**Differences between PaaS, SaaS and IaaS**

Basically they present three modern but different ways to describe how you can use the cloud for your business. IaaS (infrastructure as a service) is a model where you don't have to worry about setting up and configuring hardware, but you manage everything from the OS level and up. Typically, IaaS offerings are virtual machines, such as Azure VM or AWS EC2. PaaS (platform as a service) provides you with a pre-configured environment that you can run your apps on. This is usually the optimal choice when your application is packaged and deployed as a Docker container. Examples include Heroku, Netlify, Now.sh, Azure App Service, AWS Elastic Beanstalk. SaaS (software as a service) model abstracts away infrastructure and environment and simply provides a solution for a specific problem. This includes pretty much all software solutions hosted in the cloud, e.g. GitHub, GMail, Trello, Slack, Dropbox.

**Differences between ETL and ELT**

ETL (Extract, Transform, Load) and ELT (Extract, Load, Transform) are data integration methods. ETL and ELT differ in two primary ways. One difference is where the data is transformed, and the other difference is how data warehouses retain data.

- ETL transforms data on a separate processing server, while ELT transforms data within the data warehouse itself.
- ETL does not transfer raw data into the data warehouse, while ELT sends raw data directly to the data warehouse.

Here is the chart listing the pros and cons of them:

| Category | ETL | ELT |
|---|---|---|
| **Definition** | Data is extracted from a source system, transformed on a secondary processing server, and loaded into a destination system. | Data is extracted from a source system, loaded into a destination system, and transformed inside the destination system. |
| **Extract** | Raw data is extracted using API connectors. | Raw data is extracted using API connectors. |
| **Transform** | Raw data is transformed on a processing server. | Raw data is transformed inside the target system. |

| | | |
|---|---|---|
| **Load** | Transformed data is loaded into a destination system. | Raw data is loaded directly into the target system. |
| **Speed** | ETL is a time-intensive process; data is transformed before loading into a destination system. | ELT is faster by comparison; data is loaded directly into a destination system, and transformed in-parallel. |
| **Code-Based Transformations** | Performed on secondary server. Best for compute-intensive transformations & pre-cleansing. | Transformations performed in-database; simultaneous load & transform; speed & efficiency. |
| **Maturity** | Modern ETL has existed for 20+ years; its practices & protocols are well-known and documented. | ELT is a newer form of data integration; less documentation & experience. |
| **Privacy** | Pre-load transformation can eliminate PII (helps for HIPPA). | Direct loading of data requires more privacy safeguards. |
| **Maintenance** | Secondary processing server adds to the maintenance burden. | With fewer systems, the maintenance burden is reduced. |
| **Costs** | Separate servers can create cost issues. | Simplified data stack costs less. |
| **Requeries** | Data is transformed before entering destination system; therefore raw data cannot be requeried. | Raw data is loaded directly into destination system and can be requeried endlessly. |
| **Data Lake Compatibility** | No, ETL does not have data lake compatibility. | Yes, ELT does have data lake compatibility. |
| **Data Output** | Structured (typically). | Structured, semi-structured, unstructured. |

| Data Volume | Ideal for small data sets with complicated transformation requirements. | Ideal for large datasets that require speed & efficiency. |