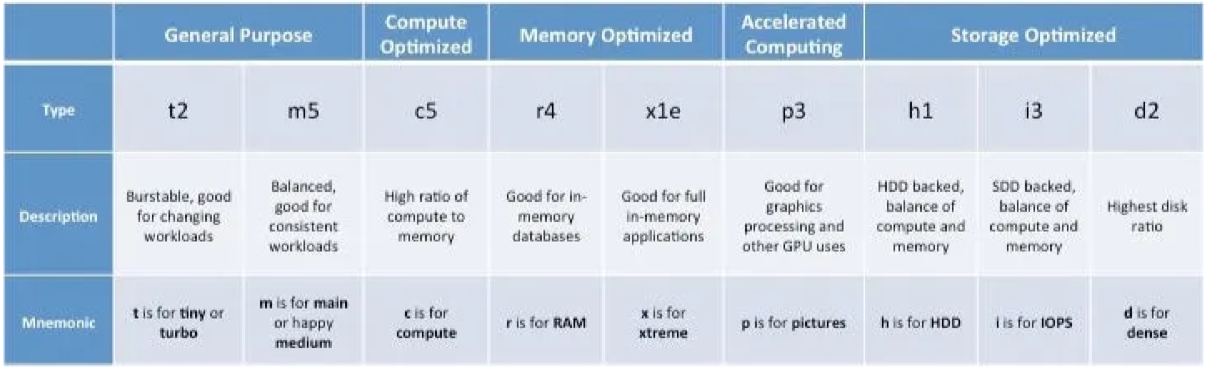
Types of EC2 Instances

Because EC2 Instances are used to power a wide variety of workloads with similarly varying resource needs, Amazon provides us with different types of EC2 instances. A table showcasing all the different EC2 instance types is given below:



	General Purpose		Compute Optimized	Memory Optimized		Accelerated Computing	Storage Optimized		
Type	t2	m5	c5	r4	x1e	p3	h1	i3	d2
Description	Burstable, good for changing workloads	Balanced, good for consistent workloads	High ratio of compute to memory	Good for in-memory databases	Good for full in-memory applications	Good for graphics processing and other GPU uses	HDD backed, balance of compute and memory	SDD backed, balance of compute and memory	Highest disk ratio
Mnemonic	**t** is for **tiny** or **turbo**	**m** is for **main** or happy **medium**	**c** is for **compute**	**r** is for **RAM**	**x** is for **xtreme**	**p** is for **pictures**	**h** is for **HDD**	**i** is for **IOPS**	**d** is for **dense**

It might seem intimidating at first, but the designation names are there to help you, and allow you to make the best decision in regards to the type of EC2 instance that is best suited for your needs.

Understanding the different EC2 Instance Types

Amazon EC2 instances power a wide variety of workloads, each with unique resource needs. To accommodate these differences, Amazon provides several types of EC2 instances, each designed to optimize for a specific workload type. At first glance, the various EC2 instance designations might seem intimidating, but their naming conventions are designed to help you make the right choice for your specific requirements. Here's an overview of the key instance types:

General Purpose Instances

General Purpose instances provide a balanced mix of compute, memory, and networking resources, making them versatile for a variety of workloads. These instances are suitable for applications such as web servers, small databases, and development environments.

Example Types:

- **T2**: Burstable instances ideal for applications with low to moderate baseline performance needs that can benefit from burst capability. The mnemonic "t" stands for tiny or turbo, reflecting the burstable performance nature.
- **M5**: Balanced instances for consistent workloads, offering a stable balance between compute, memory, and network resources. The "m" is for main or medium.

Memory Optimized Instances

These instances are designed to deliver fast performance for workloads that process large data sets in memory. They typically have a higher amount of RAM compared to other instance types, making them suitable for memory-intensive applications like in-memory databases, real-time big data analytics, and high-performance computing.

Example Types:

- **R4**: Designed for applications needing a high memory-to-CPU ratio, such as in-memory databases and real-time analytics. The "r" in R4 stands for RAM.
- **X1e**: Optimized for extremely large memory needs, often used for full in-memory applications like SAP HANA. The "x" represents xtreme memory (slightly childish I know, but it is what it is).

Compute Optimized Instances

These instances are optimized for compute-bound applications that require high-performance processors. They typically have a high ratio of vCPUs to

RAM/Memory, making them suitable for applications that require intensive computational processing such as gaming servers, scientific modeling, batch processing, and media transcoding.

Example Type:

- **C5**: These instances provide a high ratio of compute to memory. The "c" stands for compute, making them ideal for tasks that need powerful processors but do not require a lot of memory.

Storage Optimized Instances

These instances are designed to deliver high storage performance for workloads that require high I/O performance. They come with local instance storage optimized for high-speed, low-latency access, making them suitable for applications that require frequent and fast access to large data sets, such as NoSQL databases, distributed file systems, data warehousing, and data processing.

Example Types:

- **H1**: These instances use HDD-backed storage and offer a balance between compute and storage performance. The "h" stands for HDD.
- **I3**: SSD-backed instances, ideal for applications needing high IOPS, such as transactional databases. The "i" stands for IOPS.
- **D2**: Offers the highest disk throughput with dense storage for applications like data warehousing. The "d" stands for dense, representing the storage capacity.

Accelerated Computing Instances

Accelerated Computing instances are designed for applications that benefit from hardware accelerators (such as GPUs). These instances are perfect for computationally expensive applications, such as machine learning, graphics rendering, and gaming.

Example Type:

- **P3**: Ideal for GPU-heavy workloads like graphics processing or deep learning. The "p" stands for pictures, reflecting its role in image and graphics processing.

Choosing the Right Instance

Now, with so many different EC2 instance types one might think that the process of selecting an EC2 instance might be tedious and confusing, but it is actually quite simple once the workload requirements have been decided upon. Are you running a memory-intensive in-memory database? An **R4** instance might be ideal. Need to process large datasets stored locally? Consider a **D2** or **I3** instance. By understanding the key characteristics of each EC2 instance type, you can optimize performance and cost efficiency for your applications. Not to mention, in cases where workload requirements are yet to be identified one can simply start off with a general purpose instance with the intention of shifting to another instance type in the future.