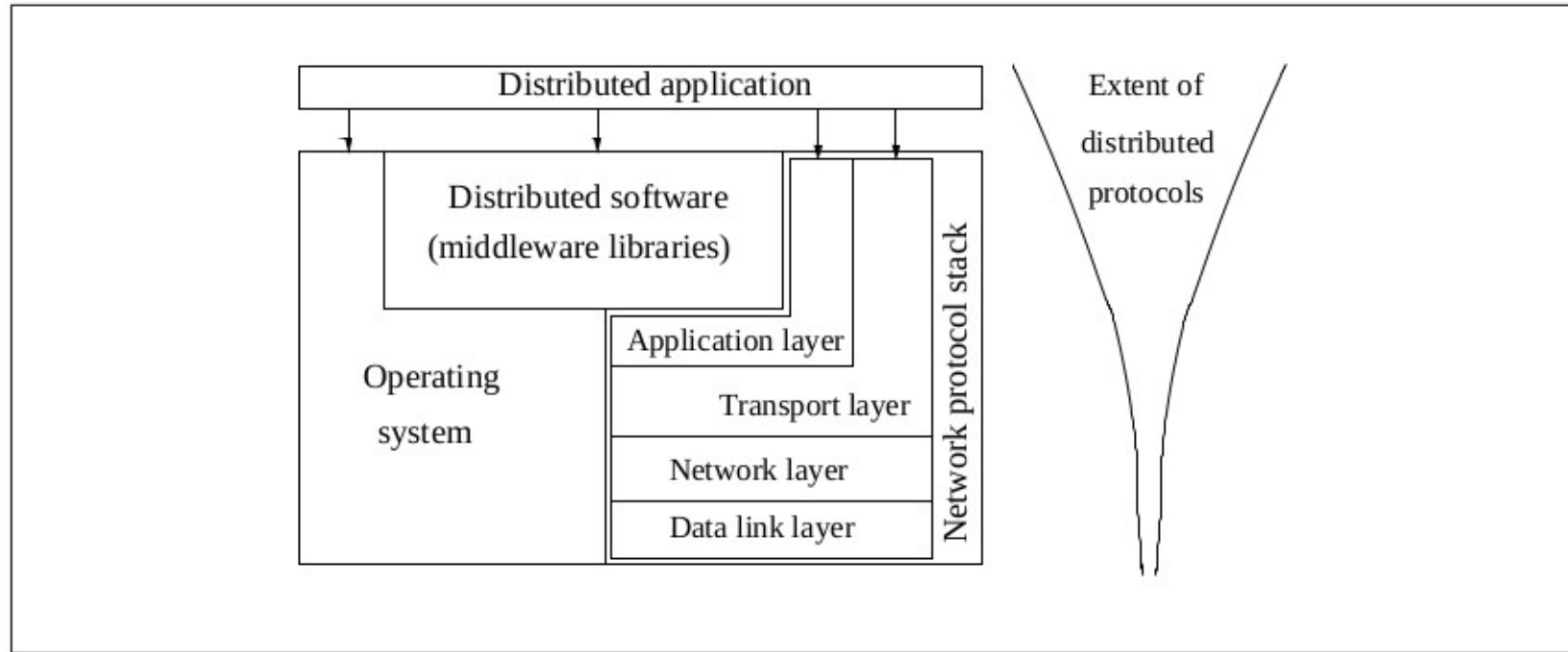


Distributed Computing

Mahesh C

Relation to Computer System Components



Motivation

- Inherently distributed computations (bank)
- Resource sharing
- Access to geographically remote data and resources.
- Enhanced reliability
- Increased Performance/Cost ratio
- Scalability
- Modularity and incremental expandability

Design Issues and Challenges - System Perspective

- Communication
- Processes
- Naming.
- Synchronization.
- Applications Programming Interface (API) and transparency
- Data storage and access
- Consistency and replication
- Fault tolerance
- Security

Design Issues and Challenges - Algorithmic Perspective

- Designing useful execution models and frameworks
- Dynamic distributed graph algorithms and distributed routing algorithms
- Time and global state in a distributed system.
- Synchronization.
- Synchronization/ coordination mechanisms.
- Reliable and fault-tolerant distributed systems
- Real-time scheduling.
- Performance
- Group communication, multicast, and ordered message delivery.
- Monitoring distributed events and predicates.
- Distributed program design and verification tools.
- Data replication, consistency models, and caching.
- World Wide Web design – caching, searching, scheduling.
- Distributed shared memory abstraction.
- Load balancing.

Applications of Distributed Computing

- Mobile systems
- Ubiquitous or pervasive computing
- Publish-subscribe, content distribution, and multimedia.
- Security in distributed systems.
- Sensor networks
- Peer-to-peer computing
- Distributed agents
- Distributed data mining
- Grid computing