


Connection Pooling

Herve Roggero
<http://www.herveroggero.com>
hroggero@bluesyntax.net



pluralsight 
hardcore developer training

Topics Covered

- **ADO.NET connection pooling overview**
 - Objectives of connection pooling and limitations
- **Common mistakes with connection pooling**
 - Review common mistakes leading to connection pooling issues
- **Programmatic control over connection pooling**
 - Options to reset connection pools
-  **Analyze behavior of connection pooling in .NET**
 - Demo on how an application can affect connection pooling

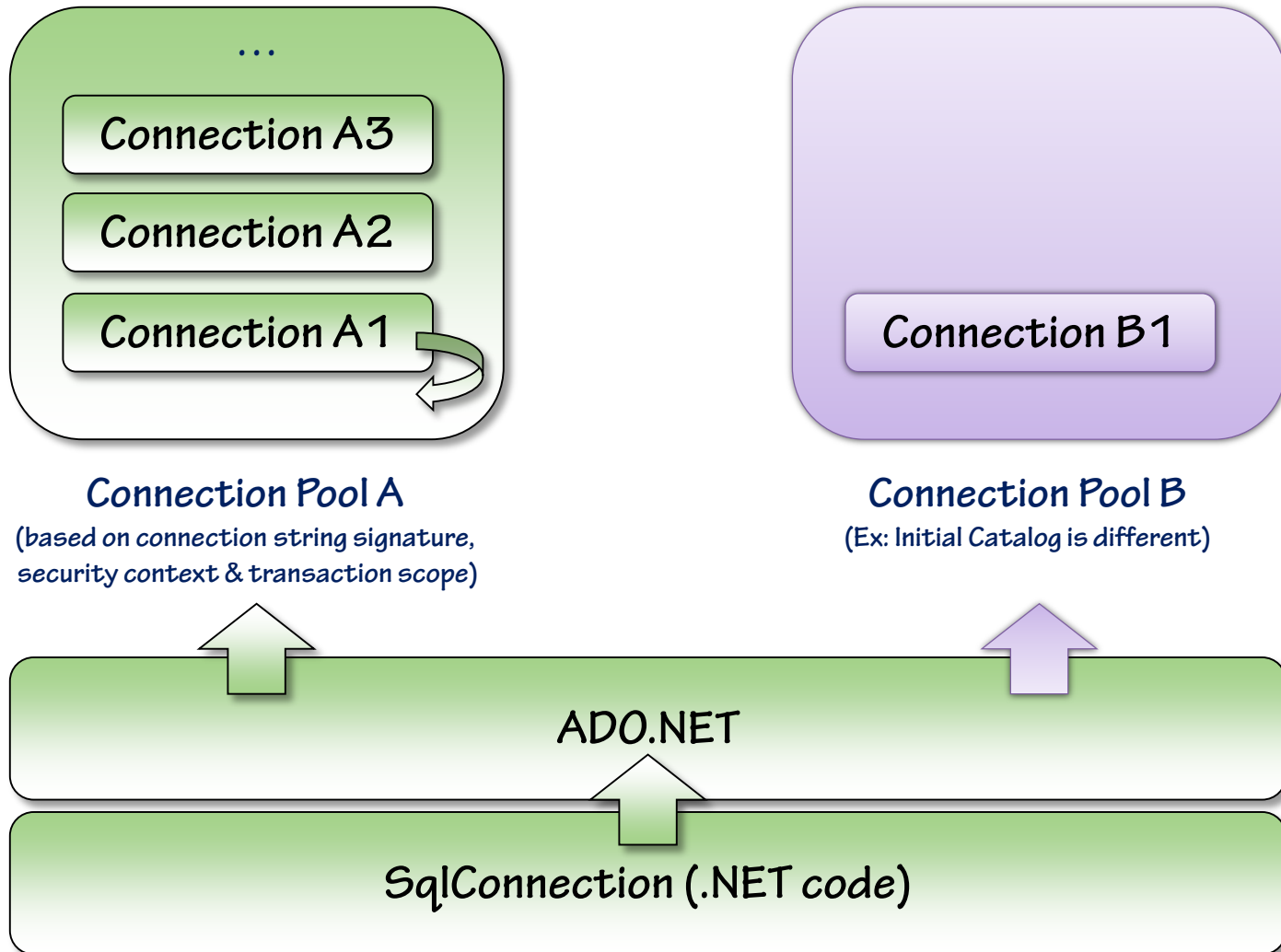
Overview

- **Reasons for connection pooling**
 - Improves application performance
 - Improves database scalability
- **What controls connection pooling**
 - Connection String
 - Currently logged on user for connections with Integration Security
 - Transaction state
- ➡ ■ **Multiple connections are needed when...**
 - Application uses multi-threading to connect to a database
 - Different connection strings are used over time
 - Different security contexts/users are used (WinForm & ASP.NET)



(*) *Connection strings must match exactly for connection pooling to work properly*

Connections and Connection Pools



Common Mistakes

- **Use a connection string as a means to pass a variable to SQL Server**
 - Use the Application Name to send a user key, or license key
- **Impersonate users that have a corresponding database login**
 - Change the UID/PWD fields in the connection string for each user
- **Use network impersonation in IIS or WinForm**
 - Change the security context of a user on the running thread and use Network Authentication
- **Overly distribute data across a large number of databases**
 - The Initial Catalog property of the connection string changes too often



Objectives are to release connections quickly & minimize the number of connection pools

Viewing Connection Pools

- **SQL Trace**

- Audit:Login and RPC_Completed (sp_reset_connection)

- **Performance Counters (using perfmon.exe)**

- .NET Data Provider for Sql Server
 - .NET Data Provider for Oracle



- **Important Counters**

- HardConnectsPerSecond (physical database connections)
 - NumberOfActiveConnectionPools (*) (number of connection pools)
 - NumberOfFreeConnections (*) (available connections in the pools)
 - SoftConnectsPerSecond (*) (connections leveraging connection pooling)



(*) *These counters are off by default; they must be activated in the application config file*

Summary

- Overview of Connection Pooling
- Using SQL Profiler and PerfMon to monitor connection pools
- Programmatic control of pooling
- Common mistakes with connection pooling

