

# Hierarquia de Classes para Suporte da Compressão de Dados

Artur Ferreira Nuno Pereira Paulo Pereira David Coutinho  
 Centro de Cálculo, DEEC, ISEL

## OBJECTIVOS

Construção de uma *framework* para compressão de dados, baseada no modelo teórico da compressão sem perdas.

A modelação é realizada segundo o Paradigma da Programação Orientada por Objectos.

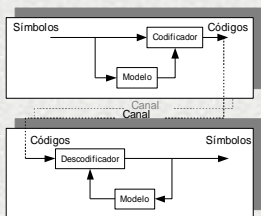
Características fundamentais:

- identificação dos componentes do modelo teórico;
- flexibilidade e facilidade de utilização e expansão.

## MODELO DA COMPRESSÃO DE DADOS

O modelo teórico da compressão sem perdas é composto por:

- modelo;
- codificador / decodificador.



Modelo da compressão de dados.

## CODIFICAÇÃO E COMPRESSÃO

Tipos de codificação:

- estatística;
- baseada em dicionário.

Formas de compressão:

- estática - com utilização de modelo pré-estabelecido;
- semi-adaptativa - criação do modelo em função da amostra a codificar;
- adaptativa - utilização de modelo pré-estabelecido, com actualização dinâmica.

## FRAMEWORK

A composição interna da *framework* reflecte os componentes do modelo teórico.

## MODELOS

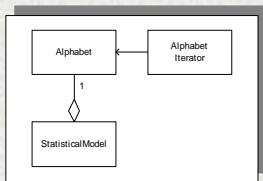
Cada tipo de codificação tem o seu próprio modelo:

- estatístico;
- dicionário.

## ESTATÍSTICO

O modelo estatístico reflecte a distribuição de probabilidades do universo de símbolos da fonte, denominado por alfabeto.

Este alfabeto é representado pela entidade *Alphabet*. A entidade *StatisticalModel* agrega o alfabeto.

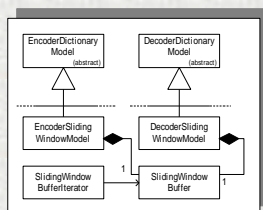


Micro-arquitetura do Modelo Estatístico.

## DICIONÁRIO

Na codificação baseada em dicionário, o alfabeto de símbolos é intrínseco ao modelo.

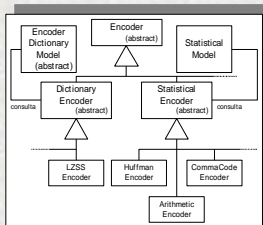
Cada codificador baseado em dicionário está associado a um modelo específico.



Micro-arquitetura do Modelo de Dicionário.

## CODIFICADORES

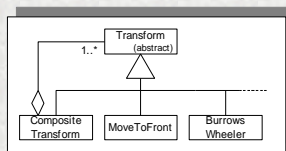
A entidade *Encoder* é a base da hierarquia de codificadores. Esta reflecte a separação entre os dois tipos de codificação.



Micro-arquitetura de codificadores.

## TRANSFORMADAS

Expansão do modelo teórico através da introdução de pré-processamento aos dados a comprimir.



Micro-arquitetura de transformadas.

É possível a concatenação de transformadas através da entidade *CompositeTransform*.

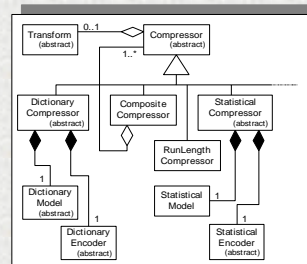
## COMPRESSORES

A classe abstracta *Compressor* é a base da hierarquia de compressores.

São estabelecidos os tipos de compressão:

- estatística;
- baseada em dicionário;
- *ad-hoc*.

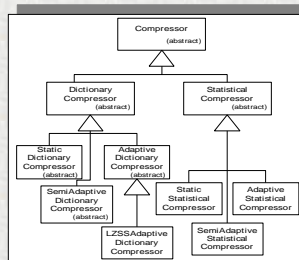
Prevê-se a utilização de sequências de compressores, através da entidade *CompositeCompressor*.



Tipos de compressor.

A especialização para as formas de compressão é obtida a partir das entidades:

*StatisticalCompressor*  
*DictionaryCompressor*.



Formas de compressão.

## APLICAÇÃO

Implementou-se o compressor constituído pelos componentes indicados na figura seguinte.

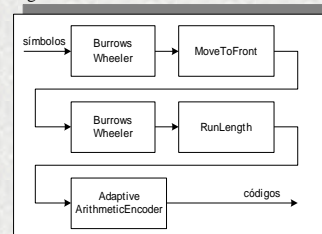


Diagrama de blocos do compressor *Comp*.

O código de criação do compressor e respectiva utilização apresenta-se a seguir.

```
// Criar a transformada composta.
CompositeTransform comp;
BurrowsWheeler bwt;
MoveToFront mt;
ct.add( bwt );
ct.add( mt );
ct.add( bwt );

// Criar os compressores
// Run Length Compressor
RunLengthCompressor r1Comp;

// Adaptive Arithmetic Compressor
Alphabet alphabet;
AdaptiveArithmeticEncoderSpeed arithEncoder;
AdaptiveStatisticalCompressor artComp
( alphabet, arithEncoder );

// Composite Compressor.
CompositeCompressor comp;
comp.add( r1Comp );
comp.add( artComp );
comp.add( artComp );

// Efectua a compressão.
comp.compress( "infile", "outfile" );
```

Na tabela seguinte apresentam-se os resultados (capacidade de compressão) deste compressor juntamente com os de alguns produtos comerciais conhecidos, aplicados ao conjunto normalizado de ficheiros *Calgary Corpus*.

	COMP	PKZIP	GZIP	BZIP2
Total	55586	59724	59444	48138
Removido	60%	64%	65%	69%
Bits/Byte	2.74869	2.83500	2.80894	2.48981

Verifica-se que o pré-processamento sobre o texto a comprimir é vantajoso porque aumenta a redundância ao nível do *n-grama*, permitindo maiores taxas de compressão.

## COMENTÁRIOS

• O Paradigma da Programação Orientada por Objectos conjuntamente com padrões de desenho é adequado ao desenvolvimento de plataformas deste género.

• Obteve-se uma plataforma expansível e flexível, possibilitando boas taxas de compressão.