

*Innovation in Action*



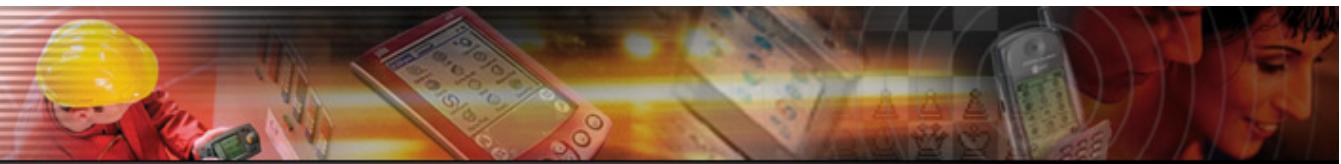
PalmSource®  
Conference and Expo

# PalmSource® Conference and Expo

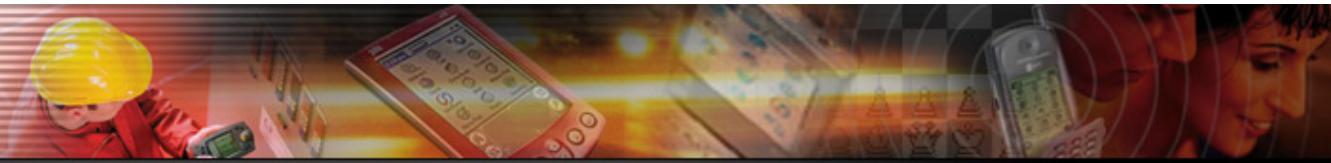


## APIs for the Next Generation Desktop

Raja Venkataraman  
Software Architect



Learn how the Palm Desktop  
helps you add value by extending  
existing applications and creating  
new ones to provide more  
services to the user.



## Agenda

---

Palm Desktop 4.1

---

Addins, Extensions, Short Cuts

---

Developing Addins in VB and C++

---

Demo

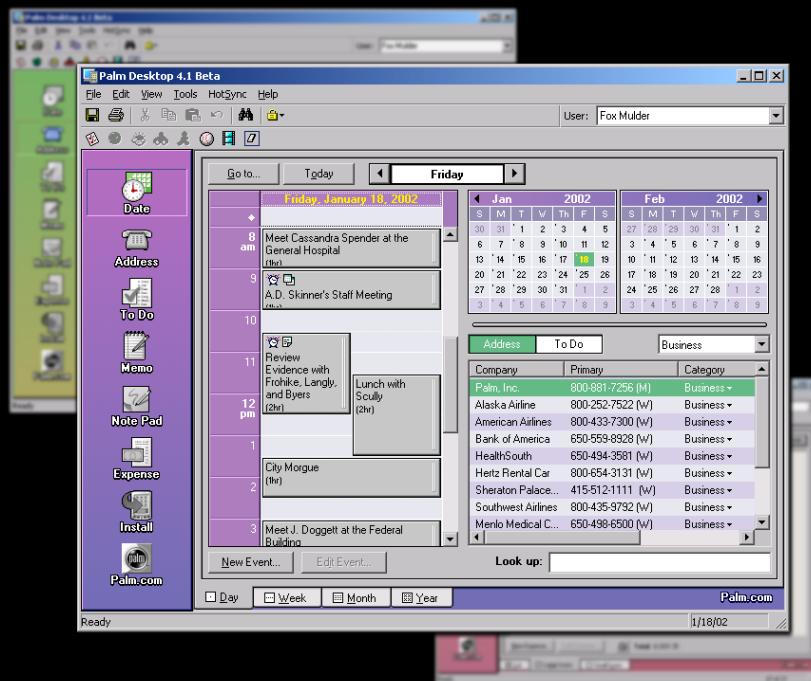
---

Q & A

---

# Palm Desktop 4.1

- UI Enhancements
- Performance
- Extensibility Framework
- Expense
- Note Pad
- No data format changes



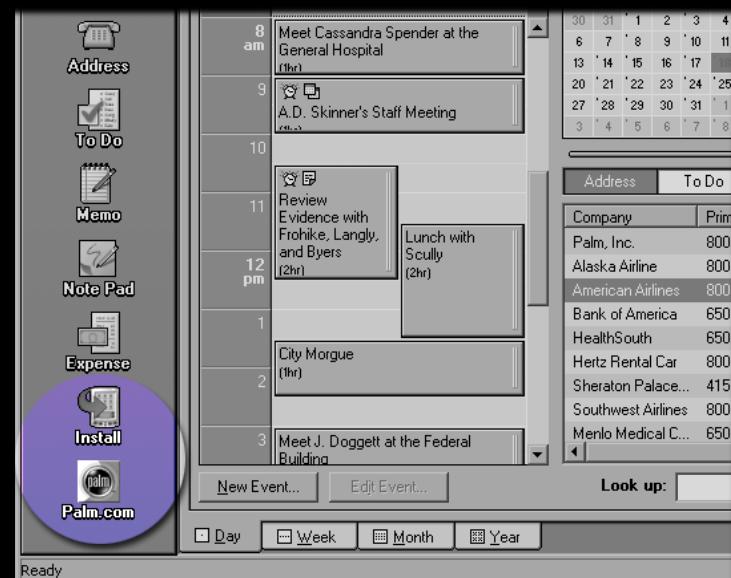


## Extensibility

- Short Cuts
- Addins
- Extensions

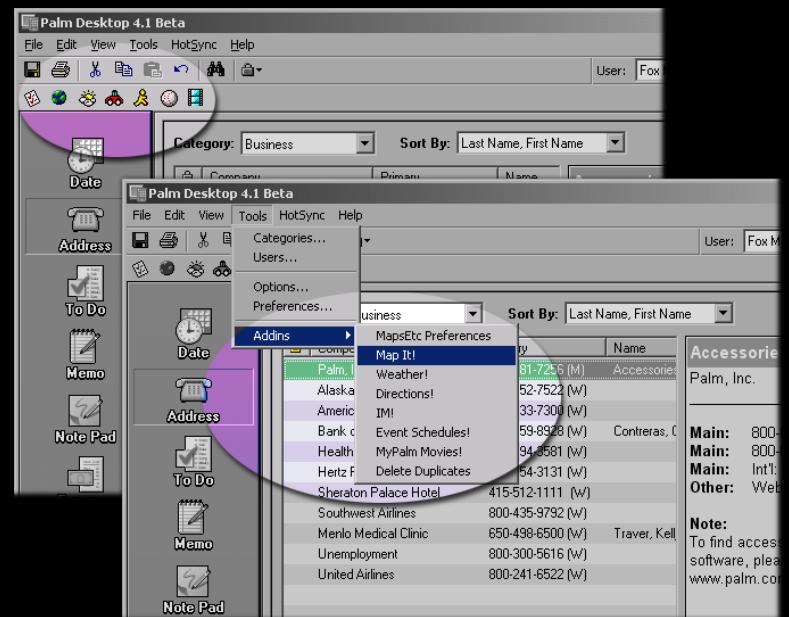
## Short Cuts

- Launches 3<sup>rd</sup> party apps
- No extra coding required
- Can handle all registered file types
- Typically used for launching web browsers, doc viewers, etc.



## Addins

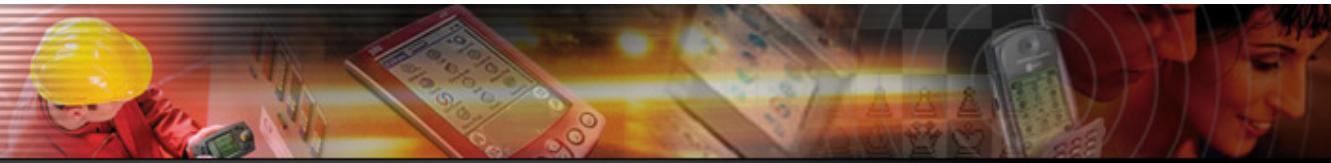
- Extend functionality of existing PIMs
- Read and write data from all PIMs using documented interfaces
- Provide new and unique services based on existing data
- Can be enabled for 1 or more PIMs



## Extensions

- Used to create new PIM applications that run within the Palm Desktop environment
- Typically have a device counterpart
- Seamless integration
- Simplifies coding, allowing developer to concentrate on core functionality





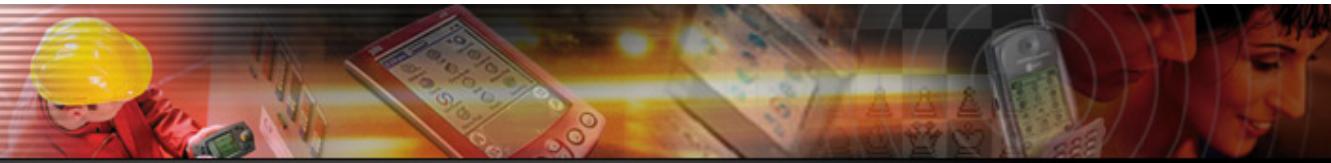
## So, what exactly is an Addin?

- In-proc (.dll) COM server
- Implements 2 interfaces
  - IDesktopAddin
  - IDesktopCommand
- Can have 1 or more commands
- Accesses Palm Desktop Object Model
- Registers itself with Palm Desktop
- Written in Visual Basic or C++



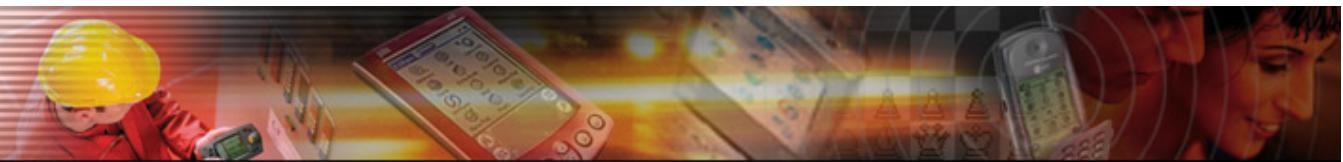
## Addin Interfaces - IDesktopAddin

- **OnInitialize**
  - Called once, at startup
  - Application Object Model passed in as a parameter
  - Stores Application pointer away
  - Creates and adds Command objects
- **OnUninitialize**
  - Called once, at shutdown
  - Releases all objects

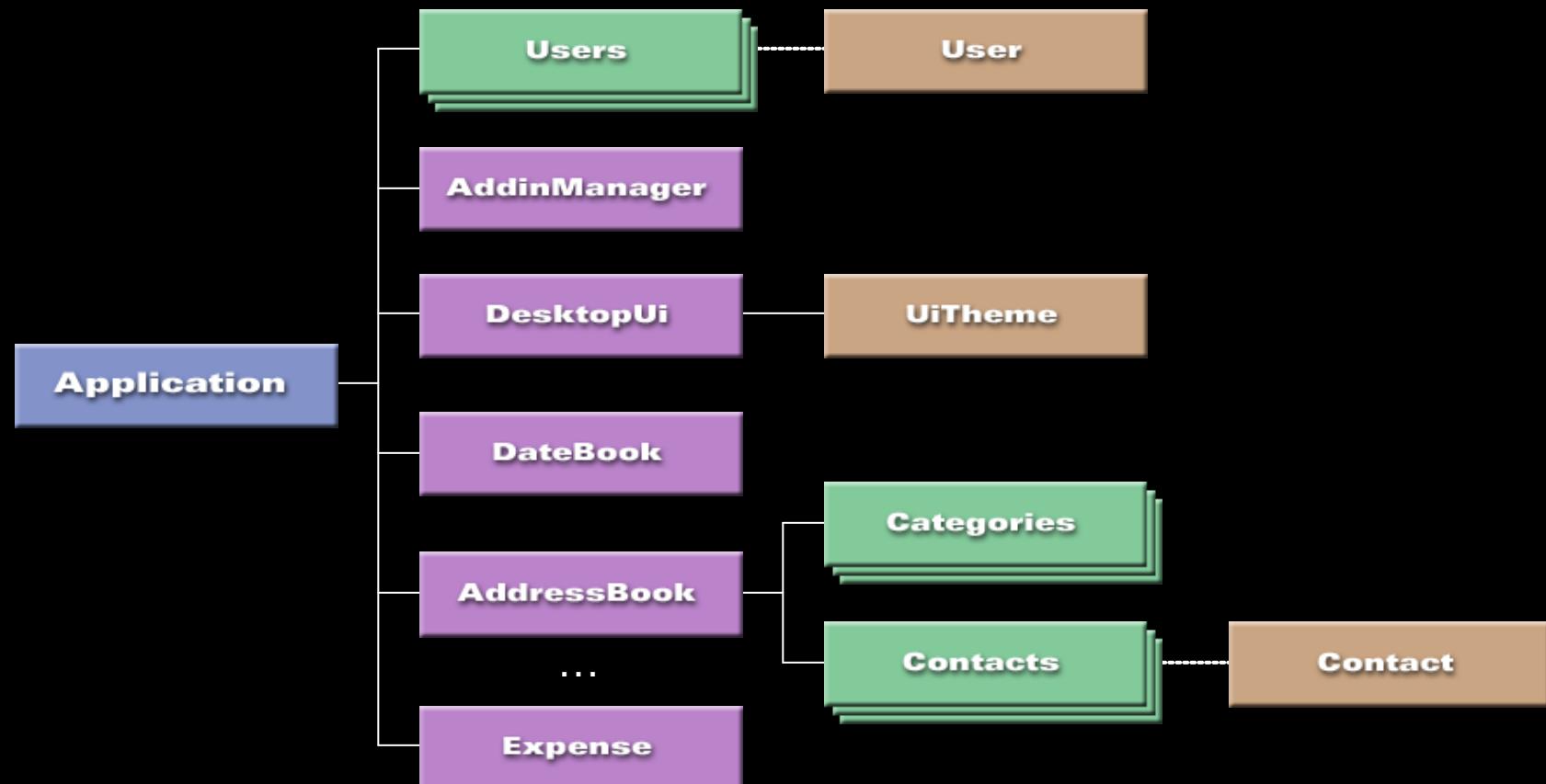


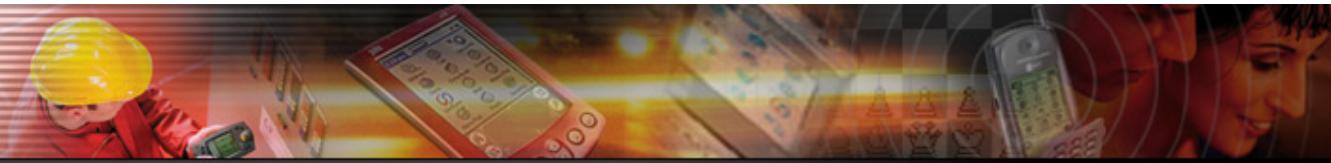
## Addin Interfaces – IDesktopCommand

- OnCommand
  - Called when user selects command
  - Typically accesses data using the Application object pointer stored away in OnInitialize
  - Can display modal UI
  - Release all objects before returning from function



## The Object Model - Partial





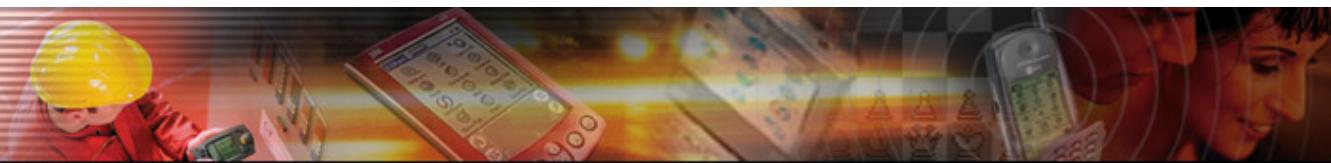
## VB Sample – List Users

```
' gPalmApp As PalmDesktop/DesktopApplication
' set in OnInitialize

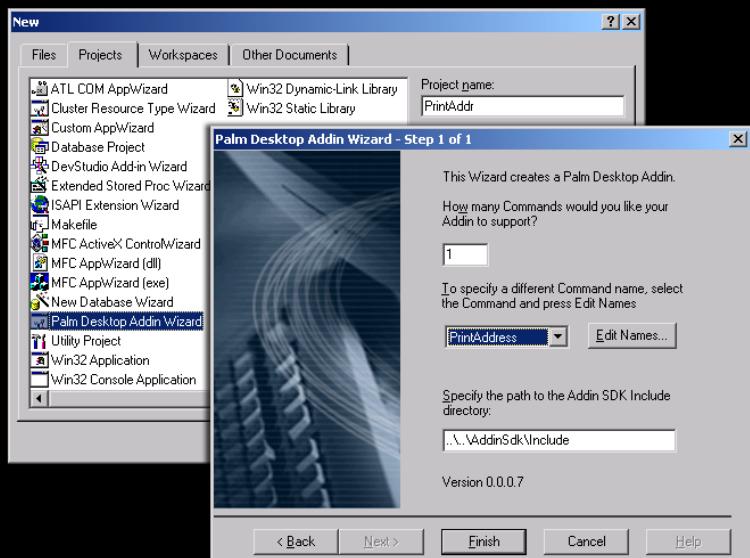
Dim UserList As New Users
Dim ThisUser As New User

Set UserList = gPalmApp.Users
For i = 0 To UserList.Count - 1      'Note: 0 based Index
    Set ThisUser = UserList.Item(i)
    MsgBox ThisUser.Name
Next i

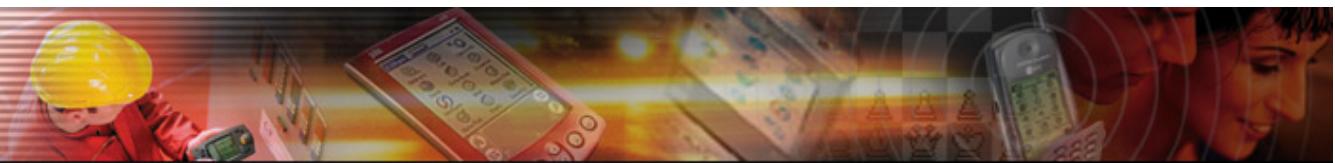
Set ThisUser = gPalmApp.CurrentUser
MsgBox "Current User " & ThisUser.Name
```



## C++ Addin Wizard



- Supports Visual Studio 6
- Generates simple Addin
- Allows developer to specify number of commands
- Has Registration code
- Code snippets



## C++ Sample – Get Selected Contact

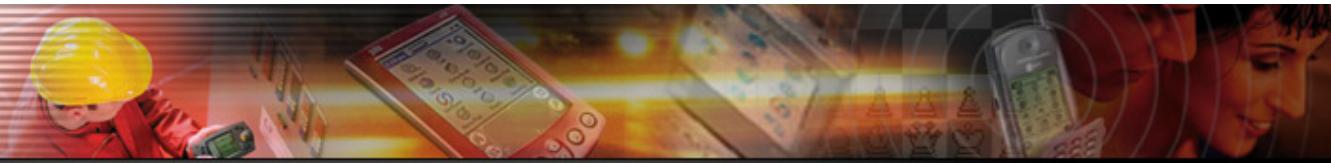
```
// spAddressBook is retrieved using GetAddressBookPim
try {
    IContactsPtr spContactsList;
    spContactsList = spAddressBook->GetSelectedContacts();

    // Assumes at least one contact selected
    IContactPtr spContact = spContactsList->GetItem(0);
    _bstr_t bstrFirstName = spContact->GetFirstName();
    _bstr_t bstrLastName = spContact->GetLastName();
    // Do something with the name
}
catch (_com_error e) {
    AfxMessageBox(e.Description());
}
```



## Handling Private Records

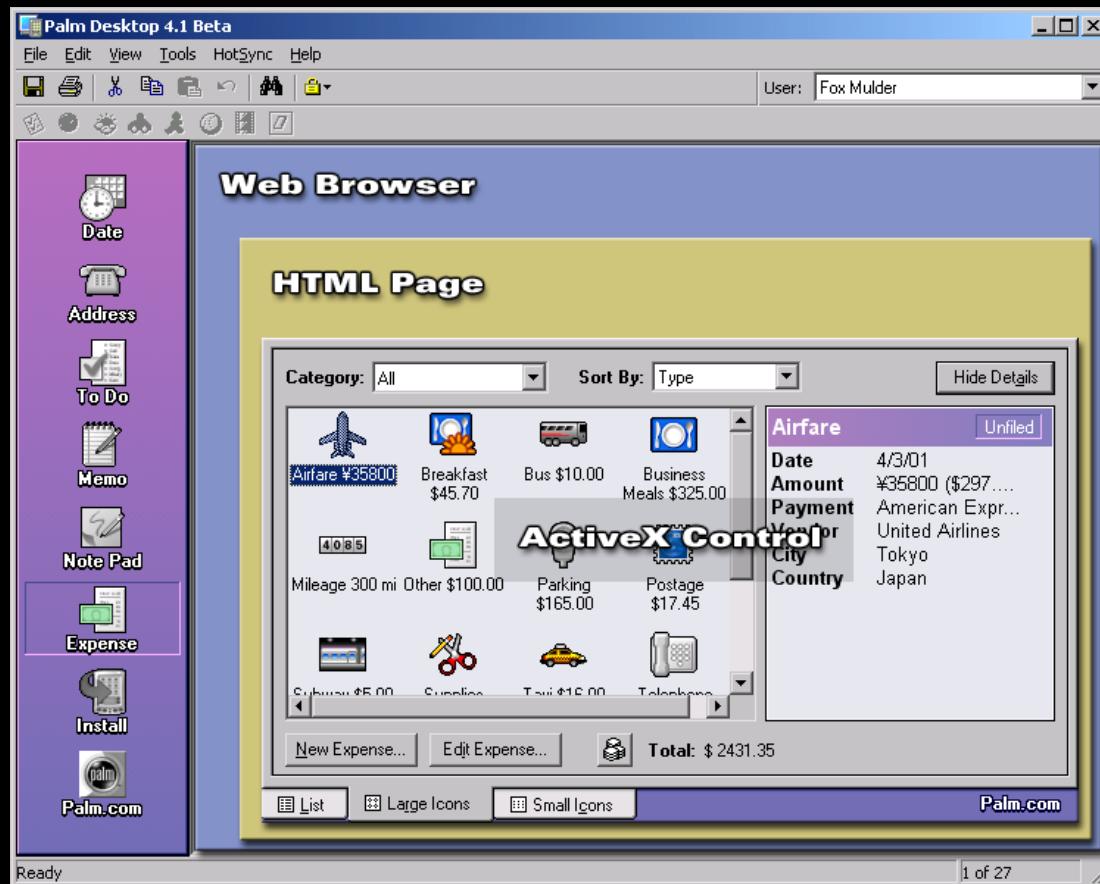
```
// Get App's Privacy State  
LONG lAppPrivacyState = spApp->GetPrivacyState ();  
  
// Get Record's Privacy State  
LONG lPrivate = spRecord->GetPrivate ();  
  
// Deal with record only if record is not private  
// or Show Private Records is set  
if ((lPrivate == 0) ||  
    (lAppPrivacyState == prvShown)) {  
    // Do something with record  
}
```

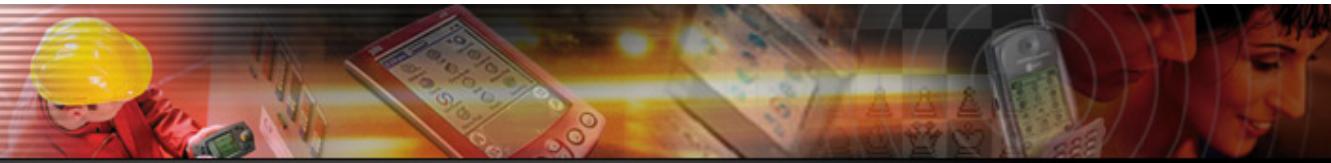


## What is an Extension?

- ActiveX control embedded within a HTML page
- Has an icon in the App bar
- Interacts with Desktop using Object Model
- Receives Event notifications
- Can expose its own Object Model
- Can work within Outlook with some changes

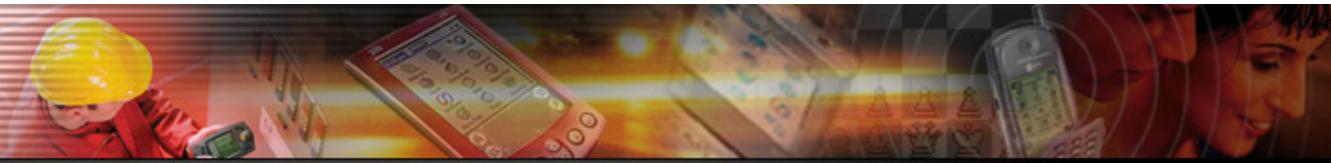
## Extensions – Hosting





## Why design Apps as Extensions?

- Seamless integration – access to Menu, Toolbar, even Theme information
- Simplifies programming – no need to access users.dat, or write HotSync Notifier DLLs
- Receive Notifications (User change, HotSync, Theme change, etc.)
- Help support
- Participate in Global Find
- Addins can access your Object Model



## Developing Extensions

- Write an ActiveX control
- Test with Internet Explorer
- Register with Palm Desktop
- Access Object Model, sign up for events
- Integrate UI
- Expose Object Model (if you want)



## Registration Services

- COM based API to register components
- ProgID - **PalmDesktop.RegistrationServices**
- IRegistrationServices
  - IsExtensionRegistered
  - GetExtensionInfo
  - RegisterExtension
  - Similar functions for Addins and ShortCuts
- Callable from InstallShield, Microsoft Installer



## Installer Services

- Do **not** use *FindWindow*, *WM\_CLOSE*.
- Use **PalmDesktop.InstallerServices**
  - *IsDesktopRunning*
  - *IsAlarmAppRunning*
  - *KillPalmDesktop*
  - *LaunchPalmDesktop*
  - *DesktopVersion*
  - *InstallDirectory*
- Callable from InstallShield, Microsoft Installer



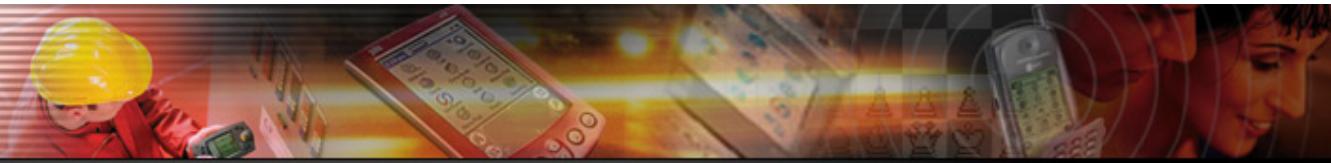
## Addin SDK

- Released as Beta
- Beta SDK includes
  - Documentation
  - Header files, Type Libraries
  - Addin Wizard
  - Sample code
- Requires Palm Desktop 4.1 to develop
  - Beta available on Developer web site
- Need your feedback



## What will be in the Final SDK?

- Updated documentation
- More samples
- Bug fixes
- Event object renamed to DateBookEvent
  - In Beta SDK, use PalmDateBook.Event
- Incorporate feedback



## Palm Desktop Developer Website

- <http://www.palmos.com/dev/tools/desktop/>
- You can download
  - Palm Desktop 4.1 – Beta
  - Palm Desktop SDK – Beta
- Developer Forum
  - “Win Desktop Forum”
  - <http://www.palmos.com/dev/support/forums/>



# Demo

*Innovation in Action*



PalmSource®  
Conference and Expo