Whitepaper

How good is your Disaster Recovery Planning?

David Cowell, May 2020

Table of Contents

Table of Contents 2
Introduction 3
Current State 3
Assessment 4
Processes: 4
Technology: 4
Gap Analysis 5
Closing the Gaps 5
Process 5
Technology 6
Finally, Implementation 7
Fail over Plan 7
'Run in DR' Plan 7
Fail Back Plan 7
What if I am running in the Cloud? 8
DR as a Service 8
Summary 8
About the Author: 8

Introduction

As organisations exercise (components of) their Business Continuity Plan (BCP) in response to the global pandemic, now is the time to also focus on IT continuity. Just as many of us have thought the likelihood of ever executing our BCP is low, recent events have demonstrated that this may no longer be the case. These events have also re-enforced that the impact of them is severe. We need to be prepared. For example, many companies have scrambled to ensure key staff and systems can work remotely and have suffered more than they may have had these systems and processes already been in place.

The same applies for IT continuity, perhaps better known as Disaster Recovery (DR). Many organisations have paid lip service to this over the years and are exposed such that, if a core system suddenly becomes unavailable, business operations that this system supports will cease. This may mean loss of revenue, compliance issues, reputation impact and, ultimately adversely impact the future of the business.

As we come out of the pandemic and organisations look inwardly at their resilience, they would be well placed to consider IT resilience at the same time. This paper provides an overview of what many organisations will see when they look at the current state and provides an example of what they could do to navigate out. It focusses on the processes and technology elements required to be managed in order to implement a solid Disaster Recovery capability.

Current State

DR is often the 'special project' that no one really worries or cares about. Many companies will have implemented a DR capability over the years and trot out management one-liners such as 'yes, we have DR'. A look under the hood will often reveal the opposite to be the case. The following are examples that I have seen that may strike a chord with you regarding your organisation:

- We have DR for some but not all core systems
- Our DR is tested at the component rather than enterprise level (e.g. have we tested multiple applications in the same 'eco-system' at the same time?
- Have we tested data centre site resiliency?)
- We test annually but there are always issues which may impact our recovery time
- Our tests are simulated, they run offline to Production and do not actually fail a Prod system
- We can fail over to our DR site, but we can't fail back
- We had a DR regime at one time but there were organisational changes that lead to changes that negatively impacted our DR capability
- Our documentation is out of date
- Our DR infrastructure has not been maintained

I could go on. Yes, some companies do a great job and technology and mature processes are in place to provide technology resilience. If yours is not one, read on.

Assessment

It is a bit like having an addiction, once you accept that there is an issue and something needs to be done, the battle victory is in sight.

I would look at the following in a DR assessment.

Processes:

Business Continuity Management (BCM) is the process that provides a framework for building organisational resilience to business disruptions, with the capacity for an effective response that safeguards financial health, business operations, regulatory compliance and brand reputation.

As part of a DR assessment the following should be considered

- Are regulatory and compliance obligations met?
- Does Executive Management support an initiative to improve DR?
- Do business owners understand the current state?
- Is there a DR team who are responsible for operations and the continuous improvement of DR capability?
- What is the process to declare a disaster, and who makes the decision?
- Are roles and responsibilities in a DR scenario documented and accepted?
- Are Maximum Allowable Downtime, Recovery Time Objective (RTO) and Recovery Point Objective (RPO) metrics per application in place?
- Does the BAU support team run DR tests and is the documentation current?
- Does the Incident Management process cater to DR scenarios?
- Do DR metrics meet the Business Continuity Management (BCM) framework requirements.
- Finally, what scenarios are catered for?
    - Data centre loss
    - component failure
    - human error
    - cyber security considerations

Technology:

Disaster avoidance is so much better than disaster recovery and it is often not that difficult or expensive to introduce or improve infrastructure resiliency. Having a ‘high-availability’ construct also allows for operational improvements such as concurrent maintenance where one half of an eco-system can be taken offline for patching etc whilst the other half continue to provide the service.

When it comes to DR the following must be taken into consideration to understand the risk profile –

- Is there a DR data centre, where is it (the proximity to the Production data centre is a critical factor)?
- Is there a network infrastructure that allows the maintaining of IP addresses when failing over to DR?

- Is the required infrastructure installed, operational and maintained?
- Are maintenance changes (e.g. patches/fixes) that are applied to Prod also applied to DR?
- Is critical data being replicated?
- are staff access policies in place to allow immediate unplanned access to the DR data centre?
- Finally, but critical, is how resilient is Production infrastructure?

Gap Analysis

Now that the effort has been made to understand the current state, a gap analysis including the following ought to be performed:

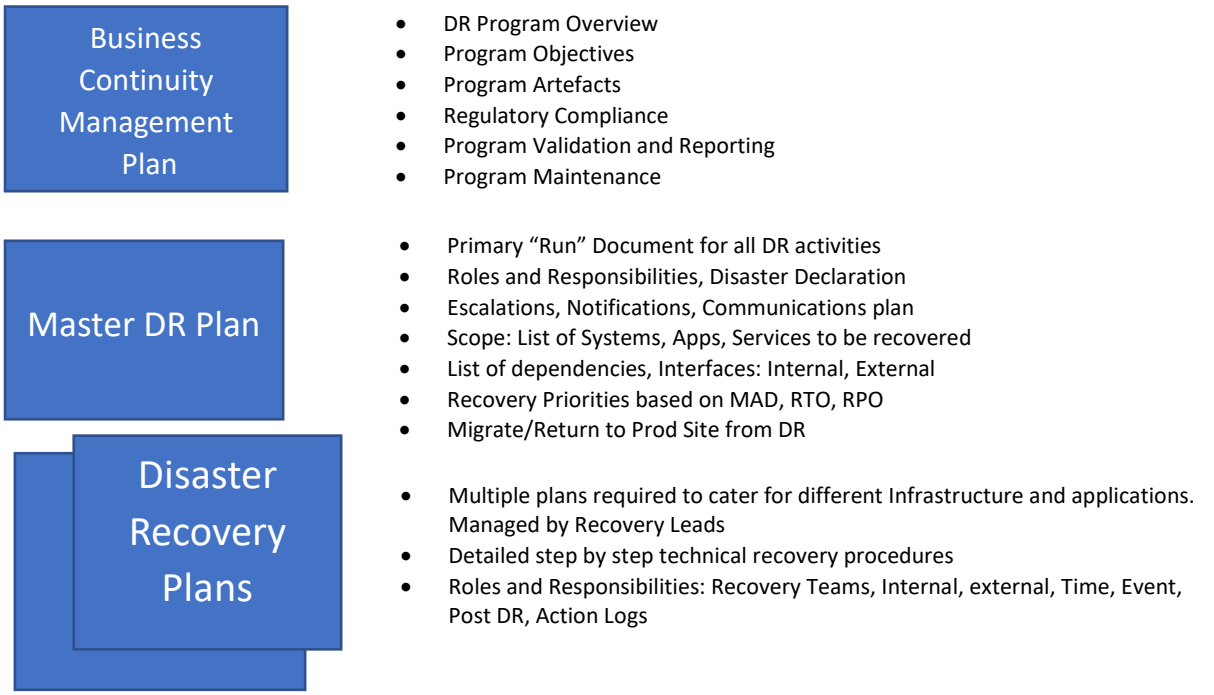
1. Does the technology in place provide the capability to meet the business recovery requirements?
2. Do processes align to industry best practices?
3. Are government and industry compliance requirements being met?
4. Is the right team in place to manage and continuously improve?
5. Is documentation up to scratch?
6. Is the test regime and test defect management process up to scratch?
7. What is the DR position with third-party vendors, particularly those who are providing software-as-a-service?
8. Are business owners aware of the risks? Do they accept them? Will they support (and fund?) improvement plans?
9. Is there funding available to bridge the gaps?

Closing the Gaps

In keeping with the technology/process flavour of this paper, I would address the gaps the same way, by looking at processes and technology. Before doing this however, perhaps this is the time to brief executive management as there is likely to be a big spend required to close the gaps. Is there appetite and funding?

Process

First and foremost is the process. The technology is great but difficult to use effectively without a proven process. The following diagram shows the key documents for consideration and what they might contain. Spend time with the business continuity team ensuring that any DR changes align to BC requirements then prepare/improve these documents starting at the top.



Treat this as a framework and modify it to suit your organisation. Make sure it aligns to industry best practices and regulatory and compliance requirements. Consult with the Service Operations and DR teams as they are the teams who will be responsible to execute, document and maintain.

Keep the key business stakeholders informed and ensure they support what you are doing.

Technology

In parallel to this and focussing on critical infrastructure first (e.g. network, storage, databases, middleware components), document what hardware is required to be deployed either in Prod and/or in DR to allow RTOs and RPOs to be met. The lower the RTO (less time available to recover) and RPO (less data loss acceptable) the more expensive the solution may be (more-so RTO than RPO) so these metrics are important. Look at high availability opportunities as part of the solution. Make sure any changes are in line with your architectural standards and best practices. Document solution options, maybe with your infrastructure vendor partners, and make decisions on a risk vs reward basis.

Look at how your applications communicate to their infrastructure and to each other. Using dns, either natively or via an F5, is a cheap way to allow flexibility to re-point applications to different hosts as required. Applications and Middleware would ideally be changed to use a hostname rather than an IP address and a process implemented to allow re-pointing as required.

Consider software licensing and engage software vendors early on. It is often a sticking point as certain vendors may assume you will be running DR in parallel with Prod and will want to charge for this luxury. This will not be the case as licencing should only apply to the 'live' or production instance running at the time. Also, many software licensing agreements allow a certain number of days of

parallel run (e.g. for a DR test) per annum. Do not accept, prima-facie, that there will be additional software costs, push back.

The last critical component is interfaces. The following infrastructure elements will likely need to be changed during a fail over event: Firewalls, Control M (or other batch processing applications), Applications and Database connection strings and File Transfer settings. Once all this is done, you are ready to test.

Finally, Implementation

Once you have policies and processes in place and technology solutions designed, it is time to work on the implementation and execution. Development of the DR lifecycle elements is critical and is inclusive of, but not limited to

- Failover plan
- Run plan in DR and
- The fail back plan

Let's look at each one in more detail -

Fail over Plan

Build a DR failover implementation plan or runbook. This will consist of the technical steps required to fail over an application from the Prod data centre to the DR data centre. Preparation is the key here so get an initial version ready then send to the relevant technical teams to improve. The runbook must include: any pre-requisites (e.g. terminate active sessions); who does what and when, how long each task takes and, critically, the dependencies between steps, checkpoints for rollback and rollback steps and duration. Make sure you book the change outage to include additional time to resolve issues and rollback time.

A purely linear runbook is an inefficient runbook as there will be many tasks that can be executed in parallel (e.g. end-user network failover can generally occur at the same time as database failover). Once the runbook has reached a level of maturity, review it with the team. This needs to be the team doing the work on the night not just any person from the team. Review it several times until you feel confident it is ready to go and will include several iterations.

'Run in DR' Plan

Once a fail over is completed, your application is now running either in part or in whole from a secondary data centre. End users are connecting to different infrastructure and data is being updated. There may be compromises. DR capacity and performance may not be the same as Prod (probably due to the cost to do so). If this will be the case, the business must accept this in advance. Keep an eye on this and, if required, limit the number of online users. Have the 'A team' on standby as gremlins will surface. The first test may be at a quiet time and may not include full BAU operations, but this will still give you confidence of capability.

Fail Back Plan

This is a 'trap for young players'. Some organisations focus on the recovery component and forget that, at some point, ideally sooner rather than later, they will have to return operations to the Primary site. This is pretty much running the same runbook as the fail over but in reverse. A critical element is reverse data replication. SAN systems need to be configured to allow two-way replication. More progressive organisations are now focussing on the ability to run Prod from either site and switch Prod from site to site on a regular (e.g. every 6 months) basis. This process is gold-plated DR.

What if I am running in the Cloud?

Cloud-based applications can either be built with resilience in mind or for the types of changes discussed here to be more easily implemented via DevOps processes and automation. Using AWS as an example, Elastic Load Balancers and multiple availability-zones / multi geographical regions allow for a high availability construct to be delivered with an outage measured in seconds not minutes or hours in a failover scenario. These can be tested in spun-up non-prod environments cheaply and effectively. The key here is the solution architecture. It is much better to incorporate resilience in any cloud migration rather than retrospectively.

DR as a Service

There are options available to provide DR as a Service (DRaaS) either in a private cloud (likely to be hosted at a managed services provider's data centre) or in a cloud provider. The main challenges to consider here are network connectivity (again, keeping IP addresses) and data migration/synchronisation. If Cloud is preferred, just remember to treat DR as Prod and, before trying to get DRaaS operational, get all your cloud operations gremlins out of the way and have some legacy apps already running there. Also consider that older technologies (e.g. old Oracle DBs or Windows operating systems) may not work *out-of-the-box.*

Summary

Many organisations will find, upon inspection, that their ability to recover from an IT disaster is not what they thought it would be or what it should be. Once a current state assessment is made that focusses on aligning to business continuity principals and industry best practices and that has the endorsement of senior IT and business stakeholders (e.g. accepting that risks must be treated, there is funding available and that there will be some business impact along the way), improvements can be implemented that will be tested regularly and continuously improved. The end result is IT and business resilience that will protect your business when the unexpected happens.

About the Author:

David Cowell is a Melbourne, Australia based data centre and IT infrastructure SME and Program Manager who specialises in large and difficult infrastructure transformations. He has, for example, run many data centre migrations, applications transformations and helped many organisations improve their Disaster Recovery capability. He can be contacted at dgcowell@live.com.au or +61 (0) 438 569822.