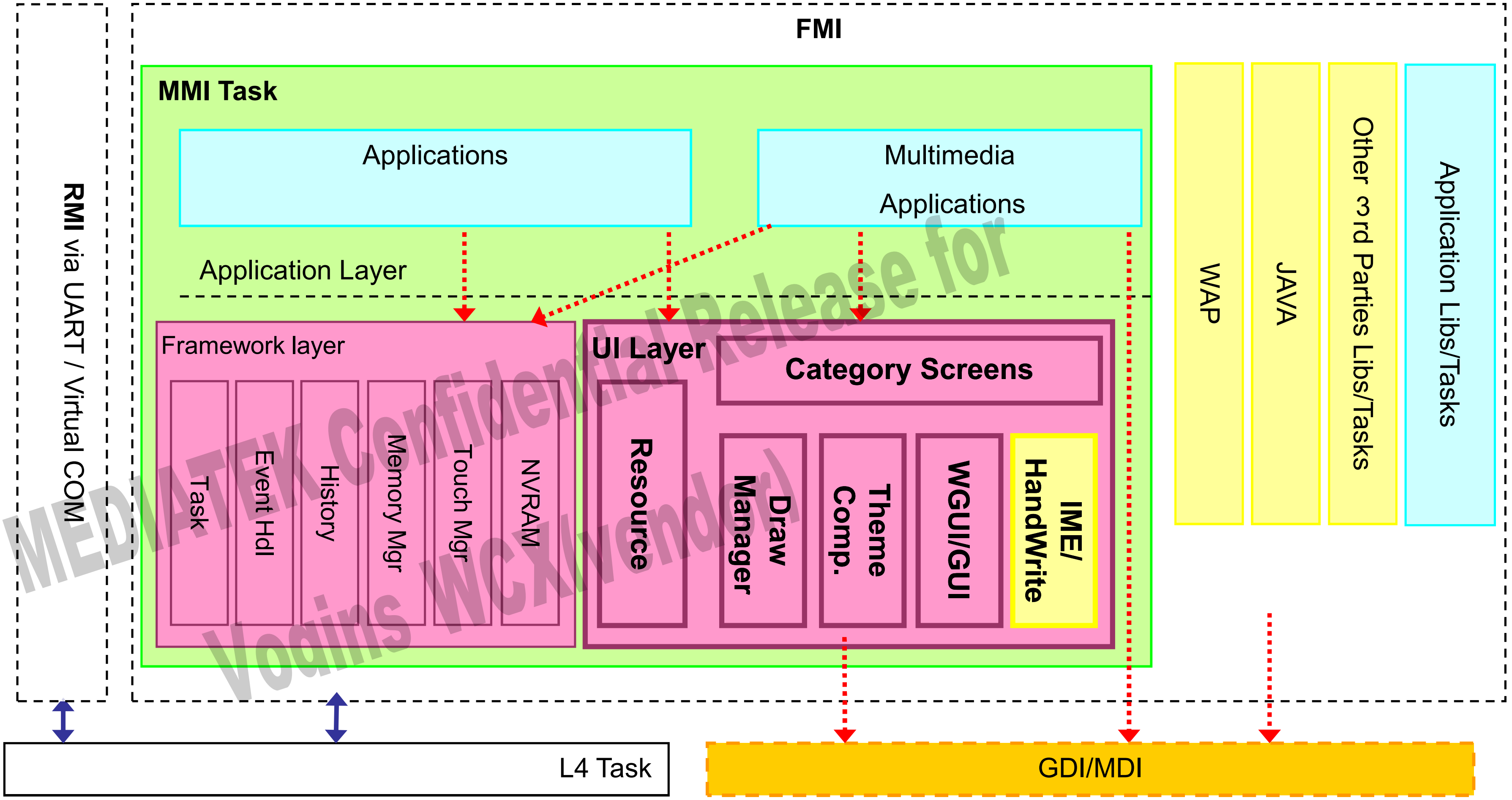
Confidential B

MMI UI Layer

Confidential B

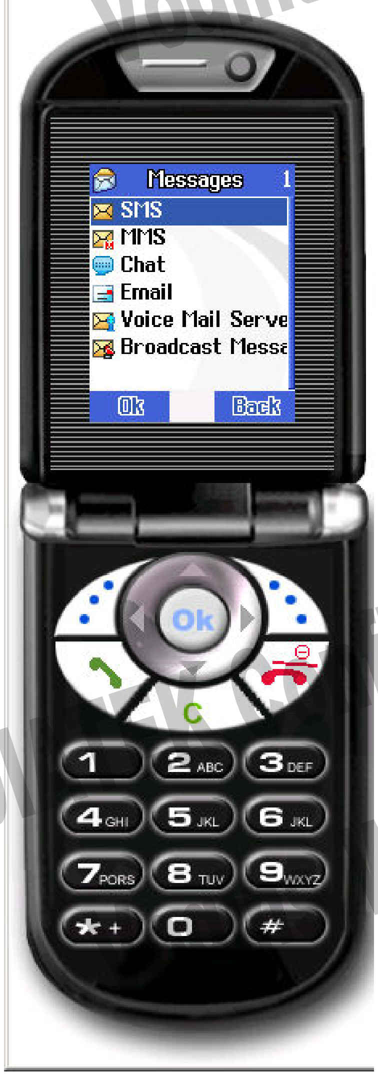
🟨 3rd Party Components
🟦 Native Apps
— Primitive Based Interface
⋯ Function-Call Based Interface

MMI Architecture

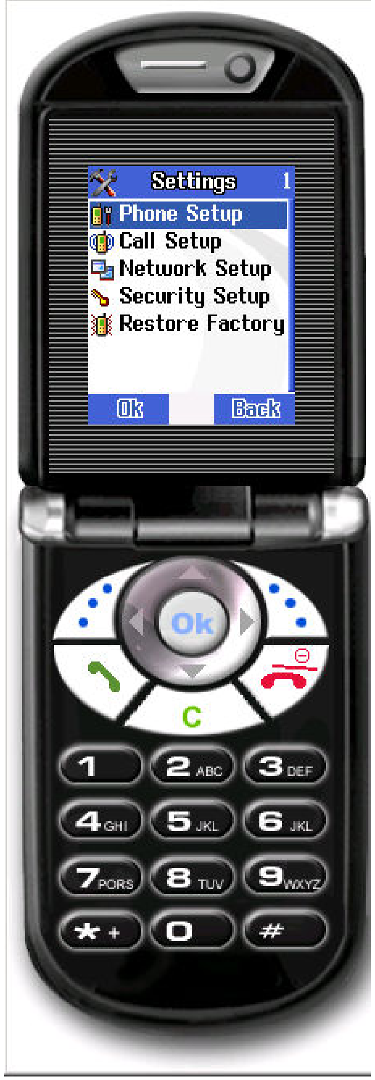


MMI Task – UI Layer

- Provides UI display functions to applications
- Components of UI Layer
  - Category Screens
    - Intelligent wrappers to draw the screens of applications
    - Accept resources such as String IDs and Image IDs from applications.
    - Keep the application independent of the layout and the look-and-feel of screens
    - Provide interfaces of history



The same Category Screen



MMI Task – UI Layer

- Draw Manager
  - Purpose
    - Keep the decision of position and size independent of GUI kernel code
  - Control set
    - Define all components displayed on screen
  - Coordinate set
    - Define position and size of all components
  - Each control set has one default coordinate set

MMI Task – UI Layer

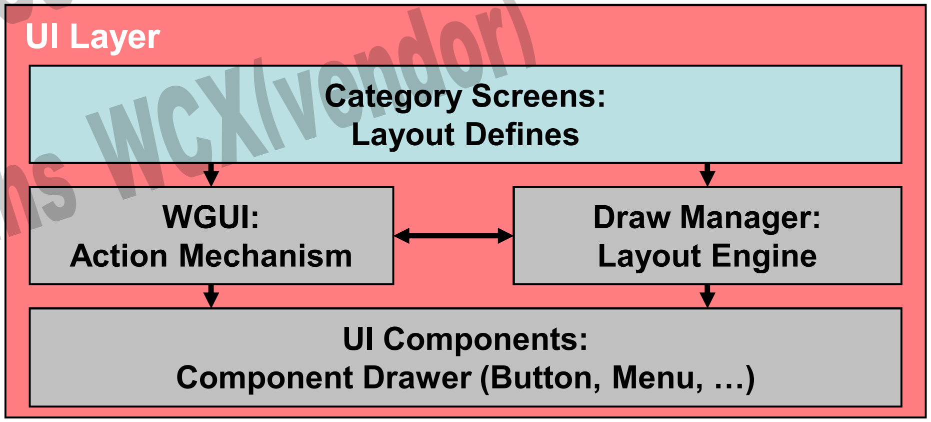
- WGUI
  - UI Element's wrapper: Manage UI element, and allocate memory for UI element.
- UI Elements
  - This layer provides ability to display and manipulate various UI elements such as buttons, scroll bars, menus etc.
  - These are the building blocks of the User Interface
- Themes
  - Responsible for applying fixed set of look-and-feel features on UI elements
  - Themes consist of
    - Object shape and color
    - Font to be used (Font type, style, text color)
    - Other display attributes

MMI Task – UI Layer

- Fonts
  - This is the data that is used by the graphics library to render characters on the display
- Images
  - Set of device independent images used as Icons, Splash screens and Wallpapers
- Graphics Library (GDI)
  - Provides the support for graphics primitives
  - Contains functions to display Fonts and Images

MMI Task – UI Layer

- Category Screen Functions
  - The category layer consists of a set of functions for applications to show its User Interface
  - Each Category screen contains the following functions
    - ShowCategoryxxxScreen
    - ExitCategoryxxxScreen
    - GetCategoryxxxScreenHistory
    - GetCategoryxxxScreenHistorySize



Category Screen Functions

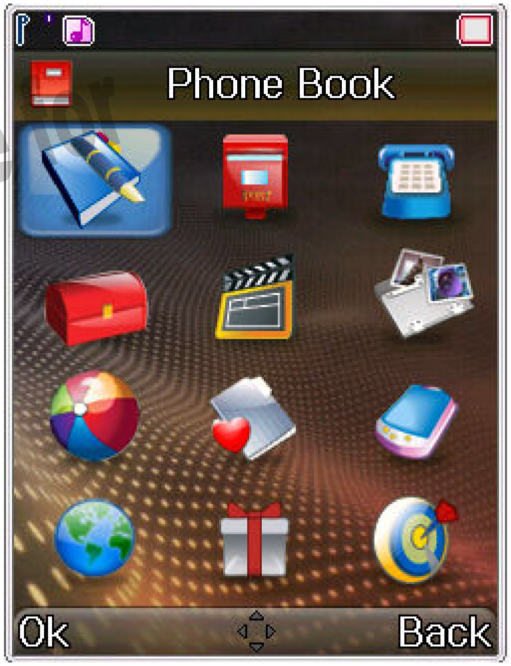
- ShowCategoryxxxScreen
  - Entry function for a Category screen
  - Called by applications
  - Arguments vary from category screen to category screen.

GUI

Category Screen #14: Matrix Menu

mcu\plutommi\mmi\gui\gui_inc\wgui_categories_MM.h

```
void ShowCategory14Screen(
        U16 title,
        U16 title_icon,
        U16 left_softkey,
        U16 left_softkey_icon,
        U16 right_softkey,
        U16 right_softkey_icon,
        S32 number_of_items,
        U16 *list_of_items,
        U16 *list_of_icons,
        S32 flags,
        S32 highlighted_item,
        U8 *history_buffer)
```



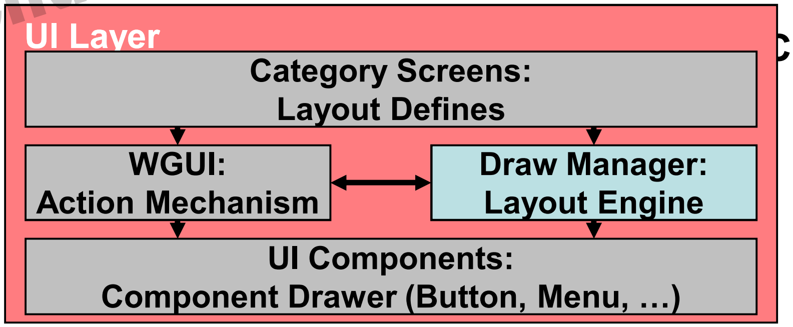
Category Screen Functions

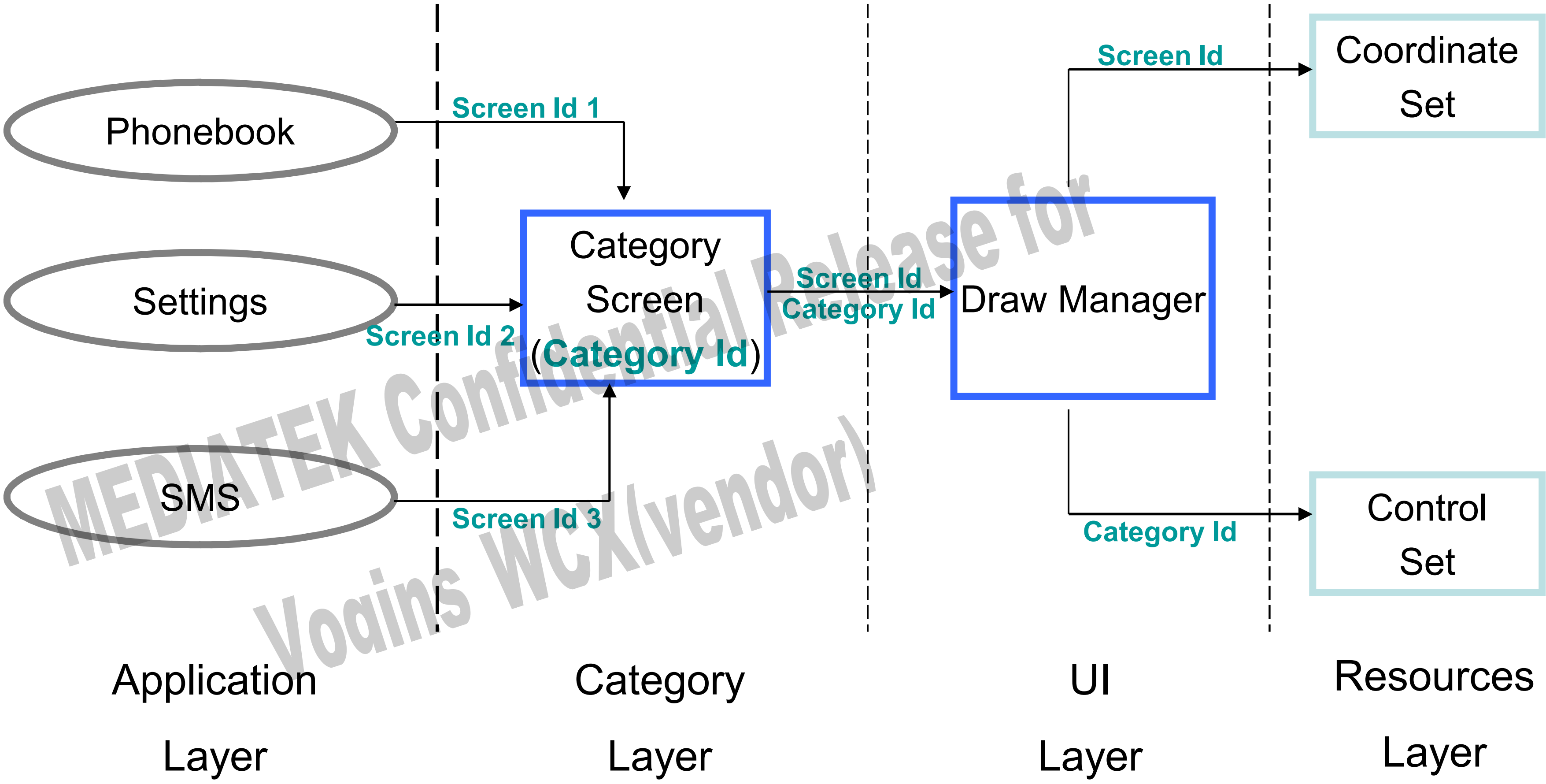
- ExitCategoryxxxScreen
  - Exit function for a Category screen
  - Called by the application when the Category screen is no longer or temporarily not needed
  - Does not clear all the screen
  - Restore some global variables and reset UI components attributes

Category Screen Functions

- GetCategoryxxxScreenHistory
  - Gets the history data of the Category screen
  - Useful in handling Asynchronous events that update the screen
- GetCategoryxxxScreenHistorySize
  - Used to get the size in bytes of the space required to store the history data of the Category screen

Draw Manager





Draw Manager

- dm_redraw_category_screen
  - Replace the redrawcategorysreen function of all category screen
- dm_get_history/dm_get_history_size
  - Replace the getcategoryhistory/ getcategoryhistorysize function of all category screens
- dm_exit_category_screen
  - Replace the exitcategoryscreen function of all category screens

Draw Manager

- Control Set [CustCoordinates.c]
  - Define all components displayed on screen.
    - For each category screen
  - Ex:

```
const U8 category157[]=
{
        5,
        DM_BASE_LAYER_START,
        DM_BASE_CONTROL_SET1,
        DM_NEW_LAYER_START,
        DM_INLINE_FIXED_LIST1,
        DM_CATEGORY_CONTROL_AREA1
};
```

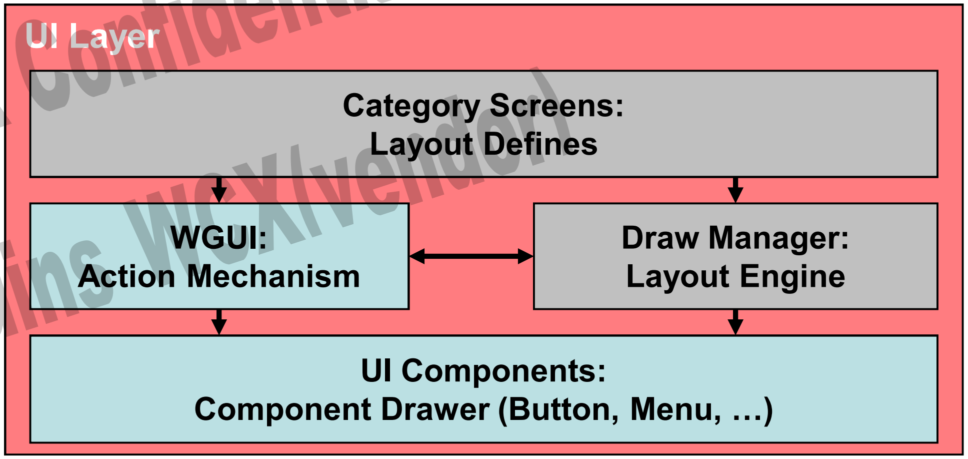
Draw Manager

- Coordinate Set [CustCoordinates.c]
  - Define position and size of all components
    - For different screens
  - Ex:

```
const s16 coordinate_set157[]=
{
            0,0,176,220,DM_NO_FLAGS,
            0,42,220,158,DM_NO_FLAGS,
            0,0,176,158,DM_NO_FLAGS,
            0,0,176,220,DM_NO_FLAGS
};
```

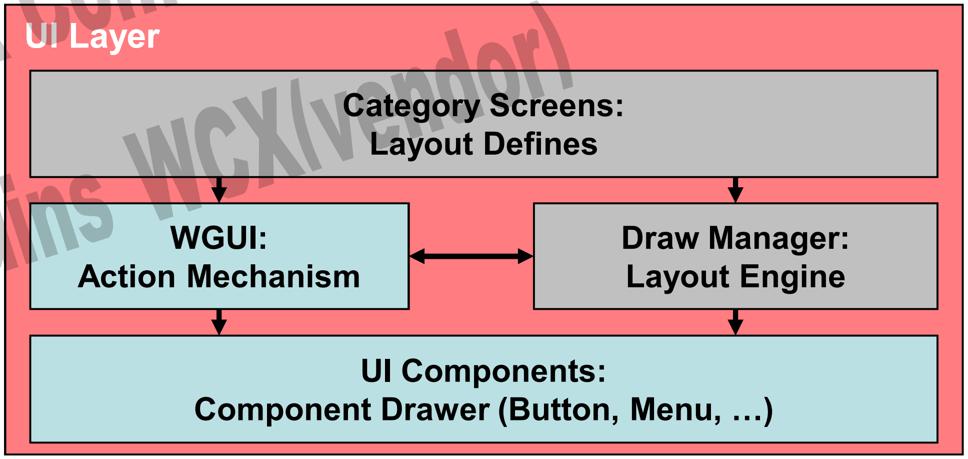
MMI Task – UI Layer

- UI elements
  - These are the building blocks of the user interface
  - They are implemented using data structures and functions
  - The structure contains all the information necessary to display the UI element
  - Provides function to perform operations on UI elements, and take instance as input



MMI Task – UI Layer

- Theme
  - These are data structures that contain values that define the appearance of UI elements
  - Structures contain information for each UI Element such as
    - Fill Color, Fill Style
    - Border Color, Border Style
    - Font used
    - Scrollbar width



MMI Task – UI Layer

- Images
  - Supported types
    - BMP, PBM, BMP sequence: s/w decoder
    - GIF :
      - "restore to previous" NOT support
      - h/w, s/w decoder
    - JPG: h/w decoder
    - PNG
      - above MT6228
      - h/w decoder
    - ABM

MMI Task – UI Layer

- Graphics library (GDI)
  - A graphic interface to perform drawings
  - The low level graphics interface. It support basic 2d graphics drawing (draw line, fill rectangle, draw image…).
  - Take care all hardware acceleration.
  - Detect hardware capable and try to use hardware function if available.

MEDIATEK

www.mediatek.com