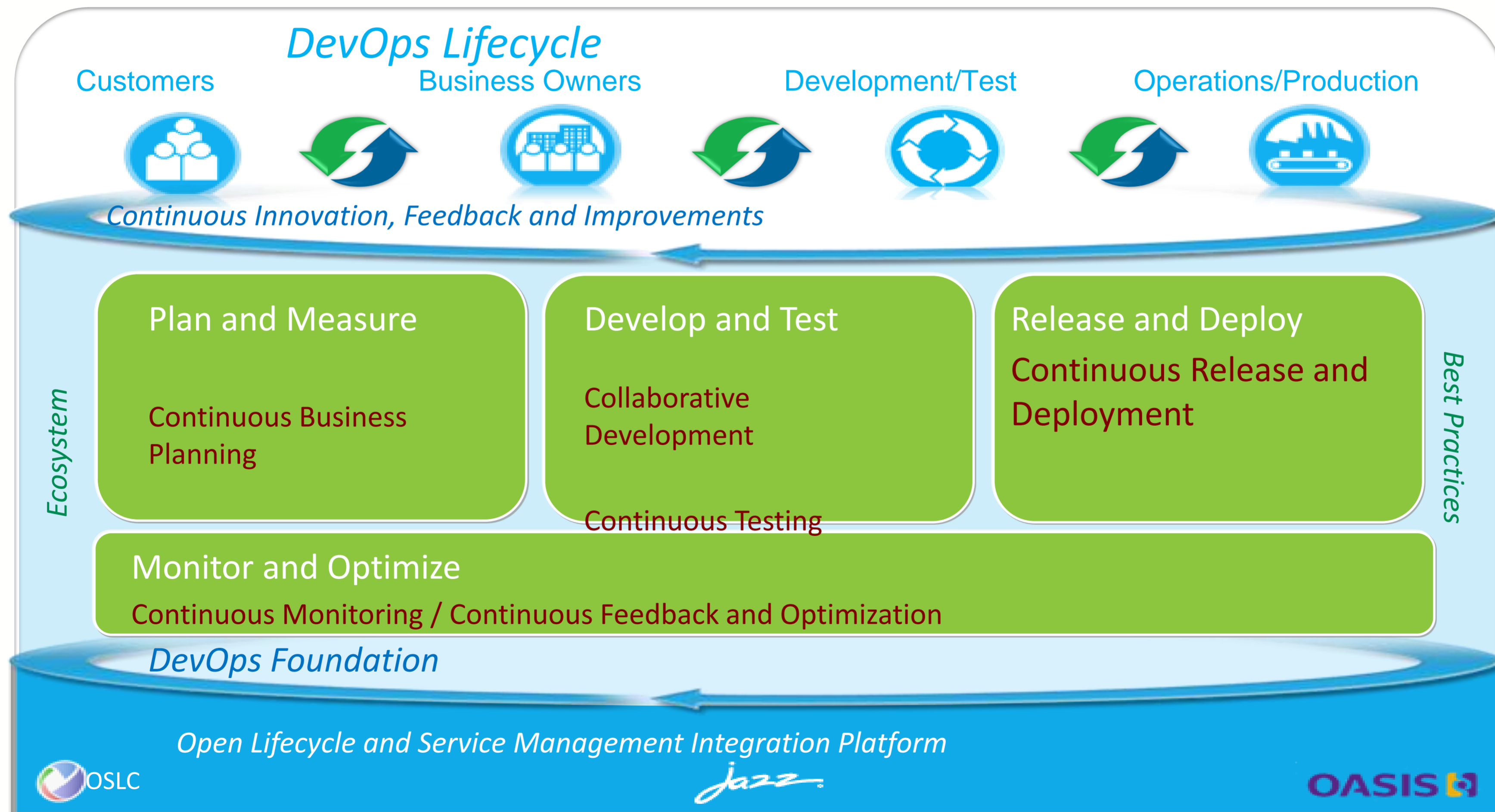
Goal: Plan and Start adopting Continuous Delivery

- DevOps and Continuous delivery maturity
- Identify where to start: assessment workshop
- Evaluate your assessment results
- Scale the deployment of the solution Enterprise wide: Examples
- Quick Startup: get started in 2 weeks time : What can you achieve in 2 months?



Capabilities to enable a DevOps approach

Practices





DevOps Adoption (1 of 2)

- Requirements Management
 - Requirements Management and communication across Development and Operations
- Versioning of all DevOps assets
 - Versioning of Deployment Scripts and Source Code
- Access to Production-like Environments
 - Documentation of Production-like environments as Patterns
 - Developers have ability to launch and destroy production-like environments from these patterns
- Deployment Automation
 - Pattern based reusable deployment scripts
 - Ability to deploy applications in One-step
 - Daily deployment and verification of applications to a production-like environment

Source: [*12 Steps to Better DevOps – Michael Elder*](#)



DevOps Adoption (2 of 2)

- Change Management
 - Linking bugs, issues and work items to application changes
 - Linking production issues to associated deployment bugs
- Automated Testing
 - Automated testing is used to validate application and platform function and characteristics
- Monitoring
 - Monitoring Deployed applications to validate performance and reliability
- Delivery Pipeline
 - Having a dashboard to track application stages thru the delivery pipeline and track deployment velocity

Source: [12 Steps to Better DevOps – Michael Elder](#)



Continuous Delivery Adoption Maturity

➤ Common Source Control

➤ Automated Builds (Build Definitions)

➤ Continuous Integration (CI)

➤ Automated Delivery

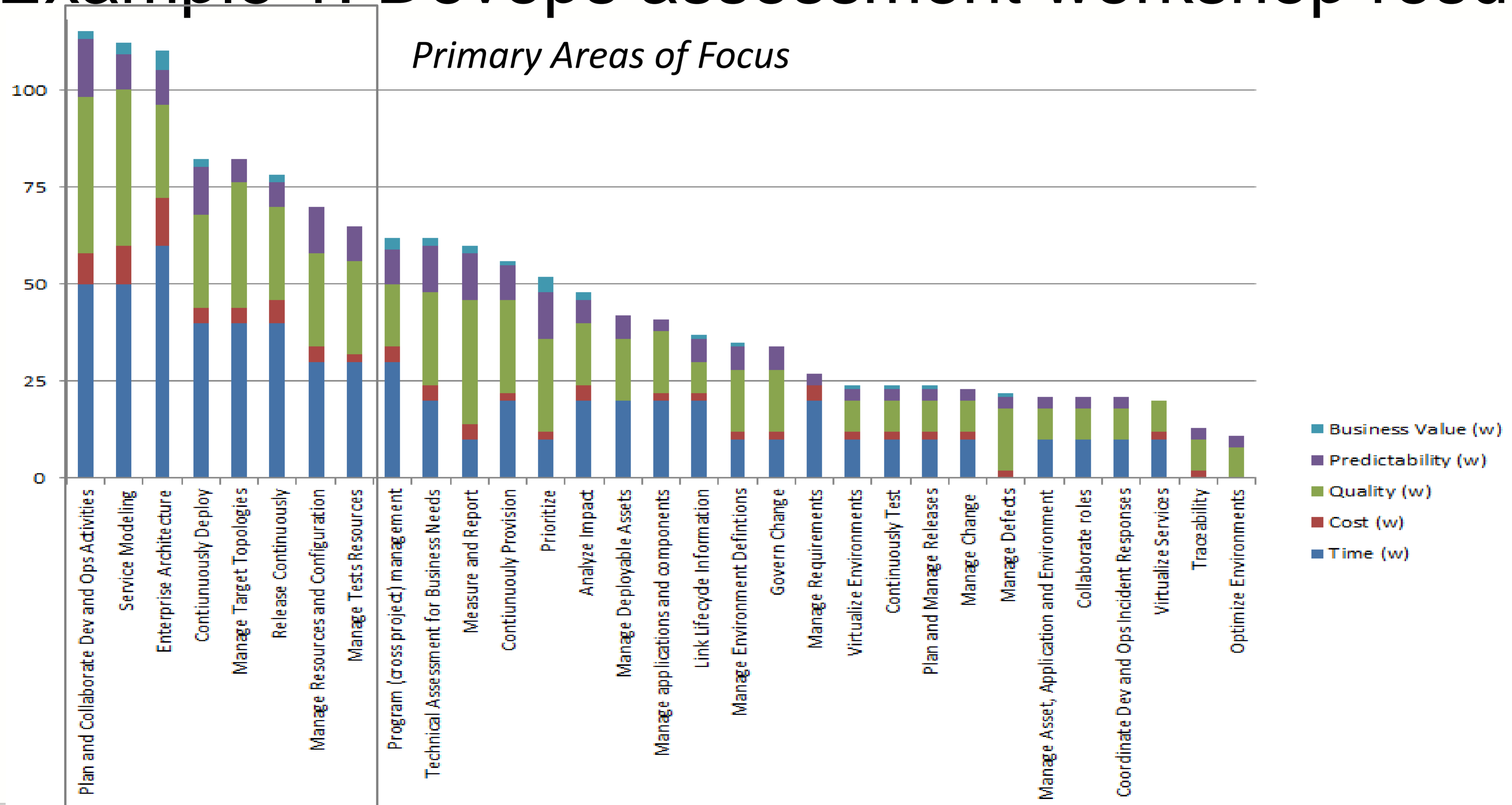
➤ Continuous Delivery to Test (CD)

➤ Continuous Delivery to Production-like Systems (Middleware as Code)

➤ Continuous Delivery thru Prod (Cloud based)

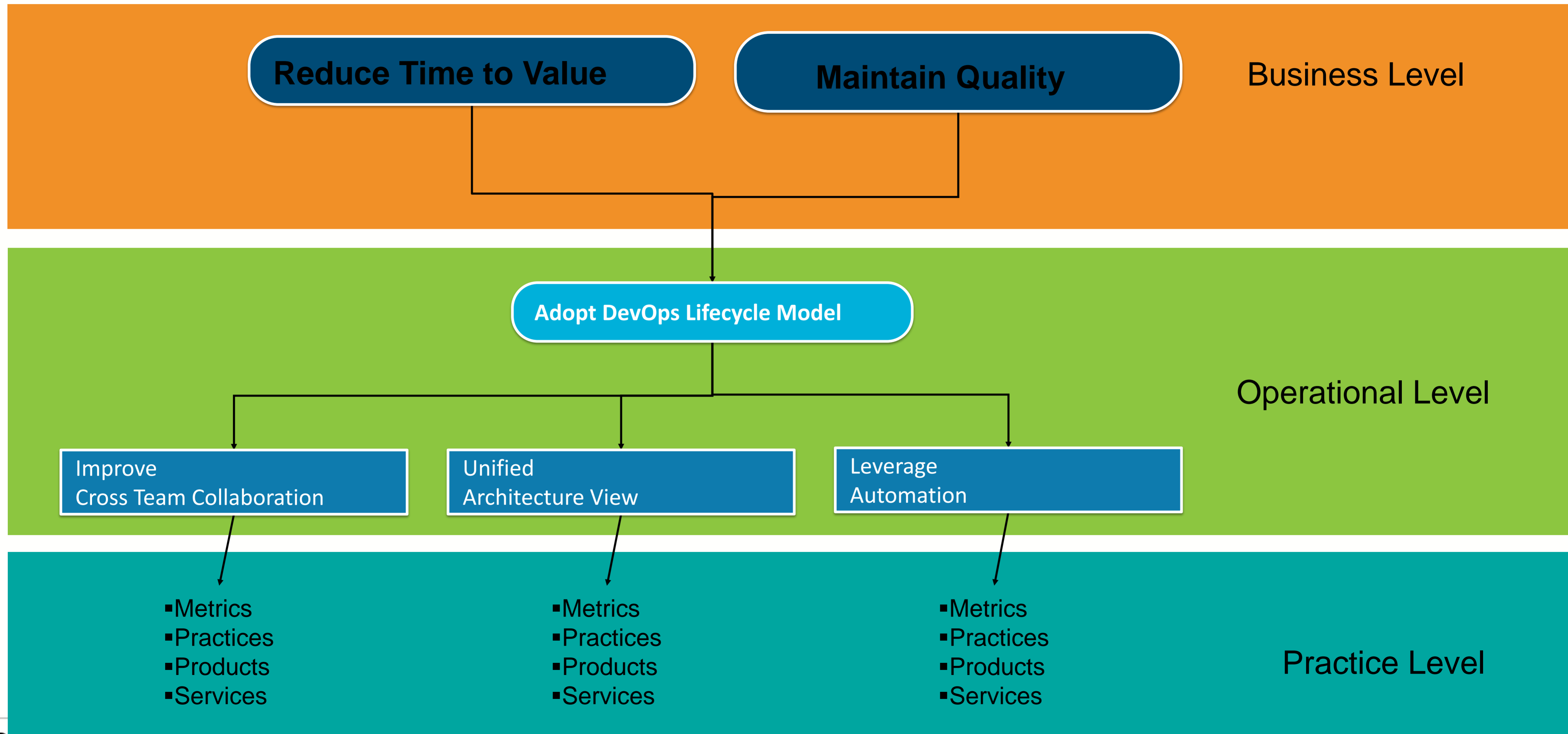


Example 1: Devops assessment workshop results





Example 2: Devops assessment workshop results (large governmental org.)





Example 3: Continuous Delivery workshop outcome from a bank in Europe

Release and deploy

- a- Release Traceability and governance
- b- Automated release and deploy capability
- c- Team member on boarding and collaboration
- d- Authority and gated release to production
- e- Provisioning of data in test environments
- f- Manual movement of data sets between database environments

Development and testing

- a- Standardization of automated testing
- b- Service virtualization and back-end simulation
- d- Code refactoring
- e- Common SCM

Organisational

- a- Formation of a steering committee: Essential in order to lead the efforts and increase awareness and adoption of DevOps in the organization.
- b- Standardization of terminology across the organization



DevOps Maturity Model – Traction around Release and Deploy

Industry norm

	Plan / Measure	Development / Test	Release / Deploy	Monitor / Optimize
<i>Scaled</i>	<ul style="list-style-type: none"> Define release with business objectives Measure to customer value 	<ul style="list-style-type: none"> Improve continuously with development intelligence Test Continuously 	<ul style="list-style-type: none"> Manage environments through automation Provide self-service build, provision and deploy 	<ul style="list-style-type: none"> Automate problem isolation and issue resolution Optimize continuously
<i>Reliable</i>	<ul style="list-style-type: none"> Plan and source strategically Dashboard portfolio measures 	<ul style="list-style-type: none"> Deliver and integrate continuously Manage data and virtualize services for test 	<ul style="list-style-type: none"> Standardize and automate cross-enterprise Automate patterns-based provision and deploy 	<ul style="list-style-type: none"> Optimize applications Use enterprise issue resolution procedures
<i>Repeatable</i>	<ul style="list-style-type: none"> Link objectives to releases Measure to project metrics 	<ul style="list-style-type: none"> Deliver and build with test Centralize test management Link lifecycle information 	<ul style="list-style-type: none"> Plan departmental releases and automate status Automated deployment with standard topologies 	<ul style="list-style-type: none"> Monitor using business and end user context Centralize event notification and incident resolution
<i>Practiced</i>	<ul style="list-style-type: none"> Document objectives locally Manage department resources 	<ul style="list-style-type: none"> Manage Lifecycle artifacts Schedule SCM integrations and automated builds Test following construction 	<ul style="list-style-type: none"> Plan and manage releases Standardize deployments 	<ul style="list-style-type: none"> Monitor resources consistently Collaborate Dev/Ops informally



Example 4: A proof of concept on continuous deployment (Large chemical engineering company)

Challenges:

- The legacy Java development does not have a CI server.
- It takes 3 hours for a user to build all projects in RAD and then deploy them.
- They do this type of deployment 10 times a week per test environment.
- The process is error prone.
- The shell script part was also manual and error prone.



Example 4: A proof of concept on continuous deployment (Large automotive company)

The use cases for the proof of concept.

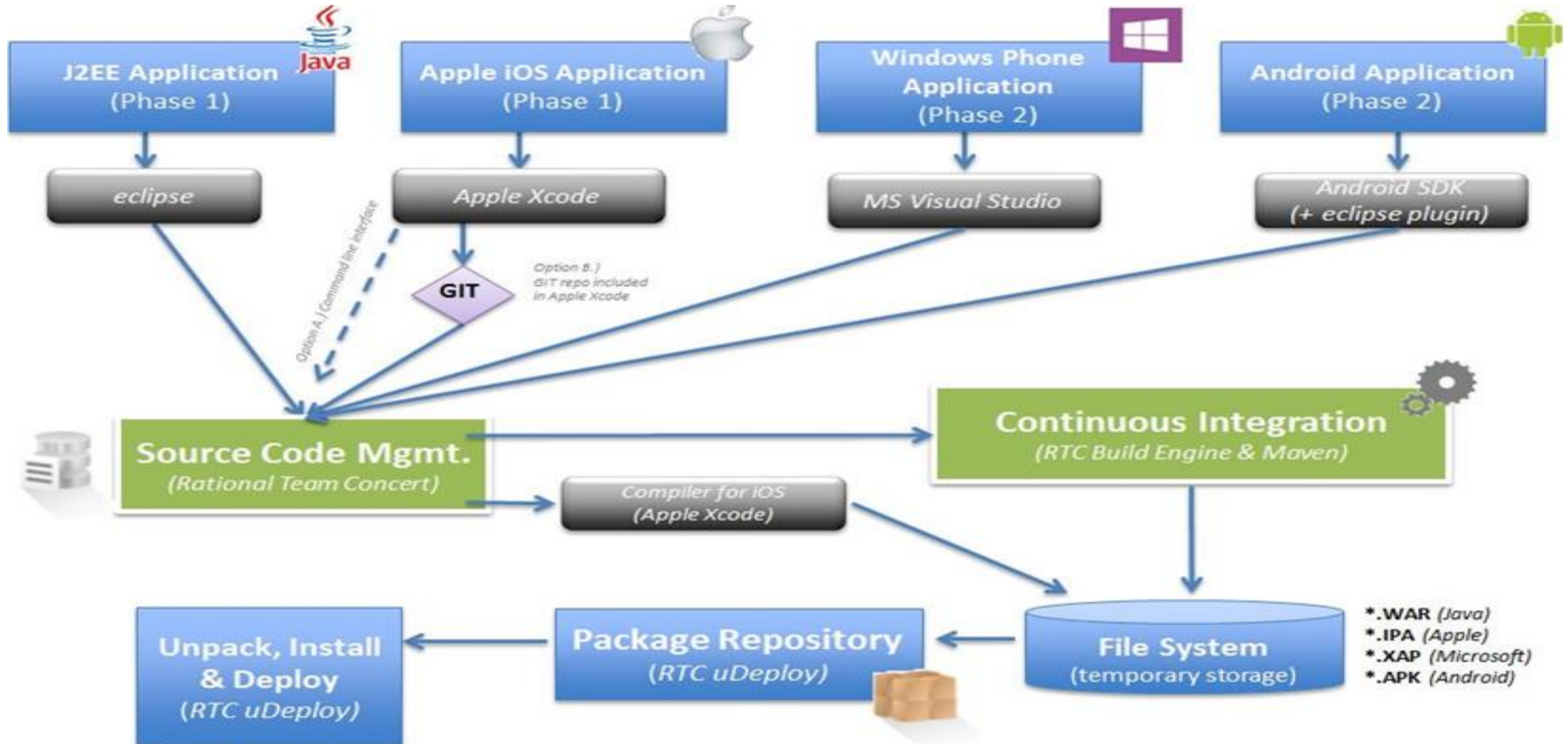
- (1) Java app deployment on WAS 7
- (2) Incorporate the Cobol deployment shell scripts.
- (3) Install & setup R.A.D Build Engine.
- (4) Demonstrate release orchestration.
- (5) Integrate Build & Deploy to setup „Continuous Delivery“

Results:

- **Deployment time down from 3 hours to 5 minutes**

- **Client got a CI server also for the legacy environment. They already used Jenkins for their modern environment.**

Example 5: A continuous delivery model with Rational tool chain for a Chemical industry client





Example 6: Deployment costs and savings model for the business case

ODAY: TOTAL COST OF DEPLOYMENT EFFORT PER YEAR		€ 2,525,400	
SAVINGS PER YEAR		€ 1,826,100	
REDUCE DEPLOYMENT EFFORT	75%	€ 1,485,000	
REDUCE ERRORS	90%	€ 153,900	
REDUCE SCRIPT WRITING	50%	€ 187,200	
DEPLOYMENT COST PER YEAR		€ 1,980,000	
DEPLOYMENT TO PRODUCTION			
# APP DEPLOYMENTS PER YEAR	AVERAGE APP DEPLOYMENT TIME	APP DEPLOYMENTS TOTAL TIME (HOURS)	COST
7000	4	28000	€ 1,680,000
DEPLOYMENT TO DEV \ TEST			
# APP DEPLOYMENTS PER YEAR	AVERAGE APP DEPLOYMENT TIME	APP DEPLOYMENTS TOTAL TIME (HOURS)	COST
2500	2	5000	€ 300,000
DEPLOYMENT ERROR COST PER YEAR		€ 171,000	
# APP DEPLOYMENTS PER YEAR	# ERRORS	DEPLOYMENT ERROR FIX TOTAL TIME	COST
9500	950	2850	€ 171,000
COST OF WRITING DEPLOYMENT SCRIPTS		€ 374,400	
SCRIPT WRITERS		FULLY LOADED FTE COST	COST
3		€ 124,800	€ 374,400



A good technical evaluation also needs a solid business case.

- IBM.com registration page:
- https://www14.software.ibm.com/webapp/iwm/web/signup.do?source=swg-rtl-sd-calc&S_PKG=ov26141&S_TACT=C25600PW

The screenshot shows the IBM UrbanCode Deploy ROI Calculator registration page. The page has a dark navigation bar with the IBM logo and links for Industries & solutions, Services, Products, Support & downloads, and My IBM. A search bar is also present. The main heading is "IBM UrbanCode Deploy ROI Calculator". Below this, there is a section titled "Assessing the value of Deployment Automation" with a small graphic of a rising line graph. The text explains that the ROI calculator helps in calculating the benefits and return of deployment automation across three main categories:

- Automating manual work: The simple reduction of deployment effort from automating a previously manual process.
- Reducing script writing and maintenance: Benefits from using pre-built integrations over writing your own scripts and tools to help with a deployment.
- Error prevention: Manual processes are error-prone. Benefits in both production and pre-production are examined.

Over the next several screens, you will be asked questions about deployment times, the number of releases per year, labor cost, outage costs, and such. You will have the opportunity to calculate the typical ROI on a given application, such as ERP, .Net, eCommerce, your Core Application, etc.

Get started!
→ No thanks, take me to your IBM UrbanCode Deploy page instead.

Asterisks (*) indicate fields required to complete this transaction.

How are you using this information?*

Which of the following best describes your company or organization?*

What role do you play in the acquisition of software solutions?*

Would you like an IBM representative to contact you regarding this IBM Software information?

Submit



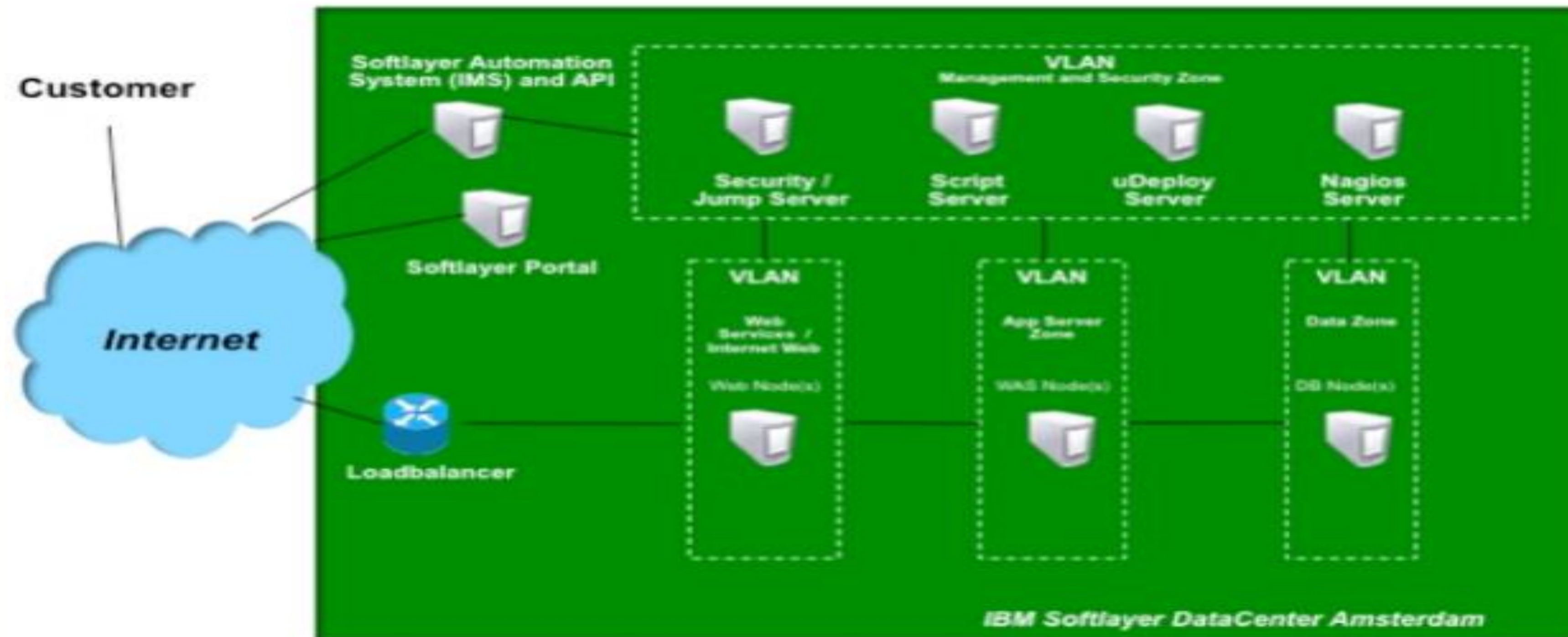
Deployment Challenges to take into consideration

- Considerations for creating appropriate standards and naming conventions
- Managing database deployments and rollback: Liquibase
- Incremental vs Full deployment (use of synch, synch and clean, etc)
- More effective code promotion and use of versioning
- Plugin, creation, usage and management.
- Opportunities for automating admin and housekeeping tasks
- Management metrics and reporting (custom reporting).
- Audit logging ; management and archiving
- Interfacing to other products: the integration ecosystem
- Server and agent performance considerations (e.g. relays)
- CLOUD provisioning

Example 7: Cloud Provisioning (automotive client) IBM UrbanCode Deploy and Softlayer IaaS



30 minutes to deploy a fully redundant WebSphere Application Server cell connected to an IBM DB2 cluster and deploy the latest version of the application on top of it within the IBM SoftLayer cloud data center in Amsterdam all controlled by IBM UrbanCode Deploy

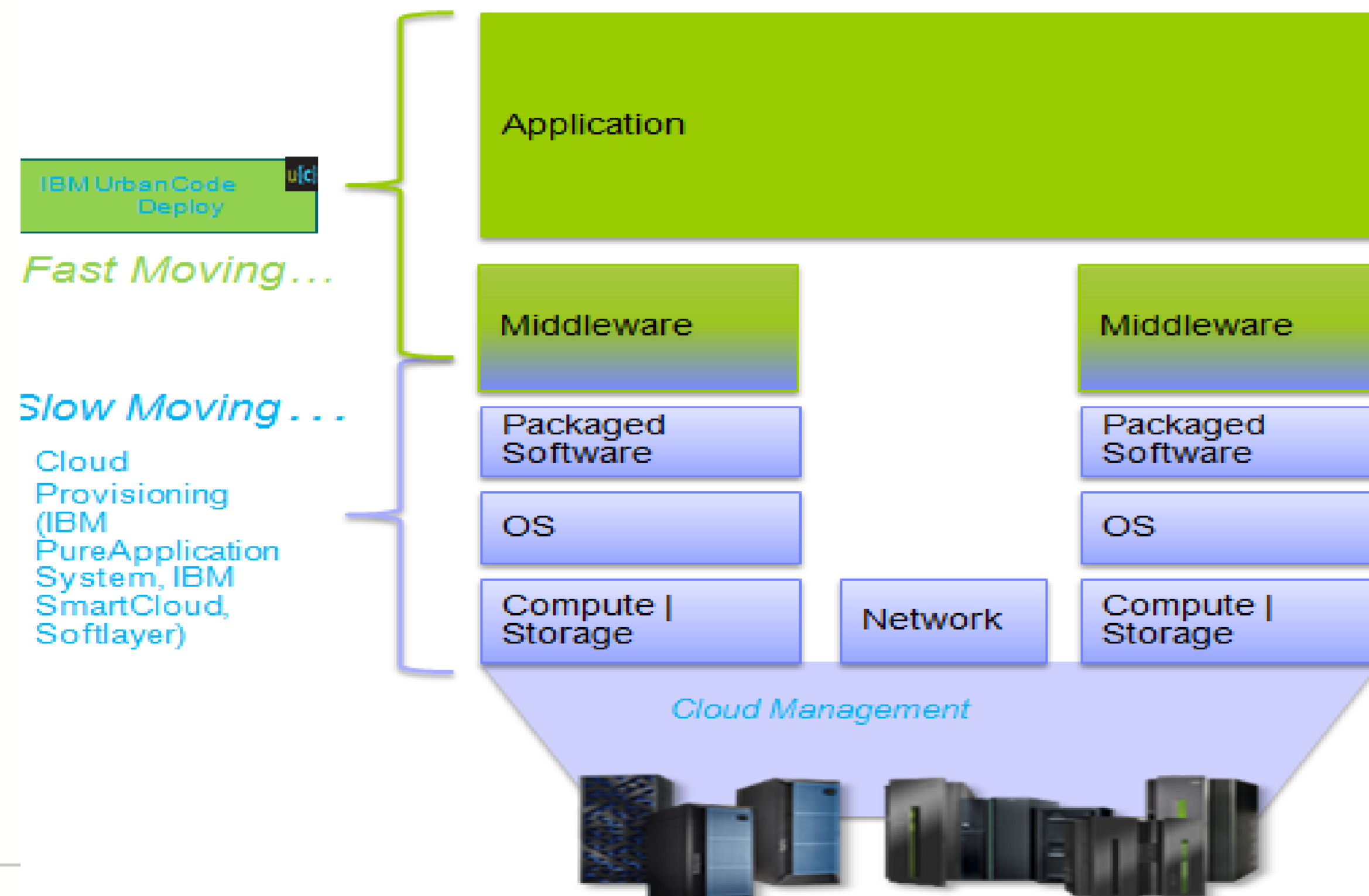


DevOps on IBM SoftLayer: Some real life experiences
Michael Brokmann,
IBM Executive IT Architect, Cloud Architecture



Example 7: Cloud Provisioning (automotive client) IBM UrbanCode Deploy and Softlayer IaaS

- **IBM UrbanCode Deploy:** Application deployment server and orchestrator of the other servers
- A **Nagios monitoring server** to monitor all systems that were provisioned automatically.
- A **script server** The scripts effectively were using the SoftLayer application programming interfaces, for example to instantiate cloud image templates. We were using SoftLayer Flex Images to do so.
- A **security server** that we were using for Secure Sockets Layer (SSL) tunneling to the provisioned environments and systems, which were not allowed to be reachable through the Internet.





Example 8: Leading International provider of investment funds

urban{code}

An IBM company



Fidelity Customer Success Story

Fidelity|Worldwide Investment Has Predictable Release Schedules with Assured Regulatory Compliance

Fidelity Evaluation Criteria

- ▶ Mapping of Applications and Environments
- ▶ Auditability and Reporting
- ▶ Security Model
- ▶ Reliability
- ▶ 3rd Party Integrations

Fidelity Benefits with uDeploy

- ▶ Cost Avoidance of over £1.5M (\$2.3M) a year
- ▶ Assured regulatory compliance
- ▶ Predictable release schedules for business stakeholders
- ▶ Test team “down-time” virtually eliminated
- ▶ Release processes take 1-2 hours versus 2-3 days
- ▶ Developers gain autonomy and self-service for deploying applications



Example 8: Leading International provider of investment funds

uDeploy Dashboard Components **Applications** Configuration Resources Deploy

Home > Applications > IM CRD

Application: IM CRD

Description: IM CRD Application

Environments History Edit Properties Components Snapshots Processes Calendar Manual Tasks Security

Environments

Drag environments by their label boxes to order them.

DV3 Request Process Compare Inactivate Copy

Latest Desired Inventory						
Component	Version	Snapshot	Properties	Status	Compliance	
IM CRD Batch	37		Version 5	Active	Compliant (2/2)	
IM CRIL Publisher Windows	1065		Version 4	Active	Compliant (1/1)	
IM CRD Batch	179		Version 1	Active	Compliant (2/2)	
IM CRIL EvtMgr	1.0.318	CRIL v1.2.2.00	Version 6	Active	Compliant (1/1)	

10 per page 21 records - [Refresh](#) [Print](#)

UA1 Request Process Compare Inactivate Copy

Latest Desired Inventory						
Component	Version	Snapshot	Properties	Status	Compliance	
IM CRD EventManager	249		Version 1	Active	Compliant (1/1)	
IM CRIL Publisher Windows	782	CRD IL 28 2.006	Version 1	Active	Compliant (2/2)	

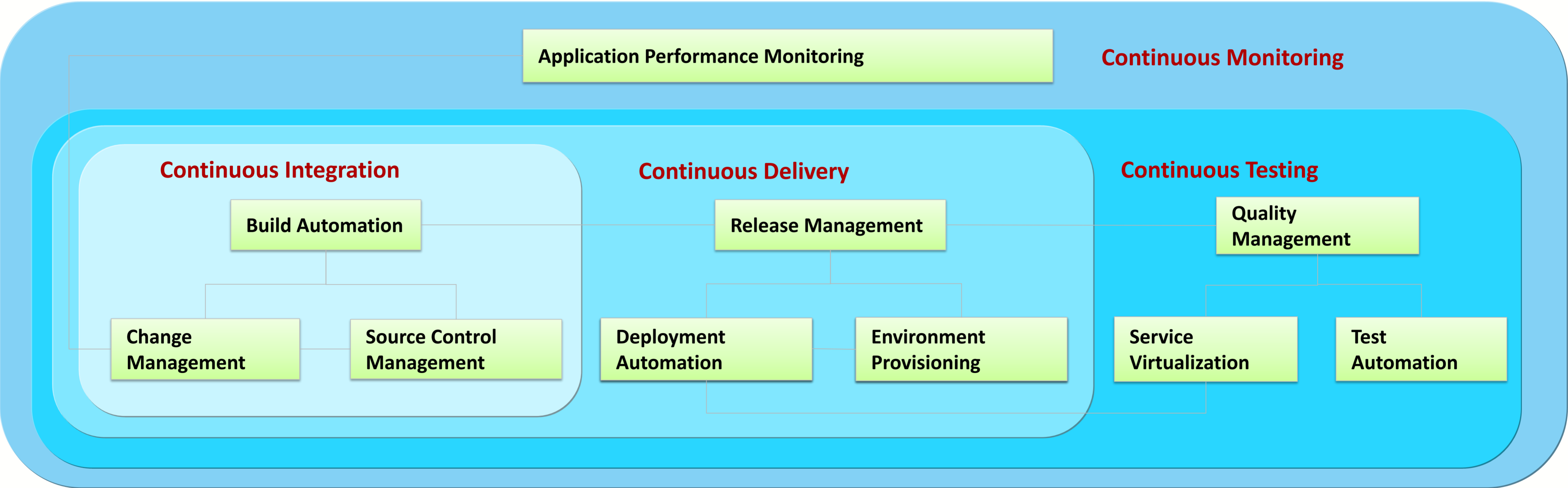
10 per page 19 records - [Refresh](#) [Print](#)

ST1 Request Process Compare Inactivate Copy

Latest Desired Inventory						
Component	Version	Snapshot	Properties	Status	Compliance	
IM DTRG	1.0.241		Version 2	Active	Compliant (2/2)	



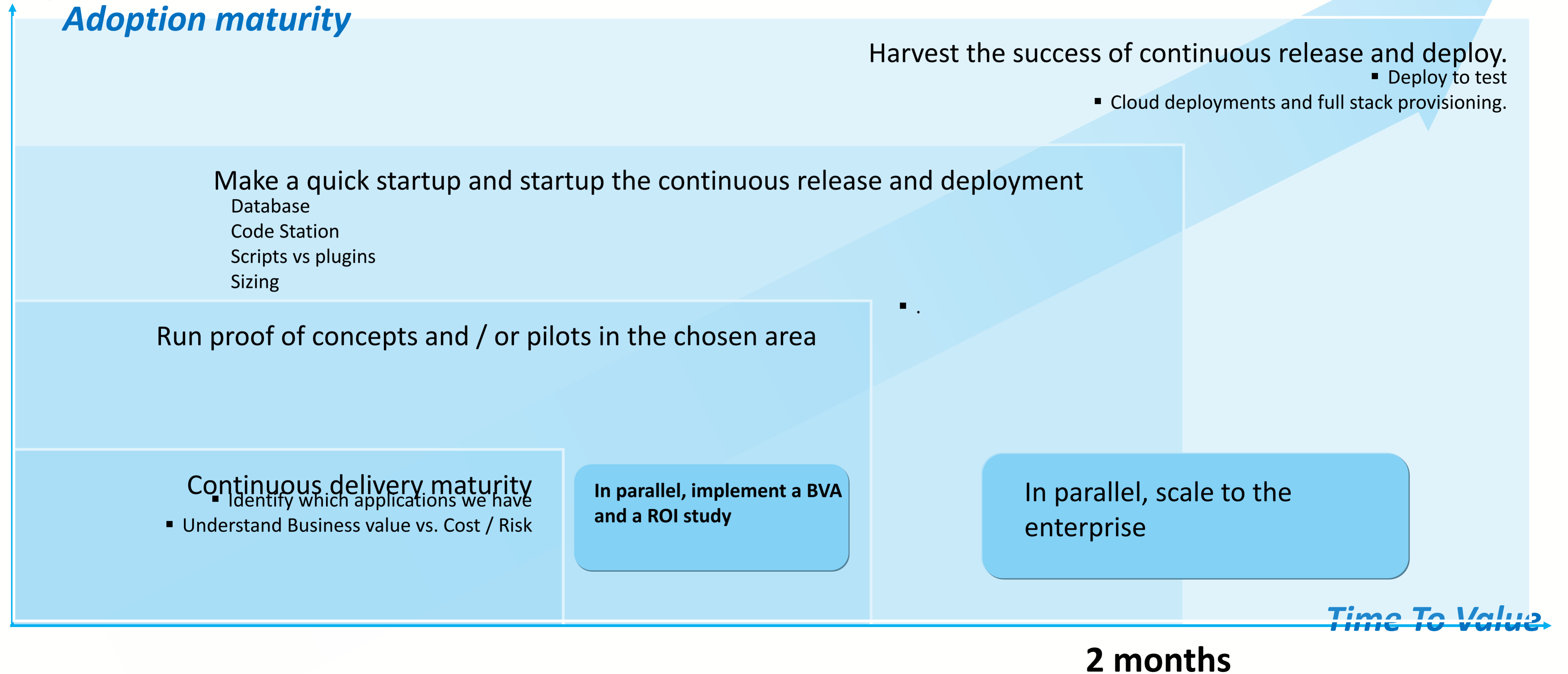
Summary: The road to DevOps





Adopting Continuous Delivery

staged and incremental approach





Fidelity Customer Success Story

Fidelity | Worldwide Investment Has Predictable Release Schedules with Assured Regulatory Compliance

*“Fidelity’s application release process has been fully automated with uDeploy. Applications that took **days** to release now take **just an hour!**”*

Tony Green Technology Architecture and Engineering

“UrbanCode is considered a strategic partner at Fidelity International (FIL). We work together closely on release automation product requirements and directions”

Tony Green Technology Architecture and Engineering



IBM Release and Deploy Solutions

Value Proposition

Time to Value

Benefits (actual customer results)

Immediate

(under 2 months)

Time-to-value

- Reduce deployment time by up to 95%
- Augment the number of deployment targets

Cost Reduction

- Cost avoidance.
- Eliminate test team downtime

Reduce Risk

- Regulations compliance
- Reduce deployment outage by 90%

Short Term

Mid Term



Thank You

© Copyright IBM Corporation 2014. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.

Innovate2014





References

- [12 Steps to Better DevOps:](https://www.ibm.com/developerworks/community/blogs/devops/entry/12_steps_to_better_devops11?lang=en)
https://www.ibm.com/developerworks/community/blogs/devops/entry/12_steps_to_better_devops11?lang=en
- IBM Enterprise DevOps blog
 - <http://ibm.co/JrPVGR>
- 6 Ways for Enterprises to Adopt DevOps blog
 - <http://ibm.co/xq71xY>
- Understanding and Adopting DevOps (Series on my Blog)
 - <http://bit.ly/MyDevOps>
- DevOps learning circle community: (on developerworks):
<https://www.ibm.com/developerworks/community/groups/service/html/communityview?communityUuid=860ff390-6cab-4f95-ab37-66d2ca7521b4>
- Automative client example:
 - https://www.ibm.com/developerworks/community/blogs/invisiblethread/entry/devops_on_ibm_softlayer_some_real_life_experiences?lang=en
 - https://m.youtube.com/watch?utm_medium=social&v=q24NGfWx56Q&utm_source=linkedin&utm_content=4699715