OUTLINE
DELPHI 2010: MASTERING CLIENTDATASETS

A ClientDataSet provides you with an in-memory data store that is both high-performance and rich in features. The widespread availability of ClientDataSet has important implications for most applications, especially database applications. In database applications, the ClientDataSet can be used to provide a layer of abstraction between your code and the underlying data access mechanism. Used in this way, the ClientDataSet provides your applications with a consistent set of features for both your data-aware components and your programmatic manipulation of data. It also provides a lightweight alternative to database servers and local database engines for single-user applications. For non-database applications, the ClientDataSet provides a flexible alternative to TStringLists, linked lists, INI files, and other temporary and persistent local storage mechanisms.

This seminar provides you with the information that you need to be productive with ClientDataSet today. It begins with an overview of the ClientDataSet, including how to configure it and how it works. It then continues with an in-depth and detailed look at using the ClientDataSet features including aggregates, cloning ClientDataSet cursors, caching lookup tables, nested datasets, UpdateMode and ProviderFlags, and cached updates. We conclude with Delphi's DataSnap, the multitier distributed application framework, and cover creating DataSnap servers and clients, DataSnap servers as DLLs, remote data module pooling, and DataSnap connection components.

CARY JENSEN, PH.D.
COPYRIGHT 2004-2010. CARY JENSEN. JENSEN DATA SYSTEMS, INC.
ALL RIGHTS RESERVED.
JENSEN DATA SYSTEMS, INC.
HTTP://WWW.JENSENDATASYSTEMS.COM
PHONE: 281-359-3311
EMAIL: INFO@JENSENDATASYSTEMS.COM

COURSE INCLUDES A COURSE BOOK PLUS ALL CODE EXAMPLES PRESENTED ALONG WITH ADDITIONAL CODE EXAMPLES.

MASTERING CLIENTDATASETS

OVERVIEW OF THE CLIENTDATASET

- How Data Is Saved
- Saving Local Data
- Saving Data with a DataSetProvider
- Loading Data with a DataSetProvider
- ApplyUpdates and MaxErrors

DEFINING THE STRUCTURE OF A CLIENTDATASET

DEFINING A TABLE'S STRUCTURE USING FIELDDEFS

- Creating FieldDefs at Design Time
- Creating FieldDefs at Runtime
SAVING DATA

DEFINING A CLIENTDATASET'S STRUCTURE USING TFIELDS

Creating TFFields at Design Time
Adding a Calculated Virtual Field
Adding a Virtual Aggregate Field
Creating Nested DataSets

CREATING A CLIENTDATASET'S STRUCTURE AT RUNTIME USING TFIELDS

TFFields and FieldDefs Are Different
An Example with a Complex Structure Defined Using TFFields

UNDERSTANDING CLIENTDATASET INDEXES

DEFAULT INDEXES
CREATING INDEXES
Temporary Indexes
Persistent Indexes

USING PERSISTENT INDEXES
Creating Persistent Indexes at Runtime

DEPLOYING WITH NO DLLS

MANIPULATING THE CHANGE LOG

NAVIGATING CLIENTDATASETS

NAVIGATING WITH DATA-AWARE CONTROLS
DETECTING CHANGES TO RECORD STATE
NAVIGATING PROGRAMMATICALLY
SCANNING A CLIENTDATASET
EDITING A CLIENTDATASET
DISABLING CONTROLS WHILE NAVIGATING
NAVIGATION DEMONSTRATION

SEARCHING A CLIENTDATASET

SCANNING FOR DATA
FINDING DATA
GOING TO DATA
LOCATING DATA
USING LOOKUP

USING RANGES

SETTING A RANGE
USING APPLYRANGE
CANCELLING A RANGE
A COMMENT ABOUT RANGES
USING FILTERS

USING THE ONFILTERRECORD EVENT HANDLER
NAVIGATING USING A FILTER
USING RANGES AND FILTERS TOGETHER

AGGREGATES AND GROUPSTATE

UNDERSTANDING AGGREGATES
CREATING AGGREGATE FIELDS
  Adding the Aggregate Field
  Defining the Aggregate Expression
  Setting Aggregate Index and Grouping Level
  Making the Aggregate Field Available
CREATING AGGREGATE COLLECTION ITEMS
UNDERSTANDING GROUP STATE

CLONING CLIENTDATASET CURSORS

HOW TO CLONE A CLIENTDATASET
  Cloning a Filtered ClientDataSet: A Special Case
  The Shared Data Store
CLONING EXAMPLES
  Creating Multiple Views of a Data Store
  Self-referencing Master-Details
  Deleting a Range of Records

CACHING LOOKUP TABLES

USING NESTED DATASETS

UNDERSTANDING UPDATEMODE AND PROVIDERFLAGS

CLIENTDATASETS AND CACHED UPDATES
  A More Complex Example
  Controlling How Updates Are Applied
  Resolving to a DataSet
  Handling Errors
  Applying Updates when Using NestedDataSets

UNDERSTANDING DATASNAP

DATASNAP OVERVIEW
CREATING A DATASNAP SERVER
CREATING A DATASNAP CLIENT
INSTALLING A DATASNAP SERVER
CREATING DATASNAP SERVERS AS DLLS
  Creating a DataSnap DLL Project
Registering a DLL-based DataSnap Server

MANAGING DATA TRANSFER BETWEEN CLIENT AND SERVER

PacketRecords and FetchOnDemand Properties

Controlling Which Records Are Downloaded

REMOTE DATA MODULE POOLING

USING DELPHI'S DATASNAP CONNECTION COMPONENTS

Using a Shared Connection