



The aim of this session is to help 4D Developers to move their applications to 4D v16 safely and painlessly, while using the advantages of this very powerful version.

Many 4D applications have been started years ago, and have been moved from version to version without the real need for working on the bones and guts of the application's core. But there is more and more data to handle, everything is changing around apps (machines, systems, way of use, and more than all, Users).

We are going to see how to move step by step in order to get advantages of 4D v16 without the need to redo everything from scratch, and to be prepared for the future.

ADDING MORE OBJECTS IN YOUR CODE

- Objects in programming.
- Objects as Parameters.
- Hashing Functions & Arrays.
- Associative Arrays.

THE DATA STRUCTURE FROM SCHEMAFUL TO SCHEMAFLEX

- Objects as Fields
- Dealing with variable-structure Fields
- Adding User-parametrable Fields to an application in few clicks.
- Query, Sort, and use of this fields: Now and tomorrow.
- The traps of Key-Value Stores.

FROM 32-BIT TO 64-BIT

- Reasons why to use 64-bit.
- Dealing with resources on Mac: PICT, cicc, STR#.

FROM CALL PROCESS TO CALL FORM

- Getting rid of the Outside Call event.
- Getting rid of the On Timer event.
- Getting rid of the Callback Process.
- Multiple windows without multiple processes.

GOING WEB WITH 4D AND IONIC/ANGULAR? IT'S EASY!

- Adding a simple Web and Mobile web interface to a 4D database with basic HTML knowledge.
- Introduction to the Free Ionic Creator design studio.
- Creating a REST interface with a 4D application.
- Coding specific methods to use your Web app (REST API).

FROM NUMBERS TO NAMES

- Moving slowly from Table number, Field number, Process number, Record number, etc. to Names, why and how

FROM STATIC TO DYNAMIC VARIABLES

- Form Object Names.
- Form Objects & Data Sources.
- Extensive use of ListBoxes.
- Callbacks.
- Concepts on Contexts.

FROM PROCESS TO WORKER

- Similarities & Differences between Process and Worker.
- Advantages of Workers.

FROM COOPERATIVE TO PRE-EMPTIVE

- Getting prepared for V16 Multithreading.
- Parallel & Asynchronous programming.
- Pre-emptive Multithreading: How to get prepared.
- Understanding Workers, Processes, and Windows contexts.
- Messaging, Slicing, Chunking, without (too much) Head-scratching.
- How to go on breathing without Inter-process Variables.



Angelo Caroprese
4D SAS Trainer
@Angelo4D

DEVELOP MORE THAN AN APPLICATION, DEVELOP YOUR BUSINESS

