

## How everything works

- The server assigns ihab's and hwnd's - threads' and windows' indexes.
- Windows get messages.
- Messages are got from either queues, assigned to ihab's, or directly,
  - if they were sent from the same thread, or indirectly, but without placing into a queue,
  - if they were sent from another thread of a process. Indirectly - that means it calls a window proc from inside WinGetMsg.
- If a message is sent to another process, it is sent via server.
- If message is "Sent" - then it's waiting, if it is "Post'ed", then it put into the process message queue, also inside WinGetMsg

A server by F\_CMD\_GET\_IHAB client's request issues a free ihab, and clientId

## How to write programs

- Let's take a PM program source. The includes are changed from os2.h to F\_OS2.hpp and defines from INCL\_DOS, INCL\_WIN to F\_INCL\_DOS, F\_INCL\_WIN; and #include "PM\_FPM.h" is added. And adding the FreePM client libraries to a project.

## Backward compatibility

- For a FreePM program to be working under PM and if we don't want to change all F\_xxx calls to WinXXX or GpiXXX, we add the \FreePM\lib\PMcompat\PM\_api.cpp file into the project

From:

<http://osfree.ru/doku/> - **osFree wiki**

Permanent link:

<http://osfree.ru/doku/doku.php?id=en:docs:freepm:workflow&rev=1399749538>

Last update: **2014/05/10 19:18**

