

# Programação de Aplicações

---

**VERSÃO 0.1 (INCOMPLETA)**

Isabel Azevedo

# ÍNDICE

<b>1</b>	<b><u>VISÃO GERAL DO SISTEMA</u></b> .....	<b>3</b>
1.1	<u>PERSPECTIVA HISTÓRICA</u> .....	3
1.2	<u>OBJECTOS</u> .....	4
1.3	<u>BIBLIOTECAS</u> .....	5
1.3.1	<i>Acesso qualificado e não qualificado à objectos</i> .....	7
1.3.2	<i>Lista de bibliotecas</i> .....	9
1.4	<u>FICHEIROS</u> .....	10
1.5	<u>MEMBROS</u> .....	12
1.6	<u>JOBS</u> .....	12
1.7	<u>LINGUAGENS DO SISTEMA</u> .....	13
1.8	<u>UTILIZAÇÃO DE MENUS</u> .....	13
<b>2</b>	<b><u>PDM</u></b> .....	<b>17</b>
2.1	<u>TRABALHO COM BIBLIOTECAS</u> .....	18
2.2	<u>TRABALHO COM OBJECTOS</u> .....	20
2.3	<u>TRABALHAR COM MEMBROS</u> .....	20

# 1 Visão Geral do Sistema

---

*Neste capítulo é apresentado o AS/400. Abordam-se alguns conceitos gerais que não existem ou que são ligeiramente diferentes dos que existem noutros ambientes.*

## 1.1 Perspectiva histórica

O AS/400<sup>1</sup> é a arquitectura utilizada pela IBM para sistemas de tamanho médio e foi criado em 20 de Junho de 1988 para substituir a família System/3x, desde o System/3 (1969), passando pelo System/32 (1975), pelo System/34 (1977), System/38 (1978), até ao System/36 (1983).

Tal como os seus antecessores, o AS/400 é um sistema multi-utilizador. O AS/400 reúne as características de facilidade de operação do System/36, com a arquitectura avançada e a produtividade do System/38. Por isto, muitos dos programas desenvolvidos para o System/36 e o System/38 puderam ser migrados e usados nos sistemas AS/400 apenas aplicando as ferramentas de migração disponíveis.

Ao longo dos anos a IBM tem convertido o AS/400 de um sistema “centrado em servidor”, com terminais ligados ao servidor, que tem todos os dados, programas e capacidade de execução, para um sistema “centrado em aplicações”, que já permite que dados e programas possam estar em sistemas separados, e a aplicação (que é composta por dados e programa) pode executar noutro sistema. Por este

---

<sup>1</sup> A designação actual é iSeries.

motivo, hoje o AS/400 é muito utilizado como servidor de base de dados, utilizando a filosofia cliente-servidor. Uma outra vantagem para utilizar o AS/400 como servidor de base de dados é que ele já vem com uma versão da base de dados relacional DB/2 integrada no seu sistema operativo. O AS/400 também é muito utilizado em redes de PC's, uma vez que existem componentes para ligá-lo a praticamente qualquer outro sistema, de PC's a estações RISC com UNIX.

O AS/400, tal como a sua designação sugere (AS de Application System), é um sistema voltado para o desenvolvimento de aplicações, que visa, através das suas funcionalidades como a arquitectura baseada em objectos do sistema OS/400, a base de dados própria do sistema e o armazenamento em único nível de memória<sup>2</sup>, aumentar a produtividade do programador de aplicações.

Em Outubro de 2000, a IBM alterou o nome de AS/400 para iSeries 400 da IBM. Nestes apontamentos será utilizado preferencialmente o termo AS/400.

O OS/400 (sistema operativo para o ambiente AS/400) é fácil de utilizar graças a sua natureza integrada e a sua estrutura orientada por menus. É um dos sistemas operativos mais fáceis de utilizar da sua classe. Existem mais de 28.000 soluções e aplicações comerciais para o AS/400 - [referencia](#).

## 1.2 Objectos

No AS/400 tudo o que existe no sistema AS/400 e que pode ser guardado ou obtido é um objecto. Um objecto combina dados e os métodos válidos para utiliza-los numa só

---

<sup>2</sup> No AS/400 existe um sistema de armazenamento compartilhado no qual todas as porções do armazenamento principal e auxiliar são endereçados como se estivessem numa única área (ou nível). O sistema usa o nome do objecto para determinar onde ele existe no sistema. Assim, o utilizador pode identificar objectos pelo nome, sem mencionar o local onde reside. Quando as operações não podem ser efectuadas num objecto que não esteja na área de armazenamento principal, o sistema move parte ou todo o objecto para a área de armazenamento principal conforme necessário, e move de volta para a área de armazenamento auxiliar quando ele não é mais necessário. Esta transferência é controlada pelo sistema, sem intervenção do utilizador ou programador. Por exemplo, se existirem 300 discos rígidos, para o utilizador é como se houvesse um único.

entidade. Deste modo, os objectos permitem que os utilizadores façam o seu trabalho sem terem de intervir directamente no funcionamento da máquina.

Cada objecto tem características diferentes. Por exemplo, um programa é um objecto que contém instruções, enquanto que um ficheiro (físico) é um objecto que contém dados.

O sistema AS/400 tem uma arquitectura baseada em objectos. Todas as entidades no AS/400 são tratadas pelo sistema operativo como objectos, sejam terminais, impressoras, ficheiros ou ecrans, por exemplo.

Um objecto é construído por um conjunto de atributos que o descreve e um valor. Os atributos de um objecto incluem seu nome, tipo, tamanho, a data em que o mesmo foi criado, uma breve descrição atribuída pelo utilizador que o criou, e o nome da biblioteca no qual o mesmo é armazenado. O nome do objecto e o tipo do objecto são utilizados para a identificação de um objecto. O nome de um objecto é explicitamente indicado pelo sistema ou pelo utilizador quando é criado. O tipo do objecto é determinado pelo comando usado na criação do objecto. Por exemplo, se um programa foi criado e a ele foi dado o nome de PGMXXX, o programa sempre será referenciado por este nome. O sistema operativo OS/400 usa o nome do objecto (PGMXXX) e o seu tipo (programa ou \*PGM) para localizar o objecto e executar operações de cópia, remoção ou quaisquer outras.

### 1.3 Bibliotecas

Bibliotecas são objectos tal como programas e bases de dados. Bibliotecas contém outros tipos de objectos, como bases de dados, ficheiros de sources e programas compilados. Bibliotecas são as estruturas organizacionais principais do AS/400. Também existe a noção de folders - estruturas que contém documentos do Office/400.

Descrição de dispositivos para terminais ou impressoras, bases de dados, programas e ecrans são objectos O sistema operativo do AS/400 (OS/400) guarda descrições de todos objectos.

Uma biblioteca pode conter tudo o que for manipulável no sistema. Uma biblioteca é um grupo de objectos relacionados<sup>3</sup>. Trata-se de um objecto que permite a identificação de grupos de objectos, auxiliando na organização de objectos. Uma das principais vantagens deste conceito é a conveniência: por exemplo, é possível efectuar a cópia de segurança de uma biblioteca inteira com um só comando. Existem as seguintes categorias gerais de bibliotecas:

- ? **Bibliotecas de sistema** (SYS)- Contém objectos de sistema definidos pela IBM, como o sistema operativo, compiladores de linguagens do sistema, utilitários e descrições de outras bibliotecas. Existem duas bibliotecas de sistema: QSYS e QSYS2, e são estas bibliotecas que identificam os objectos necessários a execução do sistema.
- ? **Bibliotecas de utilizadores** (USR)- Guardam objectos definidos pelos utilizadores como bases de dados e programas. Uma biblioteca de utilizador pode ser criada pela IBM ou por utilizadores. Exemplo de uma biblioteca de utilizador criada pela IBM: General Purpose Library (QGPL). Uma biblioteca de utilizador pode ser de um dos seguintes tipos:
  - ? \*PROD – Uma biblioteca de produção, ou seja, de trabalho;
  - ? \*TEST – Uma biblioteca de testes.

A biblioteca corrente é a biblioteca assumida que o sistema utiliza para a criação de objectos. O valor CURLIB (biblioteca corrente) pode ser utilizado na maioria dos comandos para representar uma biblioteca. Se a lista de biblioteca não tiver o nome de uma biblioteca corrente, o sistema utiliza a General Purpose Library (QGPL).

Numa mesma biblioteca, diferentes objectos podem ter o mesmo nome desde que tenham tipos diferentes. Em bibliotecas distintas poderemos ter objectos com o mesmo nome e tipo.

Uma vez definidos, os objectos são manipulados pelo sistema de uma maneira bastante simples. Além disso, o sistema operativo disponibiliza uma grande variedade

---

<sup>3</sup> O conceito é um pouco semelhante as directórias que existem nos sistemas operativos UNIX e Windows, por exemplo, mas uma biblioteca em AS/400 só permite um nível, ou seja, não é possível haver uma biblioteca dentro de outra.

de utilitários para auxiliar o utilizador na definição, criação e manipulação dos objectos.

Dentro da estrutura IFS (Integrated File System) do AS/400 temos a biblioteca QSYS. Na QSYS estão os objectos do sistema operativo. Além disso, esta biblioteca contém as descrições de todas as bibliotecas do sistema. A criação de bibliotecas no AS/400 é importante para a organização de objectos. Um objecto do AS/400 possui vários atributos. Os principais são: Biblioteca, Nome e Tipo.

É importante saber que os objectos não estão fisicamente armazenados numa biblioteca. Uma biblioteca contém nomes e endereços de outros objectos.

Uma biblioteca é um objecto que é usado para agrupar objectos relacionados e para permitir a localização de um objecto pelo seu nome. Todos os objectos do sistema estão contidos em bibliotecas, as bibliotecas do sistema armazenam os programas e funções do sistema operativo. As bibliotecas do utilizador, criadas pelos próprios utilizadores, contém programas do utilizador, documentos, membros e bases de dados.

### **1.3.1 Acesso qualificado e não qualificado à objectos**

O nome de uma biblioteca proporciona um outro nível de identificação para um objecto. Como descrito anteriormente, um objecto é identificado pelo seu nome e pelo seu tipo. A combinação do nome do objecto com o nome da biblioteca é denominada de nome qualificado do objecto. Um nome não-qualificado contém somente o nome do objecto. Para aceder a um objecto não-qualificado o sistema utiliza a lista de bibliotecas (Library List) para procurar o objecto. Caso o nome do objecto e seu tipo não sejam únicos dentro da lista de bibliotecas de pesquisa, a primeira ocorrência da lista é a que será usada pelo sistema.

Na Figura 1 pode-se ver o exemplo de um acesso qualificado a um objecto. O objecto é MYMENU e é especificada a biblioteca onde pode ser encontrado (MYLIB). Neste caso a lista de bibliotecas não é percorrida para se encontrar o objecto, visto que é especificada a biblioteca que o contém.

```
MAIN                                AS/400 Main Menu
System:  HOMEPAGE
Select one of the following:

    1. User tasks
    2. Office tasks
    3. General system tasks
    4. Files, libraries, and folders
    5. Programming
    6. Communications
    7. Define or change the system
    8. Problem handling
    9. Display a menu
   10. Information Assistant options
   11. Client Access/400 tasks

    90. Sign off

Selection or command
====> CALL PGM(MYLIB/MYMENU)

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F23=Set initial menu
(C) COPYRIGHT IBM CORP. 1980, 1999.
```

**Figura 1 – Exemplo de um acesso qualificado a um objecto**

Na Figura 2 pode-se ver o exemplo de uma acesso não-qualificado a um objecto (MYMENU). Note-se que não é especificada a biblioteca que o contém.

```
MAIN                                AS/400 Main Menu                                System:  HOMEPAGE
Select one of the following:

    1. User tasks
    2. Office tasks
    3. General system tasks
    4. Files, libraries, and folders
    5. Programming
    6. Communications
    7. Define or change the system
    8. Problem handling
    9. Display a menu
   10. Information Assistant options
   11. Client Access/400 tasks

    90. Sign off

Selection or command
====> CALL MYMENU

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F23=Set initial menu
(C) COPYRIGHT IBM CORP. 1980, 1999.
```

**Figura 2 - Exemplo de um acesso não-qualificado a um objecto**

Dois objectos com o mesmo nome podem existir na mesma biblioteca, somente se os seus tipos forem diferentes. Entretanto, dois objectos com o mesmo nome e tipo poderão existir em bibliotecas distintas. Lembre-se que se o utilizador se referir a um objecto usando o seu nome não-qualificado, o sistema percorrerá toda a lista de bibliotecas, até encontrar um objecto com o nome e tipo especificados ou até pesquisar todos os objectos da última biblioteca listada na lista de bibliotecas.

### 1.3.2 Lista de bibliotecas

Como já foi referido anteriormente, um objecto pode ser acedido através de um nome qualificado (tanto o nome do objecto quanto o nome da biblioteca são especificados) ou um nome não-qualificado (a biblioteca não é especificada). Se um nome qualificado for utilizado, o sistema operativo irá recuperar o objecto em questão na biblioteca especificada.

Se um nome não-qualificado for especificado, o sistema operativo irá pesquisar na lista de bibliotecas (library list) até encontrar a primeira ocorrência do objecto com aquele nome e daquele tipo, ou então até pesquisar todas as bibliotecas da lista de biblioteca sem encontrar o objecto desejado. Assim, a lista de bibliotecas especifica as bibliotecas que são percorridas quando se pretende encontrar um objecto em particular quando a biblioteca não é mencionada.

As bibliotecas serão pesquisadas de acordo com a ordem pela qual foi montada a lista de bibliotecas. O sistema operativo OS/400 automaticamente cria uma lista de bibliotecas por defeito, incluindo as bibliotecas de sistema, as de produto, a corrente e as de utilizador:

- ? **Bibliotecas de sistema** – São as bibliotecas necessárias para executar funções do sistema, por exemplo, a biblioteca QSYS, a biblioteca QUSRSYS e a biblioteca de help (QHLPSYS). As bibliotecas do sistema são pesquisadas antes das outras bibliotecas que compõem a Library List.
- ? **Bibliotecas de produtos** - A parte de biblioteca de produtos é alterada pelo sistema assim que os utilizadores executam comandos ou utilizam menus. As

bibliotecas de produtos irão variar enquanto um job estiver a ser executado. Por exemplo, se um programa CL também incluir comandos para os compiladores COBOL e Pascal, as bibliotecas produto para aqueles compiladores serão acedidas quando aqueles jobs forem chamados.

- ? **Biblioteca corrente** – É a biblioteca que é utilizada por defeito para a criação de objectos. Por exemplo, os utilizadores poderão especificar a biblioteca corrente para um job através de comandos CL (`CHGCURLIB`), no perfil de utilizador (`CHGUSRPRF`, no programa CL que chama o job, ou no ecran de `sign on`). A biblioteca corrente pode ser uma das bibliotecas da lista de bibliotecas do utilizador ou poderá ser qualquer outra biblioteca para a qual o utilizador tenha autoridade.
- ? **Bibliotecas de utilizador** - A parte de utilizador contém as bibliotecas usadas pelos programas de aplicação para executar suas funções. Quando o sistema é instalado, esta parte contém a biblioteca de propósito geral (QGPL) e a biblioteca temporária de jobs (QTEMP).

Lembre-se que se o utilizador se referir a um objecto através do seu nome não-qualificado, o sistema percorrerá toda a Lista de bibliotecas, até encontrar um objecto com o nome e tipo especificados ou até pesquisar todos os objectos da última biblioteca listada na Lista de bibliotecas.

É possível ter-se dois objectos com o mesmo nome em diferentes bibliotecas. Neste caso, o objecto que estiver na biblioteca com mais alto nível na Library List, terá prioridade para uso.

## 1.4 Ficheiros

Um ficheiro é um objecto numa biblioteca que contém código-fonte ou dados. Por exemplo, todos os códigos-fonte RPGLE para um utilizador podem ser guardados num ficheiro denominado QRPGLSRC dentro da biblioteca do utilizador. O uso do termo ficheiro pode ser um tanto ou quanto confuso, pois pode significar um objecto simples

como uma base de dados ou programa, ou um grupo de membros (reunidos num ficheiro de sources), com códigos-fonte de programas.

Um **ficheiro de dados** AS/400 é um objecto que contém dados. É geralmente compilado a partir de um código fonte DDS (Data Description Specifications), mas pode ser criado a partir de um comando CRTPF. O código fonte DDS contém definições e descrições de campos, com especificação dos campos-chave.

Existem dois tipos de definições de ficheiros de dados disponível para o programador:

- ? Ficheiros físicos
- ? Ficheiros lógicos

Note-se que um ficheiro físico (physical file) não é a mesma coisa que um **ficheiro físico de sources** (source physical file). Ficheiros físicos são objectos que contém registos de dados, enquanto ficheiros físicos de sources são objectos que contém código-fonte a ser compilado. Um ficheiro físico é do tipo \*FILE com o atributo PF.

Um ficheiro lógico não contém realmente dados, mas sim redefinições de um ficheiro físico. Assim, uma vista pode ser criada a partir de uma base de dados (ficheiro físico) que é distinta da sua definição original.

Alem dos ficheiros de dados, há outros dois tipos de ficheiros que serão bastante utilizados nas aulas de Programação de Aplicações:

- ? **Ficheiros de ecran** – Proporcionam ecrans para o utilizador. Tal como ficheiros de bases de dados, os ficheiros de ecran podem ser criados pela introdução de código fonte em membros que depois podem ser compilados, ou pela utilização de um utilitário do sistema: SDA. De facto, o sistema não faz distinção entre ficheiros de ecran e ficheiros de bases de dados. O código fonte de um ficheiro de ecran é tratado tal como uma DDS.
- ? **Ficheiros de programa** – Um programa é criado pela introdução de código fonte num membro de um ficheiro físico de fonte como QRPGSRC utilizando o SEU e depois compilando o código-fonte, que cria um objecto executável.

O AS/400 verifica a existência de qualquer referência a ficheiros externos (como bases de dados ou ecrans) durante a compilação para verificar se existe alguma referência a ficheiros ou campos não definidos.

## 1.5 Membros

A maior parte do trabalho realizado por um programador faz-se com membros de ficheiros físicos de sources.

Um membro contém código-fonte e é guardado dentro de um ficheiro. Um ficheiro de source (source file) pode conter um número de membros que são código-fonte RPG prontos a serem compilados. Assim, um código-fonte de um programa é um membro de um ficheiro que pode conter outros membros de código-fonte do mesmo tipo. Os tipos de membros mais comuns são:

- ✍ PF - Ficheiro físico (physical file)
- ✍ LF - Ficheiro lógico (logical file)
- ✍ DSPF - Ficheiro de ecrans (display file)
- ✍ PRTF - Ficheiro de relatórios (printer file)
- ✍ RPGLE - Programa RPGLE
- ✍ CLP - Programa de controle (control language program)

## 1.6 Jobs

Job é a unidade básica de processamento no OS/400, trata-se de algo em execução num subsistema. Um job é um conjunto de processos que podem ser pedido através de comandos da linguagem de controlo ou através de um ou mais programas. Existem vários tipos de jobs, separados entre jobs do sistema e do utilizador. Entre os de utilizador podemos ter, por exemplo, jobs interativos, batch, de comunicação, etc. Os jobs batch utilizam um subsistema já definido no OS/400, que é o QBATCH.

## 1.7 Linguagens do sistema

O sistema AS/400 suporta todas as linguagens standards do SAA (Systems Application Architecture), ou seja, um programa escrito para um dos seguintes sistema: 370, PS/2 ou AS/400, funcionará sem problemas em qualquer dos outros dois. Algumas das linguagens são as seguintes:

- ? **RPG/400** - Nasceu no antigo sistema 36/38. A orientação por colunas de um programa RPG (Report Program Generator) torna algo difícil a sua leitura e manutenção, mas trata-se de uma alternativa poderosa e breve.
- ? **COBOL/400** - É uma linguagem bastante parecida com o Inglês, ou seja, é fácil de ler, mas trata-se de uma linguagem bastante "palavrosa".
- ? **C/400**
- ? **SQL/400**

Além das linguagens que fazem parte do SAA, existe uma variedade de outras disponíveis para AS/400, como Java, ILE RPG, ILE C, VisualAge C++, O VisualAge C++, por exemplo, proporciona um ambiente de programação visual para o desenvolvimento de aplicações C++ que funcionam em Windows 95/NT, OS/2 ou no próprio AS/400. As aplicações podem ser construídas de forma a acederem aos dados do AS/400, processar a lógica em ambos os sistemas e ter uma interface gráfica, a fim de explorar a capacidade gráfica de uma estação Windows ou OS/2, e a integridade de dados do sistema AS/400.

## 1.8 Utilização de menus

O sistema AS/400 pode ser utilizado através da selecção de opções apresentadas em menus. Mas para se aceder ao sistema é necessário ter-se um perfil de utilizador que é validado através de uma palavra-passe. Na Figura 3 é apresentado o ecran inicial que faz a validação de um utilizador. É obrigatória a especificação do utilizador e da

palavra-passe, mas não o programa que se quer trabalhar logo após a validação, o menu inicial ou a biblioteca actual (e geralmente não são especificadas).

```

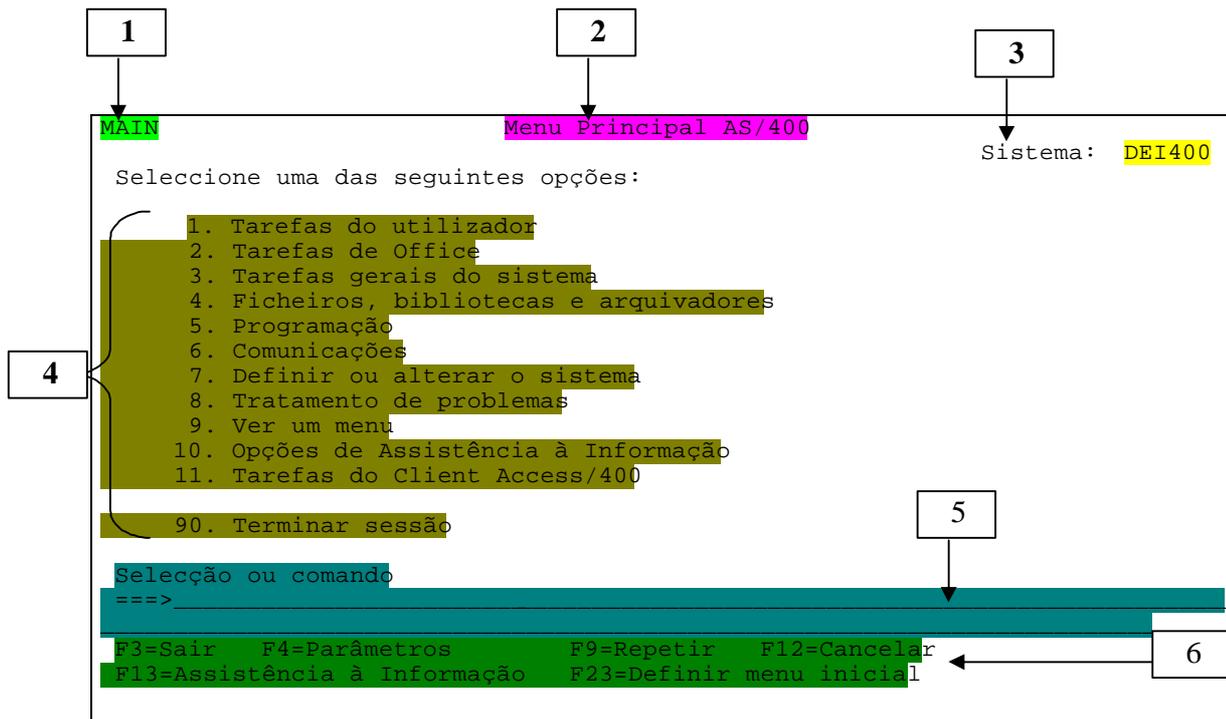
                                Iniciar Sessão
                                Sistema . . . . . : DEI400
                                Subsistema. . . . . : QBASE
                                Terminal. . . . . : QPADEV000J

Utilizador . . . . . _____
Palavra-passe . . . . . _____
Programa/procedimento . . . . . _____
Menu . . . . . _____
Biblioteca corrente . . . . . _____

(C) COPYRIGHT IBM CORP. 1980, 2000.
```

**Figura 3 – Ecran inicial de validação de utilizadores**

Na Figura 4 é apresentado o menu inicial do AS/400, a partir do qual podem ser executados comandos e utilitários através da selecção das opções que vão sendo apresentadas. Todos os menus têm a mesma estrutura, razão pela qual este é apresentado com algum detalhe (as cores servem apenas para tornar mais fácil a explicação).



**Figura 4 – Menu principal do AS/400**

Na parte superior do ecran estão disponíveis as seguintes informações:

- ? **1** – Nome do menu tal como designado pelo sistema. Em qualquer altura pode-se aceder a determinado menu, simplesmente escrevendo-se o seguinte comando CL: GO <nome\_do\_menu>
- ? **2** – Descrição ou título do menu
- ? **3** – Identificação do sistema
- ? **4** - Opções disponíveis e que podem ser escolhidas pelo utilizador
- ? **5** – Linha de comandos - é a linha que permite a introdução de comandos ou a selecção de uma das opções apresentadas
- ? **6** – Funcionalidades disponíveis através das teclas de função - Na parte inferior do menu estão as teclas de funções que podem ser utilizadas e que têm não só neste écran inicial, mas em todos os ecrans, funcionalidades atribuídas, que em geral são as seguintes:
  - ? F3 – Permite voltar para o nível de menus anterior
  - ? F4 - Fornece alguma ajuda na entrada de comandos, mostrando todas as opções possíveis para determinado comando
  - ? F5 – Actualiza a informação disponível

- ? F9 - Mostra o comando executado anteriormente ou os vários comandos executados anteriormente, de acordo com o número de vezes que se carrega nesta tecla
- ? F10 - Comando de entrada
- ? F12 - Permite voltar para o menu anterior (com a tecla de função F3, vai-se vários menus para trás e não um apenas).
- ?

## 2 PDM

---

O PDM (Programming Development Manager) é uma ferramenta do AS/400 que facilita o controle de bibliotecas, objectos e membros, de uma forma rápida e eficiente. O PDM integra um grupo de ferramentas de apoio do AS/400 conhecidas como ToolSet/400.

Utilizando PDM o utilizador poderá dispor dos seguintes utilitários ou facilidades, que por sua vez pertencem também ao ToolSet/400:

- ? Source Entry Utility (SEU)
- ? Data File Utility (DFU)
- ? Screen Design Aid (SDA)
- ? Report Layout Utility (RLU)

Para iniciar o PDM utiliza-se o seguinte comando: `strpdm`. O ecran inicial da ferramenta PDM pode ser visualizado na Figura 5. A partir deste ecran podem ser iniciadas as ferramentas mencionadas anteriormente.

```
AS/400 Programming Development Manager (PDM)

Select one of the following:

1. Work with libraries
2. Work with objects
3. Work with members

9. Work with user-defined options

Selection or command
===>

F3=Exit      F4=Prompt      F9=Retrieve      F10=Command entry
F12=Cancel   F18=Change defaults
```

**Figura 5 – Ecran inicial da ferramenta PDM**

## 2.1 Trabalho com bibliotecas

Na Figura 6 está representado o ecran que permite trabalhar com bibliotecas através do PDM:

```
Specify Libraries to Work With
Type choice, press Enter.
Library . . . . . *LIBL          *LIBL, name, *generic*, *ALL
                                *ALLUSR, *USRLIBL, *CURLIB
F3=Exit      F5=Refresh      F12=Cancel
```

**Figura 6 – Manipulação de bibliotecas através do PDM**

Neste ecran pode-se introduzir a biblioteca com a qual se pretende trabalhar (o valor do parâmetro Library) de diferentes formas:

- ? \*LIBL - Mostra a lista de bibliotecas do utilizador
- ? \*ALLUSR - Mostra unicamente as bibliotecas do utilizador
- ? \*ALL - Mostra todas as bibliotecas
- ? \*USRLIBL - Mostra todas as bibliotecas que não pertencem ao sistema
- ? \*CURLIB - Mostra a biblioteca definida como corrente
- ? Library Name - Mostra somente a biblioteca especificada
- ? \*generic\* - Mostra as bibliotecas cujo nome obedeça a especificação apresentada, com \* a representar qualquer conjunto de caracteres.

```

Work with Libraries Using PDM                                     S106CB4A
List type . . . . . *LIBL
Type options, press Enter.
  2=Change          3=Copy          5=Display          7=Rename
  8=Display description  9=Save          10=Restore         12=Work with
...
Opt  Library      Type      Text
---  ---          ---      ---
---  QSYS         *PROD-SYS System Library
---  QSYS2        *PROD-SYS System Library for CPI's
---  QHLPSYS      *PROD-SYS
---  QUSRSYS      *PROD-SYS
---  QPDA         *PROD-PRD
---  QGPL         *PROD-USR
---  QTEMP        *TEST-USR
                                           Bottom
Parameters or command
===>
---
F3=Exit          F4=Prompt          F5=Refresh          F6=Add to
F9=Retrieve      F10=Command entry F23=More options   F24=More keys

```

**Figura 7 – Trabalho com bibliotecas através do PDM**

Como se pode observar na Figura 7, as seguintes opções estão disponíveis para se trabalhar com bibliotecas:

- ? 2 (Change) - Permite alterar o tipo de biblioteca (\*PROD - biblioteca que se encontra em produção / \*TEST - biblioteca de testes), bem como alterar a sua descrição de texto.
- ? 3 (Copy) - Possibilita a cópia de uma biblioteca para outra de todas as suas características.
- ? 4 (Delete) - Permite eliminar uma ou mais bibliotecas
- ? 5 (Display) - Mostra todos os tipos de objectos contidos na biblioteca.
- ? 6 (Rename) - Permite alterar o nome de uma biblioteca.
- ? 8 (Display description)- Mostra informação sobre a biblioteca: data de criação e da última alteração, por exemplo
- ? 12 - Mostra os objectos contidos na biblioteca.
- ? 13 - Altera a descrição da biblioteca.



- ? Library: É a biblioteca que contém o ficheiro de sources especificado
- ? Member:
  - o Name : Nome da membro à trabalhar, pode ter um dos seguintes valores:
  - o \*ALL: Todos as membros
  - o Name: O nome da membro com o qual se pretende trabalhar.
  - o Type: Tipo de membro.

```

                                Specify Members to Work With
Type choices, press Enter.
File . . . . . JDEMTY      Name, F4 for list
Library . . . . . MTY1      *LIBL, *CURLIB, name
Member:
Name . . . . . P43500      *ALL, name, *generic*
Type . . . . . *ALL        *ALL, type, *generic*, *BLANK

F3=Exit      F4=Prompt      F5=Refresh      F12=Cancel

```

**Figura 9 – Ecran Trabalhar com membros**

É possível procurar uma string num membro de duas formas distintas:

- ? Utilizando-se o PDM: Trabalhar com membros (WRKMBRPDM) e escolher a opção 25 (find String) para achar a string desejada.
- ? Utilizando-se o comando FNDSTRPDM