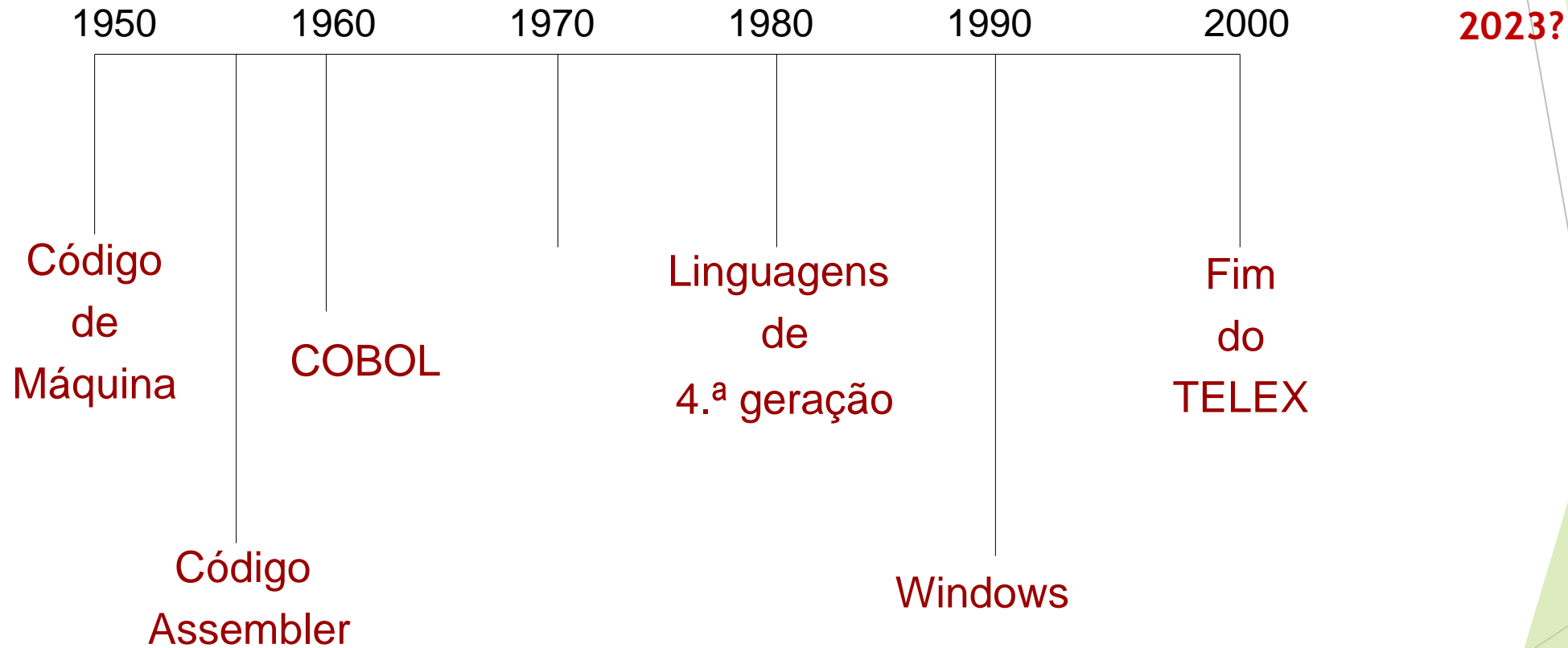


The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect.

Curso de Natural/ADABAS

Introdução à Linguagem de 4ª geração



Natural - Software AG

NATURAL é uma linguagem de programação de Quarta Geração, uma geração após o *Cobol*, desenvolvida e distribuída pela Software AG no Brasil. É considerada Quarta Geração por trazer diversos comandos de controle automático do fluxo do programa, além de editor de tela de fácil uso, programação de eventos, dentre outras funcionalidades. É formado por dois componentes principais: o **Sistema** (Server) e a **Linguagem**.

Fonte: wikipedia

Características do Natural

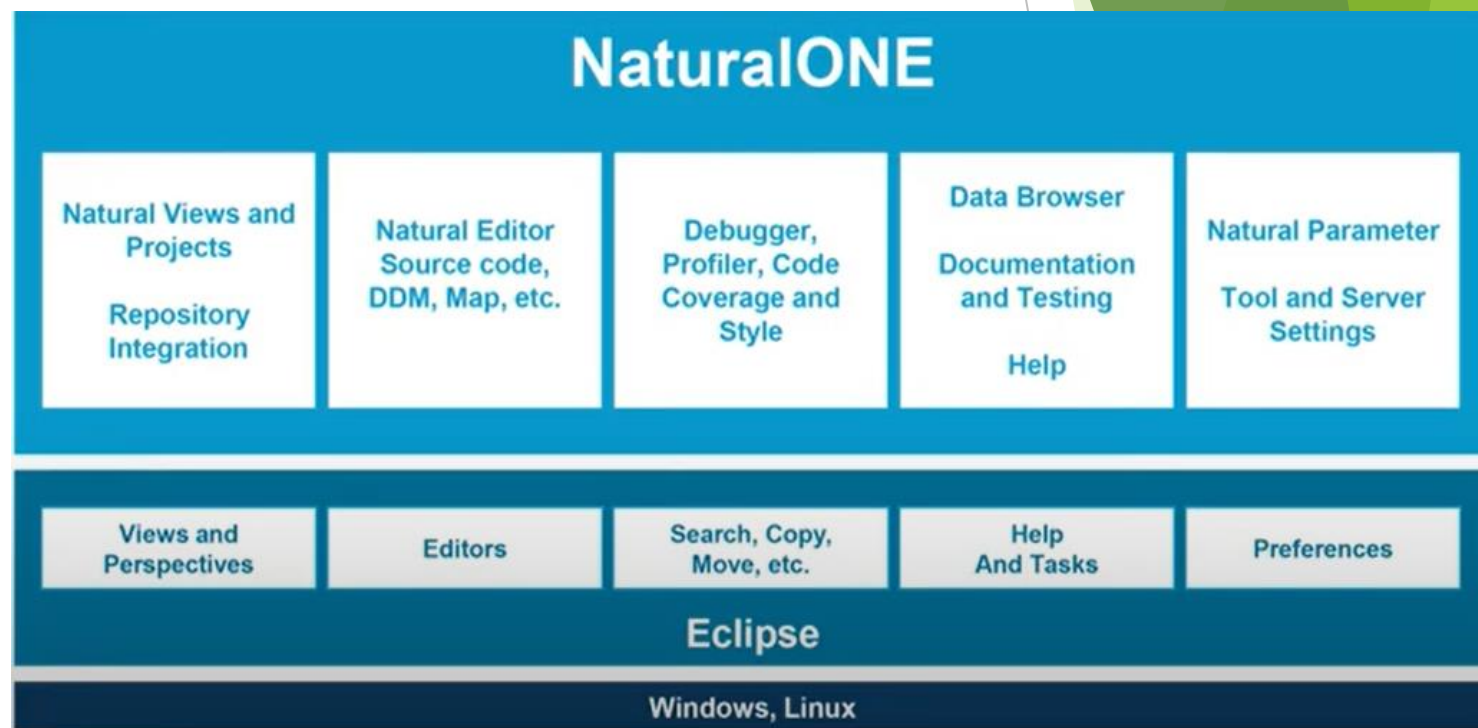
- ▶ Linguagem de programação com acesso a diversos bancos de dados: ADABAS (Hierárquico e relacional), Oracle, DB2, etc.;
- ▶ Utiliza diversos editores para criar programas, funções, telas pré-formatadas e áreas de dados;
- ▶ Permite uma programação modular;
- ▶ Utiliza funções e variáveis do sistema;
- ▶ Permite execução on-line e batch;
- ▶ Possui utilitário de testes de programação;
- ▶ Possui utilitário de criação/manutenção de mensagens de erros de aplicação;
- ▶ Qualquer aplicação pode ser facilmente portada para várias outras plataformas (Mainframe, Windows, Linux)
- ▶ Instruções (list, edit, check, stow, ...) e comandos
- ▶ Modo Report vs. Modo Estruturado

Objetos Natural

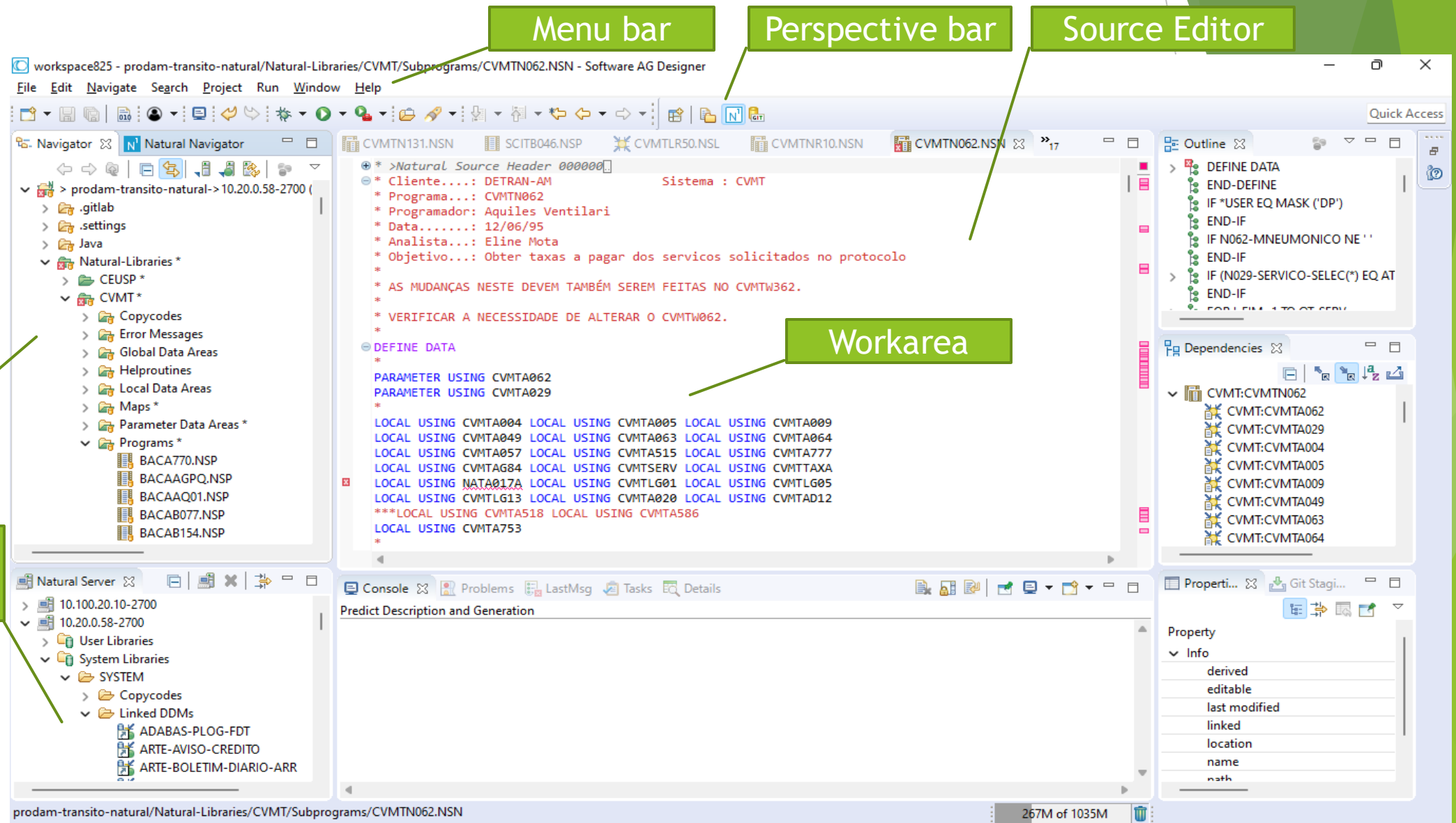
- ▶ São armazenados em bibliotecas (Libraries)
- ▶ Nomes com tamanho de 8 caracteres, não repetem
- ▶ Há três grupos principais de objetos Natural:
- ▶ **Objetos Programáticos:** Programas (P), Subprogramas (N), Subrotinas (S), Help-rotinas (H) e Copycode (C);
- ▶ **Área de Dados:** Global (G), Parameter (A) e Local (L);
- ▶ **Mapas:** Mapa (M) e Mapa de Help (H).
- ▶ Cada um desses grupos têm o seu próprio editor, que é aberto quando você escolhe o tipo de objeto com o qual você irá trabalhar.

NaturalONE - a nova geração de desenvolvimento Natural

- ▶ Baseada na popular plataforma Eclipse
- ▶ Ambiente de Desenvolvimento Integrado (IDE) com uma série de ferramentas de edição para aplicações Natural (all-in-one)
- ▶ Plugins e funcionalidades com perspectivas como: Team (versionamento), Natural Runtime (Execute/Debug/test), Natural Data Browser, Natural for Ajax, etc.
- ▶ Criação de webservices (Entirex)
- ▶ Com suporte Natural em Mainframe, Linux Windows e plataformas container.



NaturalOne ou Natural Designer



Estrutura de um programa

```
DEFINE DATA
LOCAL          /* ou GLOBAL, PARAMETER, Independent
1 #I (N2) /* variável numérica c/ 2 dígitos
END-DEFINE
* Imprime Hello World 3 vezes
FOR #I = 1 TO 3
    PRINT 'Hello World'
END-FOR
*
END
```


Visão Geral dos Tipos de Dados Disponíveis

- ▶ Área de Dados (data areas)
 - GDA (Global Data Areas) - define os dados que podem ser compartilhados por múltiplos objetos na aplicação.
 - PDA (Parameter Data Area - referencia os dados que estão definidos tanto na GDA quanto na área de dados local (LDA).
 - LDA (Local Data Area) - define os dados que podem ser usados somente pelo objeto.
 - AIVs (Application-Independent Variables) - São variáveis independentes de qualquer estrutura de dados específica
- ▶ Variáveis do Sistema e Definidas pelo Usuário
- ▶ Dados do Banco

Variáveis do Sistema

Nome	Formato/Tamanho	Formato dos Conteúdos
*DATN	N8	YYYYMMDD
*DATE	A8	DD/MM/YY
*DATI	A8	YY-MM-DD
*DATJ	A5	YYDDD
*DATX	D	Formato interno
*TIME	A10	HH:MM:SS.T
*TIMN	N7	HHMMSST
*TIMX	T	Formato interno

Para cada variável de sistema DATA corresponde outra que inclui o ano com 4 dígitos, exemplo:
*DAT4E, *DAT4I, *DAT4J.

Nome	Conteúdo
*NUMBER	Número de registos
*COUNTER	Número de vezes que o loop foi executado
*PAGE-NUMBER	Valor atualizado do número da página
*LIBRARY-ID	Nome da biblioteca Natural
*PROGRAM	Nome do objeto Natural
*USER	Identificação do usuário
*LANGUAGE	Idioma corrente (1 – Inglês) (11 ou 38 – Português)
*CURSOR	Posição do cursor
*ERROR-NR	Número de erro Natural

Área de Dados interna

- ▶ Na codificação de um programa é necessário definir todos os dados que se pretende utilizar.

```
DEFINE DATA  
  GLOBAL USING GDA2  
  LOCAL  USING LDA3  
  LOCAL  
    1 CARS VIEW OF VEHICLES  
      2 MAKE  
      2 MODEL  
      2 YEAR  
      2 COLOR  
    1 #NAME (A20)  
END-DEFINE  
...
```



Área de Dados Interna

Variáveis Definidas pelo Usuário - formatos e tamanhos

Format	Explanation	Definable Length	Internal Length (in Bytes)
A	Alphanumeric	1 - 1073741824 (1GB)	1 - 1073741824
B	Binary	1 - 1073741824 (1GB)	1 - 1073741824
C	Attribute Control	-	2
D	Date	-	4
F	Floating Point	4 or 8	4 or 8
I	Integer	1 , 2 or 4	1, 2 or 4
L	Logical	-	1
N	Numeric (unpacked)	1 - 29	1 - 29
P	Packed numeric	1 - 29	1 - 15
T	Time	-	7
U	Unicode (UTF-16)	1 - 536870912 (0.5 GB)	2 - 1073741824

* Nas versões mais atuais foi acrescentado: variáveis e arrays dinâmicos.

Exemplos de definição de dados

```
DEFINE DATA LOCAL
1 #A1  (A10)      /* Alphanumeric, 10 positions.
1 #A2  (B4)       /* Binary, 4 positions.
1 #A3  (P4)       /* Packed numeric, 4 positions and 1 sign position.
1 #A4  (N7.2)     /* Unpacked numeric,
                  /* 7 positions before and 2 after decimal point.
1 #A5  (N7.)      /* Invalid definition!!!
1 #A6  (P7.2)     /* Packed numeric, 7 positions before and 2 after decimal point
                  /* and 1 sign position.
1 #INT1 (I1)      /* Integer, 1 byte.
1 #INT2 (I2)      /* Integer, 2 bytes.
1 #INT3 (I3)      /* Invalid definition!!!
1 #INT4 (I4)      /* Integer, 4 bytes.
1 #INT5 (I5)      /* Invalid definition!!!
1 #FLT4 (F4)      /* Floating point, 4 bytes.
1 #FLT8 (F8)      /* Floating point, 8 bytes.
1 #FLT2 (F2)      /* Invalid definition!!!
1 #DATE (D)       /* Date (internal format/length P6).
1 #TIME (T)       /* Time (internal format/length P12).
1 #SWITCH (L)     /* Logical, 1 byte (TRUE or FALSE).
                  /*
END-DEFINE
```

Definindo os Dados

A estrutura para definição é composta de:

- ▶ Nível
 - ▶ Nome
 - ▶ Formato/tamanho
 - ▶ Complemento
- ▶ Podemos definir até 7 níveis de variáveis. O tamanho máximo para um nome de variável é 32 caracteres. Os caracteres válidos para formação do nome são os alfabéticos (A-Z), números (0-9) e os especiais (#, -, _, @, \$, &). O caractere + também pode ser utilizado, desde que apenas na primeira posição. Para efeito de **padronização**, utilizamos o caractere “#” como caractere inicial para designar variáveis definidas pelo usuário.
- ▶ Vetores e arrays
 - ▶ Redefinição de campos
 - ▶ Constantes e valores iniciais
 - ▶ Registros

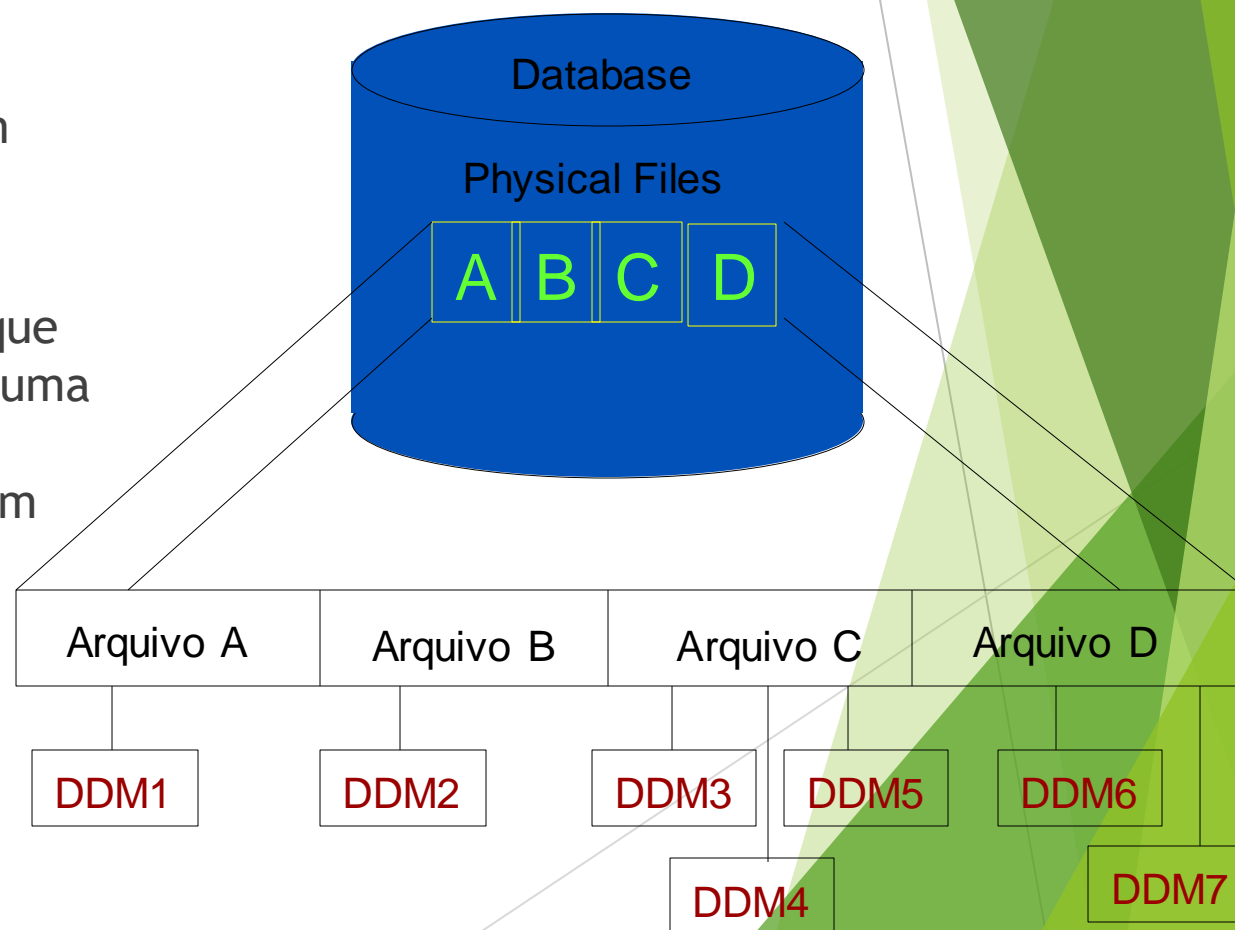
Redefinição de Campos

- ▶ O Natural permite a redefinição de um grupo ou de um único campo da área de dados interna. Uma das razões para redefinir campos é a necessidade de usar trechos de campos, tais como o ano, o mês e/ou o dia de um campo *DATA-ACQ*, por exemplo.
- ▶ A cláusula para redefinição de campos é *REDEFINE*. Ela irá definir os trechos que compõem o campo a ser redefinido. Esses trechos também podem ser campos definidos pelo usuário ou um *FILLER* (bytes que não têm nenhum significado específico para o objeto em questão).

```
0010 *****
0020 * ILUSTRA DECLARACAO REDEFINE COM A OPCAO DE FILLER
0030 *****
0040 DEFINE DATA
0050 LOCAL
0060 1 CAR VIEW OF VEHICLES
0070 2 PERSONNEL-ID
0080 2 MAKE
0090 2 DATE-ACQ
0100 2 REDEFINE DATE-ACQ
0110 3 FILLER 4X
0120 3 #MONTH (A2)
0130 3 FILLER 2X
0140 END-DEFINE
0150 *
0160 FIND (1) CAR WITH PERSONNEL-ID = '11100106'
0170 WRITE NOTITLE / 10T '=' MAKE
0180           / 10T '=' DATE-ACQ 5X '=' #MONTH
0190 END-FIND
0200 END
```

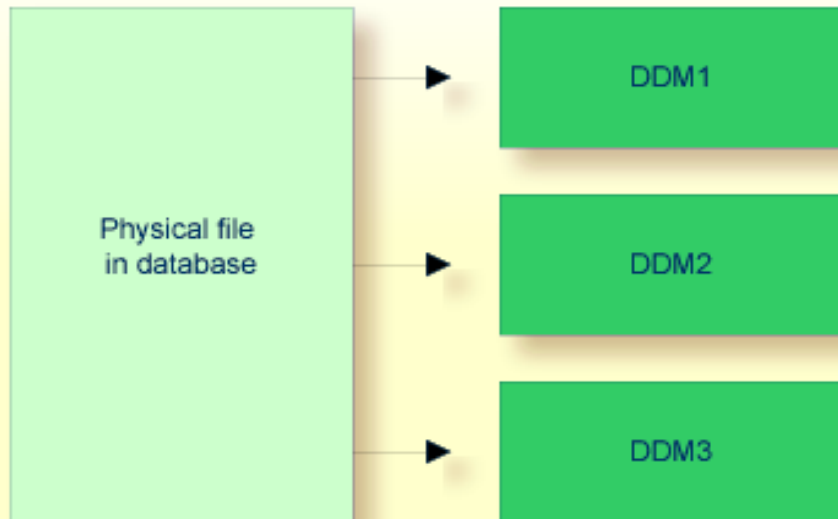
Arquivos do Banco de Dados e DDMs

- ▶ Arquivo Físico vs. Arquivo Lógico
- ▶ Os dados são armazenados no seu banco em arquivos físicos (tabelas).
- ▶ A D.D.M. se trata de uma visão lógica do arquivo físico. Define os campos do banco que vão ser usados no programa. Os campos de uma DDM podem compreender todos os campos definidos no arquivo do banco, ou apenas um subconjunto desses campos.



Acessando dados de um B.D. Adabas

- ▶ Arquivo Físico (Adabas)
- ▶ Data Definition Modules - DDM (Objeto)
- ▶ View (Data Local)



Physical File In Database

Fields:
AA
BB
CC
DD
EE
...

DDM "XYZ"

Fields:
AA PERSONNEL-ID I8
BB NAME A20
CC FIRST-NAME A20
DD BIRTH N8
EE JOB-TITLE A25
...

View

```
DEFINE DATA LOCAL  
1 ABC VIEW OF XYZ  
  2 NAME  
  2 FIRST-NAME  
  2 PERSONNEL-ID  
END-DEFINE  
...
```

Entrada/Saída de Dados e Mapas

- ▶ Parâmetros: AD=, EM=, IP=, SG= e AD=
- ▶ Comando Print/Write
- ▶ Comando Display
- ▶ Comando Input
- ▶ Set Key
- ▶ Set Window
- ▶ Comando Reinput

Parâmetros:

AD=[r][a][i/o][c][f]

r → apresentação: Blink, Default, Intensified, Non-display

a → alinhamento: Left, Right

i/o → modo: Input (A), Modifiable, Output

c → tratamento de caracteres: I minúsculas para maiúsculas, W aceita minúsculas

f → caracter de preenchimento (filler): definido pelo usuário.

```
INPUT (AD=IMLT'.'') //// 10T 'AGENCIA.....' #AGENCIA
INPUT /// 'AGENCIA' (I) #AGENCIA (AD=M)
INPUT (AD=M) ' ' #AGENCIA ' ' #CONTA
INPUT MARK *#CONTA /// #AGENCIA #CONTA
INPUT MARK 2 /// #AGENCIA #CONTA
```

INPUT com mensagem na linha de mensagem:

```
INPUT WITH TEXT 'Digite a agência. ' 'AGENCIA:' #AGENCIA (AD=MI' )
```

INPUT com utilização de mapa externo:

```
INPUT USING MAP 'XXXM0001'
```

Manipulação de Campos e Op. Aritméticas

- ▶ Comando Reset [initial]
- ▶ Comando Move [substring/rounded/by name/by position/all/edited]
- ▶ Comandos: Add, Subtract, Multiply, Divide e Compute
- ▶ Comando Examine [delete/replace/giving number/position/index]
- ▶ Comando Compress [leaving no space/with delimiter]
- ▶ Comando Translate

Instruções Condicionais e operadores

- ▶ Campos Lógicos: true, false, 'S', 'N'
- ▶ Operador EQ, NE, LT, LE, GE, GT e NOT
- ▶ Operador IS, MASK, SCAN, BREAK, MODIFIED
- ▶ Comando If
- ▶ Comando Decide On/For (case)

Controle de Processamento

- ▶ Comando Include (copycode)
- ▶ Comando For
- ▶ Comando Repeat [Until/While] (* Read/Find/Histogram)
- ▶ Comando Escape [Top/Bottom/Routine/Module]
- ▶ Comando Sort
- ▶ Comando Perform / Define Subroutine
- ▶ Comando Callnat/Call
- ▶ Comando Fetch
- ▶ Comando Stop/Terminate/End

Controle de Quebras e Eventos

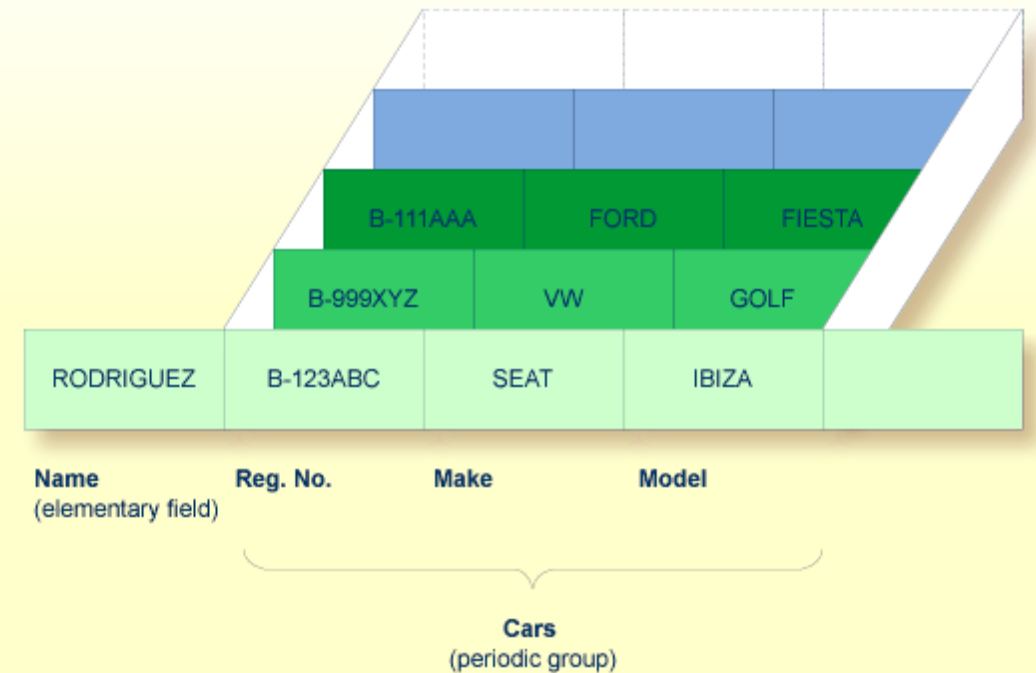
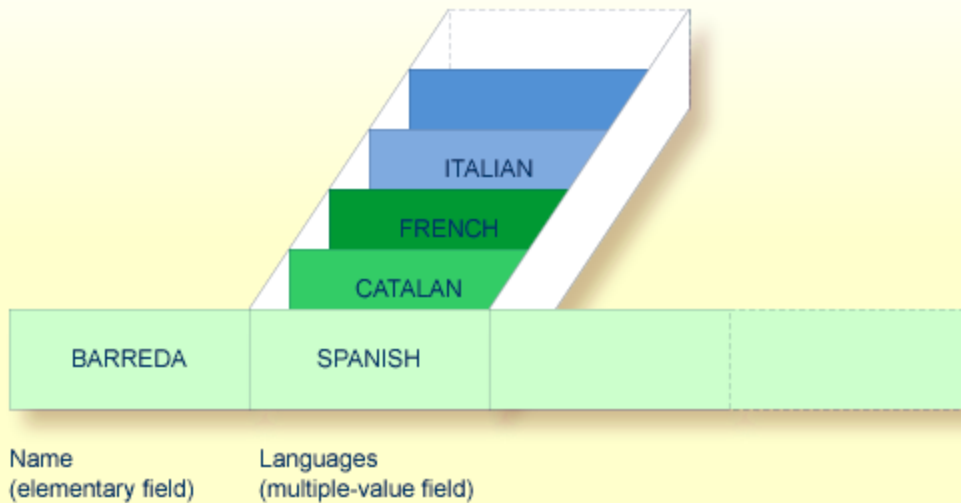
- ▶ Comando At Start of Data
- ▶ Comando At End of Data
- ▶ Comando At Top of Page
- ▶ Comando At Bottom of Page
- ▶ Comando At Break
- ▶ Funções Old. Count. Sum, Aver, Min, Max
- ▶ Tratamento de erros com On Error

Acessando arquivos texto

- ▶ Comando Define Work File
- ▶ Comando Read Work File
- ▶ Comando Write Work File

Acessando dados de um B.D. Adabas

- ▶ Descritores e Superdescritores
- ▶ Campos Múltiplos
- ▶ Campos Periódicos e Grupos



Acessando dados de um B.D. Adabas

- ▶ Comando Find
- ▶ Comando Read
- ▶ Comando Histogram
- ▶ Accept / Reject
- ▶ Comando Get
- ▶ Comando Store / Update / Delete
- ▶ End Transaction / Backout Transaction

Trabalhando com Repositório

- ▶ Natural Server
- ▶ Git
- ▶ Deploy