DIREXIONS 2007

Base Classes 101

Presented by:

Kevin Kawado

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

To receive CPE credits, please sign up at the back of the room.

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Presentation Overview

- MAS 90 Framework Review
  - Base System Classes
  - Base Application Classes
  - Implementation Classes

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Core Concepts and Terms

- Tiered Architecture
  - Separation of Business and UI
  - Each Program has a Business Rules component and a UI component
  - Important for Utilities (e.g. Visual Integrator) and Integrations
    - Provides common business logic for all processes
    - Example: Sales Order Update - same logic
      - MAS 90 Desktop
      - VI Import
      - Act! Link or CRM Integration

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Base System Classes

- Session Tracking and Core Functionality
  - SY_Session – Session Control Object
  - SY_Security – Security Control Object
  - SY_UI – General UI methods
  - SY_File – General File I/O & Dictionary routines
  - SY_Constant – Common Constants
  - SY_OpenObject – Encapsulates ProvideX “Open Object”

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Session

- Session object must be instantiated for every session of MAS 90 or 200 that starts up
  - Instantiated as coSession or %SYS_SS
    - SY_Startup.m4p for Business Framework
    - SY_StartupLegacy.m4p for Legacy apps
- Maintains several key session properties
  - Logon Information including user code, company, module, security settings, etc.
- Provides system-wide functions
  - Launch tasks, encryption, etc.

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Security

- Creates security context to run applications under
  - UI Objects setup security during startup
  - Non-UI objects must inherit or create their own security object (those being created externally)
- Security Object attaches to Session
  - Security context stored in cSecurityAccess
  - Attached Security object will change when new UI program is run

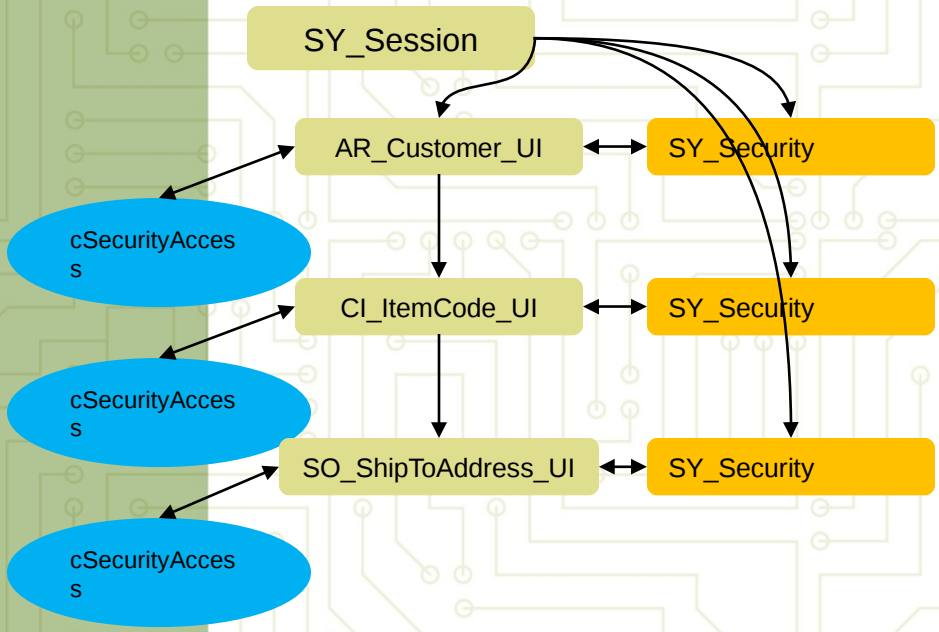
PROVIDEX

PARTNERS IN SUCCESS

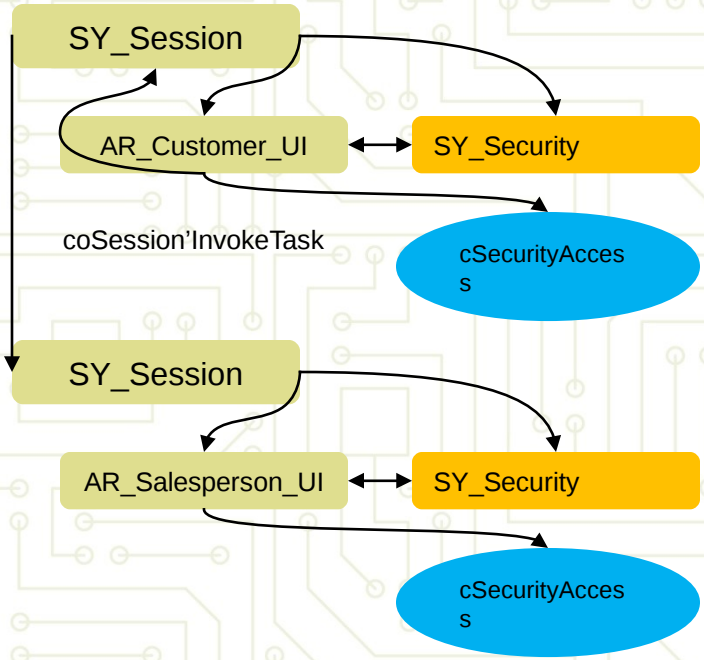
DIREXIONS 2007

SY_Security

Instantiating



Invoking



PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_UI

- Created by SY_Session when a UI Process is launched
- Adds low level properties and methods that are applicable only when a UI is present
  - Caption$
  - SystemDate$
  - MessageBox$()
  - ProgressBar()
- Note new Progress Meter for 4.30

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_File

- Provides file I/O for the Data Dictionary
  - IOList$
  - FileCreated
  - FileExists
  - OpenTable()
  - CreateTable()
  - LockTable()
  - CloseTable()
  - KeyDescription$()
- Most dictionary functions moved to external API for performance

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Constant

- Contains default properties and methods used throughout MAS 90
  - LastErrorMsg$ & LastErrorNum$
  - Return Constants
  - Message Constants
  - Edit State Constants
  - Date and Time Constants
- Inherited by virtually all objects in the Framework
  - Don’t modify SY_Constant to call other objects (creates circular references)

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_OpenObject

- Inherited primarily by SY_Service
  - Should be a part of any business object
  - Ensures correct pathing logic
- Utilizes OPEN OBJECT
  - File channels opened for exclusive use
  - Other objects (unless part of the inheritance tree) cannot access this file
  - Enforces encapsulation
- Potential Windows Vista/Server 2008 enhancements
  - Files may need to move to new location

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Class Hierarchy

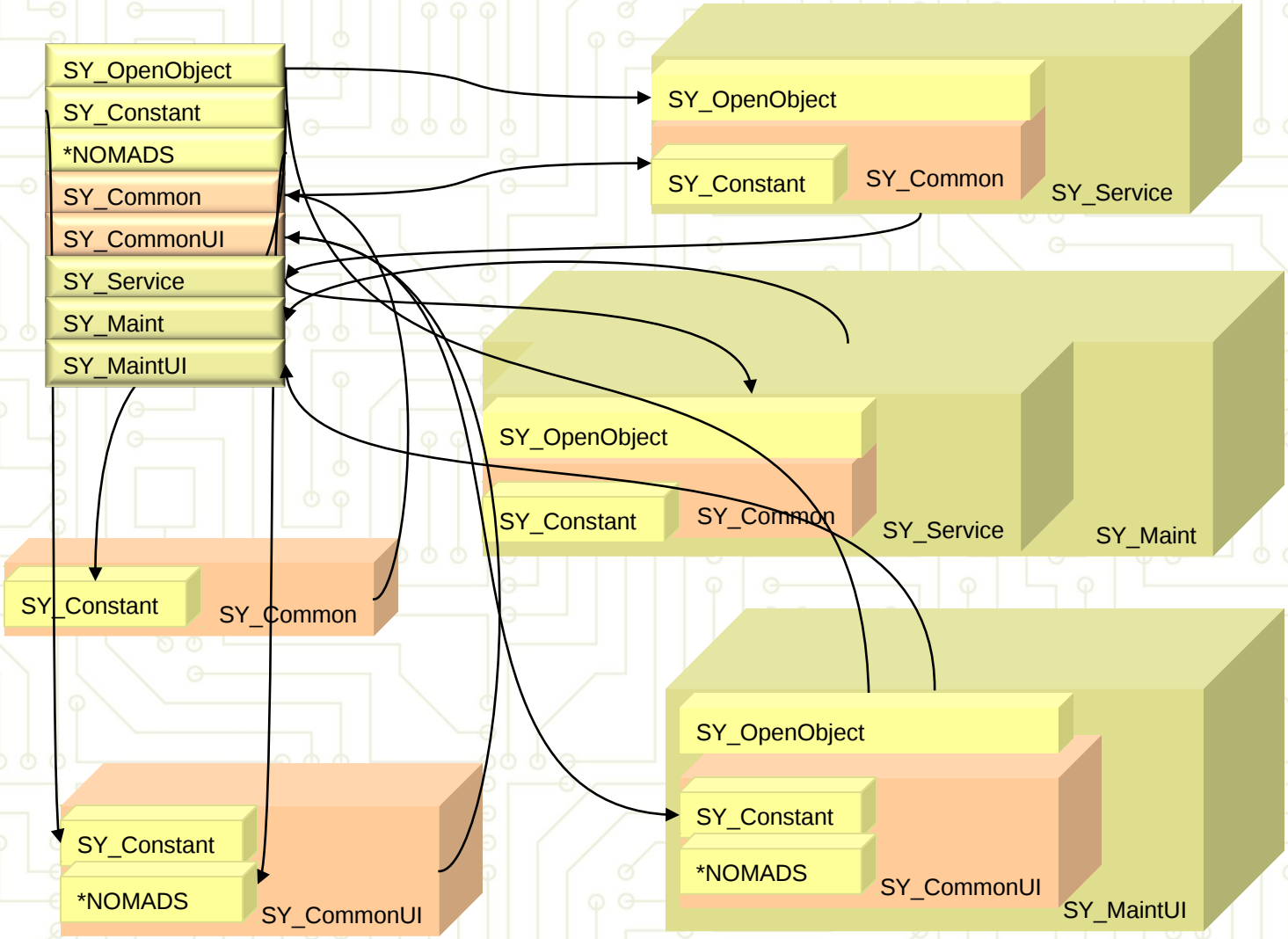
- Base or Foundation Classes
  - Core classes providing common functionality
  - Always inherited into Implementation class
  - Should not be modified
    - Equivalent to old LM public programs
    - Provides commonality between all programs
- Implementation Classes are Created
  - Objects built from Base classes
  - Exception code that is unique for program
  - Separate functionality for Service, Business and UI classes

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Hierarchal View of Base Classes



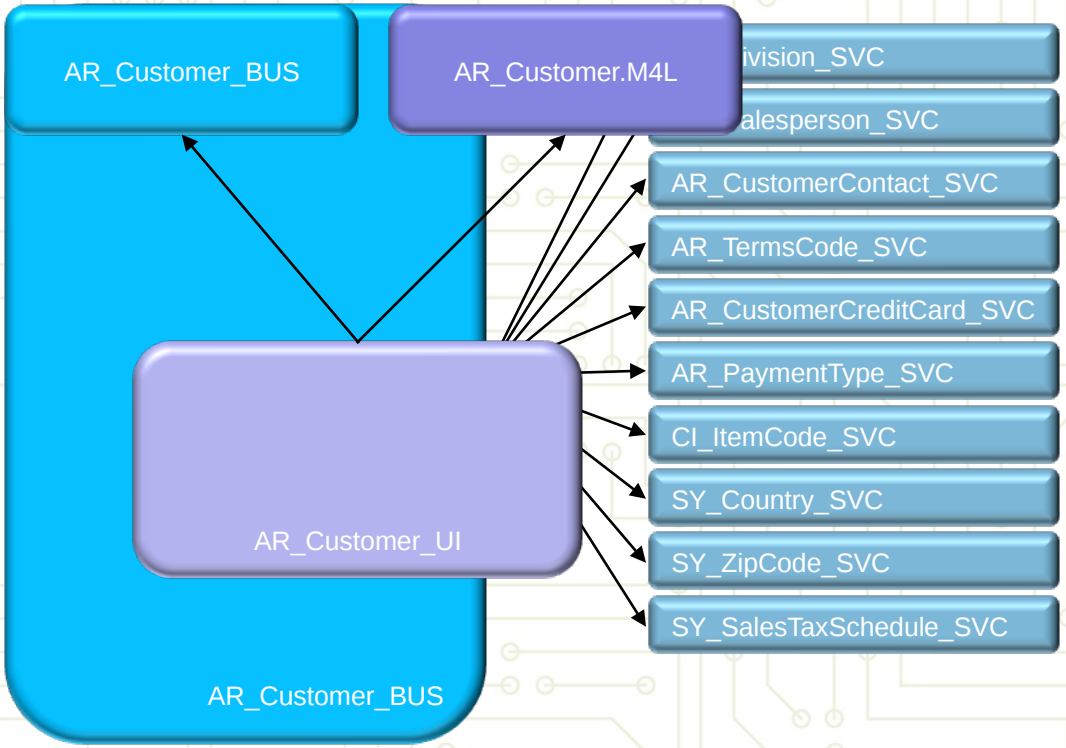
PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Building Classes

- SY_OpenObject
- SY_Constant
- *NOMADS
- SY_Common
- SY_CommonUI
- SY_Service
- SY_Maint
- SY_MaintUI



PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Base Application Classes

- Application Building Blocks
  - Business Classes
    - SY_Common
    - SY_Service
    - SY_Maint
  - UI Classes
    - SY_CommonUI
    - *NOMADS
    - SY_MaintUI
  - Validation Classes

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Common

- Inherited class
  - Inherited primarily by SY_Service
  - Also inherited by SY_Update
- Includes Common Local Methods
  - LocalizeIOList()
  - GetNextSequence()

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Service

- Provides consistent interface to access data (read only)
- Base implementation class for Service objects and inherited into SY_Maint
- Opens table specified in cMainTable$
- Key Properties and Methods
  - EOF,BOF
  - Browse Methods
    - MoveFirst(), MoveLast(), Find(), etc.
  - BindVariables()
  - GetRecord(),GetValue(),etc.

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Maint

- Provides consistent interface to access data (read and write)
  - Read interface inherited from SY_Service
- Include properties for Record State
  - EditState
  - RecordChanged
- Includes Methods used to Read/Write records
  - SetKey() / SetValue()
  - Write()
  - Delete() / Clear()

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Maint – continued

- Methods used to Read/Write records
  - SetValue()
    - Calls RetrieveCollInfo() to get column info from the dictionary
      - Override dictionary info with either:

        EVN("_OBJ'ColumnOverride"+ColumnName$)

        EVN("_OBJ'ClassOverride"+ClassName$)
    - Calls Validate() which performs dictionary validation and calls other validation routines

      retVal = _OBJ'Validate(col$,val$)
    - Sets the value in the UI if one exists

      IF coUI { coUI'SetVar(col$,VIN(col$)) }

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Maint – continued

- Methods used to Read/Write records
  - Validate()
    - Performs any validation group logic first
      EVN("_OBJ'ValidateGroup"+GrpName$+"()")
    - Dictionary validation is performed next
      _OBJ'ValidateRule(val$,colDesc$,colType$,...)
    - Column Specific Validation
      EVN("_OBJ'Validate"+Column$+"("+Val$+")")
    - Class Validation
      EVN("_OBJ‘ClassValidate"+Class$+"("+Val$+")")

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_Maint – continued

- Referential Integrity
  - _OBJ'PostWriteRec()
    - Called from SY_Maint'Write()
    - After a successful write
    - Used for updating additional information
  - _OBJ'PostDeleteRec()
    - Called from SY_Maint'Delete()
    - After a successful delete
    - Useful for cleaning up or updating additial information

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Implemented Business Object

- Inherits SY_Maint
- Add Implementation properties
  - cMainTable$ / cIsSysDB
- Add Implementation Methods
  - SetChildColl()
  - ValidateXxxxxx(val$)
- Override base class to achieve desired functionality
- Link in any Service and Validation Classes

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Implemented Business Object

- Common Implementation Examples
  - SetDefaults() – For default values need to be set programmatically when creating a new record
    - Example: SO_Invoice_bus Implements SetDefaults() to set BatchNo$, InvoiceDate$, InvoiceType$, etc.
  - Write() & Delete()– For updating additional tables when writing or deleting a record
    - Example: AR_Customer_bus updates IT_Customer and various other tables depending on setup options
  - ConfirmDelete() – For checking restrictions when deleting a record
    - Example: AR_Customer_bus will not allow Customers with Open Invoices, CRM Prospect Customers, Etc, to be deleted

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

*NOMADS

- ProvideX-provided class
  - NOMADS functionality encapsulated
  - Logic previously found in NOMADS now available as methods in the application class
    - PROCESS()
    - PreLoad()
    - PostLoad()

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_CommonUI

- Methods common to all UI objects
- Inherits *NOMADS
- Inherited by SY_MaintUI
  - Used in place of SY_MaintUI if added functionality not needed
  - Utilities, Launcher, Reporting
- Includes basic UI Methods
  - AppendTitleBar()
  - NomadsProcess()
  - ParseTagField()
  - SetControlState()

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_MaintUI

- Provides consistent UI interface for maintaining data
  - Inherits SY_CommonUI, SY_OpenObject
- Include properties for UI
  - coBusiness, cSecurityAccess, cInitMain
- Includes Methods to connect UI to Business Object
  - InitBusinessObject()
    - Invoked from ON_CREATE of Implementation UI
    - Instantiates Business Object and sets UI handle
    - Localizes (STATIC) the main table IOLIST

DIREXIONS 2007

SY_MaintUI

- Connecting UI to Business Object
  - GetBusinessObject()
    - Returns BUSOBJ= tag of the current or specific control
  - BindVariables()
    - Processes BIND= tag of each control
    - Invokes ExplodeIOL()
  - ExplodeIOL()
    - Called from KeyChange(), Browse() & BT_Cancel()
    - Read data from Record$ to IOL$
    - EVN("_OBJ'KeyChangeAdditional()")
      - Implement to override KeyChange() logic
    - EVN("_OBJ'PostReadRec()")

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_MaintUI

- Connecting UI to Business Object
  - KeyChange()
    - Invoke this method from Change method of key column(s) of Implementation UI
    - Sets the key of the business object
- Standard UI Methods
  - OnExit()
    - Method invoked when Dialog panels are closed
    - Invokes ConfirmWrite() if cSkipDialog is not set
    - Invokes BT_Cancel()
    - Clears the business object

DIREXIONS 2007

SY_MaintUI - continued

- Standard UI Methods
  - PostLoad()
    - This method is invoked for Dialog panels
    - Invokes SetFormDisplay() to hide/show controls based on security
    - Invokes BindVariables(cInitMain)
    - Invokes AppendTitleBar() to modify the Caption to include company and date
    - Calls KeyChange if a key is passed on the Process argument list
      - KeyChange(ARG_1$)
      - _OBJ'SetInitialRecord(atom)

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_MaintUI - continued

- Standard Button Methods
  - BT_Accept()
    - Calls the Write() method for the main business object and any other business object in collection
    - Calls BT_Cancel after the write is finished
  - BT_Delete()
    - Calls ConfirmDelete()
    - Calls the Delete() method for the business object
    - Calls BT_Cancel after the delete is finished
  - BT_Cancel()
    - Calls the Clear() method for the business object
    - Calls SetFormState()
    - Calls SetFocusFirstID()

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_MaintUI - continued

- More Standard UI Methods
  - ConfirmDelete()
    - Calls ConfirmDelete() from the Business Object
    - If record “can not” be deleted – Display message as to why not
    - If record “can” be deleted – Display confirm delete message

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

SY_MaintUI - continued

- DefaultChange()
  - Invoked by NOMADS if no specific change logic exists
  - Calls SetValue() for the column that matches the control name
    - Fire any dictionary and business validation logic
  - If SetValue returns retSUCCESS
    - No further action, DefaultChange returns
    - NOMADS sets focus to next control
  - If SetValue fails - retFAILURE
    - Display LastErrorMsg$ from business object
    - Restore the previous value of the control
    - Return focus to the same control
  - If SetValue returns retWARNING
    - Display message only
    - Remaining process is same as retSUCCESS

Hint: If you need additional change logic, you can still call DefaultChange() from your Change method

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Implemented UI Object

- Inherits SY_MaintUI
- Add your Implementation properties
  - cInitpXxxxx – BindVariable flags
- Set Screen properties in ON_CREATE
  - SCREEN_LIB$ = “XX_Xxxxxxx.M4L"
  - SCREEN_ID$ = "dMain“
- Call InitBusinessObject in ON_CREATE
- Add your Button Methods
  - Methods named same as Button control
    - Ex: BT_Accept(), BT_Cancel(), BT_Delete()

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Implemented UI Object

- Add Event Handling methods
  - OnFocusXxxxxx()
    - Save off values of other, affected controls
      - Example: SO_CommonEntry_UI'OnFocusUnitPrice() saves off Extension amount in case the unit price entered fails validation
  - ChangeXxxxxxx()
    - Implement control specific change logic when:
      - A Control is not part of the IOLIST
      - When more than Default Change logic is required
      - Key Columns where KeyChange() must be called
    - When possible, call DefaultChange() vs. coding default change logic yourself
      - Example: SO_CommonEntry_UI'ChangeDepositAmt

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Implemented UI Object

- Inherit UI Validation Logic
  - For all classes used on the Form
    - ClassValidate
    - ClassFormatter
    - ClassOnFocus
    - ClassChange
  - LIKE all relevant UI_VAL classes
  - Keeps consistency between various control elements

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Implemented UI Object

- Panel and Dialog methods
  - PreLoad(), PreLoadXxxxxxx()
    - Called prior to loading the Form or Panel Xxxxxxx
  - PostLoad(), PostLoadXxxxxxx()
    - Called after Form or Panel Xxxxxxx is loaded but before first control gaining focus
  - OnExit(), OnExitXxxxxxx()
    - Called when Form or Panel Xxxxxxx is closed

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Implemented UI Object

- Other common UI methods
  - PostReadRec()
    - Implement to execute logic processed when the Business Object's EditState changes
      - KeyChange()
      - Browse()
      - BT_Cancel()
    - Used to set the screen state based on the EditState of the business object and other options

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

Questions?

PROVIDEX

PARTNERS IN SUCCESS

DIREXIONS 2007

End of Presentation

THANK YOU!

PROVIDEX

PARTNERS IN SUCCESS