Zero Downtime Database Migration from On-Premises to a Cloud-based Server

Neil Comisioneru
Team 04, IS340 – Operating Systems, Masters of Science in Data Science, School of Technology and Computing, City University of Seattle

Nicole Hessner
Team 04, IS340 – Operating Systems, Masters of Science in Computer Science, School of Technology and Computing, City University of Seattle

Marie Aimee Karumuhinzi
Team 04, IS340 – Operating Systems, Masters of Science in Data Science, School of Technology and Computing, City University of Seattle

Kenneth Melnrick
Team 04, IS340 – Operating Systems, Bachelors of Science in Applied Computing, School of Technology and Computing, City University of Seattle

Gayatri Soni
Team 04, IS340 – Operating Systems, Masters of Science in Computer Science, School of Technology and Computing, City University of Seattle

Jerome Spearman
Team 04, IS340 – Operating Systems, Bachelors of Business Adminstration in Finance and Insurance,Temple University

neilc1111@cityuniversity.edu, hessnernicole@cityuniversity.edu, karumuhinzimarieaim@cityuniversity.edu, melnrickkenneth@cityuniversity.edu, sonigayatri@cityuniversity.edu, spearmanjerome@cityuniversity.edu

Abstract

Migrating databases from locally hosted servers to the cloud offers many advantages, namely the ability to pay for "right size" services and simplified scalability. Typically, with database migration, the database engine is upgraded and then the entire installed database of files is moved to a new device. Database migrations are typically more complex than storage migrations due to high data volumes and the relationality of the data. The steps involved in migrating a database include performing a backup, detaching the database from the first engine, and then migrating the files to the new engine in the

cloud, where the files can be restored to the new database and location. Since users today expect the services, they use to always be available, a key goal in this process is minimizing downtime, while ensuring no data is lost. In this paper, our team will explore setting up a new server in the cloud and the steps that will be used to migrate an on-premises database to a database accessible from the cloud.

Keywords

Database migration, cloud migration, downtime minimization, data loss prevention, source system, target system, reconciliation

1. INTRODUCTION

In this paper we proposed Database Migration from On-Premises to a Cloud-based Server. Cloud computing is a multi-sharing environment, in which resources are shared. Threats can happen from anywhere, inside or outside the common environment. Deciding whether to migrate or retain sensitive data in the workspace is one of the most important decisions faced by personal users, as well as small and medium-sized enterprises.

We have described throughout this paper several key points starting from explaining why we need data migration, its strategy, data migration phases, data migration problems, and challenges when performing data migration, data migration types, categories, methodology, risks, and solutions.

Cloud computing is one of the highly thriving data storing and data sharing mechanisms in the current computing environment. Cloud migration is relocating the existing on-premises data from the physical server to a cloud server which is controlled and monitored by the cloud service provider. In addition, the security threats associated with cloud computing may cause great deprivation to organizational secure data. By being aware of security concerns, we proposed the migration team can develop a cloud migration strategy to protect business data, operating systems, application programs.

The major goal of this paper is to ensure the existing systems meet the new business needs in cloud. Finally, we concluded Cloud computing is a cost-effective solution with many features which help enterprises to run businesses in an environment-friendly atmosphere.

1. DATA MIGRATION STRATEGY AND BEST PRACTICES

The goal of any data migration is typically to enhance performance and competitiveness. To achieve this, a thorough data migration plan must be established and some of the critical factors to consider are discussed below.

- The source data must be understood to ensure accuracy and completeness of the migrated data after migration. An audit of source data should be done to identify any issues that must be resolved before data migration process begin. This means also identifying the most logical grouping of data to be migrated.
- A migration environment with proper checkpoints, controls and audits should be established to identify and resolve any issues/errors that occur during the migration process. It is critical to identify a solution (i.e. tools and technologies) that will be used in reconciliation at different checkpoints until the migration is complete.
- Migration development and testing activities should be identified for both source and target databases. This will require an understanding of legacy and target applications (i.e. configuration, capacity). Also, it should be decided if this migration process will be manual and/or automated.

There are two major data migration implementation methods that are typically recommended depending on the organization's business needs and requirements.

- "Big Bang" Migration is the full transfer of data from one source application to a target application all at once. This can be achieved in a very short period and is ideal for smaller companies. The drawback of this implementation method is that it can be costly.
- "Trickle" Migration is implemented in multiple phases and its design is typically complex. There is typically no downtime during data migration since the source application remains available until complete transfer of data has been

accomplished. This is ideal for mid-large size companies.

1.1 Best Practices for Data Migration

Here are data migration best practices that will help turn data transfer into success.

- **Determine the project's scope.** First things first: Get a handle on what data should be migrated, then document existing and target formats. Consider working with an experienced storage partner to optimize the plan, as making the transition involves the many regulations covering your legacy data.
- **Ensure that the migration plan is compatible with existing policies.** Clearly, an important part of the process is addressing regulatory and litigation concerns. The data may be subject to specific handling requirements, such as encryption or verified chain-of-custody protection. In addition, any data migration must fit into your enterprise's overall information management strategy.
- **Establish a migration time frame.** Data move should be as minimally disruptive to current business operations as possible. Good planning is essential. Set realistic dates, with the understanding that the information may need to be de-duplicated and cleaned up before it's migrated. A careful plan should be able to avoid costly delays or cost overruns.
- **Validate and test data post-migration.** Team must make sure that all data has been completely and accurately replicated in the new format before any legacy media is destroyed.
- **Audit and document every step of the process.** Your plan must be defensible, with complete paperwork to show what was done and when.

2. DATABASE MIGRATION ROADMAP

Before the database migration begins, it is critical to know what type of data is being migrated and how the source system is structured. The outcome of the analysis will determine the migration design to the target system. The discovery process will be achieved through the following phases.

2.1 Requirement Plan

The requirement plan analysis the architecture of the legacy database. During this stage, several key activities need to be performed to ensure the project's success.

First, to determine the migration scope, the migration team needs to understand how the legacy database is structured and define the type of data that needs to be converted. Next, the team needs to know how the target database is formatted and determine how to extract the legacy data into the target system (Microsoft, 2019). This process requires an in-depth analysis and profiling of the legacy data to identify data quality, gaps, and anomalies. Data profiling is an intricate process in assessing the quality of the source data. The method identifies two potential data quality issues; non-compliance with the metadata structure and corrupted data in the database. Lastly, migration acceptance criteria must be developed and signed-off by the stakeholders.

2.2 Design Plan

Best practices recommend establishing the following activities to increase the success of the project:

- Design integration Plan
- Design Parallel Operation
- Design Mapping Plan for Data Migration

Data migration is seldom a direct transfer between the source and the target database. Utilizing a staging environment allows the migration team to validate, clean, and test the conversion of the integrated data from the source system before moving it into the target system.

2.2.1 Design Parallel Operation Plan

Best practices suggest running both databases parallel to each other for a short time frame if errors are discovered with the migrated data. (Fillinich et al., 2019). This approach allows the migration team to roll back the migration process to the point of the error and make the necessary revisions.

2.2.2 Design a Mapping Plan for Data Migration

The migration team must now map the various architectures, providing a path for data to travel from the source through the staging area and finally to the destination source. Additionally, the design must contain any data mapping to the rules, translations, and transformation processes that must be followed when the data is moved from one data place to another.

2.2.3 Development Plan

At this phase, the migration team must build technical procedures to adhere to the

requirements defined in the requirement phase. It includes two main activities:

- Development of the conversion code
- Development of test scenarios and unit test

2.2.4 Develop and Test Code Conversion Procedure

This stage implements converting the data from the source database to the target one, including mainly the development of the ETL procedures. (Sarmah et al., 2018). Part of these procedures includes data cleansing, completeness, and data validation.

In the instance where the architecture of the source database is different from the target database, data values and formats need to be transformed to meet the target source structure requirements. This process is implemented in the staging area. Once the data is loaded into the staging area, an extraction procedure needs to be developed to upload the data into the target source (Sarmah et al., 2018).

2.2.5 Develop Validate and Test Data Procedure

The migrated data must be tested and validated to comply with corporate and government rules and processes. At each stage of the migration, the migration team must create a process to check the integrity of the data. The process can be automated or done manually. However, the primes is to validate the migrated data for completeness and accuracy.

The procedure needs to confirm that data loss and accuracy are within the error tolerance threshold set in the migration plan. If 100% migration between the databases is not feasible, the migration team must consult with the leadership team and revise the accepted criteria to an achievable level without jeopardizing any compliance rules.

2.3 Testing Plan

Before deploying the migration plan, a test trial must be conducted. Two testing methods can be performed during the testing procedure:

- QA test on the complete set of data
- QA test on a representative subset

The testing environment must be configured to mirror the production environment to simulate real-world scenarios. The data must be validated upon completion of the migration with a QA test. If the outcome of the complete data set is 100% successful, the plan will be accepted, and the implementation process can begin. A similar process can be applied to a QA test performed on a subset. However, best practices recommend that multiple tests be conducted to increase confidence in the accuracy of the data migration (Google Cloud, 2020). If errors are identified during the testing trial, the necessary fixes must be made, and the test trial must be repeated.

2.4 Delivery Plan

The final stage is the deployment plan and its consists of the following activities:

- Limited Deployment and Reconciliation
- Rollback Plan
- Cutover plan

2.4.1 Limited Deployment and Reconciliation

Once the migration testing meets the accepted criteria, the migration team can begin executing the migration plan. The deployment of the migration plan should be done in waves to mitigate the risk of potentially disrupting to the rhythm of business (downtime). The first wave of deployment should be conducted on a small sample of data, for example, the company's records of a small subset of employees. The second wave of deployment will migrate a more significant subset of data- for example, the entire records of a department. The last wave of deployment will be the migration of the complete database, which was defined in the requirement phase (Google Cloud, 2020). The data will be reconciled and verified at each wave to check that the quality of the data still meets the accepted criteria requirements. All errors are captured and mitigated according to the mitigation plan.

2.4.2 Rollback Strategy

Having a rollback strategy in place is an essential safeguard if the migration fails during any wave of deployment. If the deployment fails at any wave or fails to meet the accepted requirements, the migration team can restate the database to its original form or state where the activity failed to pass. The planning of the rollback strategy should be developed throughout the testing and deployment phases.

2.4.3 Cutover Plan

The cutover procedure includes the final migration effort to the target database, the changeover to the new system, user training, and the decommissioning of the legacy database (Sarmah et al., 2018).

In the case where the plan is to shut down the source database completely, best practices recommend running both databases in parallel for

a while until gaining the level of confidence needed to decommission.

3. PHASES OF DATABASE MIGRATION

3.1 Data Assessment

The first phase of database migration is Data Assessment. In this phase a complete review of the existing data is performed which not only is crucial to ensuring that all necessary data is migrated correctly, it also is critical to defining the scope of the migration (NordicBackup, n.d.).

Basic information about the data to be migrated must be examined for both the existing data and the target platform. Critical points to consider are format, location, and schemas.

There must be a backup plan. This is vital to the success of the migration as any data loss could be potentially catastrophic. Having backups as well as a roll-back plan to restore data is necessary for any migration.

Nandini (2022) describes data mapping as needed step in assessment. In examining the data, there is high level information such as data types and low level information such as if a particular data set is required in the target, or if a required field in the target data is not null in the existing data.

Data Integrity must be maintained throughout the process. Cote (2021) states there are three main elements to data integrity: "accuracy, completeness, and quality of data."

These are only some highlights of the Data Assessment phase; a detailed and exhaustive discussion is far outside the scope of this paper. In short, however, the purpose of the Data Assessment phase is to identify all existing data and ensure that all necessary data will exist in the final product.

3.2 Data Cleansing

During the Assessment phase, issues may be identified which threaten data integrity such as inconsistency, inaccuracies, duplication of data, etc. Ensuring that you are migrating clean data has many benefits such as lower time and cost requirements. Kline (2019) suggests that in addition to identifying data to keep during the migration, it is also necessary to identify data to remove. The removal comes in two varieties: purge and archive.

In a data purge, historical data that is outdated and no longer used should be ignored. Low-quality data (data with errors, for example) should also be ignored when migrating all data.

In an archive, data that is no longer needed for operations but may have some use in examining trends or for predictive analysis can be moved to a read-only archive saving costs not only during the migration but in long-term storage.

Again these are only highlights and examples of Data Cleansing. The goal is to move only valuable data to the target, archive potentially useful (but non-changing) data, and remove all junk data. This process lowers time and cost requirements as well as helps refine the scope of the project.

3.3 Test Extract, Transform, Load

After the data has been cleaned, migration testing begins. Testing small sets of data provides troubleshooting information for fixing errors as well as verify processes work as intended. This process is most likely to use the Extract Transform Load (ETL) model (Yaddow, n.d.).

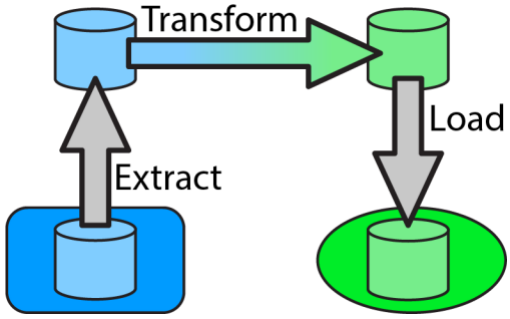

Figure 1. ETL model. IS340 Team 4, 2022.

In some cases, the legacy and target environments may be the same (such as migrating an SQL database to another SQL tool) and the Transform step may not be needed. In most cases, data will be transformed from one format to another (such from an SQL server to AWS server). Testing is vital to ensure data integrity.

- Extract data from database
- Run data through transformation software/process
- Load data to target
- Identify data translation errors
- Test for missing/unavailable data
- Measure time taken
- Identify process/tool performance and/or errors

Testing the migration process is vital to identifying errors on small data sets in order to

address problems quickly rather than after a full migration has occurred which would greatly increase the time, effort, and cost of the project. Small ETL tests can identify errors as well as give an accurate estimate of time and effort needed for the complete migration.

3.4 Final Extract, Transform, Load

This phase is almost exactly the same as the Test ETL phase with the difference that the focus is not on testing. This phase is the actual migration process, where the final product will be a fully functional target.

3.5 Migration Validation

Once the migration has been completed, it must be validated to ensure all data migrated successfully and that it can be retrieved, stored, and manipulated as specified.

The most critical aspect of validation is data integrity, which is verifying that the data came through the migration without alteration or errors. It is also critical to examine the schema changes and make sure all data can be accessed as expected.

In the validation stage, communication must be tested as well to ensure that all parts of the database from the storage itself to the end-user interface can traverse firewalls, use the correct and expected ports, etc.

In order to test validation, there are two main options. The first is a full audit in which 100% of the data is tested. This has the benefit of being the most accurate, but it is also more time and cost consuming and is recommended for relatively small data sets.

For larger data sets, or when time and cost are limiting factors, a sampling strategy can be employed. With sampling several smaller data sets are audited, and the results of those audits are used to extrapolate general results. While not as accurate as a full audit, the time and cost savings can be significant with large data sets.

The validation phase is critical to ensure that moving forward, everything works as expected and there are no surprises, which could have catastrophic consequences.

3.6 Post Migration

There must be a plan in place to resolve any migration failures or errors, as well as reconcile any mapping errors. As the goal is to fully implement the target system, not only the data needs to be reviewed, but the entire system needs to be tested and verified. Some key items to be reviewed include:

- Security
- Load handling/performance testing
- Functionality
- Data flow
- Usability

After the data migration is complete, the system must be fully vetted to make sure the data is secure and accurate, that the system behaves as expected, and that the user experience (UX) ensures end-users can actually use the new system.

4. RECONCILIATION

The reconciliation procedure plays a critical part in the delivery phase of the project. As mentioned earlier, the process reconciles and audits the migration deployment to ensure data has been accurately and completely transferred. Best practices recommend that audit scripts be prepared to compare the data's quality and quantity (number of records) between the target and the source databases. The scrips should be executed during the testing trial and deployment layers. The scrips should be run and generate the following reports at each sub-layer:

- Legacy report: identification of the data type and the count of records in the legacy system.
- Extraction report: Each time data is transferred, it is necessary to verify that it meets the acceptance requirements.
- Transformation report: validation that the transformed data meets the schema structure of the target system (if the data transformation is required).
- Target report: validation of record counts in the target system.

If any errors are identified, the migration team must analyze the issues and debug the code before reloading back into the target database.

5. COST ASSOCIATED WITH DATABASE MIGRATION

5.1 Perform a Cost Benefit Analysis

When considering a database migration, it is best practice for the organization to perform a Cost Benefit Analysis (CBA). A CBA is a process that analyzes the approximated cost and benefits of a project, to discover solution feasibility from a business point of view. It is a data driven unbiased approach that could help the decision-making process less complex.

5.2 Conduct an On-Premises Cost Analysis

Before moving from an on-premise to a cloud infrastructure, the organization should perform a current inventory assessment of all software and hardware used within the existing infrastructure. Once this information is collected, an analysis of the direct and indirect operating costs should be performed. Examples of such costs are:

- Direct Cost
  - Data Center Cost
  - Hardware Cost
  - Software Licenses
  - Operational/Labor Cost
  - Maintenance
- Indirect Cost
  - Server downtime cost and its impact on employee productivity.
  -

5.3 Cloud Cost Estimation

The cost of cloud estimation is provider dependent. However, the top providers such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Computing (GCP), offer calculators that could assist with gathering a general perspective of the cost based upon a desired tier. Keys factors that will have an impact of cost are the following:

- Number of servers needed
- Volume of data to be transferred
- Applications that may need to be rewritten native to the cloud.
- Cloud Infrastructure cost
- Labor

5.4 CapEX versus OpEX Spending Model

The on-premises infrastructure cost is based upon a Capital Expenditure (CapEx) expense model. It is the initial outlay of capital needed to support physical infrastructure, and then deducting the up-front expense over time. The upfront cost from CapEX has a value that reduces over time. (Geek for Geeks)

Whereas, Operational Expenditure (OpEX), is a pay as you go service. There is no up-front cost, you pay for a service or product as you use it. (Geeks for Geeks). The cloud infrastructure follows the OpEX model. The key difference between the two models is CapEX requires an initial outlay of funds upfront. Whereas OpEX cost are expensed over time. It is important to understand the differences between the two models as it can impact the decisions within the CBA.

5.5 Migration Cost Estimation

The greatest expense that that organizations will incur during a migration is skilled labor hours. On average, the manual effort and hours required to perform the migration process successfully are charged as low as US$1000 per server to as much as US$3000 per server and can even go up to US$15000 for complication cases. (Cloud Savvy)

6. DATABASE MIGRATION RISK

When considering moving to the cloud, there are also several risk factors that will need to consider. These include:

6.1 Data Loss Risk

During the data migration process, data loss can occur. When the data is migrated to the new system or target system, some of the data may not migrate over from the source system. This risk can be evaded by conducting data migration testing.

6.2 Incomplete deletion of data. While data loss is a real concern, there can be another issue at the other end of the spectrum. When data is spread out over multiple storage devices, this risk is even more concerning. When data deleted in a timely manner, but that data is saved on the same hardware as data from other clients, it's not as simple as just destroying the disk. In addition, because procedures for deletion may differ from one provider to another, it can be difficult to verify whether data was deleted securely and ensure that none of it remains.

6.3 Lack of a clear cloud migration strategy: Before we jump head-first into cloud computing, we will need to consider several questions and come up with a clear strategy for migrating to the cloud. Consider reasons for moving to the cloud and the benefits hoping to achieve. Think about what and how much data we want to migrate to the cloud - there might be especially critical or sensitive data that we want to keep on-premises. Also decide how much storage we require and how many different cloud providers we might need. Don't skip over this necessary planning stage because rushing in without a cloud migration strategy can lead to problems such as enormous expenses or system failures.

6.4 Security risks. These are probably the biggest risks during data migration to cloud. Migrating to the cloud involves all kinds of security risks, from compliance violations and contractual breaches to insecure APIs, accidental errors, malware, external attacks, and more.

6.5 Incompatibility of existing architecture. Cloud migration can be challenging when businesses depend on legacy applications, which may rely on programming languages, execution environments, and system libraries that aren't readily available in or aren't fully supported by the cloud.

6.6 Unwanted latency. Added latency is a risk of cloud migration that is often underestimated. A few seconds' delay by app can seriously damage business. Not only is latency a cause of frustration for customers but it can gravely impact brand reputation.

7. DATABASE MIGRATION CHALLENGES AND POSSIBLE MITIGATIONS

7.1 Poor migration strategic planning

During the project initiation phase, it is critical that business objectives and goals are well defined. 80% of projects fail due to rushed planning, lack of understanding of the business need and failure to support the need with quantifiable key performance indicators. This could be mitigated by challenging the business to clearly identify their needs and vision, through proper data gathering techniques, such as probing interview questions or system workflow reviews. A proper migration strategy would address clear goals, objectives, scope, constraints (budget, cost, human capital, time) and any known or unknown risks.

7.2 Data & System Security

Some of the data that an organization collects may contain high sensitivity information. A significant level of planning must be performed to protect the organization from data being lost intra-migration and post migration. To prevent this from occurring a proper review of the database schema should be performed. The first step in moving databases from one platform to another is to convert the schemas so that the structure of the data works within the new database.(Alooma) Once the data is moved, proper testing and validation must be conducted to ensure that transfer was completed properly and does not contain missing or null values.

7.3 Improper Testing and Results Evaluation

Testing is an important aspect of any database migration. This is where the end user can view the actual data that is transferred after the design and development phase has occurred. The detection of incompatible data after the design phase could lead to a significant amount of rework and frustration. Timely evaluation of the results during the design phase is considered best practice. A test-driven development approach would reduce a significant amount of strategy revamping, while involving the end user in test case prototype development and subsequent data output before migration occurs.

8. CONCLUSION:

The process of migrating from an on-premises infrastructure to a cloud-based solution requires an immense amount of planning, research, and resource utilization. By moving to the cloud, organizations can avoid high CAPEX expenses, while increasing the scalability of their operations. Attached to these perceived benefits, are inherent comes risks and challenges that must be considered. Such obstacles could be mitigated by the implementation of project, process and change management tools through every phase of the plan.

9. REFERENCES

[1] Alley, G. (2019, March 28). Database Migration Challenges.Alooma. https://www.alooma.com/blog/database-migration-challenges

[2] Cloud Migration Strategy – The Ultimate Guide to the 6 R's. (n.d.). Simform. Retrieved January 30, 2022, from https://www.simform.com/blog/cloud-migration-strategy/

[3] Corporate Finance Institute. (n.d.). Data Migration. https://corporatefinanceinstitute.com/resources/knowledge/data-analysis/data-migration/

[4] Cote, Catherine. (2021, February 4). What is data integrity and why does it matter? https://online.hbs.edu/blog/post/what-is-data-integrity

[5] Database migration: Concepts and principles (Part 1) Cloud Architecture Center. (2020,May 11). Google Cloud. https://cloud.google.com/architecture/database-migration-concepts-principles-part-1

[6] Kline, Kevin. (2019, October 14). What's step 1 in a cloud migration? Data cleansing. https://www.sentryone.com/blog/whats-step-1-cloud-migration-data-cleansing

[7] GeeksforGeeks. (2021, October 6). CapEx vs OpEx in Cloud Computing. https://www.geeksforgeeks.org/capex-vs-opex-in-cloud-computing/

[8] Fillinich, Mike.(2019). Data Migration Roadmap Guidance.studentaid.gov. https://studentaid.gov/sites/default/files/fsawg/static/gw/docs/ciolibrary/Data_Migration_Roadmap_Guidance.pdf.

[9] Nordic Backup. (n.d.). 7 Steps to include in your Data Migration Plan. https://nordic-backup.com/blog/7-steps-data-migration-plan

[10] The Total Economic Impact™ of Microsoft Azure Serverless Platform. A Forrester Total Economic Impact™ Study Commissioned by Microsoft. July 2019.

[11] Sarmah, Simanta (2018). "Data Migration." Alpha ClinicalSystems,8,1-10.10.5923/j.scit.20180801.01

[12] What Is a Cloud Migration Strategy? (n.d.). Cisco. Retrieved January 30, 2022, from https://www.cisco.com/c/en/us/solutions/cloud/what-is-a-cloud-migration-strategy.html

[13] What Is Cloud Migration? What To Know Before Migrating. (n.d.). CloudZero. Retrieved January 30, 2022, from https://www.cloudzero.com/blog/cloud-migration?hs_amp=true

[14] Upkin, S. (2020, September 7). How to Calculate the Average Cost of Cloud Migration. CloudySave. https://www.cloudysave.com/cloud-cost-optimization/how-to-calculate-the-average-cost-of-cloud-migration/

[15] Yaddow, Wayne. (n.d.). Foundations of Data Extraction Transform Load (ETL). https://www.ewsolutions.com/foundations-of-data-extraction-transform-load-etl