এখানে মূলত ৩টি স্লাইড আছে। আমি প্রতিটি স্লাইডের বিষয়বস্তু খুব সহজে বুঝিয়ে দিচ্ছি:

১. Parallel Processing Systems (প্যারালাল প্রসেসিং সিস্টেম)

সহজ কথায়, এটি এমন একটি সিস্টেম যেখানে একটি বড় কাজকে ছোট ছোট ভাগে ভাগ করে একসাথে অনেকগুলো প্রসেসর বা কম্পিউটিং রিসোর্স দিয়ে সমাধান করা হয়।

এর মূল বৈশিষ্ট্যগুলো হলো:

- **দ্রুত কাজ:** অনেকগুলো CPU এবং ডিস্ক একসাথে ব্যবহার করার ফলে প্রসেসিং অনেক ফাস্ট হয়।
- **একসাথে কাজ (Concurrency):** সব প্রসেসর একই সময়ে তাদের নিজ নিজ কাজ করতে পারে।
- **রিসোর্স শেয়ারিং:** প্রসেসরগুলো নিজেদের মধ্যে ডেটা, ডিস্ক এবং অন্যান্য ডিভাইস শেয়ার করে কাজ করে।
- **পারফরম্যান্স মাপার উপায়:** এই সিস্টেমটি কতটা ভালো কাজ করছে, তা মাপার জন্য দুটি জিনিস ব্যবহার করা হয়: **Speedup (স্পিডআপ)** এবং **Scaleup (স্কেলআপ)**।

২. Speedup (স্পিডআপ)

স্পিডআপ কী?

ধরে নিন, আপনার কাছে একটি নির্দিষ্ট কাজ আছে। এখন আপনি যদি কম্পিউটারের প্রসেসর বা পাওয়ার বাড়িয়ে দেন, তবে ওই একই কাজ আগের চেয়ে কত দ্রুত শেষ হচ্ছে, সেটাই হলো স্পিডআপ। অর্থাৎ, এখানে **কাজ একই থাকবে, শুধু মেশিনের শক্তি বাড়বে**।

- **সূত্র:** Speedup = TS/TL
  - $TS = ছোট মেশিনে কাজটি করতে যে সময় লাগে।
  - TL$ = বড় মেশিনে (প্যারালাল সিস্টেমে) একই কাজটি করতে যে সময় লাগে।
- **Linear Speedup (লিনিয়ার স্পিডআপ):** আপনি যদি মেশিনের শক্তি N গুণ বাড়ান এবং কাজটিও যদি ঠিক N গুণ দ্রুত শেষ হয়, তবে তাকে লিনিয়ার স্পিডআপ বলে। গ্রাফে ওপরের সোজা রেখাটি এটি নির্দেশ করে। এটি সবচেয়ে ভালো অবস্থা।
- **Sublinear Speedup (সাব-লিনিয়ার স্পিডআপ):** যদি মেশিনের শক্তি N গুণ বাড়ানোর পরও কাজটির স্পিড N গুণের চেয়ে কম বাড়ে, তবে তাকে সাব-লিনিয়ার স্পিডআপ বলে। গ্রাফের নিচের বাঁকা রেখাটি এর উদাহরণ।

৩. Scaleup (স্কেলআপ)

স্কেলআপ কী?

এখানে শুধু মেশিনের শক্তি বাড়ে না, সাথে সাথে কাজের পরিমাণও বাড়ে। অর্থাৎ, কাজের পরিমাণ বাড়ার সাথে সাথে আপনি যদি মেশিনের প্রসেসর বা পাওয়ারও বাড়িয়ে দেন, তবে সিস্টেমটি কেমন পারফর্ম করছে, সেটাই হলো স্কেলআপ। এখানে লক্ষ্য থাকে, কাজ বড় হলেও যেন সময় আগের মতোই লাগে।

- **সূত্র বা হিসাব:** ধরি ছোট মেশিন MS-এ একটি কাজ করতে $TS সময় লাগে। এখন কাজটি যদি N গুণ বড় হয় এবং সেটি করার জন্য আমরা একটি বড় প্যারালাল মেশিন M_L ব্যবহার করি, তবে তার সময় লাগে TL।
- **Linear Scaleup (লিনিয়ার স্কেলআপ):** যদি বড় মেশিনে বড় কাজটি করতে ঠিক আগের ছোট কাজের সমানই সময় লাগে (অর্থাৎ, TL = TS), তবে তাকে লিনিয়ার স্কেলআপ বলে। এর মানে হলো, কাজ যত বড়ই হোক না কেন, আপনি মেশিন বড় করে সময়টা ঠিক রেখেছেন। গ্রাফে ওপরের সমান্তরাল সোজা রেখাটি এটি নির্দেশ করে।
- **Sublinear Scaleup (সাব-লিনিয়ার স্কেলআপ):** যদি বড় মেশিনে বড় কাজটি করতে আগের চেয়ে বেশি সময় লেগে যায় (অর্থাৎ, TL > TS), তবে তাকে সাব-লিনিয়ার স্কেলআপ বলে। গ্রাফের নিচের দিকে নেমে যাওয়া বাঁকা রেখাটি এটি নির্দেশ করে।

সারসংক্ষেপ:

- **Speedup:** একই কাজ বড় মেশিনে কত *তাড়াতাড়ি* হচ্ছে, তা মাপে।
- **Scaleup:** বড় কাজ বড় মেশিনে করলে আগের *সমান সময়ে* হচ্ছে কিনা, তা মাপে।

১. Distributed Database Design (ডিস্ট্রিবিউটেড ডেটাবেস ডিজাইন)

💡 **কনসেপ্ট (বাংলায়):** ডেটাবেসের ডেটাগুলো বিভিন্ন সার্ভারে বা সাইটে কীভাবে রাখা হবে, সেটা নির্ধারণ করাই হলো ডিস্ট্রিবিউটেড ডেটাবেস ডিজাইন। এটা মূলত দুইভাবে করা যায়: ডেটাগুলো ভেঙে আলাদা আলাদা সাইটে রাখা (Partitioned) অথবা একই ডেটার একাধিক কপি বিভিন্ন সাইটে রাখা (Replicated)।

📝 **Exam-Ready Answer (English):** In a distributed database system, placing data involves two basic alternatives:

- **Partitioned (or non-replicated) Scheme:** The database is divided into a number of disjoint partitions, and each partition is placed at a different site.
- **Replicated Scheme:** This can be categorized into two types:
  - **Fully Replicated (Fully Duplicated):** The entire database is stored at each site.
  - **Partially Replicated (Partially Duplicated):** Each partition of the database is stored at more than one site, but not at all the sites.

২. Distributed Directory Management (ডিস্ট্রিবিউটেড ডিরেক্টরি ম্যানেজমেন্ট)

💡 **কনসেপ্ট (বাংলায়):** ডিরেক্টরি হলো একটা সূচিপত্র বা ম্যাপের মতো, যেখানে বলা থাকে ডেটাবেসের কোন ডেটা কোথায় আছে। এই ডিরেক্টরিটা পুরো সিস্টেমের জন্য একটাই হতে

পারে, আবার প্রতিটি সাইটের নিজস্ব ডিরেক্টরিও থাকতে পারে। এটি এক জায়গায় রাখা যায় বা সব সাইটে ছড়িয়ে দেওয়া যায়।

📝 **Exam-Ready Answer (English):** Directory management is essential for locating data within a distributed system.

- **Purpose:** A directory contains information (such as descriptions and locations) about data items in the database.
- **Scope:** A directory may be global to the entire Distributed Database System (DDBS) or local to each individual site.
- **Placement Strategy:** It can be centralized at one specific site or distributed over several sites. Furthermore, there can be a single copy or multiple copies of the directory.

৩. Distributed Query Processing (ডিস্ট্রিবিউটেড কুয়েরি প্রসেসিং)

💡 **কনসেপ্ট (বাংলায়):** ব্যবহারকারী যখন কোনো তথ্য খোঁজে (Query), তখন সেই রিকোয়েস্টটাকে কীভাবে সবচেয়ে কম খরচে এবং দ্রুত বিভিন্ন সাইট থেকে ডেটা এনে প্রসেস করা যায়, তার কৌশল বের করাই হলো কুয়েরি প্রসেসিং।

📝 **Exam-Ready Answer (English):** Query processing is critical for retrieving data efficiently across multiple sites.

- **Core Function:** It deals with designing algorithms that analyze queries and convert them into a series of data manipulation operations.
- **Primary Challenge:** The main problem is deciding on a strategy for executing each query over the network in the most cost-effective way.
- **Factors Considered:** To achieve cost-effectiveness, factors such as the distribution of data, communication costs, and the lack of sufficient locally-available information must be considered.

৪. Distributed Concurrency Control (ডিস্ট্রিবিউটেড কনকারেন্সি কন্ট্রোল)

💡 **কনসেপ্ট (বাংলায়):** যখন একাধিক ইউজার একই সময়ে ডেটাবেস ব্যবহার করে, তখন ডেটার যেন কোনো গড়মিল না হয় এবং ডেটার বিশ্বাসযোগ্যতা বা ইন্টিগ্রিটি ঠিক থাকে, তা নিশ্চিত করার জন্য এক্সেসগুলোকে সিঙ্ক্রোনাইজ করাই হলো কনকারেন্সি কন্ট্রোল।

📝 **Exam-Ready Answer (English):**

- **Definition:** Concurrency control involves the synchronization of accesses to the distributed database.
- **Goal:** The primary objective of this synchronization is to ensure that the overall integrity of the database is maintained when multiple users access data simultaneously.

৫. Distributed Deadlock Management (ডিস্ট্রিবিউটেড ডেডলক ম্যানেজমেন্ট)

💡 **কনসেপ্ট (বাংলায়):** যখন একাধিক ইউজার একই ডেটা বা রিসোর্স লক করে ব্যবহার করতে চায়, তখন এমন অবস্থার সৃষ্টি হতে পারে যেখানে কেউ কাজ শেষ করতে পারছে না (সবাই একে অপরের জন্য আটকে আছে, যাকে ডেডলক বলে)। এটি সমাধান করার উপায়গুলোই এখানে আলোচনা করা হয়।

📝 **Exam-Ready Answer (English):** Deadlock management deals with resolving system standstills caused by resource locks.

- **Cause of Deadlocks:** The competition among users for access to a set of resources (such as data) can result in a deadlock if the synchronization mechanism is based on locking.
- **Resolution Strategies:** The well-known alternatives for handling these situations—prevention, avoidance, and detection/recovery—also apply to Distributed Database Systems (DDBSs).