

Dokumentation

Projektname: OP-AWS-Migration

Klassifizierung intern
Status in Arbeit
Programmname OP-AWS-Migration
Projektnummer 69
Projektleiter Vladan Marlon Vranjes
Version 0.6
Datum 17. Mai 2024
Auftraggeber Pascal Gafner
Autoren Vladan Marlon Vranjes
Luan Stauffer
Janis Traupel
Yanis Riedo
Verteiler Pascal Gafner

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	03.05.2024	Konzept ausarbeiten	Alle Autoren
0.2	17.05.2024	Definition des Projekts	Alle Autoren
0.3	24.05.2024	Abschluss Initialisierung & Anfang von Konzept	Alle Autoren
0.4	31.05.2024	Abschluss Konzept	Alle Autoren
0.5	07.06.2024	Durchführung der Migration und Erfassung vom Vorgang mit Planabänderung	Janis Traupel, Luan Stauffer, Vladan Vranjes
0.6	14.06.2024	Abschluss Realisation & Anfang Einführung	Janis Traupel, Luan Stauffer, Vladan Vranjes

Tabelle 1: Änderungskontrolle

Inhalt

1	Initialisierung.....	4
1.1	Beschreibung.....	4
1.2	Ausgangslage.....	4
1.3	Betroffenheitsanalyse.....	4
1.4	Einführungsvorgehen	4
1.5	Ausbildungskonzept.....	4
1.5.1	Anforderungen	4
1.5.2	Kosten	4
1.6	Planung der Vorabnahme und der Abnahme	5
1.6.1	Abnahme	5
1.7	Freigabekriterien der Betriebsaufnahme	5
1.8	Ziele	5
1.9	Anforderungen.....	6
1.9.1	Funktionale Anforderungen.....	6
1.9.2	Nicht funktionale Anforderungen.....	6
1.10	Risikoanalyse	8
1.11	Risikomatrix.....	9
1.12	Schwierigkeiten.....	9
1.12.1	Migration schlechter Gewohnheiten	9
1.12.2	Fehlende Automatisierung.....	9
1.12.3	Kein Notfallplan	10
2	Konzept.....	10
2.1	Migrationskonzept.....	10
2.1.1	Ziele der Migration	10
2.1.2	Anforderungen an die Migration.....	10
2.1.3	Migrationsobjekte.....	10
2.1.4	Datenanalyse.....	10
2.1.5	Migrationsverfahren	10
2.1.6	Migrationsplan	11
2.1.7	Machbarkeit.....	12
2.1.8	Archivierung und Ausserbetriebsetzung Altsystem	12
2.1.9	Anforderungsabdeckung.....	12
2.1.10	Berechtigungskonzept	13
2.2	Testingkonzept.....	13

2.2.1	Testziele (anpassen an Anforderungen).....	14
2.2.2	Testrahmen	15
2.2.3	Testobjekte	15
2.2.4	Testarten	15
2.2.5	Übersicht Testfälle	15
2.2.6	Testfälle.....	16
3	Realisierung.....	18
3.1	Ausgangslage.....	18
3.1.1	Ausweichung auf Andere Methodik.....	18
3.2	Anleitung	18
3.3	Testfall Ergebnisse	22
4	Einführung	22
5	Tabellen Verzeichnis.....	22
6	Abbildungsverzeichnis	23

1 Initialisierung

1.1 Beschreibung

Die Firma Traupel-AG möchte ihre Datenbank, die sie selbst betreibt in die Cloud verschieben, da sie schwankende Nutzung der Datenbank haben und daher ungenutzte / fehlende Hardware haben. Ausserdem möchten sie die Hardware nicht mehr selbst betreiben möchten.

1.2 Ausgangslage

Momentan besteht eine MySQL Datenbank die zurzeit On-Prem auf einem Linux Server läuft, diese soll nun auf AWS in die Cloud migriert werden.

1.3 Betroffenheitsanalyse

Nr.	Stakeholder	Einstellung und Interessen	Betroffenheit	Intensität	Stellung und Macht
01	Nutzer	Keine	Klein	-	Klein
02	Administratoren	Positiv	Hoch	Hoch	Mittel
03	Geschäftsleitung	Positiv	Mittel	Mittel	Hoch

Tabelle 2: Betroffenheitsanalyse

1.4 Einführungsvorgehen

Die Firma Traupel-IT-AG spielt in diesem Projekt eine Rolle als Hauptentscheidungstreffer und wird demnach mit allen Änderungen benachrichtigt, welche ihre Arbeitsgewohnheiten beeinflussen. Ausserdem wird nachdem die Konzeption aufgestellt wurde, ein Meeting arrangiert, bei dem der Kunde über alle Schritte informiert wird und gegebenenfalls einwürfe vorbringt.

1.5 Ausbildungskonzept

1.5.1 Anforderungen

Das neue System wird Arbeitsalltag nichts ändern demnach bleibt es gleich wie vor der Migration.

Die Datenbank soll von der OnPrem-Umgebung in eine AWS-Umgebung migriert werden.

1.5.2 Kosten

Produkt	Kosten pro Einheit in CHF	Einheiten	Kosten Total in CHF		
RDS-Proxy (Monthly)	438.00	1x	438.00		
Storage (Monthly)			0.23	30x	6.90
RDS MySQL (Monthly)			1'162.66	1x	1'162.66
Backup Storage (Monthly)			0.095	30x	2.85
Snapshot Storage (Monthly)			0.01	30x	0.30

Arbeitszeit Luan	60.-	16x	960
Arbeitszeit Yanis	60.-	16x	960
Arbeitszeit Janis	60.-	16x	960
Arbeitszeit Vladan	60.-	16x	960
Kosten total	5'450.71		

Tabelle 3: Kosten

1.6 Planung der Vorabnahme und der Abnahme

1.6.1 Abnahme

Da der Arbeitsalltag gleichbleibt und die DB-Admins die Betriebsdokumentation erhalten, wird die Abnahme ohne Protokoll ablaufen.

Es werden noch alle nötigen Berechtigungen an die Zugriffsbewilligten Mitarbeiter ausgestellt und eine kleine Orientierungsschulung wird durchgeführt.

1.7 Freigabekriterien der Betriebsaufnahme

Die Checkliste Betriebsaufnahme beschreibt alle generellen und projektspezifischen Prüfpunkte und Freigabekriterien, die für den Entscheid Betriebsaufnahme notwendig sind. Auf das entsprechende Dokument verweisen oder Tabelle der Checkliste kopieren und hier einfügen.

1.8 Ziele

Ziel-NR.	Ziel	Zielbeschreibung
Z-01	Kosteneffizienz	Reduzierung der Gesamtkosten durch den Wechsel zu einem nutzungsbasierten Abrechnungsmodell und den Verzicht auf den Kauf und die Wartung von Hardware.
Z-02	Hochverfügbarkeit	Nutzung der globalen Infrastruktur von AWS, um Anwendungen über verschiedene geografische Regionen und Verfügbarkeitszonen hinweg zu betreiben und so die Ausfallsicherheit zu erhöhen.
Z-03	Sicherheitsverbesserungen	Zugriff auf hochmoderne Sicherheitsfunktionen und Dienste, beinhaltet Verschlüsselung, Identitätsmanagement, Zugriffsmanagement und Bedrohungserkennung.
Z-04	Datenintegrität	Gewährleistung der Sicherheit und Integrität aller Daten während der Migration, ebenfalls der Verschlüsselung und Sicherung von sensiblen Daten.
Z-05	Leistungsoptimierung	Verbesserung der Anwendungsleistung durch die Auswahl von Verwaltungsaufgaben wie Bereitstellung, Überwachung und Wartung an AWS-Tools.
Z-06	Minimierung von Ausfallzeiten	Sicherstellen, dass die Migration mit minimalen Unterbrechungen für die Endnutzer durchgeführt wird, indem geeignete Migrationswerkzeuge und -techniken verwendet werden.

Tabelle 4: Ziele

1.9 Anforderungen

1.9.1 Funktionale Anforderungen

Anforderung	Beschreibung
FA-01	Implementierung von AWS Auto Scaling, um sicherzustellen, dass die Ressourcen automatisch basierend auf der tatsächlichen Nachfrage skaliert werden.
FA-02	Implementierung von Load Balancing, um den Datenverkehr gleichmäßig über mehrere Instanzen zu verteilen. AWS Elastic Load Balancer (ELB) kann genutzt werden, um sicherzustellen, dass keine einzelne Instanz überlastet wird und bei Ausfall einer Instanz der Datenverkehr automatisch auf andere gesunde Instanzen umgeleitet wird.
FA-03	Implementierung eines robusten Identitäts- und Zugriffsmanagements (IAM), um sicherzustellen, dass nur autorisierte Benutzer und Systeme auf Ressourcen zugreifen können.
FA-04	Implementierung von Mechanismen zur regelmäßigen Validierung und Überprüfung der Datenintegrität, um sicherzustellen, dass gespeicherte Daten korrekt und vollständig sind.
FA-05	Implementierung von skalierbaren Architekturen, die es ermöglichen, Ressourcen basierend auf der aktuellen Last dynamisch zu skalieren. Implementierung von skalierbaren Architekturen, die es ermöglichen, Ressourcen basierend auf der aktuellen Last dynamisch zu skalieren.
FA-06	Durchführung von Live-Migrationen, um den Betrieb während der Migration von Ressourcen oder Diensten auf AWS aufrechtzuerhalten, ohne dass es zu Unterbrechungen oder Ausfallzeiten kommt.

Tabelle 5: Funktionale Anforderungen

1.9.2 Nicht funktionale Anforderungen

Anforderung	Beschreibung
NFA-01	Die Lösung muss in der Lage sein, je nach Bedarf zu skalieren, um Überkapazitäten und damit verbundene Kosten zu vermeiden. Dies bedeutet, dass die Architektur so entworfen sein muss, dass sie flexibel auf Lastspitzen reagieren kann, ohne unnötige Ressourcen bereitzustellen.
NFA-02	Die Systeme müssen äusserst zuverlässig sein, somit können wir Ausfallzeiten minimieren.
NFA-03	Die Integrität der Daten muss sichergestellt sein, indem Massnahmen ergriffen werden, um sicherzustellen, dass Daten während der Speicherung und Übertragung nicht verändert werden.
NFA-04	Die Daten müssen konsistent sein und korrekte Beziehungen zwischen verschiedenen Datenelementen aufrechterhalten, um inkonsistente oder widersprüchliche Informationen zu vermeiden.
NFA-05	Die Systeme müssen schnelle Antwortzeiten bieten, um eine reibungslose Benutzererfahrung sicherzustellen und die Zufriedenheit der Benutzer zu erhöhen.

NFA-06	Die Zeit, die benötigt wird, um nach einem Ausfall den normalen Betrieb wiederherzustellen, sollte minimal sein. Dies erfordert effiziente Wiederherstellungsmechanismen und Automatisierung, um den Prozess schnell und zuverlässig durchzuführen.
--------	---

Tabelle 6: Nicht Funktionale Anforderungen

1.10 Risikoanalyse

Risiken der Phase Initialisierung

Vor Massnahme							Nach Massnahmen			
Nr.	Risikobeschreibung	Auswirkung	EW	AG	Risiko	Risikominderung	EW	AG	Risiko	Risikoakzeptanz
R1	Keine Verbindung zur Datenbank	Unterbrechung des Geschäftsbetriebs, Vollständiger Datenverlust	3	3	9	Redundante Verbindungen aufsetzen	1	2	2	Akzeptabel
R2	Inkompatibilität der Datensätze vom alten System zum neuen	Funktionalitätsverluste, Fehler in der Anwendung	3	2	6	Vorabtests, Sicherstellung der Kompatibilität, Anpassung der Systeme	1	2	2	Akzeptabel
R3	Datenverlust während der Migration	Unvollständige oder inkonsistente Daten im neuen System	3	3	9	Regelmäßige Backups, Validierung der Daten vor und nach der Migration	2	1	2	Akzeptabel
R4	Systemausfall während der Migration	mögliche Dateninkonsistenzen	2	2	4	Planung von Ausfallzeiten, gründliche Tests, Notfallwiederherstellungsplan	2	1	2	Akzeptabel
R5	Sicherheitslücken während der Migration	Erhöhtes Risiko für Datenlecks oder Manipulationen	2	2	4	Sicherheitsüberprüfungen, Verschlüsselung, Zugriffssteuerungen	1	2	2	Akzeptabel
Legende: <i>EW=Eintretenswahrscheinlichkeit: 1 niedrig / 2 mittel / 3 hoch;</i> <i>AG=Auswirkungsgrad: 1 gering / 2 mittel / 3 gross; Risiko = EW x AG</i>										

Tabelle 7: Risikoanalyse

1.11 Risikomatrix

Vor Maßnahmen

Wahrscheinlichkeit			R1, R3
		R4, R5	R2
	Auswirkung		

Tabelle 8: Risikomatrix vor Massnahmen

Nach Maßnahmen

Wahrscheinlichkeit			
	R3, R4		
		R1, R2, R5	
	Auswirkung		

Tabelle 9: Risikomatrix nach Massnahmen

1.12 Schwierigkeiten

Selbst mit einem umfassenden Datenmigrationsplan und einer entsprechenden Strategie müssen mit bestimmten Schwierigkeiten gerechnet werden, die bei den meisten Migrationen auftauchen.

1.12.1 Migration schlechter Gewohnheiten

Einige Unternehmen versäumen es, schlechte Daten, Betriebs- oder IT-Praktiken bei einer Datenmigration auszukorrigieren. Genauer gesagt können Migrationsprojektmanager fehlerhafte Dateien oder Datenstrukturen übernehmen, die bei einer Migration in dieser Form zu zusätzlichen Problemen führen.

1.12.2 Fehlende Automatisierung

Datenmigrationen erfordern komplexe Mechanismen, und ein Mangel an Automatisierung und ergänzenden Softwarelösungen kann den Prozess verlangsamen oder zu Fehlern führen.

1.12.3 Kein Notfallplan

Einer der größten Fehler bei der Datenmigration ist das Fehlen eines alternativen Plans im Falle von Fehlern.

2 Konzept

2.1 Migrationskonzept

2.1.1 Ziele der Migration

Nr.	Ziel	Priorität*
01	Alle Daten werden migriert	M / 1
02	Funktionalität wird gewährleistet	M / 1
03	Nötige Funktionen bleiben erhalten bei der Migration	2
* Priorität: M = Muss / 1 = hoch, 2 = mittel, 3 = tief		

Tabelle 10: Migrationsziele

2.1.2 Anforderungen an die Migration

Nr.	Anforderung	Beschreibung
01	AWS-Account	Amazon AWS Account mit gültiger Lizenz
02	Laufende MySQL DB	Datenbank von der aus migriert werden soll

Tabelle 11: Anforderungen

2.1.3 Migrationsobjekte

Nr.	Migrationsobjekt	Beschreibung
01	Alle Datensätze Alles Definieren	Alle Daten der Firma werden mitgenommen.
02	Beziehungen der Tabellen	Beziehungen zwischen Tabellen bleiben erhalten.
03	Schlüssel (Primär- und Fremd)	Schlüssel der Tabellenverbindungen bleiben erhalten.

Tabelle 12: Migrationsobjekte

2.1.4 Datenanalyse

Grundlegend ist es möglich alle Daten und ihre Relationen zu migrieren, ohne Lücken aufzuweisen, solange es keine Unterbrüche in der Migration gibt.

2.1.5 Migrationsverfahren

Für die Migration wird Amazon AWS DMS verwendet, welches ein von Amazon AWS bereitgestellt wird. AWS DMS ist der Database Migration Service von Amazon mit dem man Datenbank- und Analyse-Workloads schnell, sicher und mit minimalen Ausfallzeiten und ohne Datenverlust zu AWS verlagern kann. Der grobe Plan sieht folgend aus:

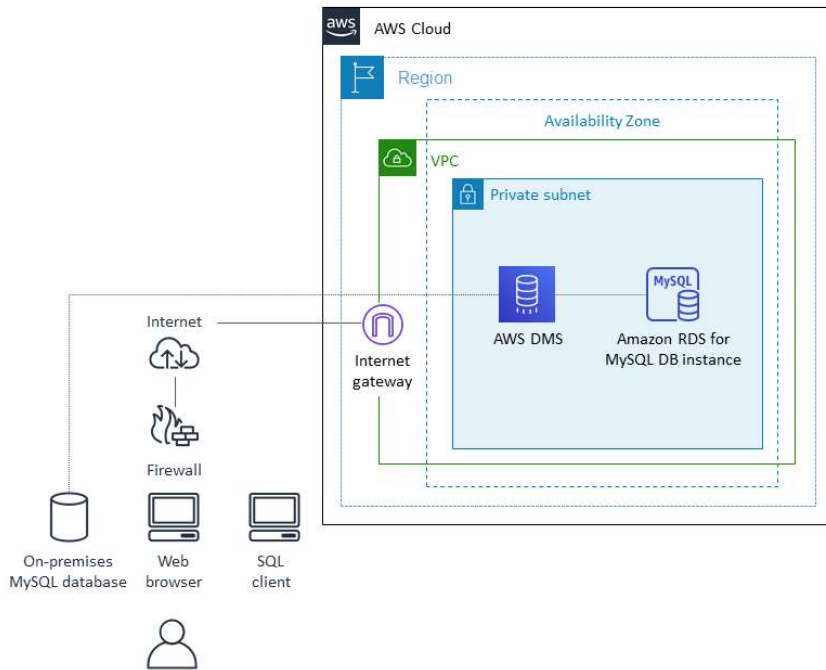


Abbildung 1 Migrationsverfahren

2.1.6 Migrationsplan

Vorgehen für die Migration, zeitlicher Ablauf, Abhängigkeiten, Qualitätsprüfungen

Datum	Migrationsschritt	Beschreibung
31.05.2024	Ermittlung zu migrierende Daten	Zusammen mit dem Auftraggeber wird ermittelt welche Daten migriert werden sollen
31.05.2024	Backup der zu migrierenden Daten	Ein Backup der Daten wird erstellt, falls ein Fehler bei der Migration auftreten sollte.
--.06.2024	AWS-Server muss funktionieren	In AWS muss der Server fertig aufgesetzt und funktionstüchtig sein.
--.06.2024	Migration der Daten durchführen (Point of no return)	Die Daten werden vom On-Prem-Server auf den AWS-Server migriert
--.06.2024	Testing	Der AWS-Server muss nach der Migration getestet werden, ob dieser Stabil läuft, Zugriff funktioniert und alle gewünschten Funktionen aktiv sind
--.06.2024	Abschaltung des On-Prem-Servers	Wenn festgestellt wird das der AWS-Server einwandfrei funktioniert und alle Daten migriert wurden, kann der On-Prem-Server abgeschaltet werden

Tabelle 13: Migrationsplan

2.1.7 Machbarkeit

2.1.7.1 Risiken

Beurteilung der Machbarkeit bzw. der Migrationsrisiken:

Nr.	Migrationsrisiken	Lösungsmöglichkeit	Restrisiko
01	Es existieren Fehler im bestehenden System:	Fehlerbeseitigung durch automatische oder manuelle Bereinigung im bestehenden System.	klein
02	Verbindungsunterbrüche	Blockweise Daten migrieren	klein

Tabelle 14: Machbarkeit

2.1.7.2 Rückfall-Szenario

Bei einem Rückfall wird die Migration zurückgestellt und wird vom Konzept aus wiederholt und wenn nötig angepasst.

2.1.7.3 Fehlerbehebungsszenario

Ab Point of no Return (Siehe Tabelle Migrationsplan) werden keine Rückfälle mehr durchgeführt und Fehlerbehebungen werden falls nötig gemacht.

2.1.8 Archivierung und Ausserbetriebsetzung Altsystem

Archivierungskonzept:

Vor der Migration werden die zu Migrierenden Daten dupliziert und sicher 10 Jahre nach hinten aufbewahrt. Diese werden bei Rechtlichen Prozessen vom IT-Spezialisten der Firma Traupel-IT-AG greifbar sein und werden so selten wie möglich hervorgenommen, um mögliche Verluste zu vermeiden.

Ausserbetriebsetzung:

Nachdem die Alte Datenbank ausser Betrieb ist, werden die Disketten in das Archiv gesichert und die restliche Hardware kann nun zu anderen Zwecken verwendet werden

2.1.9 Anforderungsabdeckung

Nr.	Anforderung	Beschreibung	Abdeckung
01	Anforderungen an das Projektteam	Sehr gute Kenntnisse in den Vorsystemen und im Zielsystem	erfüllt
02	Anforderung an das Projektteam	Technisches Know-How (Oracle Entwicklungswerkzeuge, Datenmodell)	teilweise
03	Data Cleansing	in den Schnittstellen-Tabellen	NICHT erfüllt
	Bereinigung schlechter Gewohnheiten.	Datensätze korrigieren falls vorhanden und Fehler bei der Migrationsübertragung zu vermeiden.	

Tabelle 15: Anforderungsabdeckung

2.1.10 Berechtigungskonzept

Gruppe	Globaladmin	Read auf alle Accounts	Read + Write auf eigenen Account	Read + Write auf alle Accounts	Read + Write auf alle Accounts des Teams
Admin	X			X	
GL		X			
HR				X	
Finanzen		X			
User			X		
Teamleiter					X

2.2 Testingkonzept

	Beschreibung
Einführung	Dieses Testkonzept beschreibt die Teststrategie und –Methoden, die wir anwenden, um unsere Datenbank erfolgreich migrieren zu können und dass die Datenbank korrekt in AWS funktioniert.
Testziele	<ul style="list-style-type: none"> -AWS Auto Scaling testen: Überprüfung der Skalierbarkeit der Anwendung. -AWS Elastic Load testen: Sicherstellen, dass die Lastverteilung effizient funktioniert. -Antwortzeit des Servers Prüfen: Sicherstellen, dass die Antwortzeit innerhalb akzeptabler Grenzen bleibt. -Berechtigungen (IAM) Testen: Überprüfung der Benutzer- und Rollenberechtigungen. -Integrität der Daten Prüfen: Sicherstellen, dass die Datenintegrität während und nach der Migration gewahrt bleibt.
Testmethoden	<ul style="list-style-type: none"> -Manuelle Tests: Funktionale Überprüfungen und Zugriffsrechte. -Automatisierte Tests: Nutzung von Tools zur Überprüfung der Skalierbarkeit und Lastverteilung. -Performance-Tests: Einsatz von Performance-Testing-Tools zur Messung der Antwortzeiten. -Sicherheitstests: Penetrationstests und Überprüfung der IAM-Richtlinien. -Datenvalidierung: Automatisierte Skripte zum Vergleich der Datenintegrität.
Ressourcen	<ul style="list-style-type: none"> -Tester: Zwei Cloud-Administratoren, ein Entwickler, ein Sicherheitsspezialist. -Testumgebung: AWS-Umgebung, lokale Umgebung für Vergleichstests.

	-Tools: AWS CloudWatch, AWS Load Balancer, JMeter, IAM-Analyzer, Datenintegritäts-Tools.
Testumfang	-AWS Auto Scaling: Simulation von Lastspitzen und Überprüfung der automatischen Skalierung. -AWS Elastic Load: Lasttests mit verschiedenen Workloads. -Antwortzeit des Servers: Messung der Latenzzeiten unter verschiedenen Bedingungen. -Berechtigungen (IAM): Überprüfung der Zugriffsrechte und Sicherheitsprotokolle. -Datenintegrität: Vergleich der Daten vor und nach der Migration.
Zeitplan	-Testplanung: 1 Woche -Vorbereitung und Backup: 1 Woche -Durchführung der Tests: 3 Wochen -Fehlerbehebung und Retests: 2 Wochen -Abschluss und Berichtserstellung: 1 Woche
Risiken und Gegenmassnahmen	-Skalierungsprobleme: Vorabtests und kontinuierliche Überwachung. -Lastverteilungsprobleme: Lasttests und Anpassungen in Echtzeit. -Antwortzeitprobleme: Optimierung der Serverkonfigurationen. -Sicherheitsrisiken: Regelmässige Überprüfungen und Updates der IAM-Richtlinien. -Datenintegritätsprobleme: Umfangreiche Tests und Validierungen vor dem Live-Gang.
Abnahmekriterien	-Erfolgreiche Skalierung ohne Ausfälle. -Effiziente Lastverteilung. -Antwortzeiten innerhalb akzeptabler Grenzen. -Korrekte Implementierung aller Berechtigungen. -Datenintegrität gewährleistet.
Berichterstattung	Regelmässige Statusberichte während der Testphase und ein Abschlussbericht am Ende, dieser soll die Testergebnisse, aufgetretene Probleme und empfohlene Massnahmen zusammenfassen.

2.2.1 Testziele (anpassen an Anforderungen)

Globale messbare Testziele über alle Testfälle hinweg:

Nr.	Beschreibung	Messgrösse	Priorität*
01	AWS Auto Scaling testen	Gross	M/2
02	AWS Elastic Load testen	Mittel	M/1
03	Antwortzeit des Servers Prüfen	Klein	2
04	Berechtigungen (IAM) Testen	Mittel	M/1
05	Integrität der Daten Prüfen	Gross	M/2
* Priorität: M = Muss / 1 = hoch, 2 = mittel, 3 = tief			

Tabelle 16: Übergeordnete Testziele

2.2.2 Testrahmen

Die Tests werden ausschliesslich in der AWS-Datenbank ausgeführt. Davon ausgeschlossen sind Netzwerkinfrastrukturen und Zugriffskonzepte, da an denen nichts verändert wird.

2.2.3 Testobjekte

Nr.	Objekt	Beschreibung
01	Stammdaten	Arbeitszeiten, Nutzerdaten können abgefragt werden
02	Bewegungsdaten	Welche Bewegungsdaten (Rechnungen, Gutschriften, Artikelbestände, Aufträge etc.)

Tabelle 17: Testobjekte

2.2.4 Testarten

Nr.	Testart	Beschreibung
01	Funktionstest	Überprüfen der Funktionen
02	Anwendertest	Überprüfung der Bedienung
03	Zugriffstest	Überprüfung, ob der Zugriff funktioniert.

Tabelle 18: Testarten

2.2.5 Übersicht Testfälle

Nr.	Testobjekt	Bemerkung
01	AWS Auto Scaling	Belastungstest
02	AWS Elastic Loadbalancing	Belastungstest
03	Latency Test	Ping & Erreichbarkeit
04	Berechtigungen (IAM)	Test zur Überprüfung, ob Zugriff und Berechtigungen erhalten geblieben sind.
05	Integrität der Daten	Datenintegritätsabfrage

Tabelle 19: Testabdeckung

2.2.6 Testfälle

2.2.6.1 Testfall 1

ID / Bezeichnung	T-01	AWS Auto Scaling
Beschreibung	Belastungstest	
Erwartetes Ergebnis	Bei höherer Belastung skaliert es Automatisch.	
Datum / Person	31.05.2024	Alle Autoren
Bemerkungen	keine	

Tabelle 20: Testfallbeschreibung

2.2.6.2 Testfall 2

ID / Bezeichnung	T-02	AWS Elastic Loadbalancing
Beschreibung	Belastungstest	
Erwartetes Ergebnis	Bei höherer Belastung skaliert es Automatisch.	
Datum / Person	31.05.2024	Alle Autoren
Bemerkungen	keine	

2.2.6.3 Testfall 3

ID / Bezeichnung	T-03	Latency Test
Beschreibung	Ping & Erreichbarkeit	
Erwartetes Ergebnis	Latency weniger als 20ms	
Datum / Person	31.05.2024	Alle Autoren
Bemerkungen	Ping oder Traceroute	

2.2.6.4 Testfall 4

ID / Bezeichnung	T-04	Berechtigungen (IAM)
Beschreibung	Test zur Überprüfung, ob Zugriff und Berechtigungen erhalten geblieben sind.	
Erwartetes Ergebnis	Berechtigungen verändern sich nicht	
Datum / Person	31.05.2024	Alle Autoren
Bemerkungen	Nutze verschieden berechtigte Nutzer als Test.	

2.2.6.5 Testfall 5

ID / Bezeichnung	T-05	Integrität der Daten
Beschreibung	Datenintegritätsabfrage / -abgleich	

ID / Bezeichnung	T-05	Integrität der Daten
Erwartetes Ergebnis	Integrität gewährleistet	
Datum / Person	31.05.2024	Alle Autoren
Bemerkungen	Daten mit denen vor der Migration abgleichen mit Methoden wie Hashwerte.	

3 Realisierung

3.1 Ausgangslage

Lokale Ubuntu VM in Smartlearn mit MySQL Server. Darauf sind Datensätze die vom, Filestash der GIBB sind und Personaldaten der Firma Traupel-IT-AG simulieren.

Datensätze von [Link](#) entnommen:

AWS-Account komplett leer ohne Konfigurationen.

3.1.1 Ausweichung auf Andere Methodik

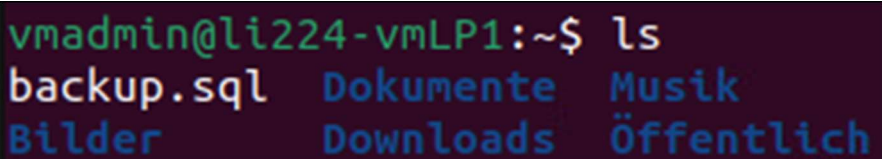
Im Prozess mit AWS DMS haben wir festgestellt, als wir das DMS Deployment ausführen, wollten das wir nicht die nötigen Rechte dazu haben, weil wir im Learner-Lab sind.

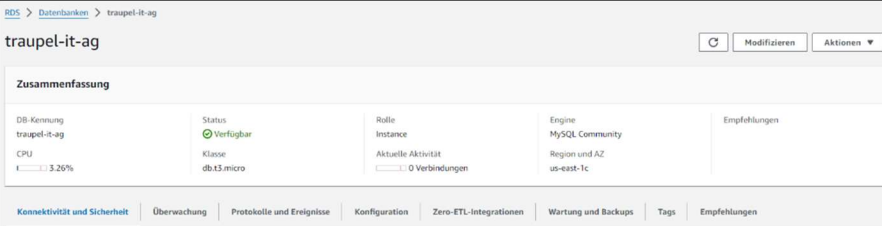
Deshalb sind wir ausgewichen auf die Methode das wir lokal Per Shell auf die AWS-Endpoint-Instanz zugreifen und dann von der Shell aus in die Daten Migrieren.

3.2 Anleitung

Jeder relevante Schritt wird hier erwähnt:

Schritt	Bild
Auf Lokaler Shell	
Öffne das Linux Terminal und starte MySQL	<pre> vmadmin@11224-vmLP1:~\$ sudo mysql -u root -p Enter password: Welcome to the MySQL monitor. Commands end with ; or \g. Your MySQL connection id is 8 Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu) Copyright (c) 2000, 2024, Oracle and/or its affiliates. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners. Type 'help;' or '\h' for help. Type '\c' to clear the current input statement. mysql> </pre>
Suche den Aktuellen Stand der lokalen Datenbank für den späteren Abgleich.	<pre> mysql> use traupel_it_ag Reading table information for completion of table and column names You can turn off this feature to get a quicker startup with -A Database changed mysql> show tables -> ; +-----+ Tables_in_traupel_it_ag +-----+ departments dept_emp dept_emp_latest_date dept_manager employees salaries titles +-----+ </pre>
Mit folgendem Befehl kann nun ein Dump von der Datenbank erstellt werden. Dabei wurde die Database:	<pre> sudo mysqldump \ --databases traupel_it_ag \ --master-data=2 \ --single-transaction \ --order-by-primary \ -r backup.sql \ -u root </pre>

«traupel_it_ag» verwendet und der Filename, der rauskommt backup.sql	<pre>sudo mysqldump \ --databases traupel_it_ag \ --master-data=2 \ --single-transaction \ --order-by-primary \ -r backup.sql \ -u root</pre>
Dieses kann man dann im aktuellen Verzeichnis sehen	

Auf AWS RDS	
Auf RDS sollte nun eine neue Datenbank Instanz erstellt werden.	
Stellen Sie sicher, dass die MySQL Version sich abgleicht.	
Auf dieser Übersicht sehen Sie unten links den Endpoint und den Port der später für die Migration entscheidend ist.	<p>Endpoint</p> <p>traupel-it-ag.c0x4t1wz7jwf.us-east-1.rds.amazonaws.com</p> <p>Port</p> <p>3306</p>

Auf Lokaler Shell	
Mit diesem Befehl verbindet man sich nun auf die installierte RDS-Datenbank-Instanz	<pre>mysql -h traupel-it-ag.c0x4t1wz7jwf.us-east-1.rds.amazonaws.com -P 3306 -u root -p</pre>
Der Befehl beinhaltet den Endpoint und den Port sowie Benutzernamen	

<p>Danach wird einem das Passwort aufgefordert und man ist verbunden mit der RDS-Instanz.</p> <p>Unterbrechen Sie nachher die Verbindung mit dem «Exit» Befehl</p>	<pre>vmadmin@li224-vmLP1:~\$ mysql -h traupel-it-ag.c0x4t1wz7jwf.us-east-1.rds.amazonaws.com -P 3306 -u root -p Enter password: Welcome to the MySQL monitor. Commands end with ; or \g. Your MySQL connection id is 45 Server version: 8.0.36 Source distribution Copyright (c) 2000, 2024, Oracle and/or its affiliates. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners. Type 'help;' or '\h' for help. Type '\c' to clear the current input statement. mysql> show databases +-----+ Database +-----+ information_schema mysql performance_schema sys +-----+ 4 rows in set (0.10 sec) mysql> exit Bye</pre>
<p>Mit diesem Befehl wird nun auf die RDS Datenbank Instanz zugegriffen und dann wird das Dump-File aus dem ersten Schritt in die RDS-Instanz deployed.</p> <p>Stellen Sie sicher, dass sie sich im gleichen Ordner wie das Dump-File befinden.</p>	<pre>mysql -h traupel-it-ag.c0x4t1wz7jwf.us-east-1.rds.amazonaws.com -u root -p < backup.sql</pre> <pre>vmadmin@li224-vmLP1:~\$ mysql -h traupel-it-ag.c0x4t1wz7jwf.us-east-1.rds.amazonaws.com -u root -p < backup.sql Enter password:</pre>

Wenn Sie sich jetzt nochmal mit RDS verbinden, dann sehen Sie mit einem einfachen SELECT Befehl, dass die Daten nun erfolgreich in die RDS-Instanz migriert sind.

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| traupel_it_ag |
+-----+
5 rows in set (0.10 sec)

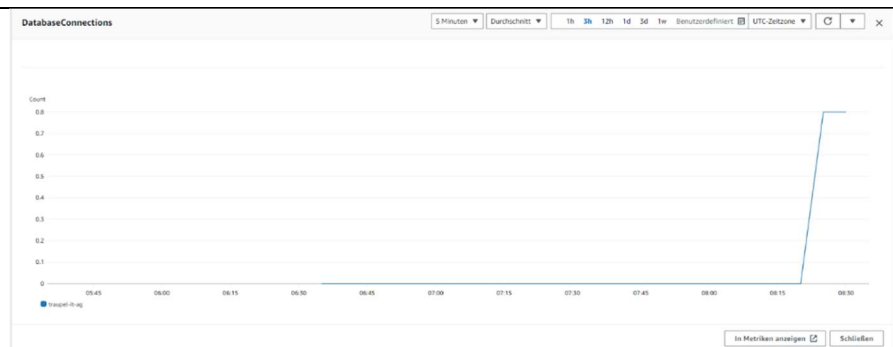
mysql> use traupel_it_ag
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_traupel_it_ag |
+-----+
| departments |
| dept_emp |
| dept_emp_latest_date |
| dept_manager |
| employees |
| salaries |
| titles |
+-----+
7 rows in set (0.10 sec)

mysql> select * from departments;
+-----+
| dept_no | dept_name |
+-----+
| d009 | Customer Service |
| d005 | Development |
| d002 | Finance |
| d003 | Human Resources |
| d001 | Marketing |
| d004 | Production |
| d006 | Quality Management |
| d008 | Research |
| d007 | Sales |
+-----+
9 rows in set (0.10 sec)
```

Auf AWS

Auf AWS kann man danach auch nachweisen, dass es sich wirklich um diese Instanz handelt.



3.3 Testfall Ergebnisse

Ergebnisse der Testfälle

Test NR	Ergebnis	Kommentar / Wie geht es weiter?
1	Erfolgreich	Automatische Skalierung eingerichtet
2	N/A	Nicht gemacht da nur eine DB-Instanz erstellt wurde.
3	Erfolgreich	Low-Latency garantiert
4	Erfolgreich	Die IAM-Rollen und Rules sind erhalten wie zuvor <div> <div>Rollen</div> <div>20</div> </div> <div> <div>Richtlinien</div> <div>6</div> </div>
5	Erfolgreich	Datenabgleich ohne Verluste <pre> mysql> show databases; +-----+ Database +-----+ information_schema mysql performance_schema sys traapel_it_ag +-----+ 5 rows in set (0.10 sec) mysql> use traapel_it_ag Reading table information for completion of table and column names You can turn off this feature to get a quicker startup with -A Database changed mysql> show tables; +-----+ Tables_in_traapel_it_ag +-----+ departments dept_emp dept_emp_latest_date dept_manager employees salaries titles +-----+ 7 rows in set (0.10 sec) mysql> select * from departments; +-----+ dept_no dept_name +-----+ d009 Customer Service d005 Development d002 Finance d003 Human Resources d001 Marketing d004 Production d006 Quality Management d008 Research d007 Sales +-----+ 9 rows in set (0.10 sec) </pre>

4 Einführung

Die Tests wurden erfolgreich durchgeführt und die Abnahme erfolgt am 21.06.2024

Es wurden nach der Anleitung keine weiteren Änderungen vorgenommen.

Die Protokolle sind allesamt dokumentiert und verzeichnet in diesem Dokument.

5 Tabellen Verzeichnis

Tabelle 1: Änderungskontrolle	1
Tabelle 2: Betroffenheitsanalyse	4
Tabelle 3: Kosten	5
Tabelle 4: Ziele	5
Tabelle 5: Funktionale Anforderungen.....	6
Tabelle 6: Nicht Funktionale Anforderungen	7
Tabelle 7: Risikoanalyse	8

Tabelle 8: Risikomatrix vor Massnahmen	9
Tabelle 9: Risikomatrix nach Massnahmen	9
Tabelle 10: Migrationsziele	10
Tabelle 11: Anforderungen	10
Tabelle 12: Migrationsobjekte.....	10
Tabelle 13: Migrationsplan	11
Tabelle 14: Machbarkeit	12
Tabelle 15: Anforderungsabdeckung	12
Tabelle 16: Übergeordnete Testziele	15
Tabelle 17: Testobjekte	15
Tabelle 18: Testarten.....	15
Tabelle 19: Testabdeckung.....	15
Tabelle 21: Testfallbeschreibung.....	16

6 Abbildungsverzeichnis

Abbildung 1: Migrationsverfahren.....	11
Abbildung 2: Vorgang.....	19