

*Ebook*

# The Blueprint for Continuous Delivery & DevOps of Oracle



# CONTENTS

---

- 1.** Who this blueprint is for
- 2.** What you'll gain from this blueprint
- 3.** Set your expectations
- 4.** The top 5 challenges in E2E oracle deployments
- 5.** How do smart enterprises resolve the challenges?
- 6.** How you make these solutions work?
- 7.** Implementation options
- 8.** About LimePoint
- 9.** About the Authors
- 10.** Terms used in this blueprint
- 11.** References

# 1.

## WHO THIS BLUEPRINT IS FOR

This blueprint is for IT specialists and operations managers of enterprises whose businesses run on Oracle technologies.

You'll get the most from this blueprint if your role is any of these:

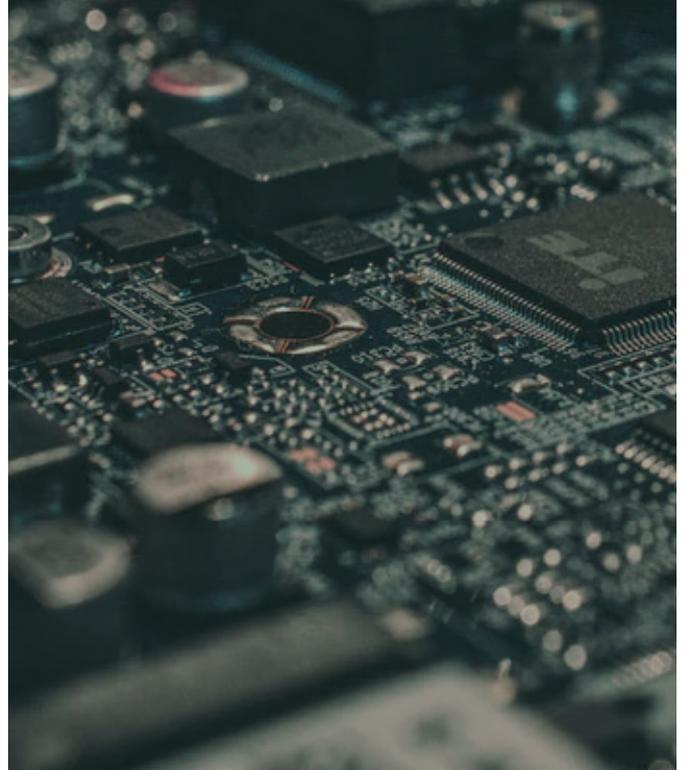
- Oracle Applications Team Leader
- Project or Program Director or Manager
- Architect or Subject Matter Expert
- CTO, CIO or IT Manager
- Environment or Operations Manager.

Your business is pushing to stay ahead of competitors through **innovations that add value, sharpen your competitive edge or improve customer experience**. Ensuring that your Oracle platforms deliver key projects reliably, consistently and with high quality is critical to your success.

**Your technical environment is more complex than ever**, increasing the risk of failures, downtime and disruptions that can harm both your business and reputation. Deployment options like public or private cloud, or on-premise/cloud hybrids provide flexibility, but add complexity.

**'Business As Usual'** is good for operations, but won't improve business agility, **Continuous Delivery** promises to improve quality and speed of deployment, and **DevOps** is touted as the Holy Grail.

*This fully-referenced blueprint will show how you can satisfy these competing needs and achieve business agility and value - without blowouts in your resources, budgets or timelines.*



“

*‘Oracle is basically selling an integrated one-stop-shop solution as opposed to an a la carte, best-of-breed approach. Oracle (like its biggest competitors, IBM, SAP and Microsoft) is aiming to serve companies that want as much technology as possible from a single source...’<sup>1</sup>*

**Mark Driver, Gartner in TechTarget**



## 2.

### WHAT YOU'LL GAIN FROM THIS BLUEPRINT

Oracle provides a comprehensive array of business solutions across myriad applications, middleware, database, virtual machines and more (*see page 17*).

To build one environment for one Oracle solution requires complicated configuration and time. With complex or diverse deployments of many Oracle solutions, the challenges mount up quickly.

*This blueprint show how to resolve these challenges and accelerate end-to-end Oracle deployments.*

In particular, it shows you how to:

- Design & build Oracle environments in hours not months
- Detect, prevent or fix Configuration Drift in real time
- Make DevOps a reality with Continuous
- Delivery Automation, and much more

“

*'...clients mention challenges due to long and cumbersome installation and setup processes, often requiring specialized skills, as well as complexity in monitoring, management and problem determination.'*<sup>2</sup>

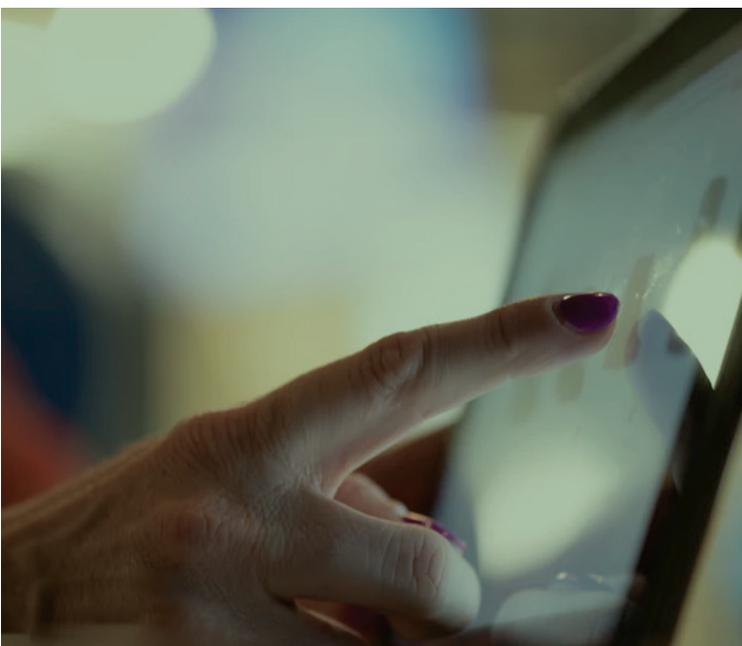
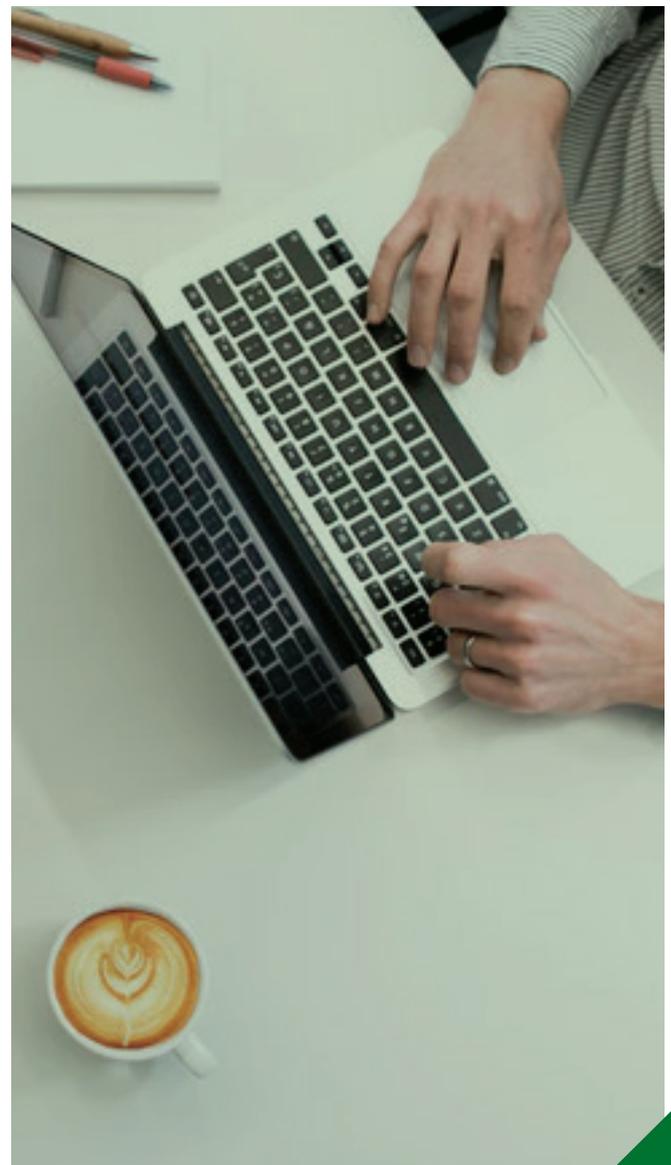
**Gartner Magic Quadrant, On-Premises Application Integration Suites 2014'**

## 3.

### SET YOUR EXPECTATIONS

For many organisations with complex Oracle deployments, the following are on their wish lists. They're probably on yours too.

- **No need to wait** for environments to be built, patched or changed
- **Ability to orchestrate Oracle deployments** without armies of experts
- **Faster delivery of projects** due to more consistent environments
- **Fewer outages or losses** caused by environment configuration errors
- **Ability to detect and prevent issues** before they impact the business
- **Lower costs, easier management** by moving to cloud or hybrid cloud
- **Continuous Delivery** of software across the whole lifecycle
- **Ability to rollout and roll back releases** with ease
- **Projects delivered on time and on budget** with high quality



# 4. THE TOP 5 CHALLENGES IN E2E ORACLE DEPLOYMENTS

## 4.1 Too long to deploy new environments

To deliver new products or services via Oracle solutions, you'll need quality environments, but **it can take months to build and deliver just one complex Oracle environment.**

That's because of the time and resources needed for manual builds and the human error they introduce. Human error leads to inconsistencies, instability and defects which, if left detected, can lead to failures in production.

Using your own internal teams for major Oracle projects might seem economical, but rarely would they have the depth and breadth of experience that consulting **SMEs** have gained over numerous projects. If the plan is to manually build anyway, the in-house option probably won't achieve the quality and speed you seek.

**Many project leaders turn to Systems Integrators, but they can bring more people to the project,** rather than the automation needed to both accelerate deployments and improve their quality.

**“**  
*'If you just deployed ten or fifteen different configuration changes across the core and per-app service infrastructure and suddenly something is “wrong”, it's going to take some time to figure out exactly which configuration change caused the problem in the first place'* <sup>3</sup>

**Lori MacVittie, Enterprise DevOps**

## 4.2 Cost blows-outs

**It's common for large IT projects to run over budget.** According to McKinsey<sup>4</sup>, this happens almost half the time.

**The risk of cost blow-outs escalates when new enterprise systems are built.** That's because unforeseen obstacles lead to architectural or design changes, that only surface as the project unfolds. These can have a major impact on project cost and resources needed for completion.

**Of course, current legacy systems have to be maintained while new systems are being developed.**

This means any overrun in project time will have a double blow-out in costs, because maintenance of legacy systems will continue beyond planned time frames.

**Other cost blowouts** range from the **need to rework deployments** because of human error, to **changes in project scope** to accommodate additional requirements, **to adding extra people** when delivery schedules are under threat.

**“**

*'...half of all large IT projects massively blow their budgets. On average, large IT projects run 45% over budget and 7% over time, while delivering, 56% less value than predicted.'* <sup>4</sup>

**McKinsey & Company, Insight Publication**



## 4.3 Devops, failures and downtime

**DevOps is a much-used term** that evokes strong and differing responses, **yet its relevance in complex or diverse Oracle projects is clear.**

*How else do you prevent the usual stand-off between the Development and the Operations team, with the latter protesting it's been thrown a 'ticking time bomb' of new Oracle deployments it can't manage, modify, remediate or enhance?*

Apart from resolving this stand-off, DevOps plays a key role in successful Oracle deployments, largely because of its key objectives:

- Delivering capabilities to business users faster
- Reducing the failure rate of new releases
- Minimising the lead time between fixes
- Improving system quality, and
- Speeding up mean-time to recovery in the event of failures introduced into production systems.

DevOps has been compared to the Holy Grail: if its objectives can be achieved by closely aligning the activities of Dev and Ops teams, it will speed up your response to the business's demands and better support its vision.

It's not that easy in practice: Recent **research from ZeroTurnaround**<sup>5</sup> found that **60%** of the failures in **DevOps projects** are caused by **human error or lack of automation**, and **IDC's 2014 survey of Fortune 1000 companies**<sup>6</sup> found that trying to adapt current tools to deliver DevOps practices has a failure rate of 80%.

These are the findings from IDC's 2014 survey of Fortune 1000 corporations<sup>6</sup>:

- Average total cost of unplanned application downtime per year: \$1.25 billion to \$2.5 billion
- Average hourly cost of an infrastructure failure: \$100,000 per hour
- Average cost of a critical application failure per hour: \$500,000 to \$1 million
- Average number of deployments per month are expected to double in two years

Reducing the incidence and cost of failures and downtime is clearly critical, yet to do so and make DevOps a reality, IDC says you'll need better tools.

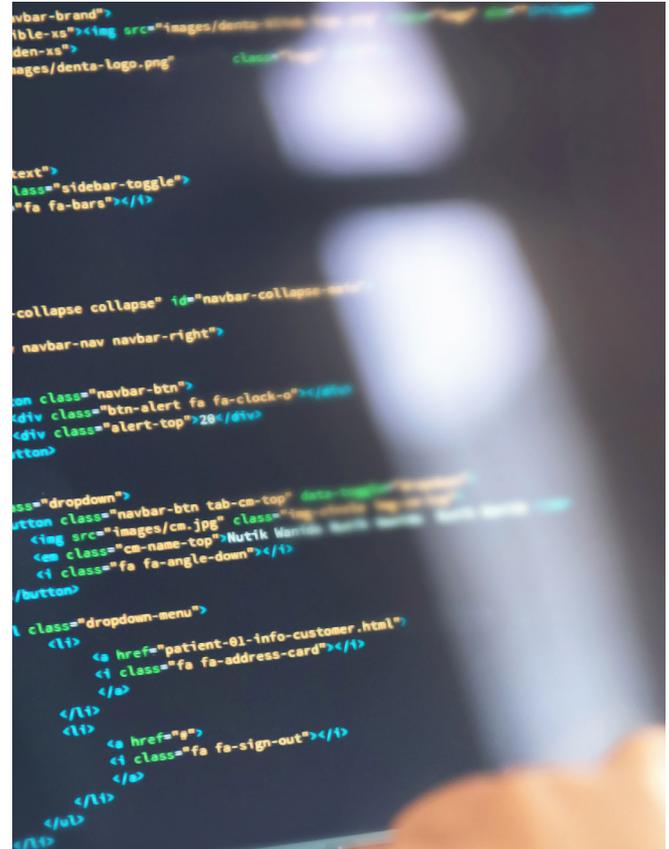
A recent study<sup>7</sup> by the Project Management Institute (PMI) shows you'll need better communications too, as poor communications account for 30% of project failures.

“

*'...Average cost of a critical application failure per hour: \$500,000 to \$1 million...'*<sup>6</sup>

**IDC**

**DevOps and the Cost of Downtime: Fortune 1000 Best Practice Metrics Quantified**



“

*'IT organisations that have tried to custom adjust current tools to meet DevOps practices have a failure rate of 80%, thus making tool replacement and/or addition a critical requirement.'*<sup>6</sup>

**IDC**

**DevOps and the Cost of Downtime: Fortune 1000 Best Practice Metrics Quantified**

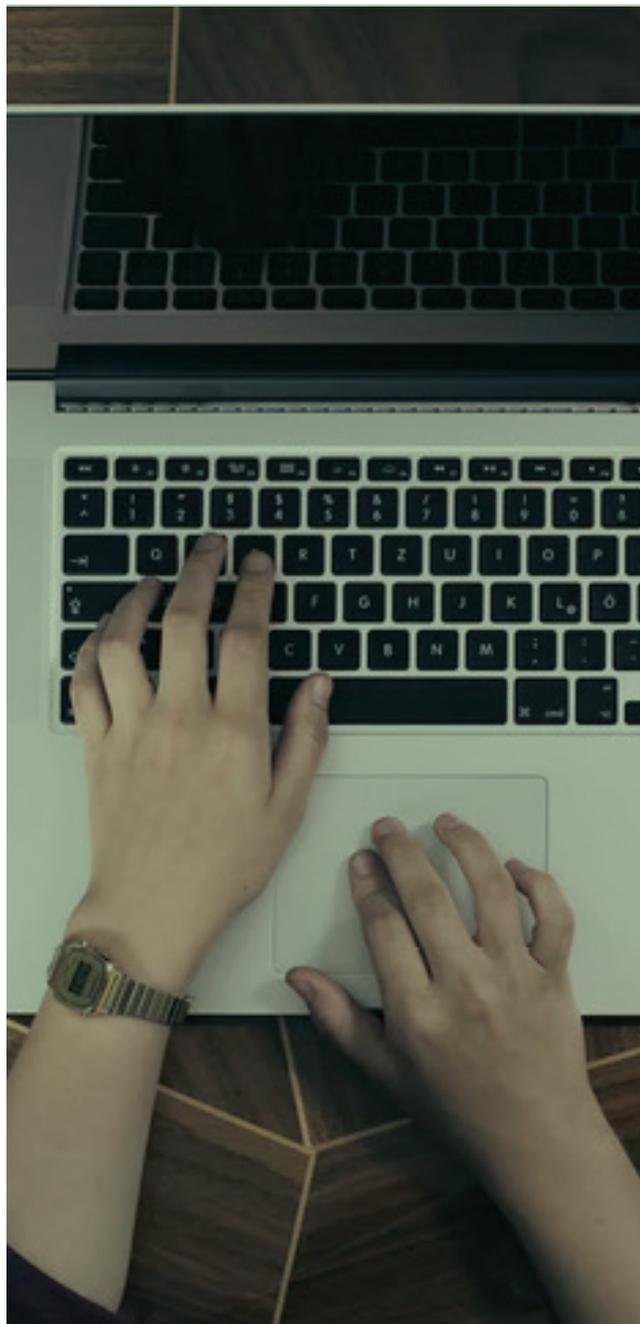
## 4.4 Increasing complexity

Oracle solutions can be very complex: in large enterprises it's common to have **50 to 100 Oracle application assets deployed**, using up to **40 distinct Oracle products and hundreds of clusters** across multiple environments.

In addition, you might have non-Oracle systems including legacy ones, too. That's why it's critical to configure consistent environments for the design, build, test and production environments of all your Oracle platforms.

Without automation, human error will cause differences between environments so that, for instance, what worked in Dev doesn't work in Test. This means that your people have to spend time figuring out what went wrong, instead of moving on to the next milestone.

The greater the complexity, the more valuable time will be lost, and this means that urgent business initiatives miss target dates.



“

*'Architects have one system for prototyping; developers have a different system to code on; test and QA have their own unique 'standard' systems; while operations have untouchable systems for production...*

*Add to this the complexity of backending into CICS or IMS applications, accessing data from DB2 or ADABAS on iSeries, or automating WebSphere deployment onto HP/UX, plus licensing for an Oracle database on AWS or an SAP client on Azure...'*<sup>8</sup>

**Andi Mann, CA Technologies**

## 4.5 Configuration Drift

You've probably experienced the **impact of Configuration Drift when a new software release fails on deployment**. The cause may be small discrepancies between environments, which can be hard to detect and harder to fix. Yet, most instances of Configuration Drift have innocent causes like these:

- Team members apply a minor fix, update software to a new version or install a conflicting package or service
- Software or operating system updates are applied without sufficient thought given to the consequences
- Small changes in the configuration of operating systems, middleware or applications

“

*'The goal is to enforce consistency, and to manage our resources so that software is deployed in the same way that a car factory builds an automobile.'*<sup>9</sup>

**Mike Fal, Devops and the DBA**

**The problem is, you can't fix serious bugs in the production environment by going back to your test environment, once configuration drift between them is an issue.** That means having to fix problems in Production which is painful; it's not designed for that. Even if you're able to correct the defect in Production, the fix won't be applied to preceding environments such as Pre-Production.

That means your change will be overwritten with Pre-Production values next time you deploy to Production. As Lori MacVittie reminds us: **'the most difficult tasks in the network are not provisioning or configuration, but troubleshooting ...** if you just deployed ten or fifteen different configuration changes across the core and per-app service infrastructure and suddenly something is "wrong", it's going to take some time to figure out exactly which configuration change caused the problem in the first place.<sup>3</sup>

According to Mike Fal, **the best way to avoid these problems is to make your environments consistent, just like a factory production line:** 'The reality is that chaos, instability, and down-time are not the result of speed, but the result of **variance** ... The goal is to enforce consistency, and to manage our resources so that software is deployed in the same way that a car factory builds an automobile. Components should be standardized, builds made consistent, and tasks automated. The result is speed ...Yes, speed, but speed that is the result of control and standardization.<sup>9</sup>



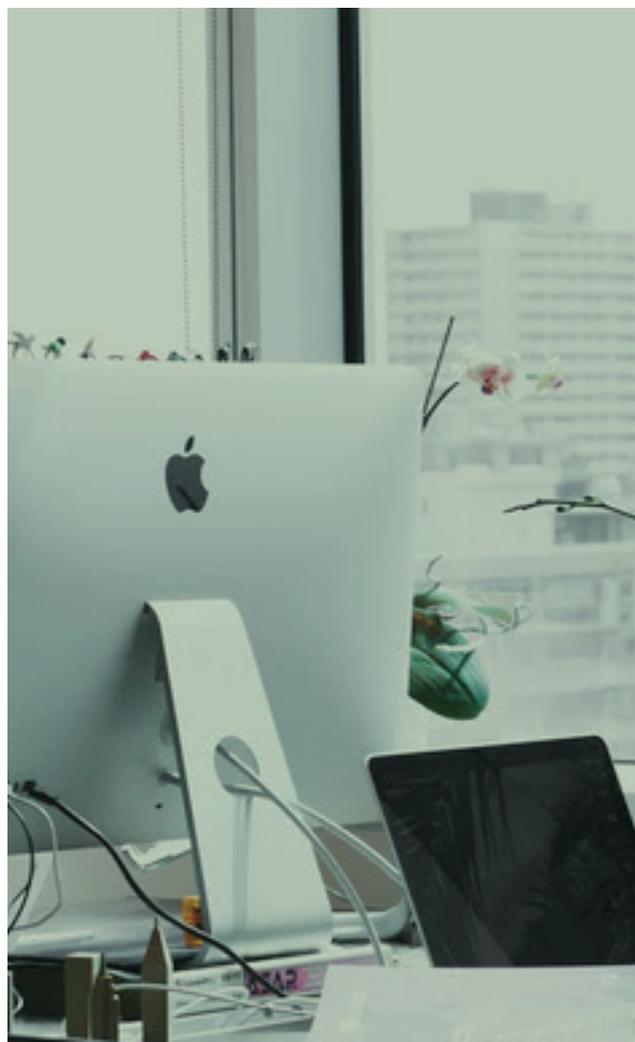
## 5. HOW DO SMART ENTERPRISES RESOLVE THESE CHALLENGES?

Many advanced, competitive enterprises rely on Oracle technologies for most of their business capability.

Yet, it takes highly skilled people and smart tools to master the discrete elements of Oracle, and tie them altogether so they work.

Smart enterprises focus on 3 crucial areas to avoid the most common challenges (see next pages):

- 1. Continuous Delivery Automation**
- 2. Real-time Diagnostics**
- 3. Specialist skills.**



“

*'...like the proverbial Swiss Army Knife, OFM has something for everyone... However, unlike that multipurpose tool, the elements of OFM are compatible, but discrete - united by a brand rather than a single look and feel'.*

**Alan R, Earls in TechTarget**<sup>10</sup>

## 6.1 Continuous Delivery Automation

**Continuous Delivery (CD) is more than an attractive ideal; it's essential for any agile, competitive enterprise.**

Continuous Delivery is the opposite of the traditional approach, where big new systems or upgrades take months or years to design, build, test and release.

**Continuous Delivery adds software enhancements and updates in small batches**, faster and more frequently, and the best CD products include a roll-back option, in case delivery causes unexpected problems.

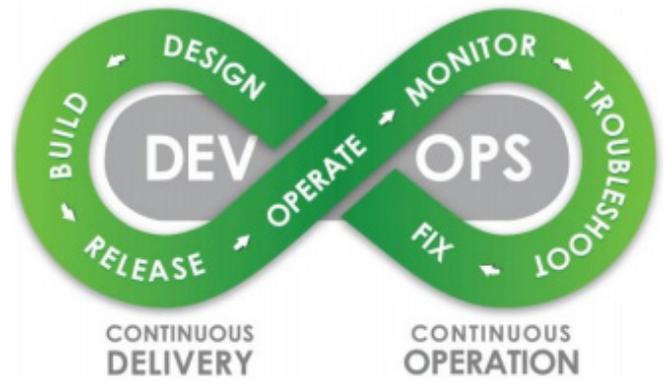
**The main reason to automate Continuous Delivery is to get new initiatives to market faster**, and to eliminate manual processes so that building of new environments is **repeatable and consistent**. Continuous Delivery is based on the principles of DevOps, which ensure closer collaboration of project teams and greater productivity.

**As the cost and complexity of your projects increases, the case for automation becomes more compelling.**

**Smart enterprises are automating everything** - including the provisioning of infrastructure, databases, OFM and applications - and look for the flexibility to do so either on premise or in the cloud, or in a hybrid of both.

**The CD product you select should enable you to:**

- Build & configure complex environments in hours
- Provision & manage high availability, clustered Oracle environments on demand
- Configure environments once and build them the same way each and every time
- Design Oracle environments via a simple user interface
- Use pre-defined pattern templates, or easily design and configure your own
- Leverage the tool as your Oracle Subject Matter Expert and reduce the need for many Oracle experts
- Deploy your infrastructure, databases, OFM and applications where you want them, including public or private clouds, hybrid clouds or in a mix of cloud and on-premise scenarios - giving you real power & choice
- Detect configuration discrepancies across environments and fix them in real time, without impacting the business

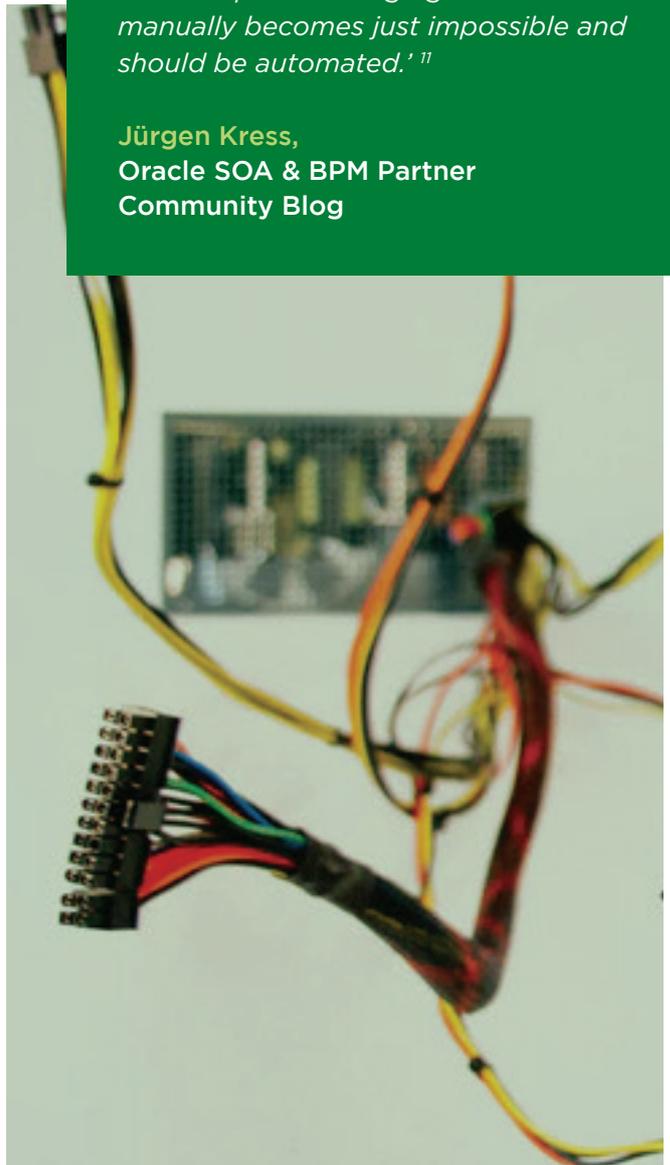


“

*'Any IT organisation sooner or later has to deal with such thing as Continuous Delivery (CD). They realize that there are various environments such as development, QA, support, UAT, production, etc. and there are a number of different systems working in those environments.'*

*At some point managing all that stuff manually becomes just impossible and should be automated.'* <sup>11</sup>

**Jürgen Kress,**  
**Oracle SOA & BPM Partner**  
**Community Blog**



## 5.2 Real-time Diagnostics

**Automating the building of complex Oracle environments is one thing; maintaining them efficiently is quite another.** So is real-time troubleshooting to ensure high availability.

**The smartest diagnostic toolsets enable easy maintenance and trouble-shooting, with advanced capabilities via real-time dashboards:**

- Deploying and verifying configuration change and measuring configuration drift
- Preventing configuration drift
- Managing user-induced changes to environments
- Detecting and preventing failures before they happen
- Eliminating inconsistencies between environments, and instability
- Harvesting configurations from existing environments and moving them back to the CD product to produce identical builds

**Real-time diagnostics provide even greater value, if they can also:**

- Be used in manually-built environments
- Be used for Oracle and non-Oracle deployments
- Enable export of detailed technical configuration reports for distribution and remediation by BAU

“

*‘...the most difficult tasks in the network are not provisioning or configuration, but troubleshooting’<sup>3</sup>*

**Lori MacVittie, Enterprise DevOps**

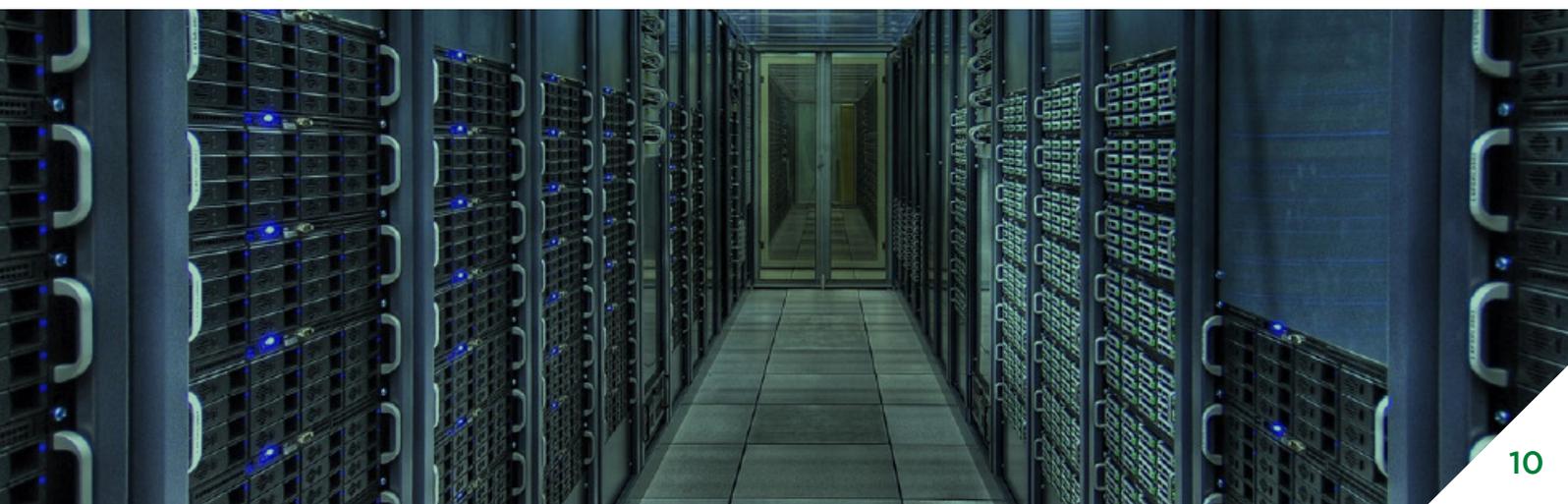
## 5.3 Specialist Skills

**When seeking help with projects based on Oracle, most organisations take advantage of in-depth expertise, either by hiring in-house or outsourcing.**

Oracle deployments typically consist of many different parts of the Oracle toolset. This includes OFM modules such as SOA, BPM, IdM plus other Oracle technologies such as Engineered Systems, Cloud and Mobility and more.

**That’s why smart organisations choose Oracle specialists whose expertise is not confined to the OFM part of the stack.**

They should be familiar with Oracle databases, operating systems, applications and industry solutions such as the Oracle Banking Platform, Human Capital Management, Enterprise Resource Planning, Supply chain Management and more. In other words, the entire Oracle Technology stack.



**Smart enterprises also choose specialists with experience that is highly relevant to their situations, such as:**

- Familiarity with the standards in their IT and business environments
- Experience with similar industries and projects
- Substantial architecture experience across many mission-critical systems
- Knowing how to make DevOps work for Oracle

**Many enterprises will hire Systems Integrators for major projects,** and then discover that SIs are good at a theoretical level, but don't have people with 'big DevOps' experience, that is, practical implementation of DevOps projects in large enterprises.

Specialist Oracle consultants can add insights and experience that make a crucial difference, right from the start. They know the Oracle space inside out, their people have worked in your industry and situation before, and they know what works and what doesn't.

**The most experienced specialists are often the architects of advanced Continuous Delivery Automation and Real-time Diagnostics toolsets for Oracle.**

That's why this is a key selection criterion.

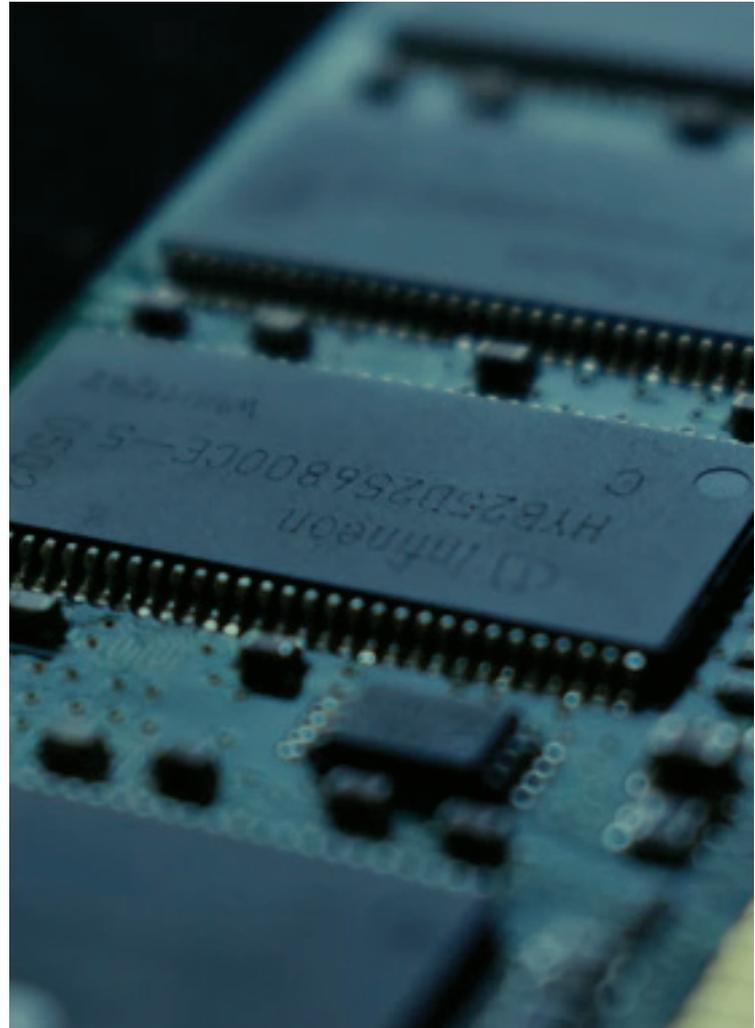
These specialists have deep technical knowledge which informed the development of very smart tools which have reality what used to be impossible: creation of quality Oracle environments in minutes, not weeks or months.

The best Oracle specialists will run a POC (Proof of Concept) to show what their automation and diagnostic tools can achieve, not in theory but in your particular situation.

They'll also train your staff, SI or other contractors in how to use them, so you can carry out ongoing management and trouble-shooting, cost-effectively, inhouse.

“

*The most experienced specialists are often the architects of advanced Continuous Delivery and Real-time Diagnostics toolsets for Oracle'sho'*



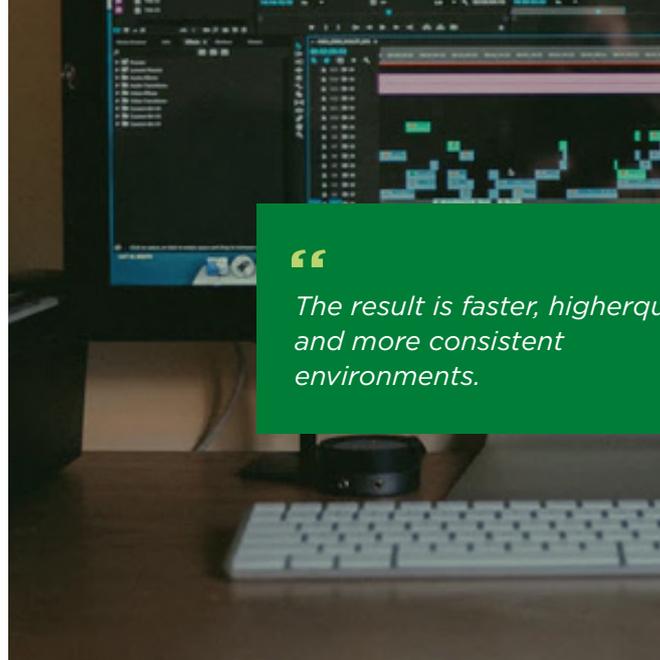
## 6. HOW DO YOU MAKE THESE SOLUTIONS WORK?

### 6.1 Small Batches

**Traditionally, delivery of large-scale systems requires a herculean effort from your IT department, involving years of building and testing.** Often this slow, 'big bang' approach threatens the very competitive advantage you hoped it would deliver.

**The key objectives of DevOps are the reverse - to deliver capabilities much more quickly,** and Continuous Delivery Automation makes that possible through frequent, small batch changes. This way, new or updated capabilities are delivered rapidly, reliably and repeatedly, with minimal manual interaction and risk, and much higher quality. The eight key differences between traditional and DevOps approaches<sup>12</sup> are shown below, courtesy of Mustafa Kapadia at DevOps.com.

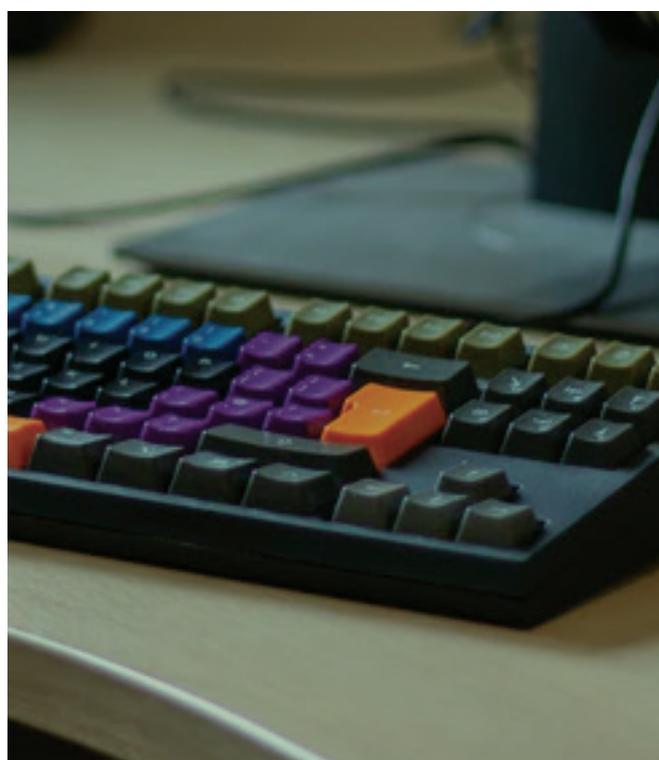
Small batch sizes are also easier to work with, and testing and deployment can be largely automated. They're less risky to deploy too, as small changes can be easily rolled back with minimal or no impact. You can also communicate them more easily and clearly to affected teams.



“  
The result is faster, higher quality and more consistent environments.”

	Dimensions	Traditional IT	DevOps
Planning & Organisation	Batch Size	Big	Micro
	Organisation	Skill Centric Silos	Dedicated Cells
	Scheduling	Centralised	Decentralised & Continuous
Performance & Culture	Release	High Risk Event	Non Event
	Information	Disseminated	Actionable
	Culture	Do Not Fail	Fail Early
Measure	Metric	Cost & Capacity	Cost, Capacity and Flow (Time)
	Define "Done"	"I did my job"	"It's ready to deploy"

Fig 1: Comparing DevOps to traditional IT: Eight key differences . (Used with permission from the author)



## 6.1 Smart Design

**To achieve the objectives of DevOps** – predictable, efficient, secure, easily-maintained operational processes – **the automation tools you choose are critical.** The best ones are easy to use via a simple user interface, and offer advanced functionality that enables:

- Design of high quality new environments in minutes
- Your architects & designers to collaborate with far less documentation and chance of errors
- Easy cloning & extending of reference templates, or building of new environments from scratch
- Simpler management of topology versions
- Validation of configurations before builds are passed onto Dev or Prod

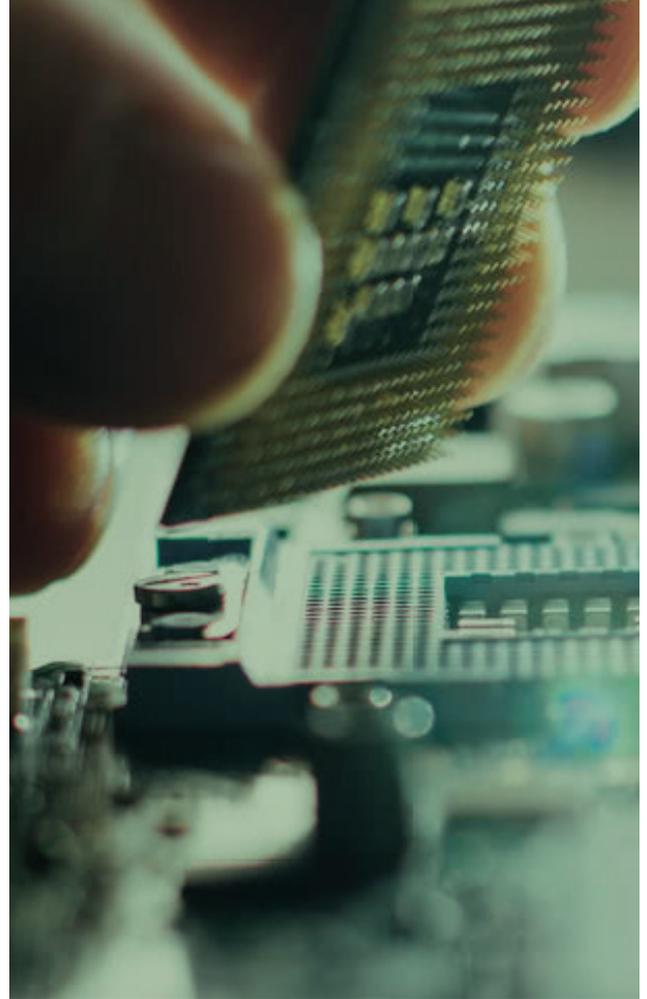
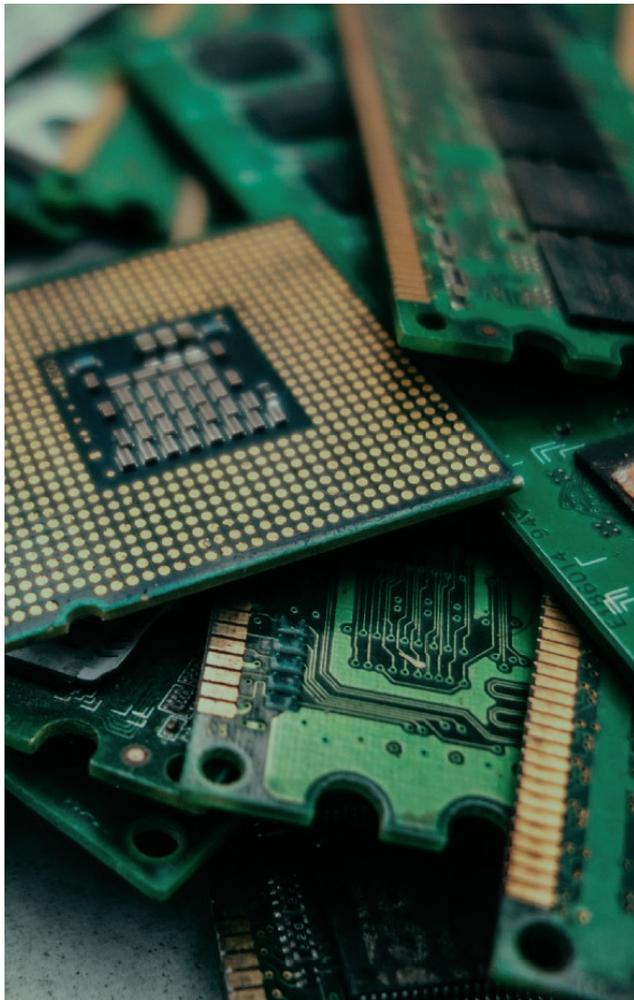
## 6.3 Automated Build

**Continuous Delivery Automation is the enabler of repeatable, reliable, high quality outcomes, and of rapid response to changing business needs.** The less manual your processes, the greater your agility and ability to adapt. **An effective Continuous Delivery Automation engine:**

- Monitors real time job activity and logs
- Lets you run & re-run all or parts of a build
- Lets you copy or clone configurations from an existing environment back into Design
- Stores all build configuration in 'Source Control', a centralised configuration repository for audit and quality assurance
- Enables integration of the Oracle technologies underpinning your solution such as infrastructure virtualisation, databases, OFM and applications
- Enables the integration of cloud services provided by Oracle.

**Keeping everything in Source Control and adopting an 'Infrastructure-as-code' approach allows you to define your infrastructure and platform through code constructs, and deliver it through automation.**

This is critical because cloud deployments - especially cloud/on-premise hybrids - make it far more difficult to keep your environments and configurations consistent across your delivery pipeline.



## 6.4 Manage & Troubleshoot

**To achieve consistency of your environment builds, a configuration console is essential:** it gives your operations and support teams a real time, in-depth view (with drill-down ability) of the technical configurations in their environments.

This is key to achieving consistency, continuous integration and meeting DevOps objectives. **The best automation tools have configuration consoles that include:**

- Detection and prevention of issues before they happen
- Environment comparison, automated change detection and configuration management
- Detailed reports on Configuration Drift and other changes
- Comparison of multiple environments with configuration snapshots
- Uncovering and eliminating operational process inefficiencies
- Enhancing project delivery certainty

Using advanced tools like this, you can detect and fix the majority of issues, before delivery.

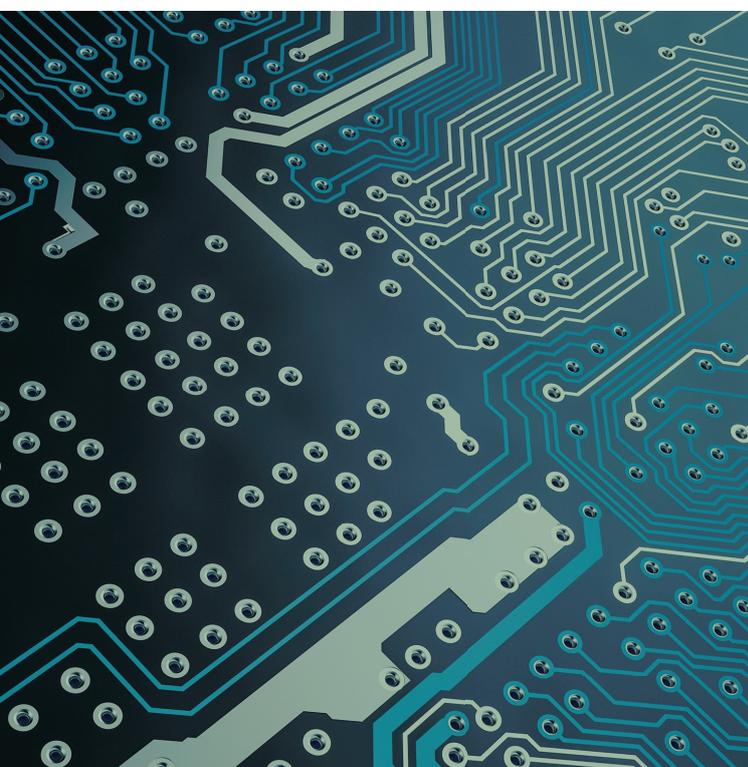
## 6.5 Cultural Change

***DevOps relies on principles that advocate close collaboration between Dev and Ops teams, which might mean a culture change in your organisation.***

The good news is that most obstacles are organisational, and trust plays a key part; seeing is believing.

Either way, faster building of environments with fewer defects makes building trust between Dev and Ops teams easier, and is beneficial for the business, anyway.

***In fact, the greater the cohesion between Dev and Ops, the greater the business value of delivered solutions and capabilities.*** The full benefits of DevOps principles are realized in maturity: when its concepts become Business As Usual (BAU).



## 7. IMPLEMENTATION OPTIONS

*This blueprint has shown how to overcome the 5 main challenges in deploying end-to-end Oracle deployments, and the role of Continuous Delivery Automation in managing quality Oracle environments and achieving DevOps.*

*When it comes to implementation, you have several options to weigh up.*

## 7.1 Doing it in-house manually

Hiring Oracle niche specialists or contracting them externally might seem an economical option, but is least likely to achieve quality outcomes if your builds are still manual. Even if you have plenty of in-house resources, you'll need to keep their Oracle knowledge up-to-date, which is a sizable, ongoing investment. You might want to weigh up the better ROI, from this versus investing in your business..

## 7.2 Developing an automated toolset in-house

This is an option if you have advanced Oracle skills in-house, and the experience to develop a sophisticated automation tool from scratch. If you don't, it's likely to be difficult to complete and manage, with uncertain performance, functionality, cost and ROI. Alternatively, you could try to combine various external tools into one automation capability, but you might find the components weren't designed to work together. Any change by the creator of any one component could compromise or render ineffective your combined toolset.

## 7.3 Hiring a systems integrator (SI)

This is a popular option especially for large enterprises. Yet, you may find some SIs are generalists who will hire Oracle specialists to help out, anyway. If their plan is to do the project manually or develop an automation tool for you, hiring a SI would have the same drawbacks as doing either of these in-house, and would also add costs. For best results, you might consider choosing an SI who uses advanced automation tools or who is happy to work with an automation specialist recommended by Oracle.

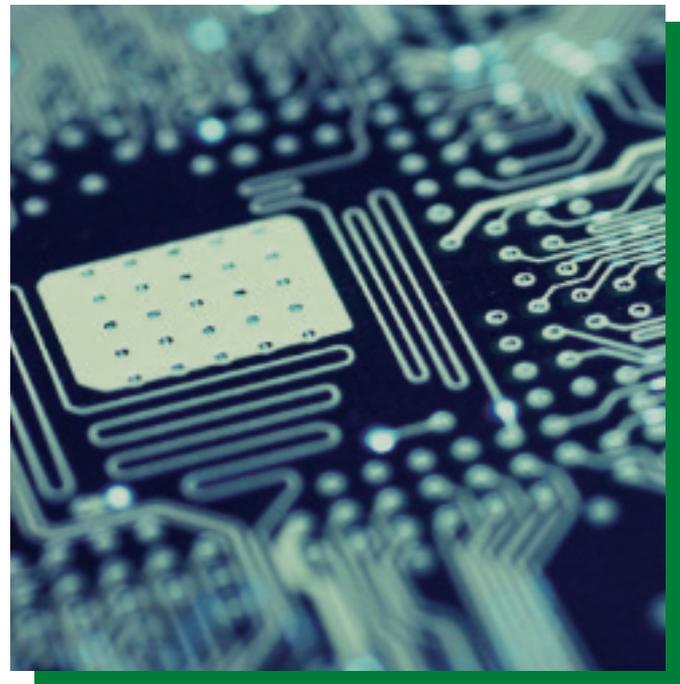


## 7.4 Buying an automation tool

These tools are mostly as good as the people who design them, and some are very good. The best tools go beyond automation to provide real time diagnostics and remediation capability, so you can detect and fix or prevent inconsistencies before they impacts your business. Buying an advanced automation tool is a good option for Systems Integrators or enterprises who routinely build and manage complex Oracle environments in-house.

## 7.5 Contracting oracle specialists

If you choose a team with proven Oracle experience who has developed advanced automation and diagnostic tools, you'll get the job done with lower overheads, in less time and with higher quality results. The right team will take responsibility for project success, and will work seamlessly with your internal team and SI to achieve it, and will remove the complications of multiple external SMEs. This is the right option if you want the peace of mind of quality environments and projects, delivered on time and on budget, and don't have limitless resources or time to do it.



## 8. ABOUT LIMEPOINT

***LimePoint is a highly experienced Oracle specialist based in Australia.***

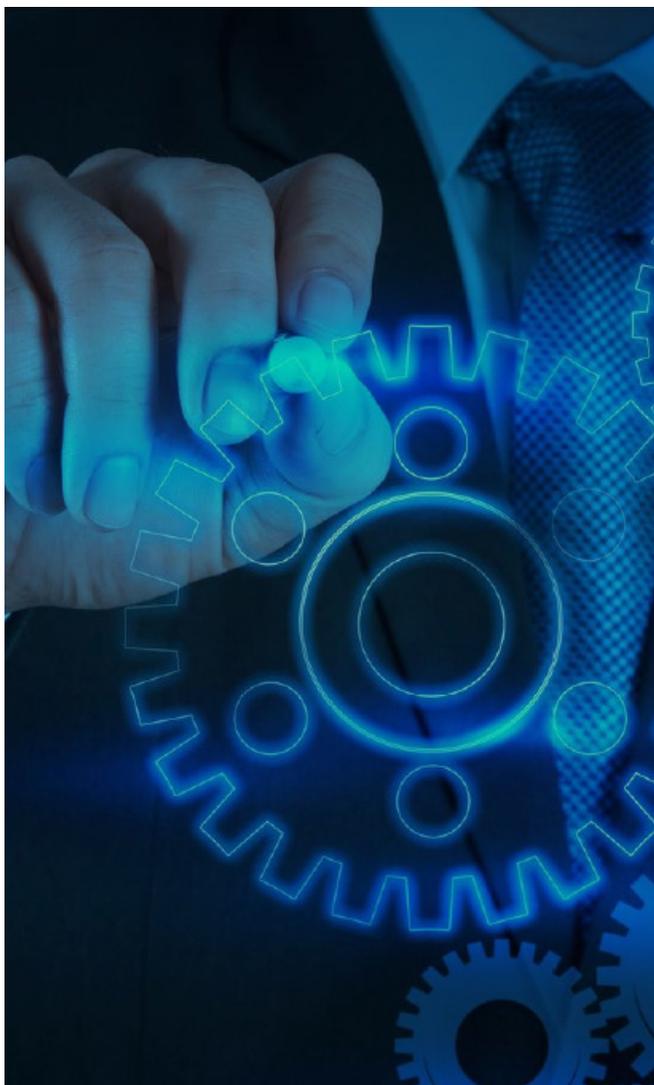
Our people have **over 20 years' experience with large scale, multi-million dollar Oracle deployments** in Finance, Government, Education, Retail and Utilities. We've helped them streamline their deployments, achieve DevOps and improve business agility and value.

What also sets LimePoint apart is our **Smart Suite for Continuous Delivery & DevOps for Oracle.**

Developed in-house, our **Smart Suite** comprises two products: **MintPress** which designs, builds and deploys high quality Oracle environments in hours not months, and **DriftGuard** which detects and fixes or prevents Configuration Drift in Oracle and other environments, in real time. Available together or separately, **MintPress** and **DriftGuard** make the impossible a reality: high quality environments that are fast to build and easy to manage and troubleshoot, in-house.

LimePoint has rare **Oracle expertise across the whole Oracle stack and the full environment life cycle.** We work from project planning to financial establishment and stakeholder management, through to implementation and measurement, providing solutions that are fit for purpose and support innovation, business value and competitive advantage for your clients.

Whether you choose our **advanced Smart Suite**, our **deep Oracle expertise** or **both**, LimePoint is the considered choice for complex or diverse Oracle environments.



## 9. ABOUT THE AUTHORS



### GORAN STANKOVSKI

***Goran is CTO, a Director and co-founder of LimePoint, and the innovator behind our technology.***

He's a recognised thought leader in Oracle architecture and had an international career with Oracle before forming LimePoint. Goran is the mentor for the LimePoint technical team and the brains behind LimePoint's Automation Suite, our advanced tool for Continuous Delivery and automation of Oracle deployments.

Goran has presented at Oracle Open World several times and is the author of many articles about Oracle architecture, automation and performance.



### MATT TUOHEY

Matt Tuohey is a Director and co-founder of LimePoint, and the driver behind our practical, client-focused approach.

He's led complex Oracle deployments for over 20 years and is passionate about making them simpler and easier to use in clients' own hands.

Matt has rare skills in communication and stakeholder management and a deep appreciation of enterprise business drivers and operations. His clear understanding of what enterprises need, how they operate and the value they seek from technology, is the force behind the LimePoint approach: delivering practical solutions that are fit for purpose.



# 10.

## TERMS USED IN THIS BLUEPRINT

### BAU

Business As Usual or maintaining normal operation for employees, managers, partners and customers of a business, while new software is being built, tested or deployed or existing software is being updated or patched.

### BPM

Business Process Management is systematic approach to management of business processes so the enterprise's workflow is more effective, more efficient and more able to adapt to change. Oracle BPM Suite is the part of Oracle Fusion Middleware that enables systems developers to accomplish this.

### Configuration Drift

Configuration Drift refers to gradual changes made over time to hardware and software configurations, which occurs unintentionally yet continuously in most data centres. If undetected and un-remedied, these changes or inconsistencies can result in serious system failure. Some sources say configuration drift is the cause of the majority of such system failures.

### Continuous Delivery

Delivery of software capability through automation so it's rapid, reliable and repeatable, resulting in higher quality due to lower manual input and human error, and lower risk to the enterprise.

### DevOps

DevOps is an IT software development terms that describes an agile, co-operative relationship between Development and IT Operations. The goal of DevOps is to change and improve the relationship by advocating better communication and collaboration.

### e2e Oracle Deployments

End to end Oracle deployments are complete, enterprise-wide Oracle solutions that include some or many items from the complete Oracle Stack.

### OFM

Oracle Fusion Middleware is a collection of software products (including Java EE and developer tools, integration, business intelligence, collaboration and content management) to develop, deploy and manage applications and systems from Oracle and other vendors

### Oracle

Oracle Corporation, the US-based global computer technology company based in Redwood City, California

### Oracle enterprise deployment

An enterprise wide deployment involving several Oracle solutions and technologies

### Oracle environment

A unique combination of Oracle Applications, Middleware, Database and Servers that needs to be built at each stage of the SDLC for each software application

### Oracle solution

A combination of Oracle technologies or applications that delivers a specific outcome e.g. User Single Sign On

### Oracle stack

The complete offering from Oracle split into 7 high level layers: Applications, Middleware, Database, Operating Systems, Virtual Machine, Servers and Storage.

### Oracle technology

An identifiable Oracle product or application that performs a specific task e.g. Oracle Access Manager or Oracle Exalogic Elastic Cloud

### SDLC

Systems (or Software or Application) Development Life Cycle is the process of planning, designing, testing, and deploying new software or updates to existing software. To minimise defects, each stage requires a unique environment which is consistent with those before and after it.

### SME

A Subject Matter (or Domain) Expert in software development is an authority in a particular area of expertise.

### SOA

Service Oriented Architecture is the framework which enables access to services (such as producing data, validating a customer, or providing analytical services). The Oracle SOA Suite is part of Oracle Fusion Middleware that enables system developers to set up, manage and orchestrate services into composite applications and business processes.

### Source Control

Source Control (or version control or revision control) is a software configuration management term that means management of changes to documents, computer programs, large web sites, and other collections of information

### Systems Integrator

A service provider that implements, plans, coordinates, schedules, tests, improves and sometimes maintains IT operations for clients, by integrating a variety of disparate products.

# 11.

## REFERENCES

- 1.** *Oracle Fusion Middleware in perspective*  
Mark Driver, Gartner, in Tech Target 2014.
- 2.** *Magic Quadrant On-Premise Application Integration Suite.*  
Gartner Inc. 2014
- 3.** *How Deploy Frequency Impacts Infrastructure Stability*  
Lori MacVittie. DevOps.com. 2015
- 4.** *Delivering large-scale IT projects on time, on budget, and on value*  
McKinsey Insights and Publications 2012.
- 5.** *Developer Productivity Report.*  
ZeroTurnaround. 2013.
- 6.** *DevOps and the Cost of Downtime: Fortune 1000 Best Practice Metrics Quantified.*  
IDC Insight December 2014
- 7.** *PMI Study Reveals Poor Communication Leads to Project Failure One Third of the Time.*  
Yvonne Monterroso. CoreWorx 2013
- 8.** *Making DevOps Work in Complex Enterprise Environments*  
Andi Mann. CA Technologies 2014
- 9.** *DevOps and the DBA.*  
Mike Fal. Simple Talk 2015
- 10.** *Oracle products: How Fusion Middleware measures up for SOA integration.*  
Alan R Earls in Tech Target 2014.
- 11.** *Continuous Delivery of Fusion Middleware applications*  
Jürgen Kress. Oracle SOA & BPM Partner Community Blog 2015.
- 12.** *Comparing DevOps to traditional IT: Eight key differences.*  
**Mustafa Kapadia.**  
DevOps.com. Feb 2015

## CONTACT US

 [www.limepoint.com](http://www.limepoint.com)

 [info@limepoint.com](mailto:info@limepoint.com)



**LIMEPOINT**