HPCC Systems

Big Data pela perspectiva HPCC Systems

Alysson Oliveira

HPCC™ SYSTEMS

O Grupo RELX

RELX

RELX é um provedor global de análises baseadas em informações e ferramentas de decisão para clientes profissionais e empresariais. O Grupo atende clientes em mais de 180 países e possui escritórios em cerca de 40 países, com um total que supera 36 mil contribuidores.

Saiba mais em www.relx.com

Científico

ELSEVIER

Análise de risco

LexisNexis® RISK SOLUTIONS

Eventos

RX

Legal

LexisNexis® | Legal & Professional

HPCC SYSTEMS

HPCC Systems: Ativos e Clientes

- 12 petabytes de dados públicos e privados

Unidade	Símbolo	Número de Bytes
Kilobyte	**KB**	**2^10 = 1024 bytes**
Megabyte	**MB**	**2^20 = 1,048,576 bytes**
Gigabyte	**GB**	**2^30 = 1,073,741,824 bytes**
Terabyte	**TB**	**2^40 = 1,099,511,627,776 bytes**
Petabyte	**PB**	**2^50 = 1,125,899,906,842,624 bytes**
Exabyte	EB	2^60 = 1,152,921,504,606,846,976 bytes
Zettabyte	ZB	2^70 = 1,180,591,620,717,411,303,424 bytes
Yottabyte	YB	2^80 = 1,208,925,819,614,629,174,706,176 bytes

HPCC Systems: Ativos e Clientes

- 12 petabytes de dados públicos e privados
- 270+ milhões de transações por hora
- Clientes em mais de 180 países
- 84% dos integrantes da Fortune 500
- 9 dos 10 maiores bancos do mundo
- 100% dos 50 maiores bancos americanos
- 98 das 100 maiores seguradoras do mundo
- Mais de 7500 órgãos governamentais: locais, estaduais e federais

A LexisNexis Risk Solutions

Estrutura no Brasil

Total de 140 colaboradores

Área de atuação

Análise de dados para organizações que buscam gerenciar riscos, encontrar oportunidades e melhorar seus resultados. Sediada em Atlanta, Geórgia, a LexisNexis Risk Solutions tem mais de 11.000 funcionários ao redor do mundo.

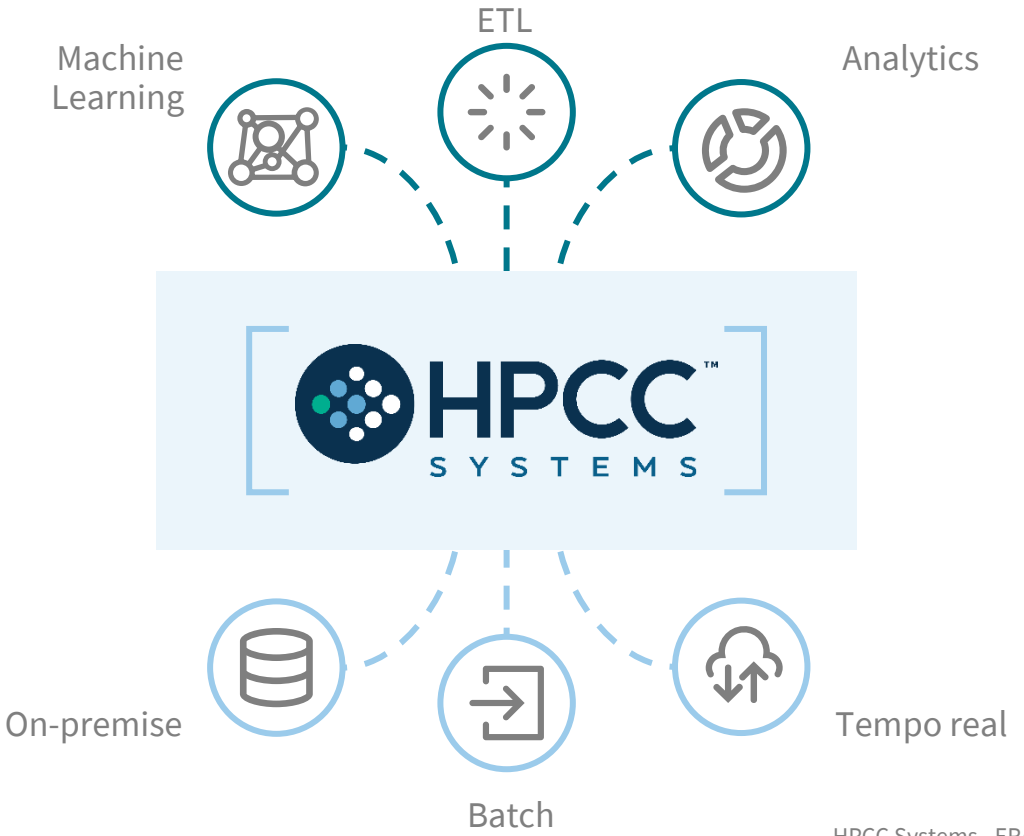
Tecnologia de código aberto

Plataforma de computação de Big Data de código aberto chamada HPCC Systems com vastos ativos de dados para proporcionar inteligência de decisão para clientes.

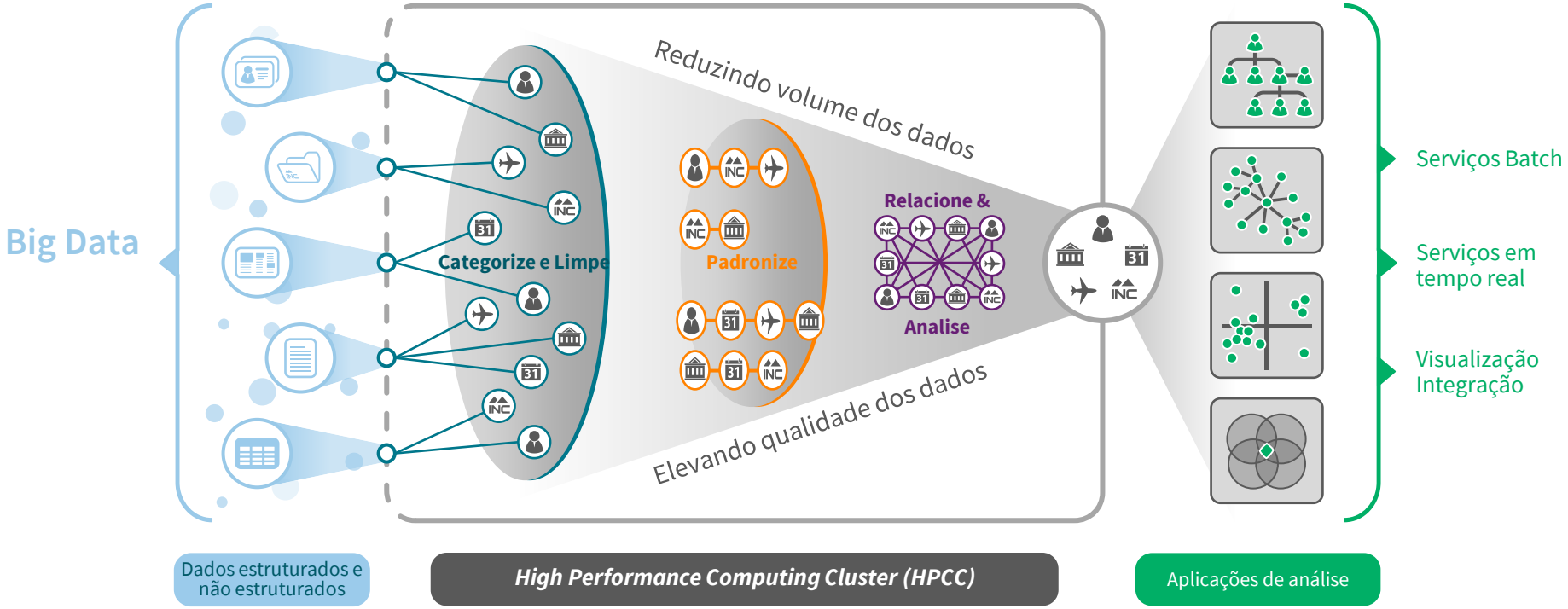
HPCC SYSTEMS

A plataforma HPCC Systems

- Stack para big data
- Processamento paralelo
- Dados distribuídos
- Código aberto
- Gratuita



“Funil” de dados no HPCC Systems



Cadeia de Big Data em HPCC Systems

Breve histórico do HPCC Systems

2001

Primeira versão da plataforma é lançada

2011

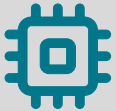
Código aberto (licença Apache e código no GitHub)

2012 – 16

Melhorias contínuas com **FOCO NA QUALIDADE**

Suporte e treinamento aprimorado

2017- Presente



Aprimoramentos de arquitetura (Cloud)

Desenvolvimentos em Machine Learning

I.A. generative e afins

Visão geral do stack

Cluster Thor
Extração, transformação e carregamento de dados

Cluster ROXIE
Entrega online de consultas em big data

Ferramentas para manipulação de dados
Perfilamento, limpeza, consolidação e linking de dados

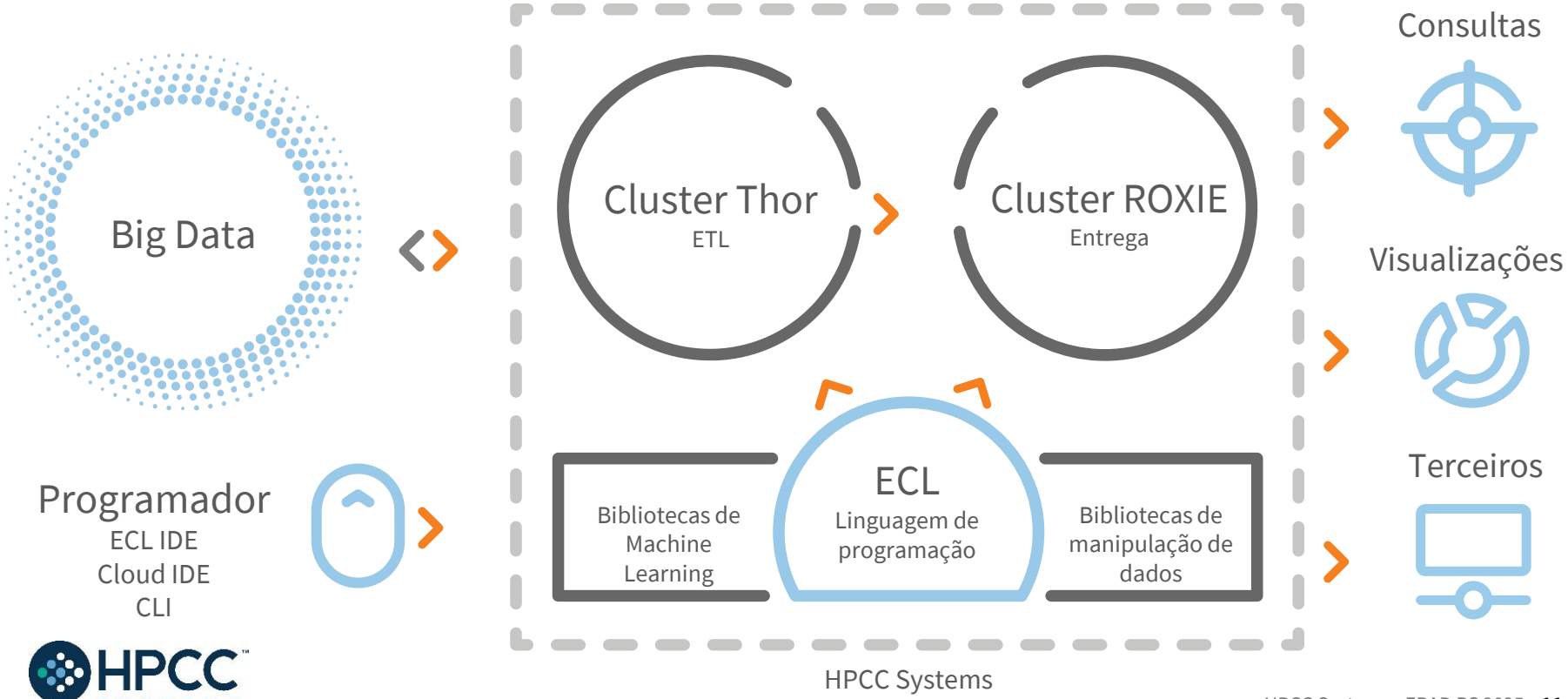
Bibliotecas de Machine Learning
Supervisionado, não-supervisionado, aprendizagem profunda

Conectividade

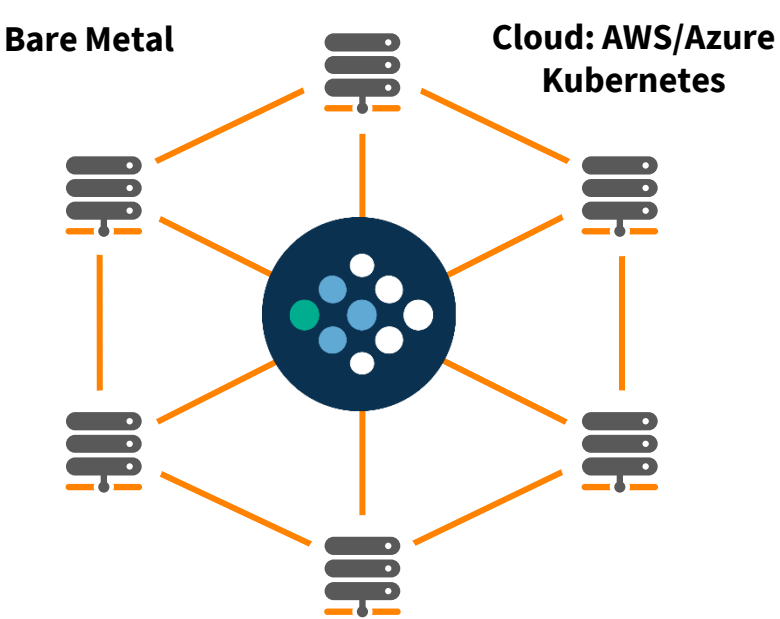
Plugins de integração com outros sistemas

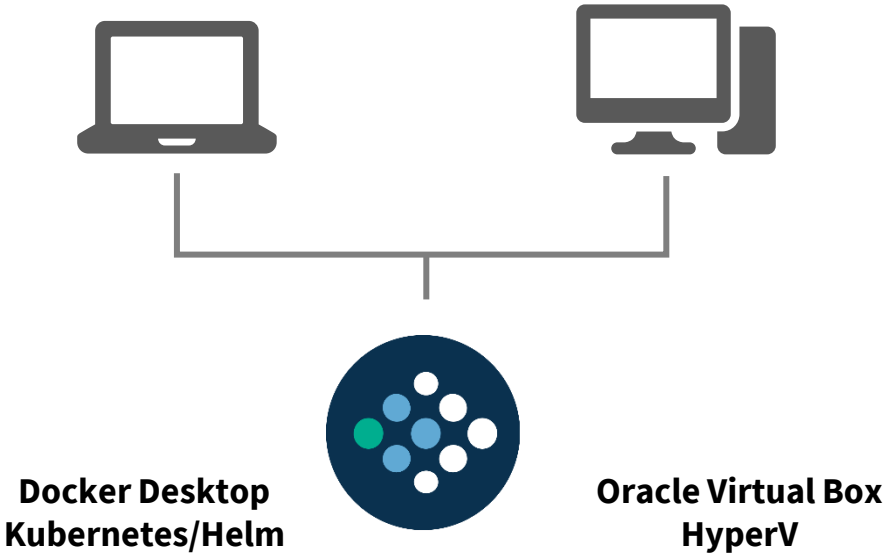
HPCC SYSTEMS

Os componentes da plataforma

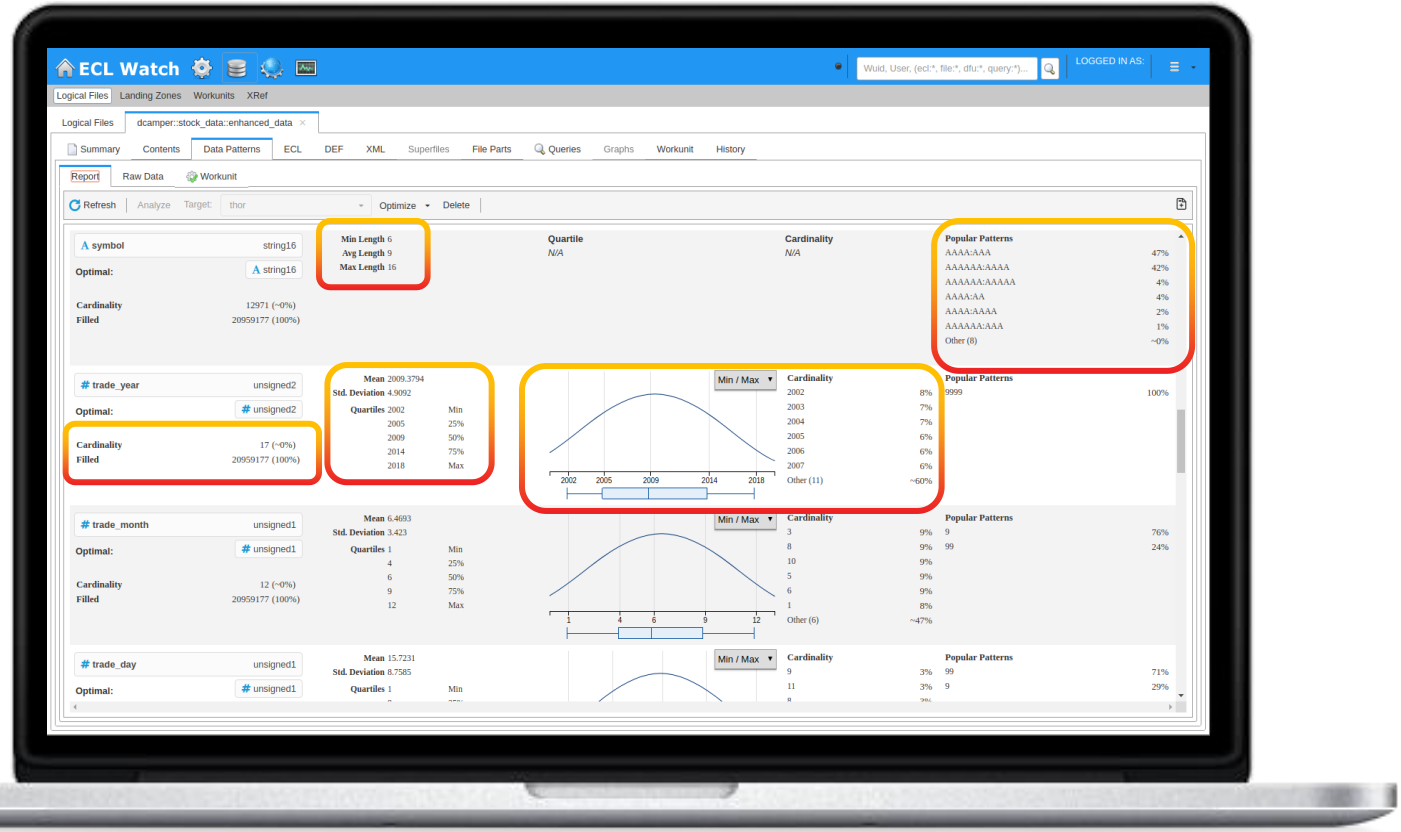


Jornada em direção à nuvem





Bibliotecas de perfilamento de dados



Bibliotecas de machine learning

Não supervisionado

Clusterização

DBSCAN

K-Means

PLN

Text Vectors

Levenshtein Deletion Neighborhood

Redução de Dimensão

PCA

Supervisionado

Classificação

SVM

Árvores de decisão

Regression logística

Classification Forest

Alocação Latente de Dirichlet (Topic Modeling)

Regressão

Regressão linear

GLM

Regression Forest

Redes neurais & Deep Learning

Autoencoders

Redes neurais convolucionais

Redes neurais recorrentes

Perceptrons

Métodos ensemble

Random Forest

Gradient Boosted Forest

Gradient Boosted Trees

Plugins para conectividade

WsSQL

TOMBOLO

SPARK

JDBC/ODBC Driver

KAFKA

HPCC

PENTAHO

Couchbase

Tableau

SQS

Java API

MEMCACHED

REDIS

Linguagens suportadas

- C++
- R
- Python
- Java
- Cassandra
- SQL/SqLite

CODE: SELECT ALL

```
IMPORT java;
STRING jcat(STRING a, STRING b) :=
   IMPORT(java,
          'JavaCat.cat:(Ljava/lang/String;Ljava/lang/String;)Ljava/lang/String;' :
classpath('/opt/HPCCSystems/classes'));

jcat('Hello ', 'world!');
```

CODE: SELECT ALL

```
IMPORT python;
SET OF STRING split(STRING text) := EMBED(python)
  return text.split()
ENDEMBED;
split('Once upon a time');
```

CODE: SELECT ALL

```
IMPORT python;
r := RECORD
  STRING word;
  UTF8 tags;
END;
DATASET(R) tag(STRING text) := IMPORT(python, './ex2.tag');
tag('Once upon a time there was a boy called Richard');
```

CODE: SELECT ALL

```
IMPORT MySQL;
stringrec := RECORD
   string name
END;
sqlrec := RECORD
   string ssn;
   string address;
END;
DATASET(sqlrec) MySQLJoin(dataset(stringrec) inrecs) := EMBED(mysql)
  SELECT * from tbl1 where name = ?;
ENDEMBED;
MySQLJoin(indata);
```

Linguagem ECL

```
// *****
// Elementos constituintes basicos da ECL
// Uma definicao
Mydef := 'Olá mundo';  // definicao do tipo "value"

// Uma acao
OUTPUT('Olá mundo');
OUTPUT(mydef);

// *****
// Estruturas de dados basicas em ECL
// Estrutura RECORD
rec := RECORD
  STRING10  Firstname;
  STRING    Lastname;
  STRING1   Gender;
  UNSIGNED1 Age;
  INTEGER   Balance;
  DECIMAL7_2 Income;
END;

// Declaracao DATASET
ds := DATASET([{'Alysson','Oliveira','M',26,100,1000.50},
               {'Bruno','Camargo','',22,-100,500.00},
               {'Elaine','Silva','F',19,-50,750.60},
               {'Julia','Caetano','F',45,500,5000},
               {'Orlando','Silva','U',67,300,4000}],rec);
OUTPUT(ds);
```

```
//Filtragem de datasets
recset := ds(Age<65);
recset := ds(Age<65,Gender='M');
IsSeniorMale := ds.Age>65 AND ds.Gender='M'; //definição do tipo "boolean"

SetGender := ['M','F'];  //definicao do tipo "set"
recset := ds(Gender IN SetGender);
recset;                          // definição do tipo "recordset"
COUNT(recset);    //Equivale a: OUTPUT(COUNT(recset));

*****
//Transformacoes basicas em ECL
//Eliminacao de campos desnecessarios
tbl := TABLE(ds,{Firstname,LastName,Income});
tbl;

// Ordenacao de valores
sortbl := SORT(tbl,LastName);
sortbl;

// Remocao de duplicidades
dedptbl := DEDUP(sortbl,LastName);
dedptbl;

// Adicao de campo no dataset
rec2 := RECORD
  UNSIGNED   recid;
    STRING10   Firstname;
    STRING     Lastname;
    STRING1    Gender;
    UNSIGNED1  Age;
    INTEGER    Balance;
    DECIMAL7_2 Income;
END;
rec2 MyTransf(rec Le, UNSIGNED cnt) := TRANSFORM
  SELF.recid:=cnt;
  SELF := Le;
END;
newds := PROJECT(ds,MyTransf(LEFT,COUNTER));
```

Macros nativas

- Processo de ingestão para qualquer conjunto de registros

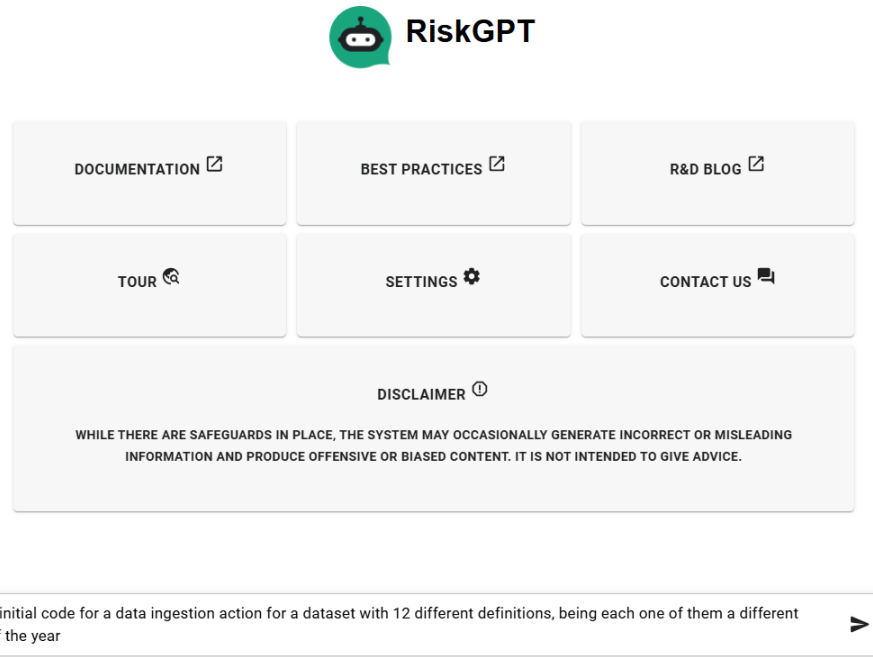
```
FM_Upper(Ds):= FUNCTIONMACRO
  #EXPORTXML(Data, RECORDOF(Ds));
  #DECLARE(newrecord)
  #SET(newrecord,'newrecord := RECORD \n')
  #FOR(Data)
    #FOR(Field)
      #IF(%'@type'% = 'string')
          #APPEND(newrecord,%'@ecltype'%+' '+%'@label'%+':= std.Str
      #ELSE
          #APPEND(newrecord,#TEXT(Ds)+'.'+%'@label'%+';\n')
      #END
    #END
  #END
  #APPEND(newrecord, '\nEND;')
  #APPEND(newrecord, '\nsaida := TABLE('+#TEXT(Ds)+', newrecord); \
RETURN %'newrecord'%;
ENDMACRO;

OUTPUT(FM_Upper(DsTeste));
```

```
newrecord := RECORD
  DsTeste.id;
  string10 name:= std.Str.ToUpperCase(DsTeste.name);
  string10 lastname:= std.Str.ToUpperCase(DsTeste.lastname);
  DsTeste.age;
  string field1:= std.Str.ToUpperCase(DsTeste.field1);
  string field2:= std.Str.ToUpperCase(DsTeste.field2);
  string field3:= std.Str.ToUpperCase(DsTeste.field3);
  string field4:= std.Str.ToUpperCase(DsTeste.field4);
  string field5:= std.Str.ToUpperCase(DsTeste.field5);
  string field6:= std.Str.ToUpperCase(DsTeste.field6);
  string field7:= std.Str.ToUpperCase(DsTeste.field7);
  string field8:= std.Str.ToUpperCase(DsTeste.field8);
  string field9:= std.Str.ToUpperCase(DsTeste.field9);
  string field10:= std.Str.ToUpperCase(DsTeste.field10);
  string field11:= std.Str.ToUpperCase(DsTeste.field11);
  string field12:= std.Str.ToUpperCase(DsTeste.field12);
  string field13:= std.Str.ToUpperCase(DsTeste.field13);
  string field14:= std.Str.ToUpperCase(DsTeste.field14);
  string field15:= std.Str.ToUpperCase(DsTeste.field15);
  string field16:= std.Str.ToUpperCase(DsTeste.field16);
  string field17:= std.Str.ToUpperCase(DsTeste.field17);
  string field18:= std.Str.ToUpperCase(DsTeste.field18);
  string field19:= std.Str.ToUpperCase(DsTeste.field19);
  string field20:= std.Str.ToUpperCase(DsTeste.field20);

END;
saida := TABLE(DsTeste, newrecord);
OUTPUT(saida);
```

I.A. Generativa




```
EXPORT monthsIngestion := MODULE
    EXPORT months := RECORD
        STRING Jan;
        STRING Feb;
        STRING Mar;
        STRING Apr;
        STRING May;
        STRING Jun;
        STRING Jul;
        STRING Aug;
        STRING Sep;
        STRING Oct;
        STRING Nov;
        STRING Dec;
    END;
    EXPORT ds := DATASET(yourFile,months,CSV);
END;
```

Relacionamento com Academia

Universidade de São Paulo
Brasil

USP

UNIVERSIDADE FEDERAL
DE SANTA CATARINA

UNIVERSIDADE FEDERAL DO PARA

Imperial College
London

FAU
COLLEGE OF ENGINEERING
& COMPUTER SCIENCE
Florida Atlantic University

UNIVERSITY OF
OXFORD
Mathematical Institute

KENNESAW STATE
UNIVERSITY

Universidad
CATÓLICA
de Pereira

CLEMSON
UNIVERSITY

2020 HPCC SYSTEMS® COMMUNITY DAY
CLOUD ATLAS
Charting Our Course

HPCC SYSTEMS

Universidades Brasileiras

Universidade de São Paulo
Brasil

- Disciplina Optativa na Poli/USP (Link para a disciplina)
- Cursos de Difusão (Fundação Vanzolini)
- Co-orientação de IC's (PIBIC)
- Co-Orientação de TCC's

UNIVERSIDADE FEDERAL
DE SANTA CATARINA

- Co-Orientação de IC's
- Co-Orientação de TCC's e Metrados
  - Artigos publicados (ERAD/RS, CotB, Fusion, etc)
  - Apresentações no HPCC Summit
- Compra de equipamentos

HPCC SYSTEMS

Universidades Brasileiras

- Cursos de Difusão
- Co-orientação de IC’s (PIBIC)
- Co-Orientação de TCC’s

Universidades Estrangeiras

Imperial College London

- Pesquisas de Doutorado
  - Deep Learning, Machine Learning, Text Mining, Natural Language Processing

CLEMSON UNIVERSITY

- Estagiários
  - Machine Learning

HPCC SYSTEMS

Projetos de Pesquisa, Mentorias e Parcerias Acadêmicas

Site: https://hpccsystems.com/community/academics

- Programa de Estágio
- Mentorias Acadêmicas
- Bolsas de Estudo
- Publicações Acadêmicas
- Treinamentos

Código Aberto

Github: https://github.com/hpcc-systems

- Linguagem: C++
- Repositório bastante ativo
  - 250+ Commits nos últimos 30 dias
- Documentação
  - Arquivos README.md dentro do repositório
  - Site do HPCC (https://hpccsystems.com/training/documentation)
- Tickets
  - https://track.hpccsystems.com/secure/Dashboard.jspa

Minicurso amanhã

Minicurso 5

Processamento e análise de Big Data para aplicação de algoritmos de Machine Learning através da plataforma HPCC Systems

Horário e local: 24/04 (8:30 - 11:30) — LAB2

Resumo: Ao longo do minicurso os participantes terão a oportunidade de conhecer os conceitos essenciais de processamento e análise de volumes massivos de dados (Big Data), e o processo de desenvolvimento de um serviço de consulta através da utilização da plataforma open-source composta por um Cluster Computacional de Alto Desempenho (HPCC Systems), assim como a aplicação de algoritmos de Aprendizado de Máquina, e a possibilidade de aplicar os conhecimentos adquiridos em um ambiente de treinamento disponibilizado em sala de aula.

Autores

Alysson Oliveira

Alysson Oliveira é formado em Engenharia de Computação pela USP e atual engenheiro de software na LexisNexis Risk. Sua principal atuação gira em torno do suporte e desenvolvimento de programas de treinamento para a plataforma HPCC Systems no Brasil, abrangendo o público acadêmico, pesquisadores e profissionais da área da computação e de dados. Também busca estabelecer parcerias com universidades a fim de oferecer aos alunos de graduação a oportunidade de trabalhar em projetos científicos.

Considerações Finais & Perguntas

Contato

- Alysson.Oliveira@lexisnexisrisk.com

- Mauro.marques@lexisnexisrisk.com