

Direção de Sistemas e

Tecnologias de Informação

NOTAS DE VERSÃO NEW TERMINALS

# Componente Millennium

|  |  |
| --- | --- |
| Sistema: | Millennium |
| Referência: | DD\_MIL\_NOTAS\_VERSÃO\_NEWTERM\_20201215 |
| Data: | 15-12-2020 |
| Versão: | V01 |
| Autor: | DISTI/GAJ |
| Descrição: | Notas de versão da componente Millennium do projeto Novos Terminais. |

Histórico do Documento:

|  |  |  |  |
| --- | --- | --- | --- |
| Versão | Data | Responsável | Descrição de Alterações |
| V.01 | 15-12-2020 | DISTI/GAJ | Criação do documento |
| V.02 | 23-09-2021 | DISTI/GAJ | Informação mais detalhada das alterações |

1. **Informação de Desenvolvimento**

|  |  |
| --- | --- |
| Sistema / VERSÂO | MILL /01 |
| DESCRIÇÃO | Alterações ao sistema de jogo Millennium no âmbito do projeto Novos Terminais. |
| COMENTÁRIOS | Não há prioridade na instalação das máquinas. |
| DETALHES de INSTALAÇÃO | 1. Após o STOPSYS do sistema Millennium; 2. Instalar em todas as máquinas do sistema Millennium os novos executáveis resultantes do LOTGEN (executáveis e memórias); 3. Instalar o script MESSAGEQCONNECTION.COM no diretório GXOLM; 4. Configurar em cada máquina do sistema Millennium o valor dos novos parâmetros globais de sistema através do programa BLDSYS (parâmetro n.º 143). |
| FICHEIROS ALTERADOS | Consultar anexo |
| ANEXOS | DD\_MIL\_SOURCE\_CODE\_LIST\_NEWTERM\_20201215\_V01.xlsx |

1. **Notas**

As alterações implementadas foram desenvolvidas no âmbito do projeto Novos Terminais.

Neste projeto foram ainda incluídas as alterações de código fonte decorrentes das alterações implementadas e do novo Programa COMOLM.

Dada a natureza das alterações foi necessário realizar o LOTGEN de todo o código fonte do sistema Millennium.

Foram modificadas as seguintes estruturas de dados:

* Ficheiro de configuração do sistema Millennium, SCF.FIL:
* Definição de novos parâmetros globais de configuração relativos ao novo programa COMOLM:
  + OLMCONF – Estado da ligação do sistema Millennium ao sistema Olimpo (através do MessageQ controlando a leitura/ou não leitura de mensagens na sua Queue “*MESSQ\_GET*”).
* Ficheiro TMF de transações do sistema Millennium:
* Criação de novos campos nas transações do tipo LOTT, SPORT e IL para além do Sign-On (estes são 3 novos campos Transaction Serial Number do Olimpo, Message ID, Channel Communication flag)
* Aumento de segmentos nas transações LI para os casos ILOT, ICAR, IORD, INMU que se encontram nos ficheiros e subroutines: DILOT, DICAR, DIORD, DINMU (nota: ver no fim do ficheiro onde está a lógica atualizar o SIZE).
* Alterações nos ficheiros SPESRV.FOR, INSPRO.FOR, WAGPRO, CANPRO, VALPRO para o TRABUF receber os novos campos do buffer (SEROLM\_OLM, CHOLM\_OLM, MESSID\_OLM) para ser guardado no TMF pelo TRALOG.
* Foi criada a seguinte memória partilhada:
  + GXOLMCOM.ESH - memória que permite partilhar dados relacionados com MessageQ entre os vários programas e tarefas aplicacionais do sistema de jogo Millennium, especificamente VISION.

1. **Código fonte**

Os ficheiros de código fonte alterados encontram-se listados no ficheiro ***MILL-NewTerminals2021-Changes***.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tipo de Ficheiro | Extensão do Ficheiro | Total de Ficheiros New Terminal Proj | | |
| **Novos** | **Modificados** | |
| Script DCL | .COM | 1 | | 2 |
| Ficheiro de Definições de Código Fonte Fortran | .DEF | 1 | | 6 |
| Ficheiro de Código Fonte Fortran | .FOR | 4 | | 23 |
| Ficheiro de Código Fonte C | .C | 1 | | 0 |
| Ficheiro de Linkagem EXE | .LNK | 1 | | 2 |
| Ficheiro de Linkagem ESH | .LSH | 1 | | 0 |
| Ficheiro de Configuração de Script DCL | .NAM | 0 | | 1 |
| Ficheiro de Opções de Linkagem | .OPT | 0 | | 1 |
|  | Total | 9 | | 32 |

Nota: Dada a natureza das alterações foi necessário realizar o LOTGEN de todo o código fonte do sistema Millennium.

1. **Programas/Executáveis**

Programa BLDSYS

Foi criado o parâmetro global de sistema Millennium nº 143 (*opção* ***1 – Change a global system parameter***) depois escolher ***m*** *de more* até chegar ao fim da lista que é a opção ***143 OLMCONF*** e alterar com o comando ***OLMCONF 1*** para ficar ativo a leitura por default.

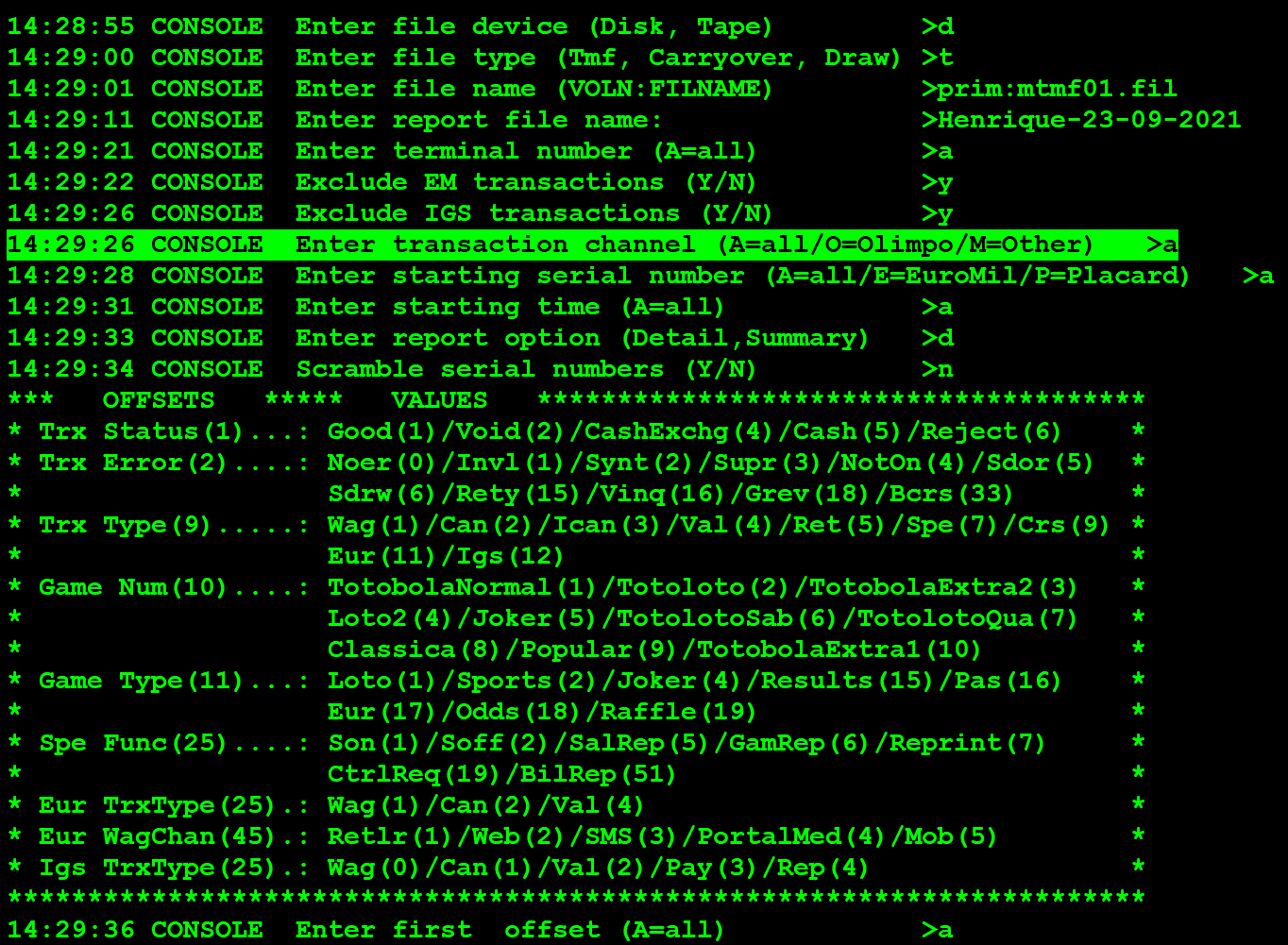
|  |  |  |
| --- | --- | --- |
| N.º do Parâmetro  Global | Nome | Função |
| 143 | OLMCONF | Ligar ou desligar a leitura de mensagens do MessageQ provenientes do Hades |

Programa TSTLOG

Este programa utilitário foi atualizado na sequência de incorporar as várias transações dos jogos LOTT, SPORT e LI que ainda estavam em falta . A sua função é testar as rotinas de escrita e leitura das transações do ficheiro TMF.

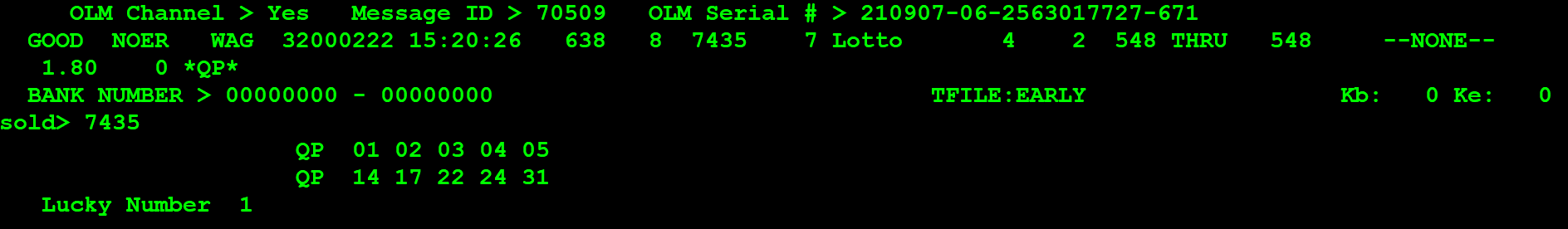
Programa TMIR

Em resultado das alterações à composição das transações do tipo LOTT, SPT e LI, o programa Tmir foi alterado para ler os novos campos e a sua escrita nos relatórios gerados. Foi adicionado um novo filtro que possibilita filtrar transações do novo canal de comunicação ou filtrar dos canais antigos de comunicação.



As modificações de que o programa foi alvo são as seguintes:

* Introdução de um novo parâmetro de entrada no programa que permite excluir todas as transações provenientes do novo canal OLM\_COM ou provenientes dos canais antigos X2X\_COM e MXS\_COM.
* Apresentação de 3 novos campos nas Transações dos Jogos LOTT, SPT e LI que são Serial Number, Message Id, Channel.



Programa VISION

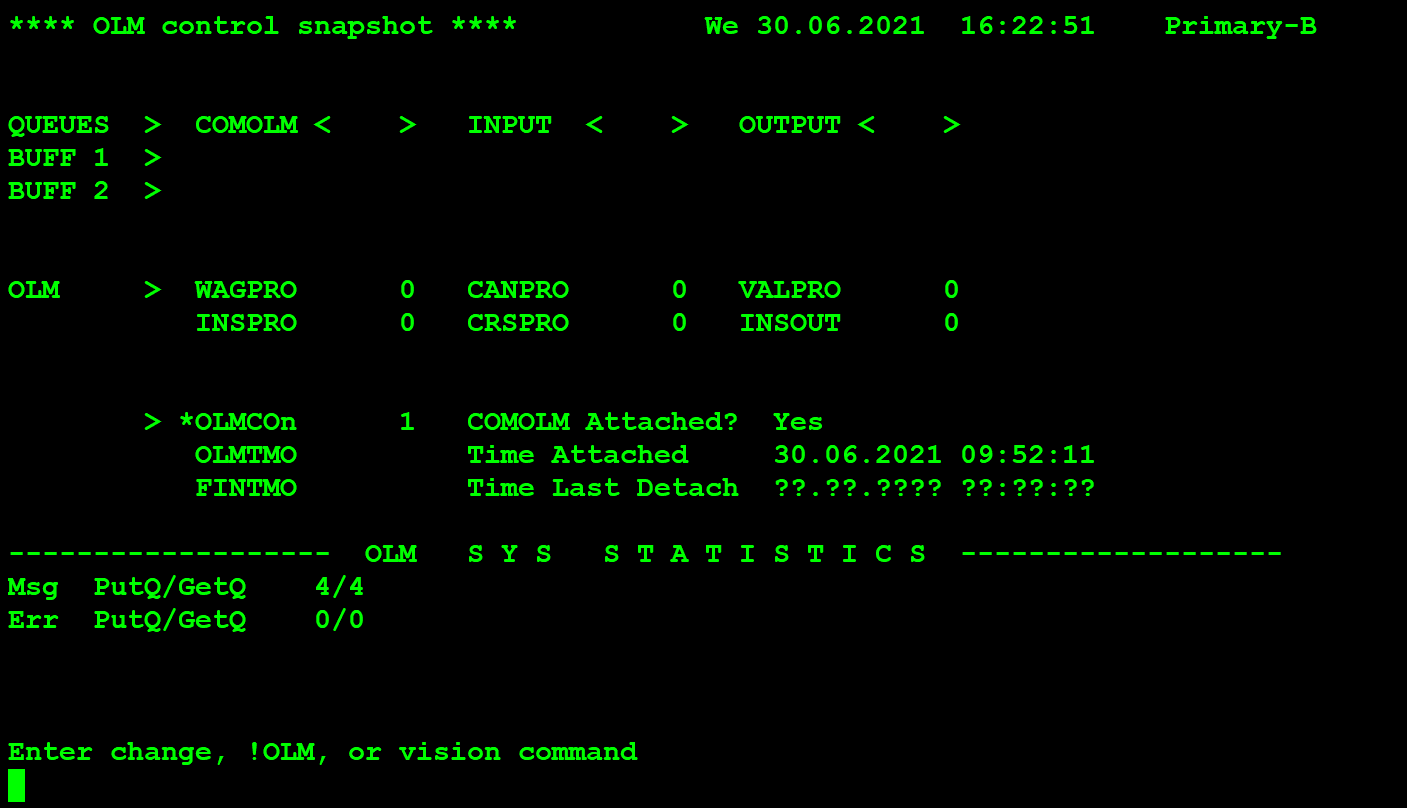
No programa VISION para a monitorização dos parâmetros troca de mensagens do MessageQ do Hades com Millennium foi criado uma nova tela ***OLMsnp***.

* P(OLMCONF) é uma variável de sistema, esta variável não controla nem faz detatch da MessageQ mas sim controla se lê ou não mensagens do MessageQ através da função **GETFROMOLM** como analogia é um function supress.
* OLMS\_ATTATCHS é uma variável normal que se encontra na memória **GXOLMCOM.ESH** (olmcom.def) esta variável sim indica se **COMOLM** está ou não ligado ao MessageQ.
* Estas duas variáveis encontram-se a ser lidas na tela **OLMsnp** onde se vê os seus estados em que **P(OLMCONF)** o seu estado pode ser alterado na própria tela com os seguintes comandos:

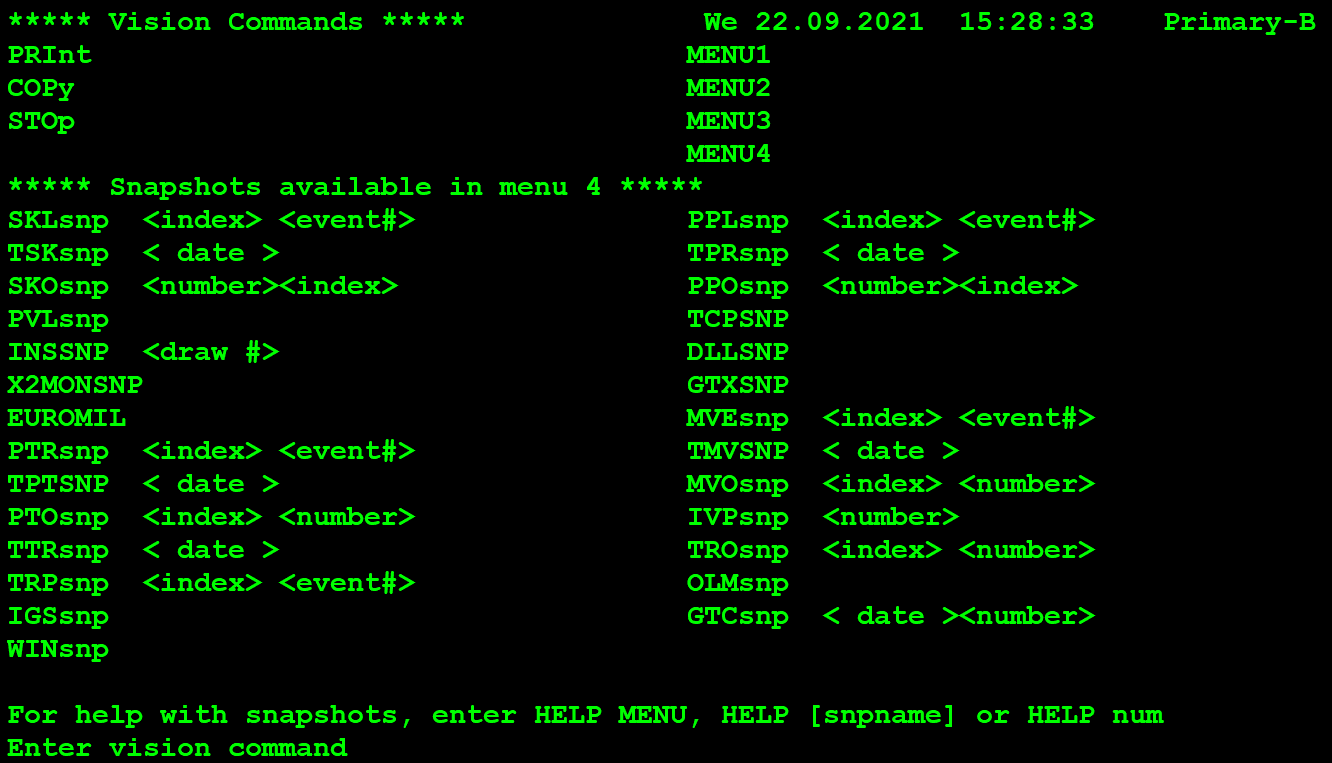
**olmcon 1** (COMOLM lê as mensagens da MessageQ)

**olmcon 0** (COMOLM não lê as mensagens da MessageQ)

Nota: Para concluir quando OLMCOn, ou seja, P(OLMCONF) está a 0 não quer necessariamente que o OLMS\_ATTACHSTS esteja a 0 ou seja “COMOLM Attached? No”.



* Na tela é possível ver os buffers em uso por 3 queues aplicacionais OLM, INQUE, GAME\_OUTQUE para além do contador total.
* É possível ver o contador de mensagens em várias queues aplicacionais que as transações passam provenientes do Olimpo.
* Informação de Mensagens lidas com sucesso do MessageQ ou que tenham falhado, tal como a data e hora que foi feita a última ligação ou que se tenha perdido ligação.
* Para ter em conta da existência de uma nova tela foi adicionado ao Menu4 do Vision o comando para abrir a nova tela.

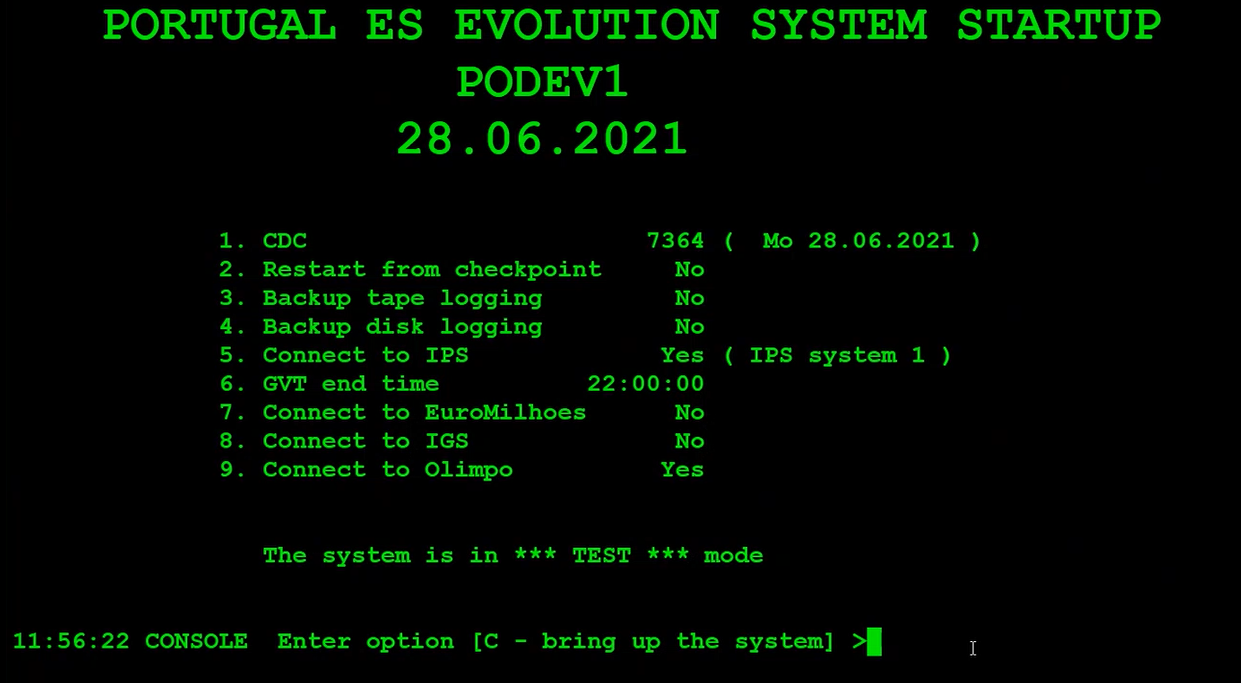


Programa RESET

Programa que inicializa as memórias para processamento transacional das tarefas aplicacionais e comunicações.

Este programa é executado através do script RUNSYS.COM ou RUNTST.COM (sendo estes dois scripts iguais exceto que o de Testes imprime uma label nos bilhetes obtidos do terminal a indicar que é para efeitos de teste).

Este programa foi alterado para alterar o novo parâmetro do sistema **OLMCONF** de modo a ligar ou desligar a leitura de mensagens no MessageQ provenientes do Hades que é a opção 9 “Connect to Olimpo”.



No RUNSYS de modo a arrancar com o novo programa COMOLM foi alterado a subroutine XRUNTSK\_DET que passou a incluir no arranque dos processos o logical name OLMLIB de modo a arrancar com o COMOLM que não se encontra na pasta default TSK mas sim numa subpasta OLM (TSK.OLM).

Programa COMOLM

Programa criado especificamente para a comunicação com o Olimpo através do uso de MessageQ, recebe os pedidos de provenientes do Hades com a leitura das mensagens no MessageQ estas vêm no formato Hexadecimal já como vêm dos canais X2X\_COM e MXS\_COM.

Neste novo canal foi introduzido 3 novos campos:

Transaction Serial Number (proveniente do Olimpo): YYMMDD-GG-NNNNNNNNNN-KKK

* YYMMDD - ano mês e dia da transação
* GG - representa o jogo
* NNNNNNNNNN - Numero interno de série do Millennium
* KKK – check digits

Para este campo foi analisado o espaço necessário para ser guardado como um todo e como composto pelos seus subcampos.

Para campo todo do Serial é necessário pelo menos 9 Bytes:

999999999999999999999 = 69,760489992634609305275265324236 bits = 8,73 bytes -> 9 bytes

Para guardar os subcampos do Serial e deste modo obter-se o Serial são necessários pelo menos 11 Bytes:

YYMMDD = 991231 = 19,918861781817377011657710110785 bits = 2,49 bytes -> 3 bytes

999999 = 19,931567126628411850257166027671 bits = 2,49 bytes -> 3 bytes

301231 = 18,200510721355401033559399955287 bits = 2,28 bytes -> 3 bytes

NNNNNNNNNN = 9999999999 = 33,219280948729353974607084478953 bits = 4,16 bytes -> 5 bytes

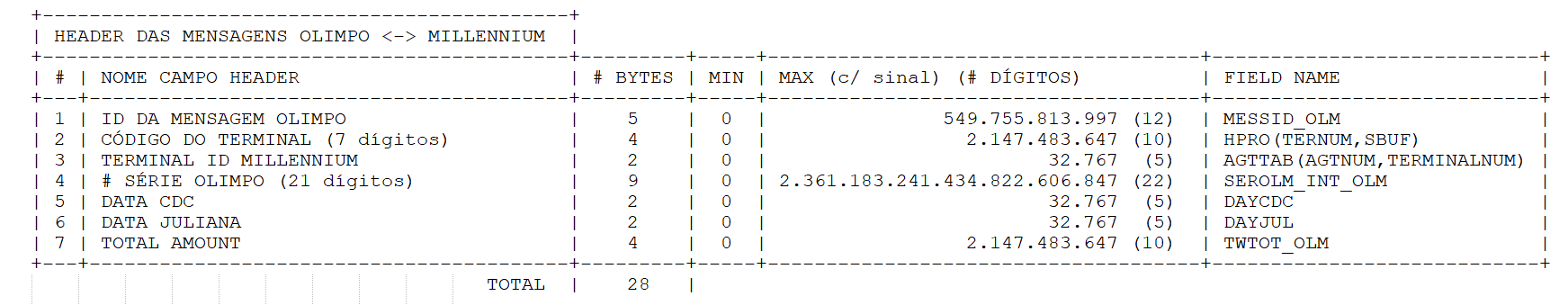
KKK = 999 = 9,9643408677924183262190210094566 bits = 1,25 bytes -> 2 bytes

GG = 99 = 6,6293566200796096191068409346214 bits = 0,83 bytes -> 1 byte

Total: -> 11 bytes

Conclusão optou-se por guardar o Campo Serial como um todo pois só ocupa 9 em vez dos 11 Bytes.

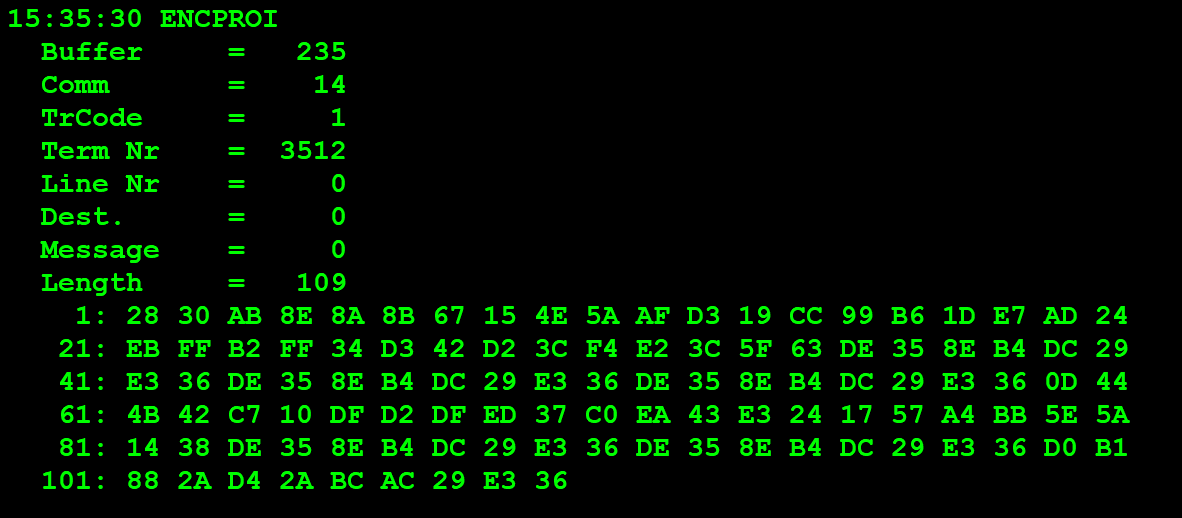
* Mensagens compostas por um novo Header na comunicação com o Hades:



* O body da Mensagem de comunicação com o Hades é composto pelo Header e Body normal que se encontra na documentação ***Terminal Message Formats\_3.22.doc***.
* Novo canal de comunicação foi criado para indicar a proveniência das novas mensagens, que se chama OLM\_COM.
* Fluxo de leitura e escrita na MessageQ é igual ao do IGS e EUR que se priorizasse a escrita em relação há leitura de novos pedidos, havendo também um período de 250ms de espera para ver se existe novas mensagens para ler após queue ter estado vazia.

Nota: antes de esperar os 250ms ainda volta a validar mais uma vez se existe alguma mensagem pendente de ser enviada que, entretanto, tenha sido processada por MILL.

* Para a mensagens respeitarem o formato já existente das diferentes transações do Millennium que passam a vir do novo canal foi feito uma extensa análise com o uso da documentação ***Terminal Message Formats\_3.22.doc.***
* Para os efeitos da análise replicou-se os pedidos com o uso do terminal code mais conhecido por agente number 0300006 terminal físico ligado ao ambiente SCML na máquina PODEV1.
* Para a leitura dos dados recebidos no Millennium foi ativado o modo debug através do programa utilitário **SETDEBUG** e filtrou-se a sua ativação para o terminal id **3512** que corresponde ao agente number 0300006, obtendo dos ficheiros gerados **GTECH$DEBUG.DAT** na pasta TSK que se localiza em gxtsk.



* Numa primeira fase para conseguir-se ver os valores corretos foi desativado a encriptação (**DESFlg=1**) na tela/snapshot **SYS**.
* Em caso de falha no envio/escrita de uma mensagem no MessageQ é logado a mensagem no seu formato Hexadecimal no ficheiro de logs aplicacional que se encontra em GXOLM ficando com o nome do processo neste caso é SCMLCOMOLM.LOG.

Nota: só depois do processo estiver parado é que se pode efetuar a leitar do ficheiro de logs aplicacional.

Programa ERRLOG

Tarefa aplicacional que apresenta no log de consola do sistema Millennium vários tipos de mensagem (relativas a sistema, a comandos, a comunicações, ao sistema Euromilhões, etc.).

Nesta nova versão verifica-se o seguinte:

* Criado uma nova mensagem do tipo TEOLM que é invocada pelo COMOLM em diferentes estados no seu fluxo, que retorna para o LOG se foi conectado com sucesso ou não ao MessageQ (importante no arranque do sistema RUNSYS) e caso falhe na escrita de uma mensagem no MessageQ.

Programa SPESRV

Tarefa aplicacional de processamento de serviços especiais (por ex., pedidos do Terminal Altura de relatórios financeiros, chaves e resultados, reimpressão, Sign-On, etc.).

Nesta nova versão foi adicionado:

* + A escrita dos 3 novos campos Serial Number, Message Id, Channel no TRABUF nas posições TSDT1, TSDT2, TSDT3, TSDT4, TSDT5 e TSDT6 para o SIGN-ON.
  + A escrita dos 3 novos campos Serial Number, Message Id, Channel no TRABUF nas posições TSDT7, TSDT8, TSDT9, TSDT10, TSDT11 e TSDT12 para o SALES REPORTS.
  + TRABUF(TSDT1)=PRO(SEROLM\_INT\_OLM,BUF)
  + TRABUF(TSDT2)=PRO(SEROLM\_INT\_OLM+1,BUF)
  + TRABUF(TSDT3)=BPRO(SEROLM\_OLM+8,BUF)
  + TRABUF(TSDT4)=PRO(MESSID\_INT\_OLM,BUF)
  + TRABUF(TSDT5)=BPRO(MESSID\_OLM+1,BUF)
  + TRABUF(TSDT6)=BPRO(CHOLM\_OLM,BUF)
  + TRABUF(TSDT7)=PRO(SEROLM\_OLM,BUF)
  + TRABUF(TSDT8)=PRO(SEROLM\_OLM+1,BUF)
  + TRABUF(TSDT9)=BPRO(SEROLM\_OLM+8,BUF)
  + TRABUF(TSDT10)=PRO(MESSID\_OLM,BUF)
  + TRABUF(TSDT11)=PRO(MESSID\_OLM+1,BUF)
  + TRABUF(TSDT12)=BPRO(CHOLM\_OLM,BUF)

Programa WAGPRO

Programa responsável pelo processamento das mensagens de apostas dos jogos mútuos que ainda se encontram em vigor que são LOTTO e SPORT.

Nesta nova versão verifica-se o seguinte:

* Foi adicionado antes do processamento das mensagens a leitura dos novos campos para o TRABUF:
  + TRABUF(TWCOLMSERL\_TLTO)=PRO(SEROLM\_INT\_OLM,BUF)
  + TRABUF(TWCOLMSERM\_TLTO)=PRO(SEROLM\_INT\_OLM+1,BUF)
  + TRABUF(TWCOLMSERH\_TLTO)=BPRO(SEROLM\_OLM+8,BUF)
  + TRABUF(TWCOLMMIDL\_TLTO)=PRO(MESSID\_INT\_OLM,BUF)
  + TRABUF(TWCOLMMIDH\_TLTO)=BPRO(MESSID\_OLM+4,BUF)
  + TRABUF(TWCOLMCOMF\_TLTO)=BPRO(CHOLM\_OLM,BUF)

Estas posições de cada campo foram obtidas através de uma análise ao ficheiro de código fonte destra.def em que se notou que entre os campos em uso (TWBORD=88) e (TWWEQP=120) aparentava várias posições disponíveis, mas que após melhor analise o campo TWBORD só indica o inicio onde se encontra a ser guardado os números da aposta que cresce dinamicamente.

Para obter o máximo que o board data pode ocupar quer para LOTTO quer para o SPORT fez-se uma aposta máxima para cada jogo.

* Aposta de LOTTO

Preencheu-se 10 boards (nº máximo de boards por bilhete) que por cada board se escolheu o números 1, 2, 18, 34 e 49. Esta escolha foi de obter o numero máximo de valores depois de aplicado o algoritmo (ver documentação ***Terminal Message Formats\_3.22.doc***) que é 1 1 16 16 15 que se converte em hexadecimal 0001 0001 0000 0001 0000 0001 1111. Que corresponde a 7 nibbles per Board = 3,5 bytes como se têm 10 boards é 3,5\*10=35 bytes para converter para inteiro 35/4=8,75 ou seja 9 inteiros.

* + (88+9) - 1=96 posição máxima ocupada por dados aposta para o Totoloto.

Nota: ler documento **Wager – TRABUF.xlsx** que se encontra na pasta **Millennium\New Terminals\TMF-Study-NewFields\Wager Totoloto\**.

* Aposta de SPORT

Preencheu-se as 10 boards (nº máximo de boards por bilhete) em que por cada board independentemente das escolhas cada board ocupa no máximo e mínimo 7 bytes sendo no total os 70 bytes mais um byte do “super 14” dando 71 bytes que são 71/4=17,75=18 inteiros necessários para guardar o board data.

* + (88+18) - 1 = 105 posição máxima ocupada por dados de uma aposta para o Totobola.

Para dar uma margem de folga para alterações futuras no board data que eventualmente se possa querer implementar, os novos campos começam a ocupar a posição 110 para frente do TRABUF para as transações de Wager e Cancellation dos jogos mútuos (LOTTO & SPORTS)

PARAMETER (TWCOLMSERL\_TLTO = 110)

PARAMETER (TWCOLMSERM\_TLTO = 111)

PARAMETER (TWCOLMSERH\_TLTO = 112)

PARAMETER (TWCOLMMIDL\_TLTO = 113)

PARAMETER (TWCOLMMIDH\_TLTO = 114)

PARAMETER (TWCOLMCOMF\_TLTO = 115)

PARAMETER (TCOLMSERL\_TLTO = 116)

PARAMETER (TCOLMSERM\_TLTO = 117)

PARAMETER (TCOLMSERH\_TLTO = 118)

PARAMETER (TCOLMMIDL\_TLTO = 119)

PARAMETER (TCOLMMIDH\_TLTO = 123)

PARAMETER (TCOLMCOMF\_TLTO = 124)

Estando as posições 120, 121 e 122 já ocupadas e em uso pelo END OF BOARD DATA, LUCKY NUMBER e SPORTS GAME CANCELATION EVENTS BITMAP.

Programa CANPRO

Programa responsável pelo processamento das mensagens de cancelamento de apostas dos jogos mútuos que ainda se encontram em vigor que são LOTTO e SPORT.

Nesta nova versão verifica-se o seguinte:

* Foi adicionado os novos campos (Transaction Serial, MessageId, Channel) para o cancelamento e reset aos novos campos guardados na Wager (pois lê o registo da wager para gerar o registo de cancelamento) que não são relevantes neste ponto e para evitar conflito nas flags do novo canal:
  + TRABUF(TCOLMSERL\_TLTO)=PRO(SEROLM\_INT\_OLM,BUF)
  + TRABUF(TCOLMSERM\_TLTO)=PRO(SEROLM\_INT\_OLM+1,BUF)
  + TRABUF(TCOLMSERH\_TLTO)=BPRO(SEROLM\_OLM+8,BUF)
  + TRABUF(TCOLMMIDL\_TLTO)=PRO(MESSID\_INT\_OLM,BUF)
  + TRABUF(TCOLMMIDH\_TLTO)=BPRO(MESSID\_OLM+4,BUF)
  + TRABUF(TCOLMCOMF\_TLTO)=BPRO(CHOLM\_OLM,BUF)
  + TRABUF(TWCOLMSERL\_TLTO)=0
  + TRABUF(TWCOLMSERM\_TLTO)=0
  + TRABUF(TWCOLMSERH\_TLTO)=0
  + TRABUF(TWCOLMMIDL\_TLTO)=0
  + TRABUF(TWCOLMMIDH\_TLTO)=0
  + TRABUF(TWCOLMCOMF\_TLTO)=0

Para as posições encontram-se referidas no tópico do programa WAGPRO.

Programa VALPRO

Programa responsável pelo processamento das mensagens de validação e pagamento dos jogos mútuos que ainda se encontram em vigor que são LOTTO e SPORT.

Nesta nova versão verifica-se o seguinte:

* Foi adicionado os novos campos (Transaction Serial, MessageId, Channel) para a validação:
  + TRABUF(TVOLMSERL\_TLTO)=PRO(SEROLM\_INT\_OLM,BUF)
  + TRABUF(TVOLMSERM\_TLTO)=PRO(SEROLM\_INT\_OLM+1,BUF)
  + TRABUF(TVOLMSERH\_TLTO)=BPRO(SEROLM\_OLM+8,BUF)
  + TRABUF(TVOLMMIDL\_TLTO)=PRO(MESSID\_INT\_OLM,BUF)
  + TRABUF(TVOLMMIDH\_TLTO)=BPRO(MESSID\_OLM+4,BUF)
  + TRABUF(TVOLMCOMF\_TLTO)=BPRO(CHOLM\_OLM,BUF)

Para este tipo de transação no destra.def os campos em uso vão até há posição TVEPTYP=53 que corresponde “IF SET IT IS AN ETICKET FLAG” começou-se a guardar os novos campos nas posições:

* PARAMETER (TVOLMSERL\_TLTO = 60)
* PARAMETER (TVOLMSERM\_TLTO = 61)
* PARAMETER (TVOLMSERH\_TLTO = 62)
* PARAMETER (TVOLMMIDL\_TLTO = 63)
* PARAMETER (TVOLMMIDH\_TLTO = 64)
* PARAMETER (TVOLMCOMF\_TLTO = 65)

Programa INSPRO

Programa responsável pelo processamento das mensagens de Lotaria instantânea a serem enviadas para o Sistema de Jogo IPS do qual a sua comunicação é feita no programa CRSPRO (CRS=Cross System).

Nesta nova versão verifica-se o seguinte:

* Foi adicionado os novos campos (Transaction Serial, MessageId, Channel) para a validação:
  + TRABUF(TVOLMSERL\_IL)=PRO(SEROLM\_INT\_OLM,BUF)
  + TRABUF(TVOLMSERM\_IL)=PRO(SEROLM\_INT\_OLM+1,BUF)
  + TRABUF(TVOLMSERH\_IL)=BPRO(SEROLM\_OLM+8,BUF)
  + TRABUF(TVOLMMIDL\_IL)=PRO(MESSID\_INT\_OLM,BUF)
  + TRABUF(TVOLMMIDH\_IL)=BPRO(MESSID\_OLM+4,BUF)
  + TRABUF(TVOLMCOMF\_IL)=BPRO(CHOLM\_OLM,BUF)
  + TRABUF(TGOLMSERL\_IL)=PRO(SEROLM\_INT\_OLM,BUF)
  + TRABUF(TGOLMSERM\_IL)=PRO(SEROLM\_INT\_OLM+1,BUF)
  + TRABUF(TGOLMSERH\_IL)=BPRO(SEROLM\_OLM+8,BUF)
  + TRABUF(TGOLMMIDL\_IL)=PRO(MESSID\_INT\_OLM,BUF)
  + TRABUF(TGOLMMIDH\_IL)=BPRO(MESSID\_OLM+4,BUF)
  + TRABUF(TGOLMCOMF\_IL)=BPRO(CHOLM\_OLM,BUF)

As posições destes campos são:

* PARAMETER (TGOLMSERL\_IL = 127)
* PARAMETER (TGOLMSERM\_IL = 128)
* PARAMETER (TGOLMSERH\_IL = 129)
* PARAMETER (TGOLMMIDL\_IL = 134)
* PARAMETER (TGOLMMIDH\_IL = 135)
* PARAMETER (TGOLMCOMF\_IL = 136)
* PARAMETER (TVOLMSERL\_IL = 130)
* PARAMETER (TVOLMSERM\_IL = 131)
* PARAMETER (TVOLMSERH\_IL = 132)
* PARAMETER (TVOLMCOMF\_IL = 133)
* PARAMETER (TVOLMMIDL\_IL = 134)
* PARAMETER (TVOLMMIDH\_IL = 135)

Que se analisou que no destra.def para a transição que ocupa mais espaço no Trabub que é a validação que ocupa até a posição 126, sendo assim para os novos campos passou-se a usar do 127 para cima, o Serial Transaction General ocupa as posições 127,128,129 e o Serial Transaction Validation ocupa as posições 130,131,132 deste modo não existe sobreposição dos dois Serial tal como na flag ocupa posições diferentes para cada Transaction.

* PARAMETER(TIVDESCR=TIVTYP+1) !NON-CASH PRIZE DESCRIPTION (107-126)

Programa LOGGER

Programa responsável pelo arranque do TMFMON (programa de monitorização do MTMF) e responsável pela escrita das transações no MTMF que invoca a função LOGOUT.

* Programa inalterado tal como a função LOGOUT, mas de referir que como as transações provenientes do novo canal são sempre regulares deixa de ser suportado:
  + SEND TO SPECIFIC TERMINAL
  + BROADCAST
  + SEND BUFFER TO ALL MXS TERMINALS (para todos terminais)

Programa ENCPROI

Programa que é responsável por encriptar e desencriptar mensagens que passam por Millennium, quando está desativo a variável de sistema **DESFlg.** Pois quando está ativo a comunicação com os sistemas externos é feita desencriptado, mas por default vai sempre encriptado (**DESFlg=0**), pode ser alterado esta funcionalidade no VISION.

Nesta nova versão verifica-se o seguinte:

* Reencaminhar as mensagens provenientes do novo canal (OLM\_COM) para a correta queue aplicacional que é **COMOLMQUE** na função SENDOUT.

Nota: o programa encpro e a função que invoca (sendout) que se encontra noutro ficheiro de código fonte já não se encontravam em uso.

* As transações provenientes do novo canal passam a ser sempre do tipo Regular **(TYPREG=1)** que no fluxo de dados quando as mensagens passam pela função logout são reencaminhadas para a queue aplicacional **GAME\_OUTQUE**, do qual são lidas no encproi.

1. **Programas Auxiliares**

Programa OLMCON

Programa utilitário que serve para ligar ou desligar a leitura de mensagens do MessageQ do COMOLM que passou a ser feito na tela *olm* do VISION. Tal como para ver os valores de algumas variáveis do sistema de jogos Millennium e a data CDC do sistema, foi mais usados para efeitos de testes.

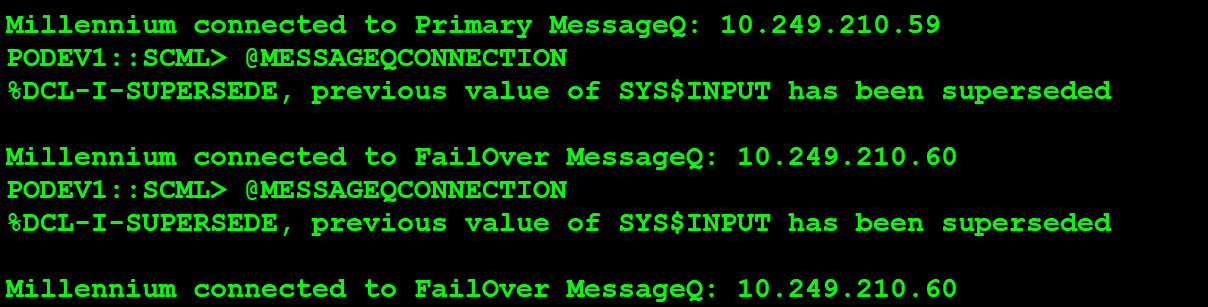
Programa REMOVEBUF

Programa utilitário que servio para efeitos de testes iniciais quando ficavam mensagens pressas na queue aplicacional do COMOLM quando ainda se estava a reaproveitar a queue aplicacional PASPRO (Lotaria Nacional) que já não se encontrava em uso. Sendo neste caso o uso da

1. **Scripts**

SCRIPT MESSAGEQCONNECTION

Script utilitário que se encontra na pasta OLM que se pode localizar pelo *logical name* GXOLM. Este script serve para indicar a qual MessageQ MILL se encontra ligado no momento que se corre o script, se é o primário ou se é o failover.



Script KILSYS

Este script tem a função de terminar abruptamente os processos relativos ao sistema de jogo Millennium.

Nesta nova versão foram adicionadas as tarefas aplicacionais que estavam em falta:

* COMOLM – tarefa responsável por comunicar as transações entre o sistema de jogo Millennium e o sistema de jogo Olimpo com recurso ao MessageQ.

SCRIPT DMQ\_OLM\_CONFIG

Este script serve para alterar o ficheiro de configuração do MessageQ que é DMQ.INI em que altera qual é máquina que é primaria e qual é a failover.

SCRIPT DSS\_LOGICALS

Este script foi atualizado para criar dinamicamente no arranque do Sistema de jogos Millennium os logical names em uso:

$ DEFINE/SYSTEM/EXEC/NOLOG GXOLM GXPROJ:[TSK.OLM]

$ DEFINE/SYSTEM/EXEC/NOLOG OLMLIB GXOLM

$ DEFINE/SYSTEM/EXEC/NOLOG OLMTSK GXPROJ:[TSK.OLM]

O script encontra-se na pasta DSS$GTECH.

(Nota: <https://wiki.vmssoftware.com/Logical_Name>)

SCRIPT DSS\_LOTGEN.COM

Este script serve para antes de fazer o lotgen remover/eliminar os binários anteriores das respetivas pastas, neste caso passou a existir também a pasta OLM no ambiente GODMIL para qual é gerado o novo binário e para remover as versões anterior em cada lotgen foi adicionado:

$ DEL OLMTSK:\*.EXE;\*

SCRIPT DELOLDBINS

Este script serve para auxiliar na remoção de binários antigo após um lotgen no ambiente de DEV e TST.

1. **Sistema de Ficheiros**

Foi criado uma nova pasta chamada de OLM que se encontra na pasta TSK ao qual está associado o logical name ***gxolm*** que contêm vários ficheiros.

* Onde se encontra o binário/executavel do COMOLM.
* O ficheiro de configuração do MessageQ DMQ.INI ao qual este processo se liga.
* O script DMQ\_OLM\_CONFIG.COM.
* O programa auxiliar olmcon.

Foram reestruturados os seguintes ficheiros:

* SCF.FIL – ficheiro de configuração do sistema Millennium.

Nota: o ficheiro TMF irá aumentar o espaço usado em disco, fruto do aumento do número de registos de log:

|  |  |  |
| --- | --- | --- |
| LI | ILOT | De 1 para 2 segmentos |
| LI | ICAR | De 1 para 2 segmentos |
| LI | IQTA | De 1 para 2 segmentos |
| LI | IORD | De 1 para 2 segmentos |
| LI | IMNU | Mais rapidamente atinge um maior número de segmentos a usar |
| LI | IVAL | Mais rapidamente atinge um maior número de segmentos a usar |
| LOTTO/SPORTS | WAG | Igual |
| LOTTO/SPORTS | CAN | Igual |
| LOTTO/SPORTS | VAL | Igual |
| LOGIN | TSON | Igual |

1. **Símbolos Lógicos (Logical Symbols)**

Não foram criados símbolos lógicos novos.

1. **Nomes Lógicos (Logical Names)**

"GXOLM" = "GXPROJ:[TSK.OLM]" (LNM$SYSTEM\_TABLE)

"OLMLIB" = "GXPROJ:[TSK.OLM]" (LNM$SYSTEM\_TABLE)

"OLMTSK" = "GXPROJ:[TSK.OLM]" (LNM$SYSTEM\_TABLE)

Para configurar estes logical names foi necessário proceder aos seguintes passos:

1. DEFINE/SYSTEM/EXEC GXOLM GXPROJ:[TSK.OLM]
2. DEFINE/SYSTEM/EXEC OLMLIB GXPROJ:[TSK.OLM]
3. DEFINE/SYSTEM/EXEC/NOLOG OLMTSK GXPROJ:[TSK.OLM]
4. **Compilação (LOTGEN)**

Os passos que se seguem devem ser realizados num ambiente de compilação do sistema Millennium.

1. Adicionar o código fonte no CMS
   1. Os ficheiros \*.FOR, \*.DEF, \*.OPT, \*.LNK e \*.LSH devem ser colocados na biblioteca GXPROJ:[CMS.SRC]
   2. Os ficheiros KILSYS.COM e SHRIMG.NAM devem ser colocados na biblioteca GXPROJ:[CMS.COM]

Nota: Para atualizar uma determinada biblioteca é necessário definir primeiro a biblioteca que se pretende (executar o comando CMS SET LIBRARY nome\_da\_biblioteca para definir a biblioteca).

1. Para ver a biblioteca em uso executar o comando CMS SHOW LIBRARY

> cms sh library

Your CMS library list consists of:

GXPROJ:[CMS.SRC]

GXPROJ:[CMS.COM]

GXPROJ:[CMS.HLP]

GXPROJ:[CMS.DOC]

1. Executar o LOTGEN (tempo aproximado de execução entre 10 e 15 minutos)
   1. Executar o comando LOTGEN ACTIVE “PROJ\_TERMINAIS\_OLM”

Os binários gerados são produzidos nas diretorias GXPROJ:[TSK], GXPROJ:[TSK.IGS], GXPROJ:[UTL] e GXPROJ:[TSK.OLM].

1. Verificar que não ocorreram erros de compilação e Linkagem no ficheiro DSS\_LOTGEN\_BATCH.LOG que se localiza em GXPROJ:[SRC]
2. Da execução do LOTGEN resultam 494 EXEs e 33 ESHs:
   * 443 EXEs localizam-se GXTSK;
   * 3 EXEs localizam-se em GXIGS;
   * 1 EXEs localizam-se em GXOLM (1 EXE é novo);
   * 48 EXEs localizam-se em GXUTL;
   * 33 ESHs localizam-se em GXTSK (1 ESH é novo).
3. **Instalação no Ambiente de Testes**
   1. Garantir que o sistema Millennium do ambiente de testes está em baixo e que as memórias não estão instaladas;
   2. Remover os binários antigos (executáveis e memórias) do ambiente de testes nas diretorias GXPROJ:[TSK], GXPROJ:[TSK.IGS], GXPROJ:[TSK.OLM] e GXPROJ:[UTL] possível usar o script DELOLDBINS.COM;
   3. Instalar os binários gerados pelo LOTGEN no ambiente de testes nas diretorias GXPROJ:[TSK], GXPROJ:[TSK.IGS], GXPROJ:[TSK.OLM] e GXPROJ:[UTL];
   4. Instalar o script KILSYS.COM na diretoria GXPROJ:[COM] do ambiente de testes;
   5. Instalar o ficheiro de configuração SHRIMG.NAM na diretoria GXPROJ:[COM] do ambiente de testes;
4. **Pós-instalação no Ambiente de Testes**

Nota: a execução deste ponto depende da realização prévia do ponto 11. deste documento.

1. Executar o programa BLDSYS, que se localiza em GXTSK, e configurar o parâmetro novo n.º 143;

A definição/consulta do valor dos novos parâmetros é efetuada através da opção n.º 1 do menu principal do programa BLDSYS (1 - Change a global system parameter).

Em baixo, sombreado a amarelo, encontra-se a configuração do novo parâmetro, relativos ao sistema Olimpo, a existir no momento do arranque do sistema Millennium (RUNSYS).

PODEV1::SCML> r bldsys

141>EUSPFIR 1 EM Suppr Fin Report 142>EUSPICA 0 EM Suppr Int Cancel

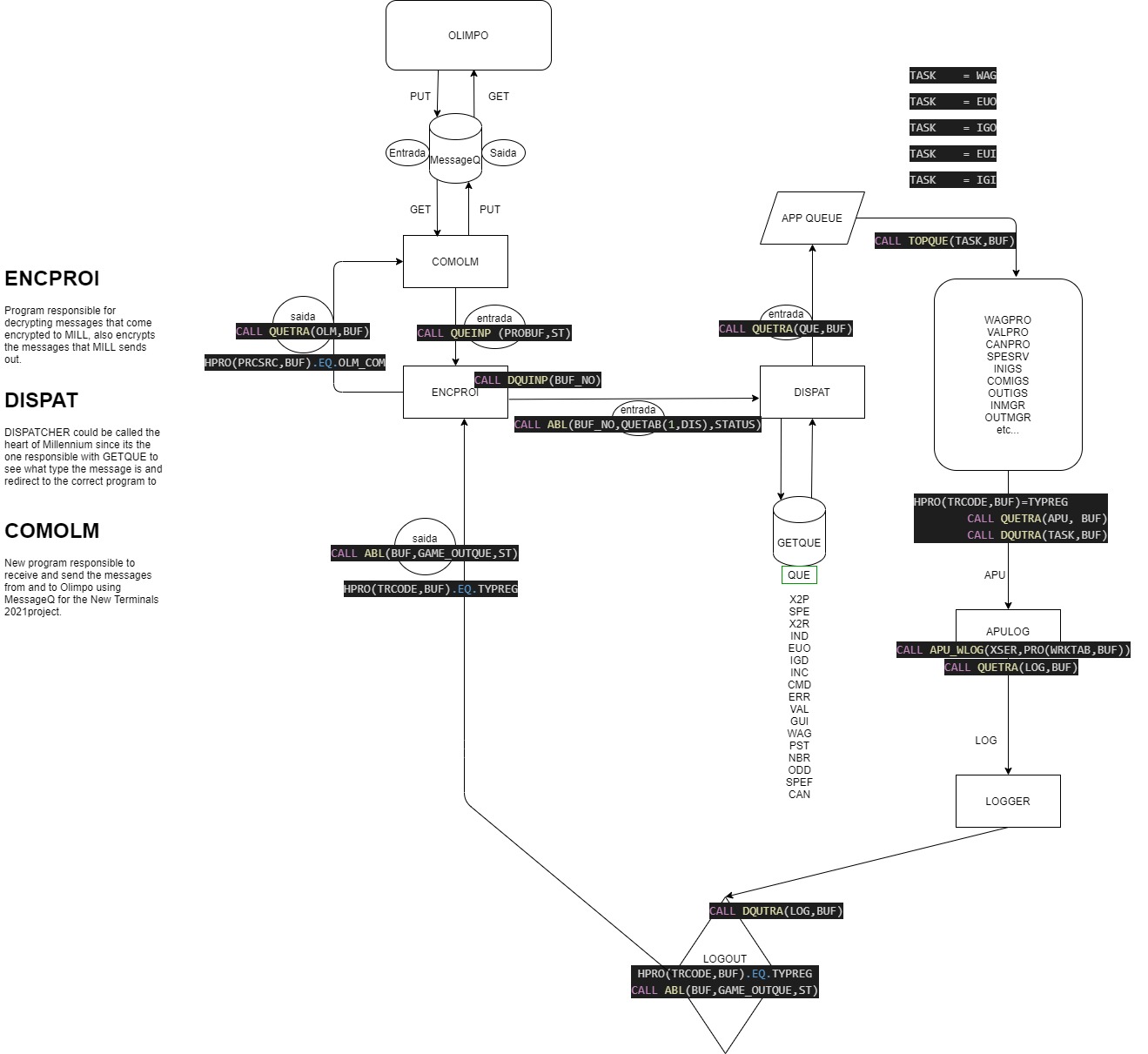
143>OLMCONF 1 OLM System Connect

11:00:52 CONSOLE Enter number to change, E-exit >

O parâmetro n.º 143 ativa/suprime a leitura de mensagens do MessageQ de onde provém as transações do Sistema de Jogo Olimpo desabilitando o processamento de novos pedidos, relembrar que é leitura não é a ligação/attach que que se perde.

Nota: após o arranque do sistema, o valor dos parâmetros pode ser alterado usando a tela OLMsnp do programa VISION.

**Diagrama do fluxo de Mensagens provenientes do novo canal**

****