Synchronize Smart Client Data and Offline Data

Brian Noyes
IDesign, Inc. (www.idesign.net)
brian.noyes@idesign.net
About Brian

• Principal Software Architect, IDesign, Inc. ([www.idesign.net](http://www.idesign.net))
• Microsoft MVP in ASP.NET
• Writing
  - MSDN Magazine, asp.netPRO, Visual Studio Magazine, .NET Developer’s Journal
  - Building Windows Forms Data Applications with .NET 2.0, Addison-Wesley, expected release spring 2005
• Speaking
  - Microsoft TechEd, Visual Studio Connections, DevEssentials, VSLive!, INETA Speakers Bureau
• Participates in Microsoft design reviews
• E-mail: brian.noyes@idesign.net
Agenda

• What is a Smart Client?
• Disconnected operations challenges
• Data communications approaches
• Client side data caching
• Connection management
• Data synchronization
• Offline Application Block
What is a Smart Client?

- Rich user interface (WinForms)
- Connects to back-end services
- Runs securely on the client
- Supports auto-deployment and update over the network
- Supports disconnected operations
Disconnected Operations Challenges

- Offline use case identification
- Online communications transport
- Connection management
- Client-side caching approach
- Offline data synchronization
- Security
Data Communication Approaches

- .NET Remoting
- Enterprise Services (COM+)
- Database
- MSMQ
- Web Services
Data Transfer Approaches

• Passing a Data Transfer Object via Method Call
  ● .NET Remoting, Enterprise Services, Web Services
  ● DataSet, custom business object (collection)

• Service-Oriented
  ● Enterprise Services, Web Services, MSMQ

• Data replication
  ● MSDE/SQL Express -> SQL Server
Connection Management

• Need to detect and control online vs. offline operations

• Detection techniques
  ● Try connected operation – handle failure
  ● Ping/connect attempt first
  ● WinInet API
  ● Offline Block Connection Management
  ● NetworkChanged class (.NET 2.0)
Client Side Data Caching

- Memory
- Saved Data Transfer Object
- Database
- Message Queues
  - MSMQ
  - Database
  - Enterprise Services
Data Synchronization

- Data oriented
  - Merge replication

- Service oriented
  - Remote method invocation
  - Message delivery
    - Confirmation return message
    - Poll for completion
Demos

Simple Offline Data Caching
Enterprise Services Queued Components
Offline Application Block

- Connection State Detection / Control
- Download / Upload Data
- Queued data requests
- Reference Data Caching
- Asynchronous request processing
- Encryption / signing of stored data
- Provider model for connection detection, data request queuing, data caching, service agents
Offline Application Block

- Application
- ConnectionManager
- Connection Detection Strategy
- ServiceAgent Manager
- ServiceAgent
- Application Service Agent
- DataLoaderManager
- QueueManager
- Queue Storage
- Executor
- Online Proxy
- ReferenceDataCache
- Cache Block

Legend:
- Customer does it
- We do it
- We do one or two
Demo

Offline Application Block Client
Summary

• Design for disconnected operations early
  ● Identify offline use cases
  ● Pick data communications technology
  ● Select caching and synchronization mechanism

• Prefer decoupled, service oriented approaches for enterprise applications

• Explore the Offline Application Block for maximum flexibility
Resources

- Smart Client Architecture and Design Guide
- SQL Express
  [link](http://lab.msdn.microsoft.com/express/sql/default.aspx)
- Offline Application Block
  Managing Offline Clients with the Offline Application Block, Klaus Aschenbrenner,
  DevX, [link](http://www.devx.com/dotnet/Article/21420/0/page/1)
- Enterprise Services / Queued Components
- Web Services
  .NET Web Services, Keith Ballinger, Addison-Wesley, 2003

- E-mail: [brian.noyes@idesign.net](mailto:brian.noyes@idesign.net)
- Blog: [link](http://www.softinsight.com/bnoyes)