ecosoft™

ADempiere
The Professional Open Source Business Suite

# Module 9
## Introduction to Application Dictionary

**By** Kitti Upariphutthiphong
Technical Consultant, **eco**soft™
kittiu@gmail.com

# Module Objectives

- ADempiere as an Application Framework
- Knowing Application Dictionary
- Creating a new simple window using Application Dictionary

# ADempiere as an Application Framework

ecosoft™

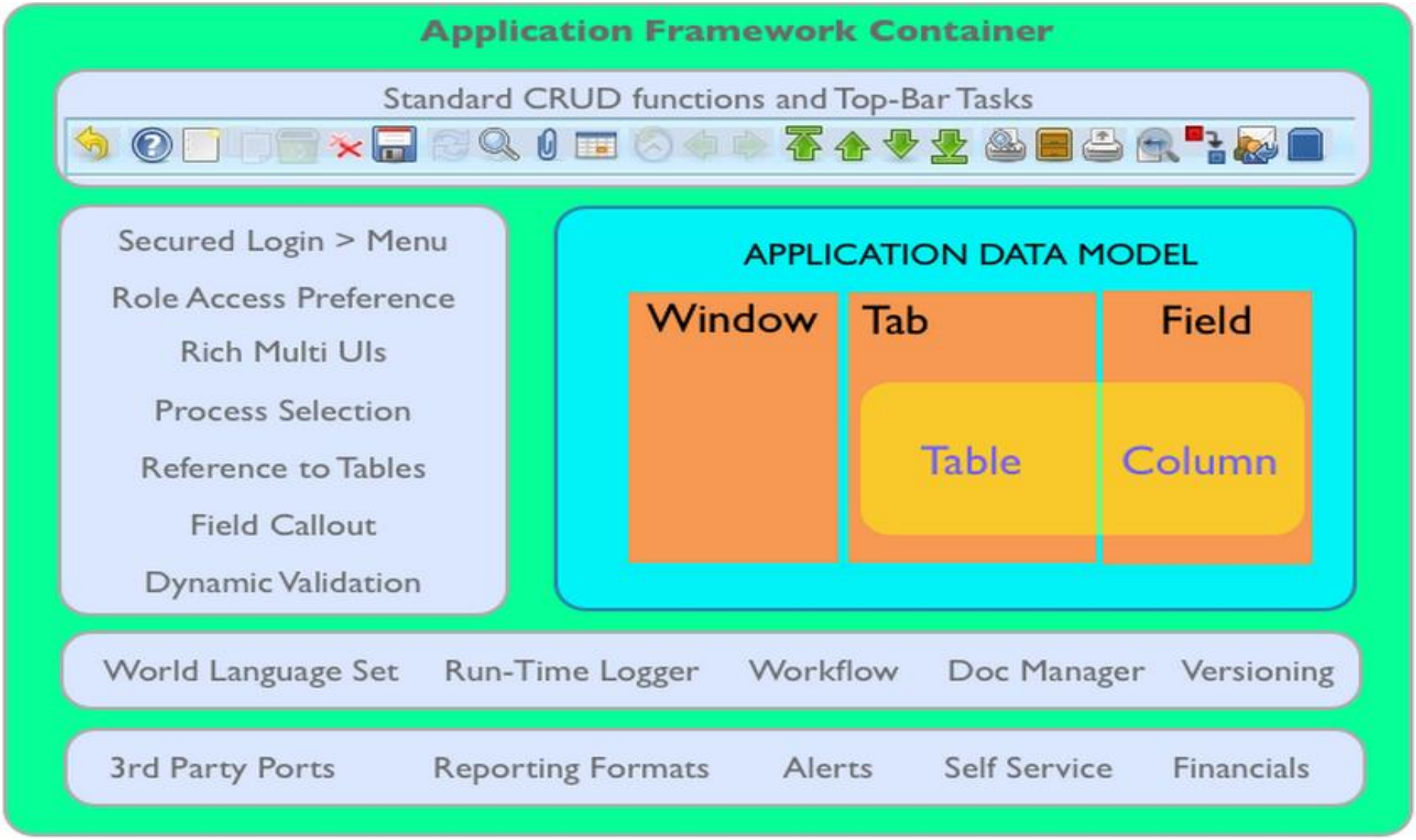
**"ADempiere is well known for its Powerful Application Dictionary Framework"**

**Planning & Configuration**

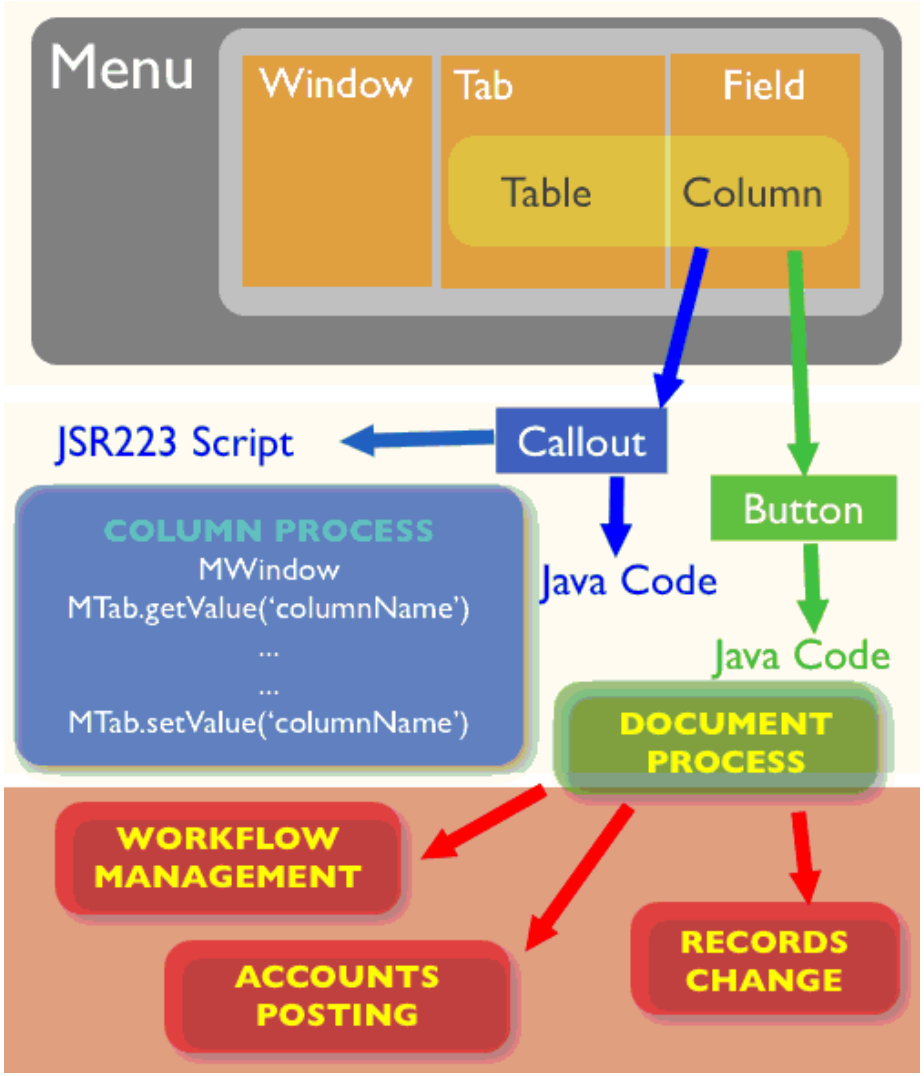
**Coding**

**This is a Big topic that require time to study, yet "Worth It!"**

# Application Dictionary Framework



# ADempeire Application Architecture




- **Interface Layer**
  - Pure Application Dictionary
  - Pure Configuration

***Note:*** *Tab = Table, Field = Column*

- **Process Callout Layer**
  - Partly managed by Application Dictionary
  - Configuration + Coding
  - Java Code or Light Weight Script

- **Document / Data Model Layer**
  - Object Model
  - Pure Java Coding
  - Base infrastructure of the system

# **Application Dictionary** Development Scenario

**Configure Database Table**

- Create & Update Table, Columns
- Syncing changes to underlining DB

**Windows, Tab, Fields, Buttons**

- Create & Update Windows, Fields, Buttons
- Organize looks and feels

**Reference Data Fields**

- Reference fields to data source
- Smart Database lookup

**Add Rules & Logics to Fields**

- Add field logics
- How data and fields will display
- Fields dependencies, etc.

**Develop Process & Reports (Coding)**

- Additional Callout or Process attached to Fields, Buttons or Actions
- Require Java or Light Weight Scripts

**Package as 2Pack (PackOut)**

- Package all changes as XML package
- For distribution to target server

**Production Deployment (PackIn)**

- Deployment Management

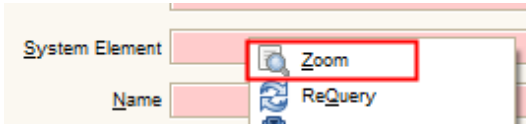
# WORKSHOP

# Scenario

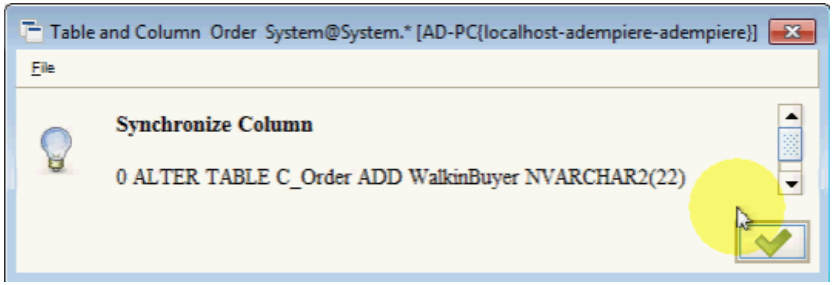
- For the POS Order only, we want a new field **Walkin Buyer Name** that list all user in the system
- This field will display only when it is a **POS Order**, when it display, display as **mandatory field!**

# Create a new simple field in Sales Order window

- Create new **WalkinBuyer** Column in table C_Order
  - Login as **System Administrator**
  - Open **Table and Column** window
  - Lookup for DB Table Name = **C_Order**
  - With record **C_Order** selects, click on **Column** tab
  - Click **New**
  - We need to create new **WalkinBuyer** element first, as this does not exists.
  - Right click on **System Element** field, and click **Zoom**



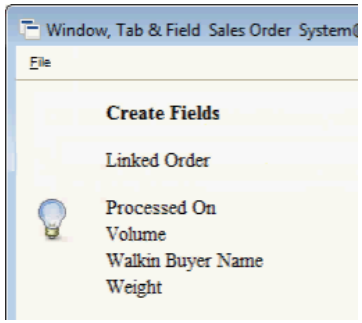

  - **Element** window open
- Create new **WalkinBuyer** element in Element window
  - Click **New**
    - **DB Column Name:** WalkinBuyer
    - **Name:** WalkinBuyer
    - **Print Text:** Walkin Buyer Name
  - Click **Save**
- Back to **Table and Column** window
  - Create new column with
    - **DB Column Name:** WalkinBuyer
    - **System Element:** WalkinBuyer
    - **Name:** Walkin Buyer Name
    - **Reference:** String
    - **Length:** 22
  - Click **Save**
- Sync with Backend Database to create new column
  - Click on **Synchronize Column** button to auto create new Column in DB.



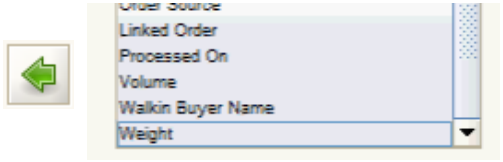

*Clicking **Synchronize Column** button will sync the Dictionary in ADempiere to the backend database. If the table exist (like in this case), it will just alter table. But if the table do not exists, it will create a new table.*

# Create a new simple field in Sales Order window

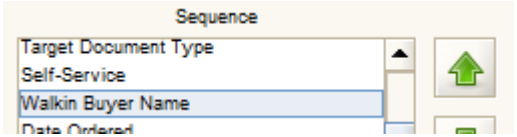
- Create the new field in Window Sales Order
  - Open **Window, Tab & Field** window
  - Lookup for Name = **Sales Orders**
  - With record **Sales Order** selects, click on **Tab** tab
  - Lookup for Tab Name = **Order**
  - Expand the page and scroll down
  - Click **Create Fields** button
  - New fields will be created,



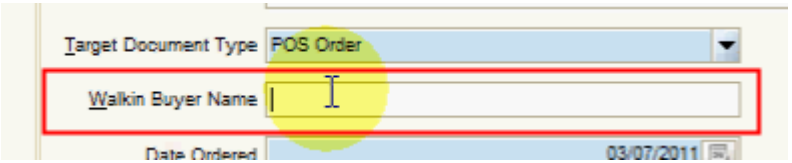

  - Note that system will create all fields that has not been created before. **Walkin Buyer Name** field is one of them.
- Arrange the new field.
  - Click on Field Sequence tab
  - Select **new fields** and move to the left pane. Except the **Walkin Buyer Name** field which remain on the right pane.




  - On the right pane, select **Walkin Buyer Name** and move it to up under the field **Self Service**



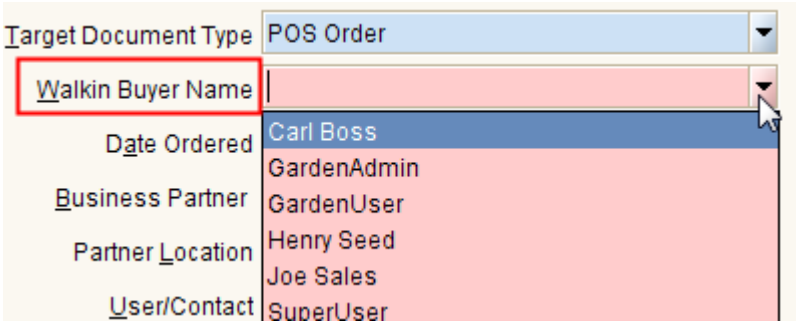

  - Click **Field** Tab, lookup for Field = **Walkin Buyer Name**
  - The new field is already in this Sales Order window
- View new field in **Sales Order** window.
  - Logoff and Login again as **GardenWorld Admin**.
  - On Sales Order window, you will see the new **Walkin Buyer Name** field.



# Adding some rules to the field

*So far, we have just create a simple string field. Now we will continue by making this field a user name dropdown list. And make it required field only when Target Document Type is POS Order*

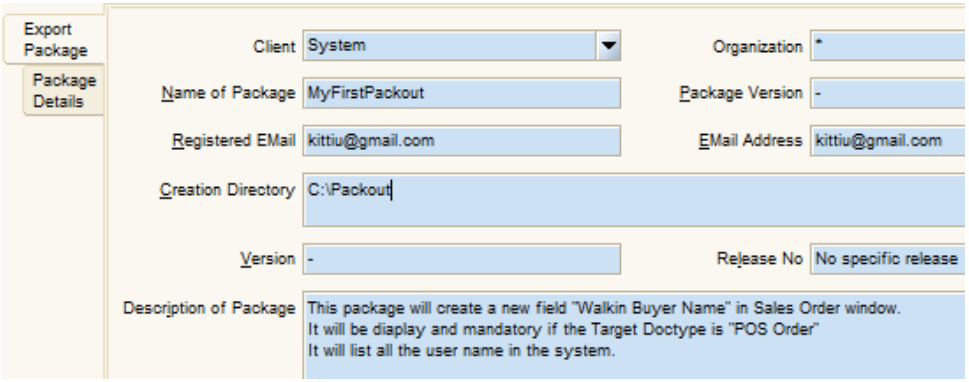
- Make **WalkinBuyer** field reference to user list in table AD_User
  - Re-Login as **System Administrator**
  - Open **Table and Column** window
  - Lookup for DB Table Name = **C_Order**
  - With record **C_Order** selects, click on **Column** tab
  - Look for column **WalkinBuyer** and change the setting as follow,
    - **Reference:** Table
    - **Reference Key:** AD_User
  - **Save** change
  - Click on **Synchronize Column** button to auto update Column in DB.
- Make this **WalkinBuyer** field visible only when **Target Document Type** is **POS Order**
  - Open **Window, Tab & Field** window
  - Lookup for Name = **Sales Orders**
  - With record **Sales Order** selects, click on **Tab** tab
  - Lookup for Name = **Order**
  - Expand the page and scroll down to **Walking Buyer Name** field.
  - Make the setting as follow,
    - **Display Logic:** @C_DocTypeTarget_ID@=135
    - **Mandatory Overwrite:** Yes
  - **Save** change
- View new field in **Sales Order** window.
  - Logoff and Login again as **GardenWorld Admin**.
  - On Sales Order window, you will see the new **Walkin Buyer Name** field now as drop down list.
  - By changing the **Target Document Type** to POS Order, it will be **Visible** and **mandatory**.



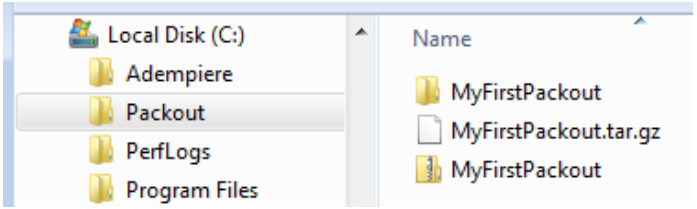

- Next step, we will package the changes as 2Pack format for reuse.

# Package the changes with 2Pack - Packout

- Package the change using Packout
  - Re-Login as **System Administrator**
  - Open **Packout** window
  - Click **New**,
    - **Name of Package:** MyFirstPackout
    - **Package Version:** 1.0
    - **Registered Email:** kittiu@gmail.com
    - **Email Address:** kittiu@gmail.com
    - **Creation Directory:** C:\Packout
    - **Version:** -
    - **Release No:** No specific release
    - **Description of Package: ….**
    - **Instruction: -**



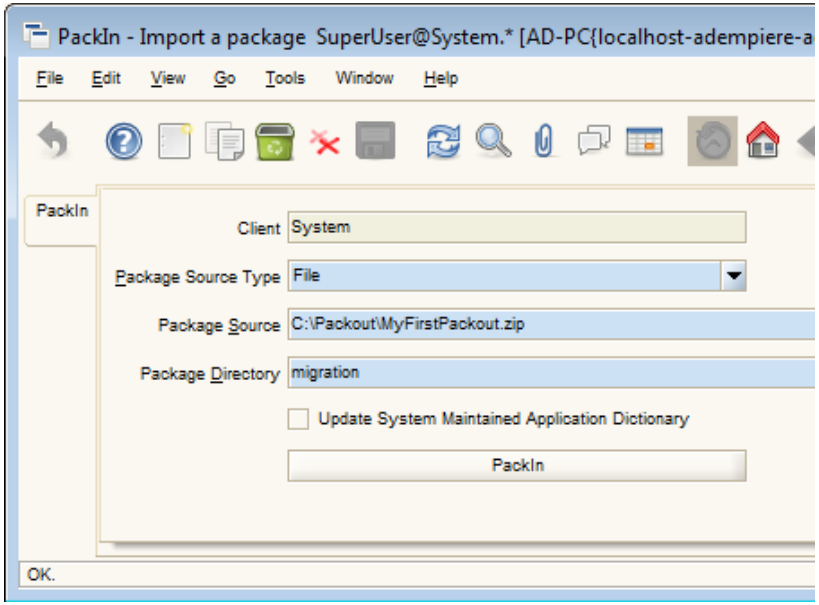

  - Click **Save**
  - Click on **Package Details** tab
  - Click **New**,
    - **Type:** Application or Module
    - **Menu:** Sales Order
    - **Note:** -
  - Click **Save**
  - Back to Export Package tab,
    - Click Export Package button
  - A package called MyFirstPackout will be created under C:\Packout




  - Package will be in zip for Windows and tar.gz for Linux. Inside package is instruction on all the change we made in XML format.
  - We can use this package to deploy the changes in Production System.

# Deploy the change to Other System - Packin

- Deploy the change to Other System using Packin
  - In target system, login as **System Administrator**
  - Open **Packin** window
  - Click **New**,
    - **Package Source Type:** File
    - **Package Source:** C:\Packout\MyFirstPackout.zip
    - **Package Directory:** migration
  - Click **Save**
  - Click **PackIn** button to deploy
  - If no error, **OK** message will be shown on the bottom left of the window.




- In Target System, view new field in **Sales Order** window.
  - Logoff and Login again as **GardenWorld Admin**.
  - On Sales Order window, you should see the new **Walkin Buyer Name** field.

*2Pack is a good way of packaging and deploying application. With Package Maintenance and Package Installed windows, you can also monitor changes you have make in a centralized way.*

***So far we only touch on the basic feature of Application Dictionary. To be an expert, you need a lot of real life experiences. ☺***

ecosoft™