**Database selection criteria**

When choosing a database, different people use different criteria:

* Database procurement staff pay more attention to **purchase costs, including storage and network requirements**.
* Database administrators (DBAs) care about:
  + **Operation and maintenance costs**:
    - A reliable monitoring and alerting system
    - Support for backup and restore
    - Reasonable upgrade and migration costs
    - An active support community
    - Ease of performance tuning
    - Ease of troubleshooting
  + **Service stability:**
    - Support for multiple data replicas
    - Highly available services
    - Support for multiple writes and multi-active architecture
  + **Performance:**
    - Latency
    - Queries per second (QPS)
    - Whether it supports more advanced hierarchical storage features
  + **Scalability:** Whether it's easy to scale horizontally and vertically
  + **Security:** Whether it meets audit requirements and prevents SQL injections and information leakage
* Application developers care about:
  + **Stable services**
  + **Performance**
  + **Scalability**
  + **Ease of developing database interface**
  + **Ease of modifying the database schema**
  + **What Is Structured Data?**

Structured data usually resides in relational databases (RDBMS). Fields store length-delineated data like phone numbers, Social Security numbers, or ZIP codes. Records even contain text strings of variable length like names, making it a simple matter to search. Data may be human- or machine-generated, as long as the data is created within an RDBMS structure. This format is eminently searchable, both with human-generated queries and via algorithms using types of data and field names, such as alphabetical or numeric, currency, or date.

Common relational database applications with structured data include airline reservation systems, inventory control, sales transactions, and ATM activity. Structured Query Language (SQL) enables queries on this type of structured data within relational databases.

Some relational databases store or point to unstructured data, such as customer relationship management (CRM) applications. The integration can be awkward at best since memo fields do not lend themselves to traditional database queries. Still, most of the CRM data is structured.

**What Is Unstructured Data?**

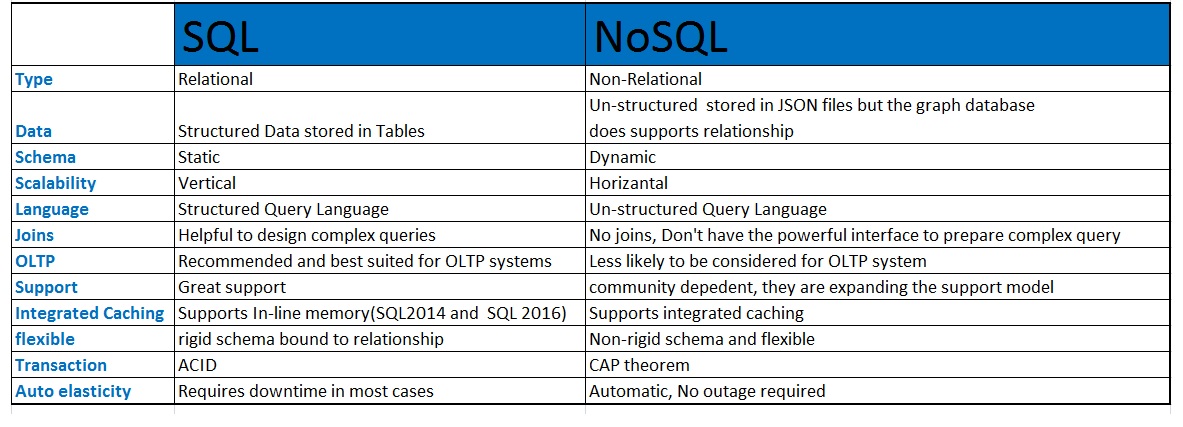
Unstructured data is essentially everything else. Unstructured data has an internal structure but is not structured via predefined data models or schema. It may be textual or non-textual, and human- or machine-generated. It may also be stored within a non-relational database like NoSQL.

Typical human-generated unstructured data includes:

* **Text files:** Word processing, spreadsheets, presentations, emails, logs.
* **Email:** Email has some internal structure thanks to its metadata, and we sometimes refer to it as [semi-structured](https://www.datamation.com/big-data/semi-structured-data/). However, its message field is unstructured and traditional [analytics tools](https://www.datamation.com/big-data/top-big-data-tools/) cannot parse it.
* **Social Media:** Data from Facebook, Twitter, LinkedIn.
* **Website:** YouTube, [Instagram](https://sixads.net/alternatives/instagram-ads/), photo sharing sites.
* **Mobile data:** Text messages, locations.
* [**Communications:** Chat](https://getvoip.com/cpaas/), IM, phone recordings, collaboration software.
* **Media:** MP3, digital photos, audio and video files.
* **Business applications:** MS Office documents, productivity applications.

Typical machine-generated unstructured data includes:

* **Satellite imagery:** Weather data, landforms, military movements.
* **Scientific data:** Oil and gas exploration, space exploration, seismic imagery, atmospheric data.
* **Digital surveillance:** Surveillance photos and video.
* **Sensor data:** Traffic, weather, oceanographic sensors.



**What's a BLOB?**

We've come a long way in computing power since the days when computers filled entire rooms. It's now possible to store and retrieve huge chunks of data. One of those chunks is really a shapeless, undefinable entity: a BLOB. It's not text, or numbers, or dates and times.

You might think this is science fiction, but it's very real. Technically a **Binary Large Object (BLOB)** is an object data type, meaning it refers to an object. Unlike a character or integer data type, the object data type only contains a pointer or reference to the value of the object. A BLOB can hold a very large block of data, anything from documents to images to videos. You could store your great American novel in a BLOB if you really wanted to (as a file).

A BLOB is really the object's agent, or handler. The database manager shouldn't need to know what's in the file or how to work it, but it can still be a part of the database.

Let's take a look at some database management systems and how they support BLOBs.

**MySQL**

MySQL supports four BLOB types:

1. TINYBLOB
2. BLOB
3. MEDIUMBLOB
4. LONGBLOB

These are all BLOBs, but they differ in how large they can be. TINYBLOB is only about 256 bytes, and LONGBLOB is 4 gigabytes! Why would we even create a TINYBLOB, since at 256 bytes, it can hardly be considered a large object? For MySQL, the focus is on the object. You could still store small text files in the database, as opposed to having to copy and paste the data from the text into another field. If you want to use a BLOB in MySQL, use the LONGBLOB option, as it supports a larger file size.

In order to save some memory and processing overhead, MySQL stores the BLOB information in a separate memory area than the normal table-processing memory.

**Framework**

A framework is defined as “[a package made up of a structure of files and folders of standardized code (HTML, CSS, JS documents etc.).](https://www.awwwards.com/what-are-frameworks-22-best-responsive-css-frameworks-for-web-design.html)”

This is a technical way of saying a set of software tools that helps people build websites easier.

(Examples: Bootstrap, Skeleton)

**Hosting**

Hosting refers to a web server where the files for your website are stored.

**Meta Tag**

A meta tag is an important HTML tag for SEO ranking as it stores information about the web page. Search engines use this information to categorize websites and display information in the results.

**Propagation**

Propagation is the amount of time required for changes to your domain name servers to take effect.

In other words, when DNS changes are made, every DNS server in the world needs to update its records of what IP is associated with which domains and this can take time.

**Server Side Scripting**

Server Side Scripting is a technique used in development which involves utilizing scripts on a web server that produce a custom response for each user’s request to the website.

**SSL**

SSL stands for Secure Sockets Layer and is a standard security protocol for establishing encrypted links. This ensures that all data transmitted between the web server and the browser remains secure.

#### . Agile

A is for agile, a major buzzword across the entire tech industry right now. Agile web development essentially refers to a particular way of working, and you’ll often hear this term in the startup world. In an agile team, web developers will work according to weekly or biweekly sprints. A sprint usually consists of five phases: design, develop, test, deploy and review.

#### Algorithm

An algorithm is basically a set of steps for carrying out certain tasks. In computer programming, algorithms are a key part of problem-solving. When creating an algorithm, developers will document all the necessary steps it took to arrive at a solution to a problem, and what each step involved.

#### 3. API

API stands for Application Programming Interface. An API enables two different programs to communicate with each other by making some parts of the website code available to developers. Developers can use this code, i.e. the API, to build tools and widgets that can be connected to that particular website. A classic example is the Facebook API, which enables apps like Facebook Messenger and Words with Friends.

#### Cache

#### A cache is a temporary storage space for data. When you visit a website, the files that you request are automatically stored in the cache. If you return to that same website in the near future, your browser will retrieve the necessary files from your cache rather than from the original server—so the webpage will load quicker.

