accelerate

palm

collaborate

innovate

communicate

PalmSource

Conference 2000

COM Sync Suite

Giovanni Marais
Software Engineer Palm Inc.

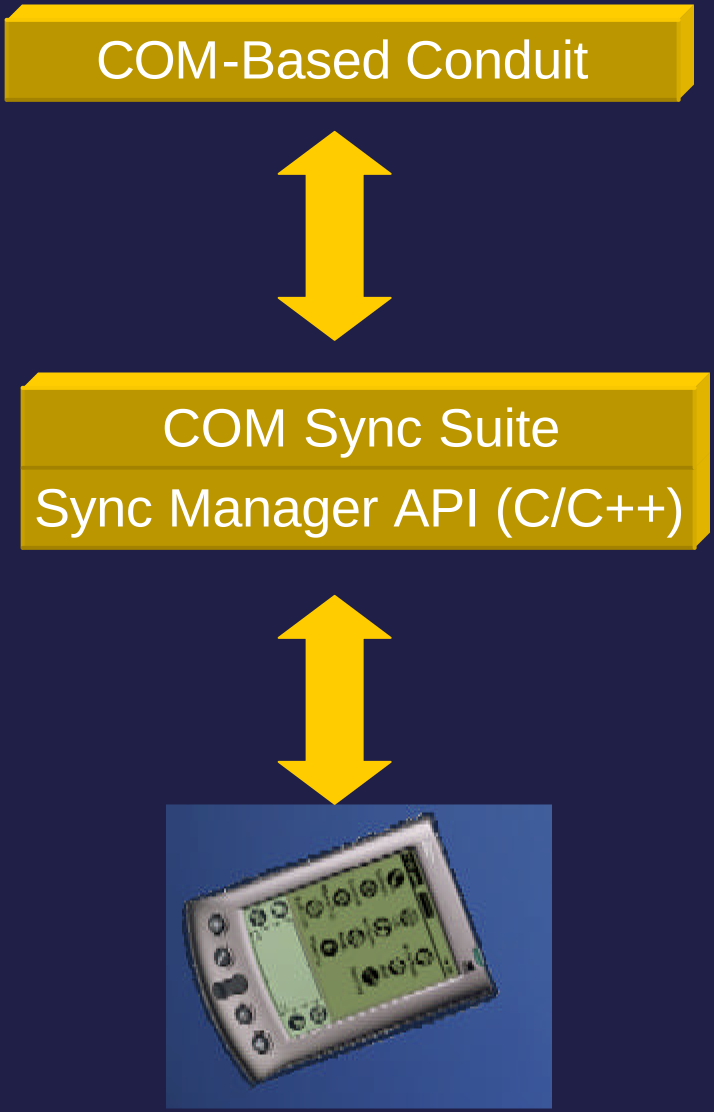
Agenda

- COM Sync Suite Overview
- COM Conduit Development Process
- Generic Conduit using Visual Basic
- Questions and Answers

Fundamentals

- COM Sync Suite is a Software Toolkit
  - Microsoft COM Based Conduit authoring tool
  - Any COM compliant language can use it
- Extends the Sync Manager API Functionality
  - Uses COM as a “wrapper” for Sync Manager API
  - Supports Sync Manager API Version 3.0
- Packaged as a set of .Dlls
  - Client : (ComStandard.dll, ComDirect.dll)
    - Exposes the COM Interface
    - Communicates to the server via Object Model
  - Server : ComConduit.dll
    - Acts as a HotSync Conduit

COM-Based Conduit



COM-Based Conduit

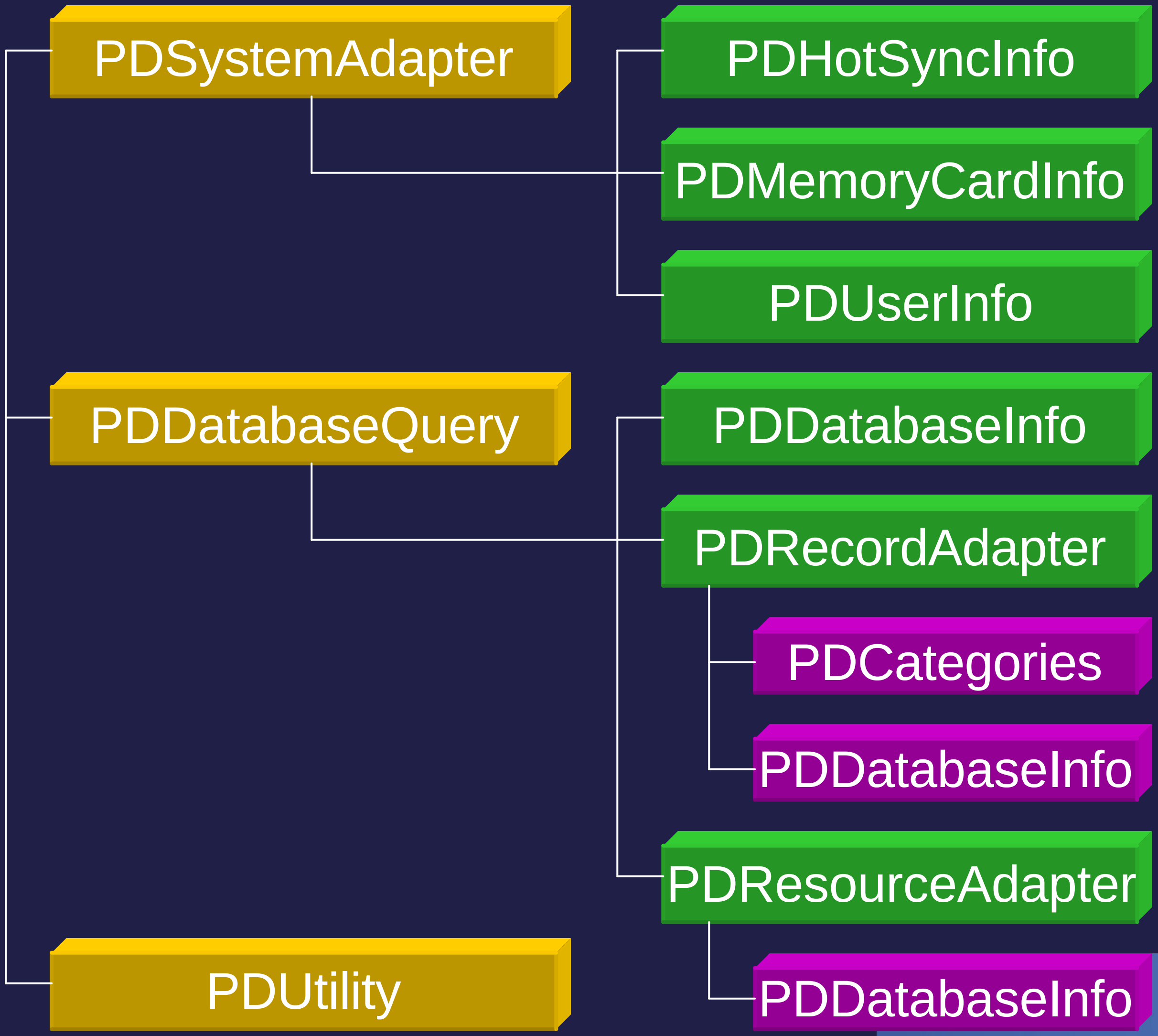
COM Sync Suite

Sync Manager API (C/C++)

Features

- Added Functionality
  - Categories Object
  - Utilities Object
- Multi-database Support
- Multi-threading Support
- Debug Conduits in real-time
- No HotSync Manager keep-alive needed
- DCOM Support

COM Sync Suite Objects



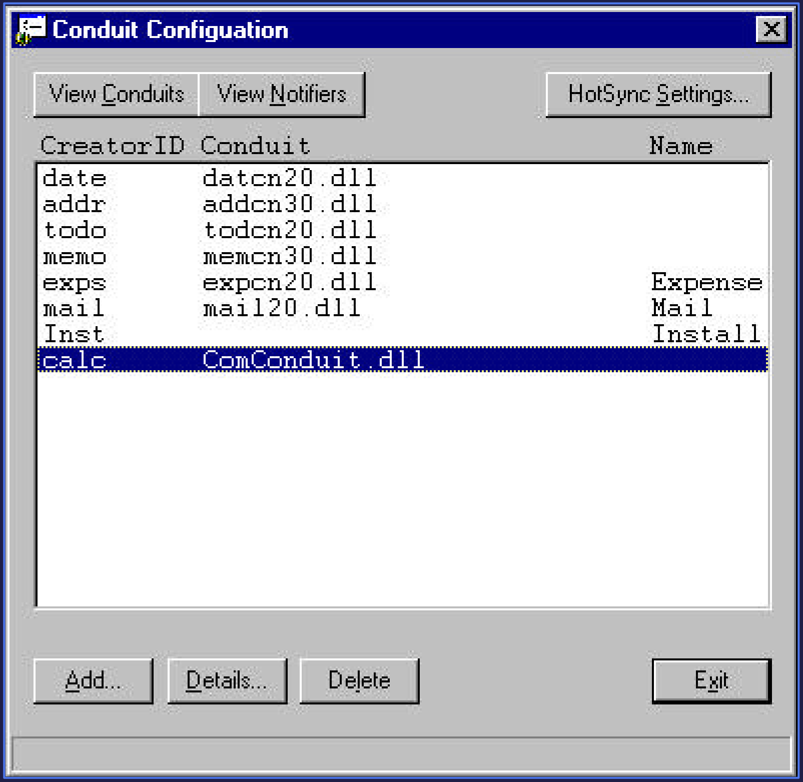
Notification Interface

- COM Sync Suite defines IPDClientNotify
  - Used by ActiveX clients only
- Purpose
  - Activate the Client during HotSync
  - Activate a configuration routine from the Custom Menu
- Usage
  - BeginProcess
  - CfgConduit
  - GetConduitInfo

COM Conduit Development Process

- Install COM Sync Suite
- Register Visual Basic IDE
  - Use CondCfg.exe
  - Add COM Sync Suite as a reference
- Create Standard Executable
- Debug Conduit
- Add IPDClientNotify
  - Set to Multiuse
- Convert Conduit to ActiveX DLL or EXE

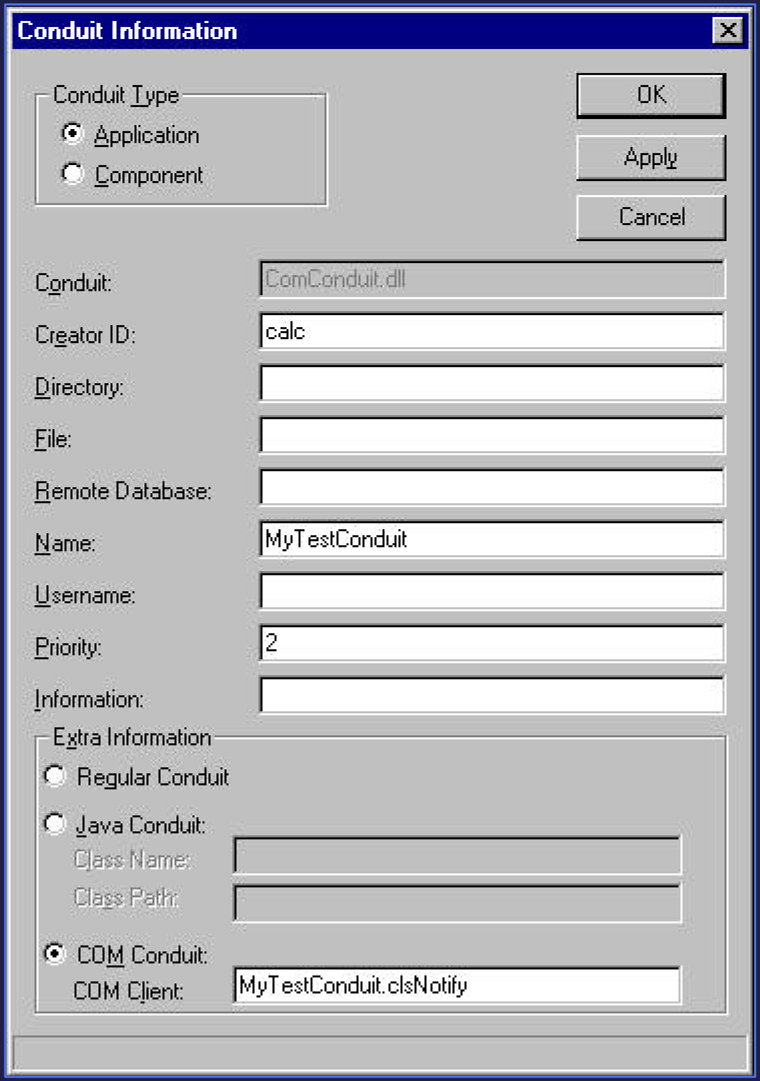
Conduit Configuration



CreatorID	Conduit	Name
date	datcn20.dll	
addr	addcn30.dll	
todo	todcn20.dll	
memo	memcn30.dll	
exps	expcn20.dll	Expense
mail	mail20.dll	Mail
Inst		Install
calc	ComConduit.dll	

- Part of the Conduit Development Kit
- Displays a list of Conduits
- Add/Edit Conduit detail

Adding a Conduit



Conduit Information

Conduit Type
● Application
○ Component

Conduit: ComConduit.dll
Creator ID: calc
Directory:
File:
Remote Database:
Name: MyTestConduit
Username:
Priority: 2
Information:

Extra Information
○ Regular Conduit
○ Java Conduit:
Class Name:
Class Path:
● COM Conduit:
COM Client: MyTestConduit.clsNotify

- Select COM Conduit option
- Add Class ID or path
- Name
- Creator ID

Code Sample : Open a Record Database

```
Dim PQuery As New PDDatabaseQuery
Dim PRecAdapt As PDRecordAdapter

Set PRecAdapt =
PQuery.OpenRecordDatabase("MyConduitDB",
"PDDirect.PDRecordAdapter")
```

Code Sample : Read a Record

```
Dim Index As Long
Dim UniqueId As Variant
Dim Category As Long
Dim Attributes As Long
Dim RecordData As Variant

PRecAdapt.IterationIndex = 0
RecordData = PRecAdapt.ReadNext(Index,
UniqueId, Category, Attributes)

' Convert and Display the record data.
MsgBox StrConv(RecordData, vbUnicode)
```

Where Do I Get It?

- COM Sync Suite is available for download
- http://www.palmos.com/dev/tech/conduits/

Generic Conduit Using Visual Basic

Michael Knopp
Developer Support Engineer
Palm Inc.

Agenda

- Why Visual Basic?
- Generic Conduit Introduction
- A Bit of Inheritance…
- Visual Basic Generic Conduit Classes

Why Visual Basic ?

- Rapid application development
- Rapid **CONDUIT** development
- Easily create data-centric applications
- Easily create data-centric **CONDUITS**

Generic Conduit Introduction

- Base Classes and functionality are provided
  - Category Classes
  - Synchronizer Classes
  - Manager Classes
  - Record Classes
- Derive classes and customize them for the data formats used on the handheld or PC
- RAPID CONDUIT DEVELOPMENT !

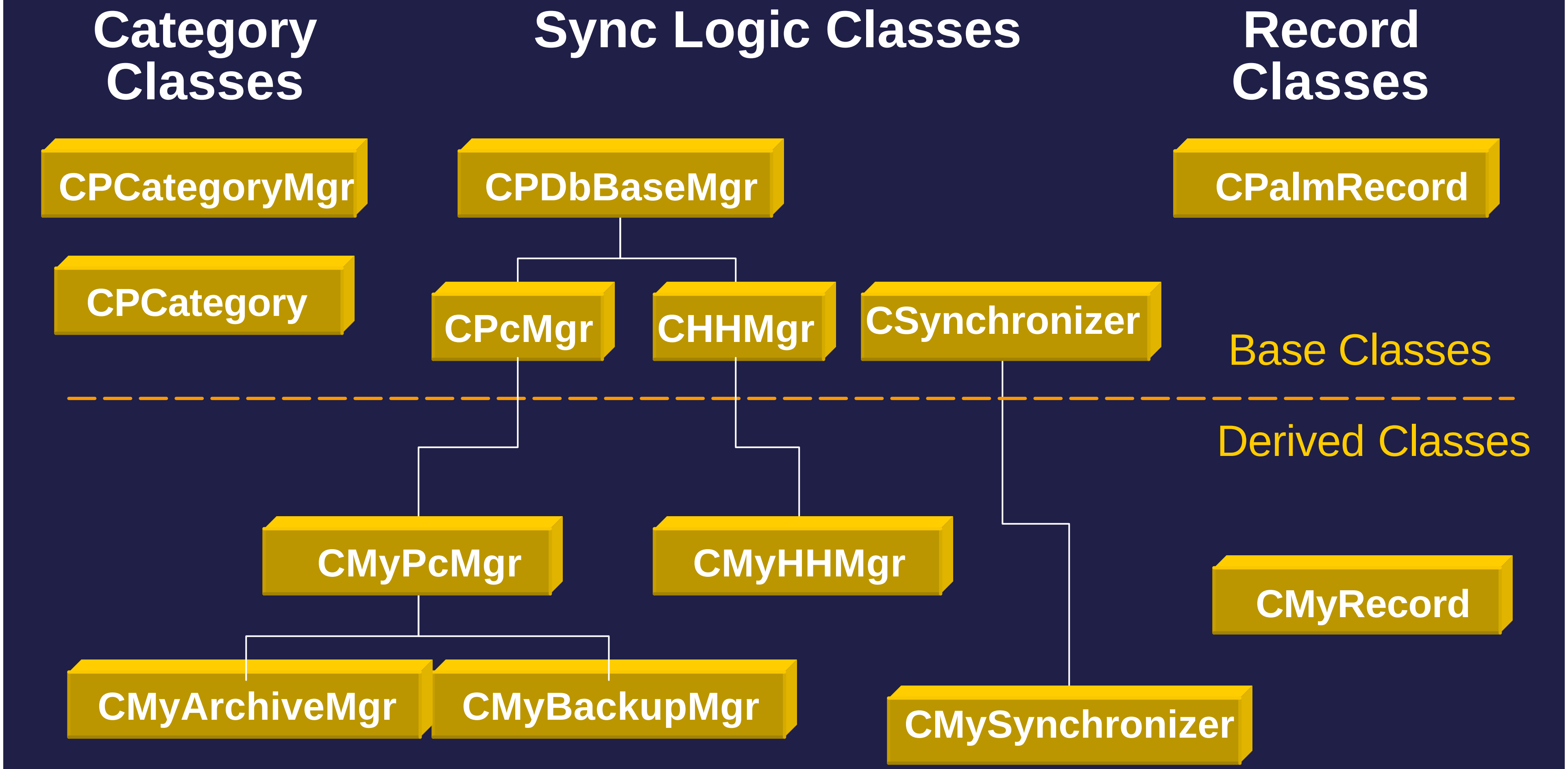
Generic Conduit Introduction

- When should I use the Generic Conduit ?
  - Not Ideal:
    - One-way download or upload
    - No synchronization required (Corp. phone book)
  - Ideal:
    - PIM Applications
    - Dynamic databases
    - Sync Logic ? What’s that ?

Generic Conduit Introduction

- Looks Good ! What are the general steps ?
  - 1. Describe your record format
  - 2. Implement storage and retrieval of your records
  - 3. Implement storage and retrieval of an archive database or backup database
  - 4. Implement the record conversion to/from the generic CPalmRecord format
  - 5. Implement any synchronization of AppInfo data

“Generic Conduit” Class Hierarchy



Inheritance

- Delegation Method:
  - Create a private variable for doing delegation:
    - *Private delegationVar* as Parent
  - In the child class Initialize procedure:
    - *Set delegationVar* = New Parent
- Implements Method:
  - Simply add “Implements Parent” in the child class
- The derived classes are implemented using both methods

Generic Conduit Classes

- Collection Classes:
  - CPCategory
    - Collection of all categories
    - Store collections for handheld and Desktop
  - CPalmRecord
    - Collection of all Palm record data
    - Common record format
    - Used for handheld and Desktop synchronization

Generic Conduit Classes

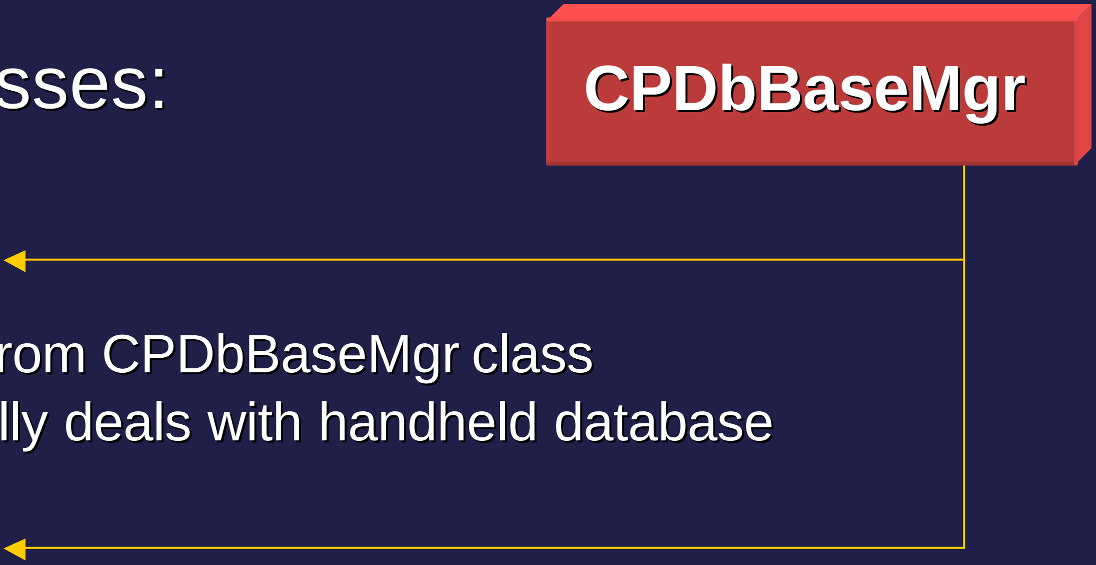
- Standalone Classes:
  - *CPCategoryMgr*
    - Manages the CPCCategory collection class
    - Contains functions which manipulate categories for synchronization

Generic Conduit Classes

- Parent Classes:
  - *CPDbBaseMgr*
    - Manages the from handheld and Desktop
    - Contains file related functions
  - *CSynchronizer*
    - Contains important Synchronization activities
    - Uses ComConduit calls

Generic Conduit Classes

- Derived Classes:

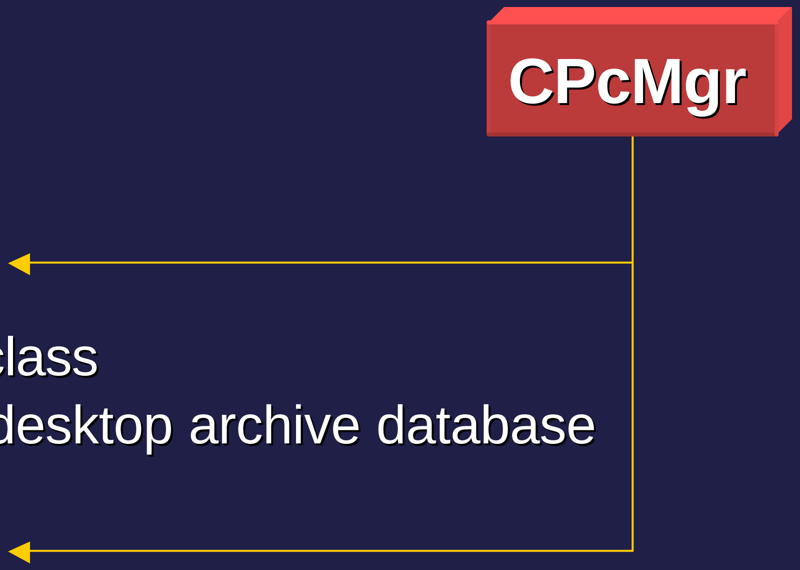


CPDbBaseMgr

  - *CHHMgr*
    - Derived from CPDbBaseMgr class
    - Specifically deals with handheld database
  - *CPcMgr*
    - Derived from CPDbBaseMgr class
    - Specifically deals with the desktop database

Generic Conduit Classes

- Derived Classes:



CPcMgr

  - *CArchiveDatabase*
    - Derived from CPcMgr class
    - Specifically deals with desktop archive database
  - *CBackupMgr*
    - Derived from CPcMgr class
    - Specifically deals with desktop backup database

Summary

- Rapid Conduit Development
  - Visual Basic + Generic Conduit
- Generic Conduit
  - Provides sync logic, logging, record reads/writes
  - Derive classes… ready to go !

Questions and Answers

- Kelly McCaw: CDK Lead Engineer
- Cole Goeppinger: CDK Engineer
- Muru Manokaran: CDK Engineer
- Geoff Richmond: Developer Support Engineer
- Ravi Duggaraju: CDK Engineer

accelerate

palm

collaborate

innovate

communicate

PalmSource

Conference 2000