**Background and History:**

* Pig was developed at Yahoo! Research and became an Apache open-source project.
* It was invented by Christopher Olston, Benjamin Reed, Utkarsh Srivastava, Ravi Murthy, and Andrew Tomkins.
* **Purpose:** Pig was created to simplify the writing of complex data transformations for Hadoop. It offers a high-level platform to express data analysis programs that can be executed on Hadoop.

**Key Differences Between Pig and Hive:**

1. **Language:** Pig uses the Pig Latin scripting language, while Hive uses a SQL-like query language called HiveQL.
2. **Schema:** Hive is schema-on-read, meaning the schema is applied when data is queried. Pig is schema-on-load, allowing more flexibility.
3. **Ease of Use:** Pig is preferred for ETL (Extract, Transform, Load) tasks and complex data transformations, while Hive is better suited for SQL-like queries.
4. **Custom UDFs:** Pig supports custom User Defined Functions (UDFs) in multiple languages, while Hive primarily uses Java-based UDFs.
5. **Performance:** Hive is often preferred for structured data, while Pig is more suitable for semi-structured or unstructured data.
6. **Data Types:** Hive supports a wider range of data types and is better for handling structured data, while Pig is more flexible in dealing with unstructured data.
7. **Optimization:** Hive provides query optimization through statistics, while Pig relies on the programmer for optimization.

**What is Pig?**

Pig is a high-level platform and scripting language developed by Yahoo! for processing and analyzing large datasets in the Hadoop ecosystem. It provides a simplified way to write complex data transformations, making it easier to work with big data. Pig is a part of the Apache Software Foundation and is available as open-source software.

**Key Features of Pig:**

1. **Abstraction over MapReduce:** Pig abstracts the complexity of writing MapReduce programs, making it easier for data analysts and engineers to work with Hadoop without needing to be experts in Java.
2. **Pig Latin Language:** Pig uses a scripting language called Pig Latin. It's a simple, high-level language designed to express data transformations concisely.
3. **Schema Flexibility:** Unlike some other data processing tools, Pig doesn't require a predefined schema for data. This flexibility is especially valuable when dealing with semi-structured or unstructured data.
4. **Extensible:** Pig allows you to write User Defined Functions (UDFs) in languages like Java, Python, or JavaScript. This means you can extend Pig's functionality to suit your specific needs.
5. **Optimization:** Pig includes optimization techniques that can improve query performance, and it can be integrated with other Hadoop components like Hive and HBase.

**When to Use Pig:**

Pig is an excellent choice when you need to perform the following tasks:

1. **ETL (Extract, Transform, Load):** Pig is often used for data preparation tasks, such as cleaning, filtering, and structuring data for analysis.
2. **Semi-structured Data:** It excels in processing semi-structured or unstructured data, which can be challenging with traditional databases.
3. **Data Transformation:** If you need to perform complex data transformations, Pig simplifies this process, particularly when the logic is too intricate for SQL.
4. **Log Processing:** Pig is suitable for log analysis, where data may not conform to a fixed schema.

**How Pig Works:**

1. **Load:** You start by loading data from various sources like HDFS, local files, or databases.
2. **Transform:** Pig allows you to transform the loaded data using a series of data manipulation operations. This is where the power of Pig Latin scripting comes into play.
3. **Store:** After processing, you can store the results in HDFS, databases, or other storage systems.
4. **Optimize:** Pig performs query optimization behind the scenes, improving the efficiency of your data processing tasks

**1. Field:**

* A field is the most granular data unit in Pig. It's akin to a column in a relational database or a key-value pair in key-value stores. Fields are used to represent attributes in your data.

**2. Tuple:**

* A tuple is an ordered set of fields. It's similar to a row in a relational database. Tuples can be used to represent individual records in your data.

**3. Bag:**

* A bag is a collection of tuples. It's comparable to a table in a relational database but can also be used to represent complex data structures. Bags can be nested, meaning you can have bags within bags.

**4. Relation:**

* A relation is a named dataset in Pig. It's a way to organize your data. You can think of a relation as a table with named attributes.

**5. Filter:**

* Filtering is the process of selecting a subset of data based on certain conditions. Pig provides filtering capabilities to work with your data effectively.

**6. Supported Operators:**

* Pig offers a wide range of operators for data transformation, filtering, and analysis. Some commonly used operators include **LOAD**, **STORE**, **FILTER**, **GROUP**, **JOIN**, and **FOREACH**.

**7. Function Library:**

* Pig provides a library of built-in functions for various data processing tasks. For example:
  + **AVG**: Calculates the average of a set of values.
  + **MAX**: Finds the maximum value in a set of values.
  + **TOKENIZE**: Splits a string into words or tokens.

**8. Relational Operators:**

* Pig supports relational operations, allowing you to work with data in a structured way similar to SQL. You can use **JOIN**, **CROSS**, **UNION**, and other relational operators to combine and manipulate data.

**9. MapReduce:**

* While Pig abstracts much of the complexity of MapReduce, it's essential to understand that Pig scripts are translated into a series of MapReduce jobs that run on your Hadoop cluster. Pig leverages the power of MapReduce for data processing.

**10. String and Math Functions:** - Pig provides a wide range of string and mathematical functions to manipulate and analyze your data. For example, you can use string functions like **CONCAT**, **SUBSTRING**, and mathematical functions like **ABS**, **ROUND**, and **SIN**.

**11. Loading and Storage:** - Pig allows you to load data from various sources like HDFS, local files, databases, and more using **LOAD** statements. You can also store the results in various formats and locations using **STORE** statements.

**12. UDFs (User-Defined Functions):** - UDFs are custom functions that you can write in languages like Java, Python, or JavaScript to extend Pig's functionality. This is particularly useful when you need to perform specialized data transformations or analyses.

The power of Pig is in its simplicity and flexibility, making it an ideal tool for ETL processes, data transformations, and data analysis in the big data world.