**Best Practices for Transactional Replication in SQL Server (Production)**

1. **Replication Topology**:
   * **Single Publisher, Multiple Subscribers**: Ideal for scaling out read-heavy workloads.
   * **Separate Distributor**: Use a dedicated server for the distribution database to offload the Publisher and Subscriber.
2. **Publisher and Subscriber Configuration**:
   * **Keep Publisher Server Optimized**: Ensure the Publisher is tuned for high transaction throughput (e.g., optimize disk I/O and CPU).
   * **Subscriber Maintenance**: Regularly update statistics and rebuild indexes on Subscriber databases to ensure query performance.
3. **Network Optimization**:
   * **Dedicated Replication Network**: Isolate replication traffic to reduce congestion and latency.
   * **Compression**: Enable replication compression to minimize network bandwidth usage.
4. **Security**:
   * **Use SSL/TLS Encryption**: Encrypt communication between Publisher, Subscriber, and Distributor.
   * **Least Privilege Principle**: Grant the minimum necessary permissions for the replication agents and database access.
5. **Transactional Consistency**:
   * **Full Recovery Model**: Ensure the Publisher database is in full recovery mode to maintain log-based replication.
   * **Log Shipping or Mirroring (if required)**: Use log shipping or database mirroring in conjunction with transactional replication for disaster recovery.
6. **Replication Agent Management**:
   * **Log Reader Agent**: Monitor the Log Reader Agent for processing latency and failures. Ensure it runs continuously.
   * **Distribution Agent**: Ensure Distribution Agents are running smoothly; configure them for higher priority if needed.
   * **Agent Scheduling**: Set appropriate intervals for the Distribution Agent based on your transactional volume and business needs.
7. **Latency Management**:
   * **Low Latency Configuration**: Set up agent schedules to minimize latency. Ensure that the Log Reader Agent picks up transactions quickly.
   * **Monitor Replication Latency**: Regularly monitor the replication delay between Publisher and Subscriber and adjust as needed to maintain acceptable latency.
8. **Performance Optimization**:
   * **Indexes**: Ensure appropriate indexing on the replicated tables to optimize transaction replication and query performance.
   * **Batch Size Adjustment**: Control the -B parameter (batch size) in the Distribution Agent to optimize network and transaction log usage.
   * **Avoid Replicating Large Objects**: Avoid replicating large objects (LOBs like images, blobs) unless necessary as they can degrade performance.
9. **Conflict Management**:
   * **Avoid Write Conflicts**: In transactional replication, write conflicts should be avoided as they can lead to data inconsistencies.
   * **Conflict Resolution**: For read-only Subscribers, configure conflict resolution strategies and ensure only one Publisher is responsible for updates.
10. **Monitoring and Alerts**:
    * **SQL Server Replication Monitor**: Use Replication Monitor to track the health of all replication agents and replication latency.
    * **Custom Alerts**: Set up alerts for key performance metrics such as replication agent failures, distribution database growth, and latency thresholds.
11. **Backup and Disaster Recovery**:
    * **Back Up Distribution Database**: Regularly back up the distribution database and replication configuration.
    * **Publisher Database Backups**: Ensure that full backups and transaction log backups are done regularly for recovery and point-in-time restores.
12. **Maintenance**:
    * **Cleanup Jobs**: Set up cleanup jobs to remove old replication history from the distribution database to prevent it from growing uncontrollably.
    * **Reinitialize Subscription**: If needed, reinitialize subscriptions when data inconsistency or replication issues arise.