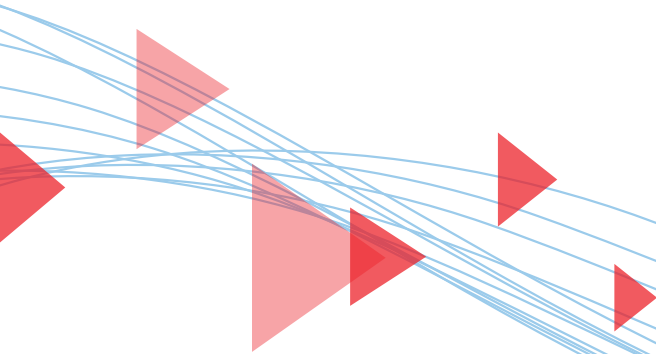


HPCC Systems

Big Data pela perspectiva HPCC Systems

Alysson Oliveira



O Grupo 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



Eventos



Análise de risco



Legal



HPCC Systems: Ativos e Clientes



12
petabytes

de dados públicos
e privados



270+ milhões
de transações por hora



Unidade	Símbolo	Número de Bytes
Kilobyte	KB	$2^{10} = 1,024$ 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

maiores
seguradoras
do mundo



órgãos
governamentais



das empresas
Fortune 500

0%
50
iores
ncos
ericanos

%

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.

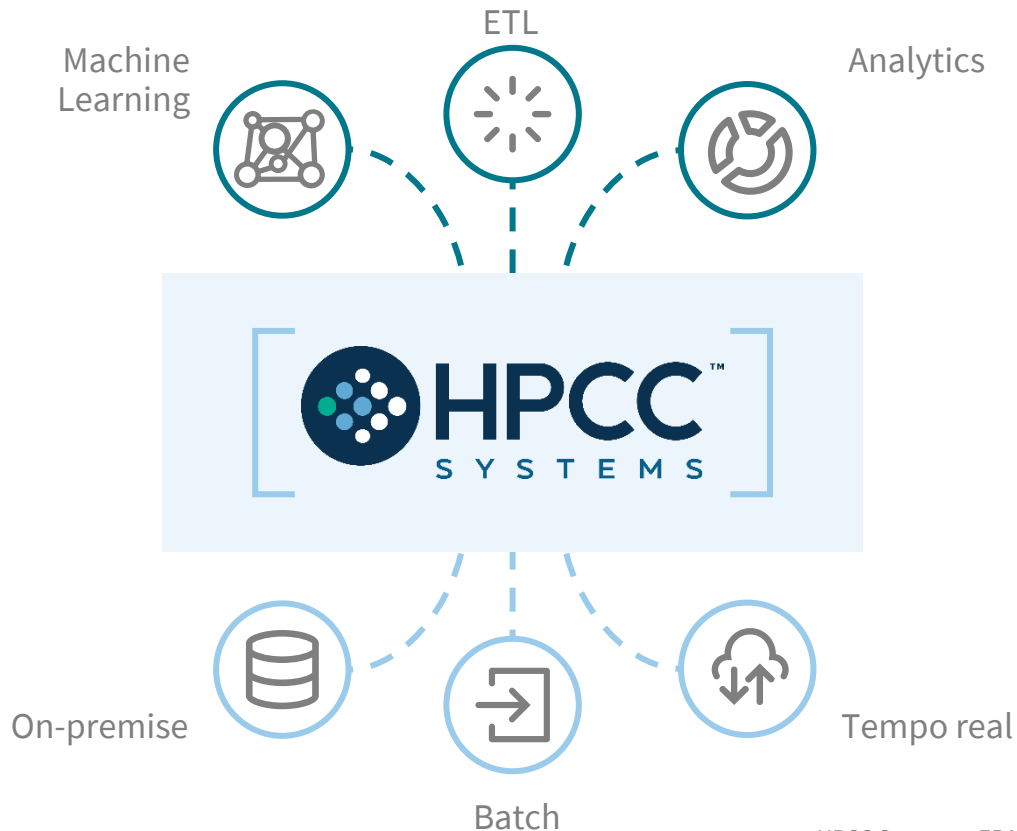
Tecnologia de código aberto

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

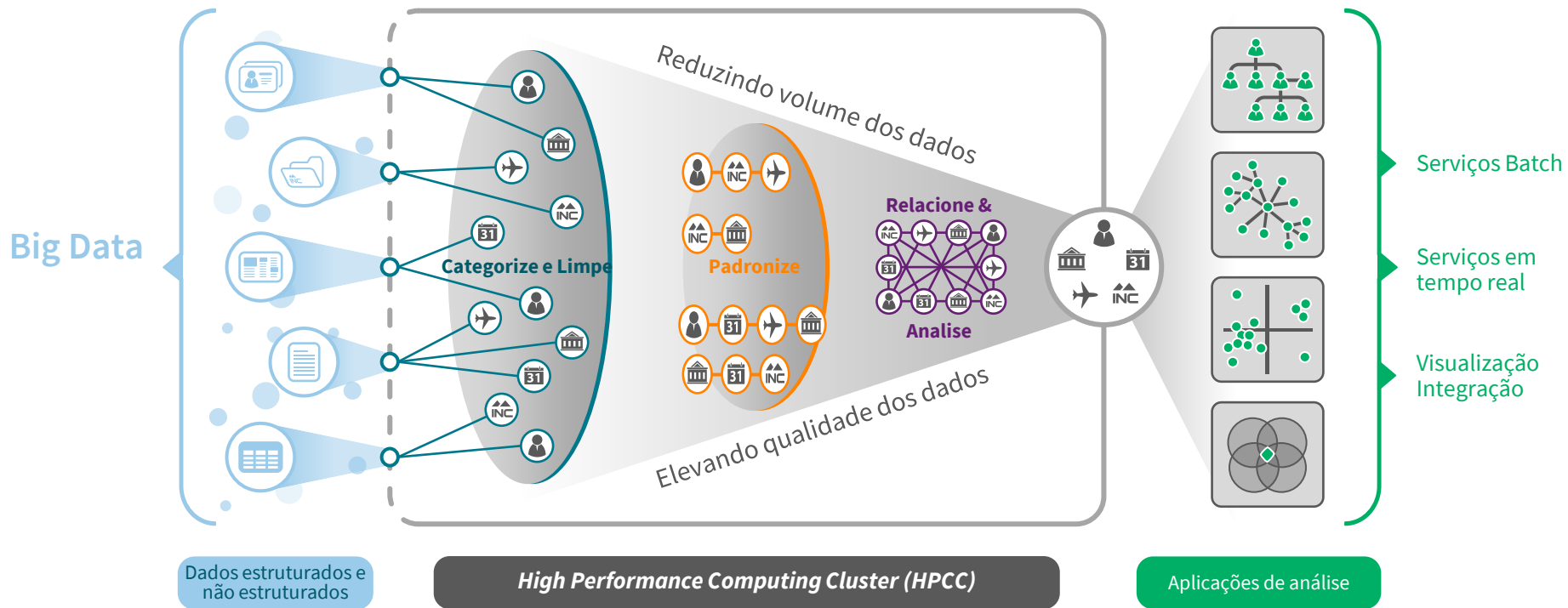


A plataforma HPCC Systems

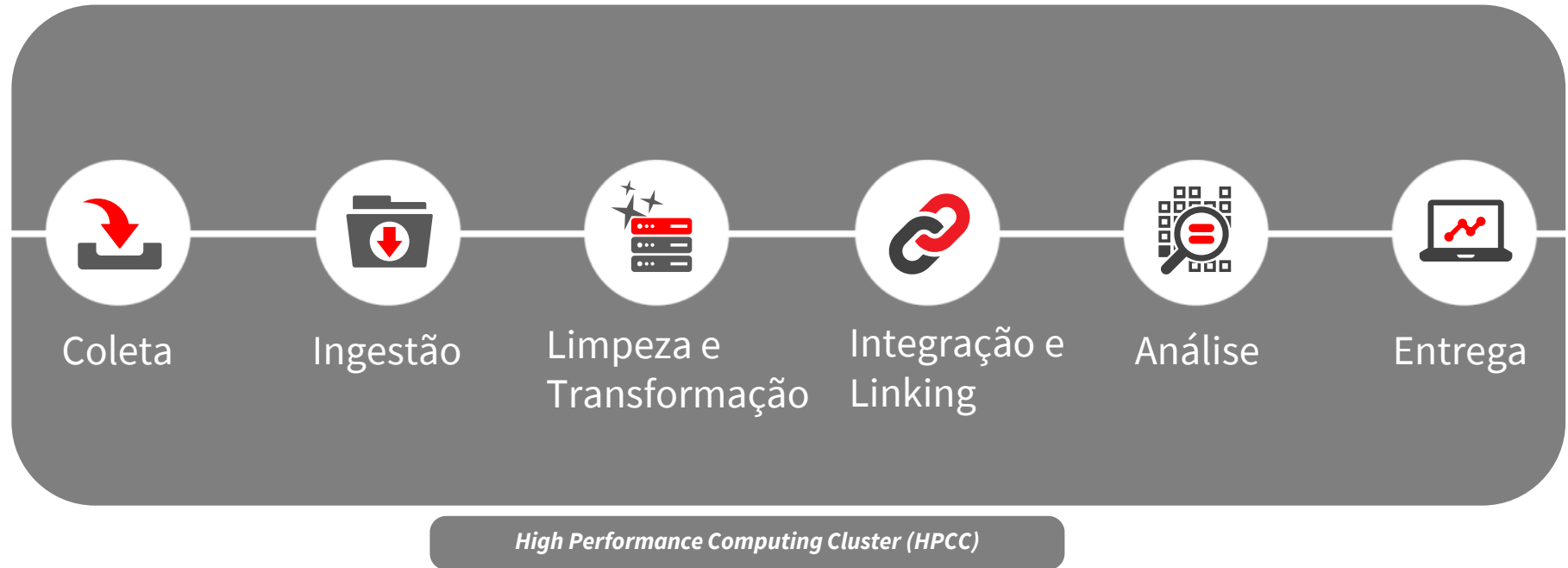
- Stack para big data
- Processamento paralelo
- Dados distribuídos
- Código aberto
- Gratuita



“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)

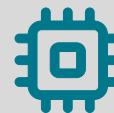
2012 – 16



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

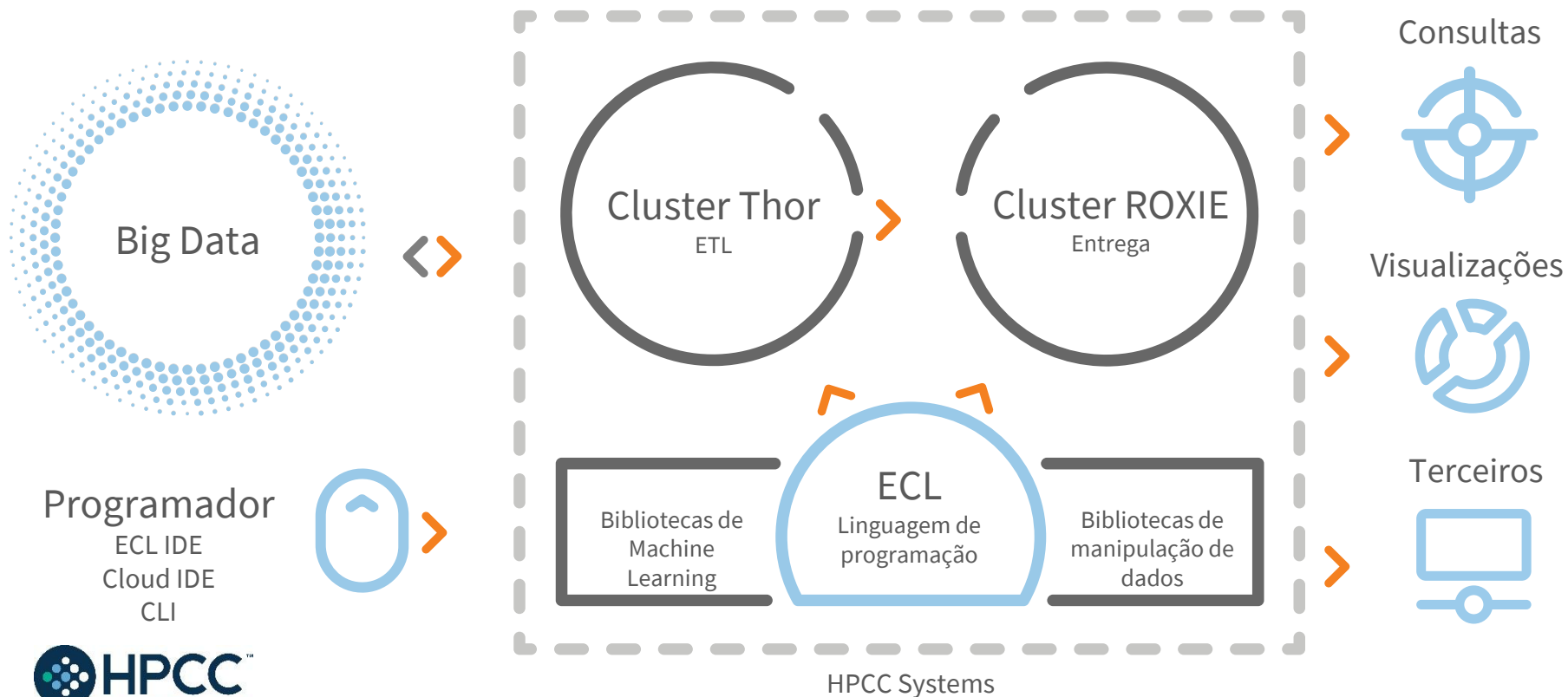
Supervisionado, não-supervisionado, aprendizagem profunda



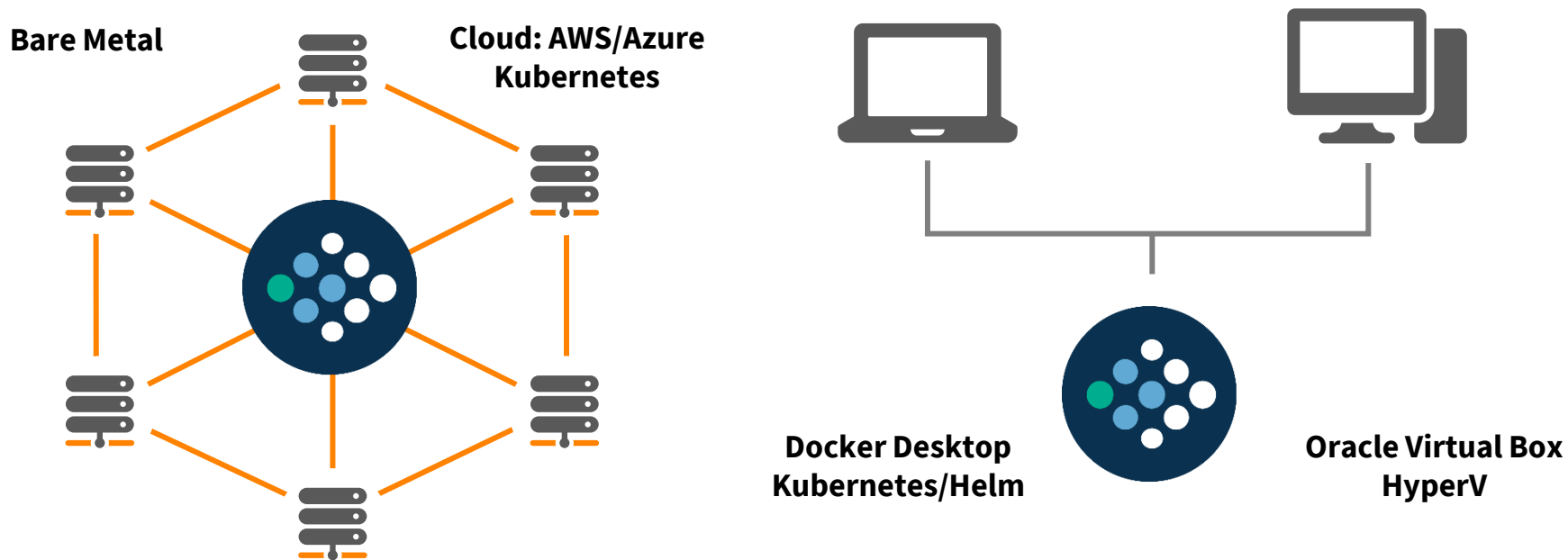
Conectividade

Plugins de integração com outros sistemas

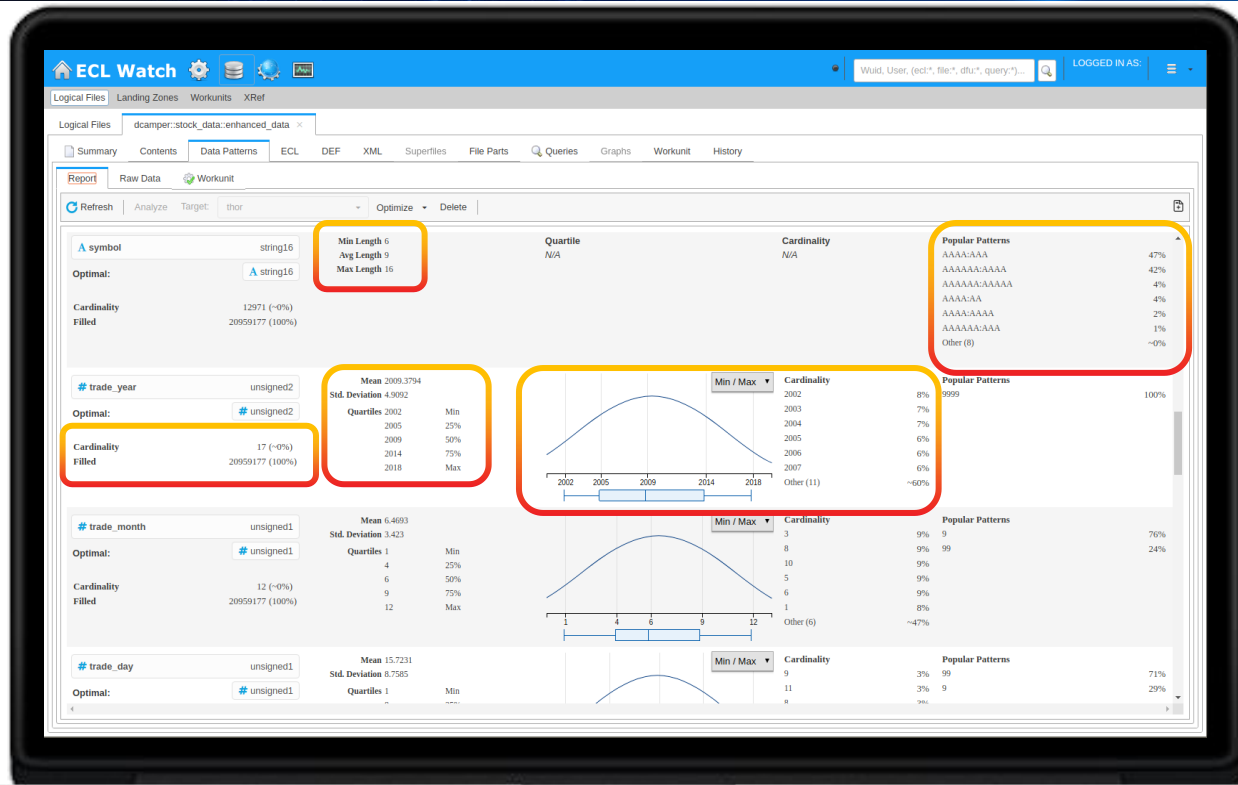
Os componentes da plataforma



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



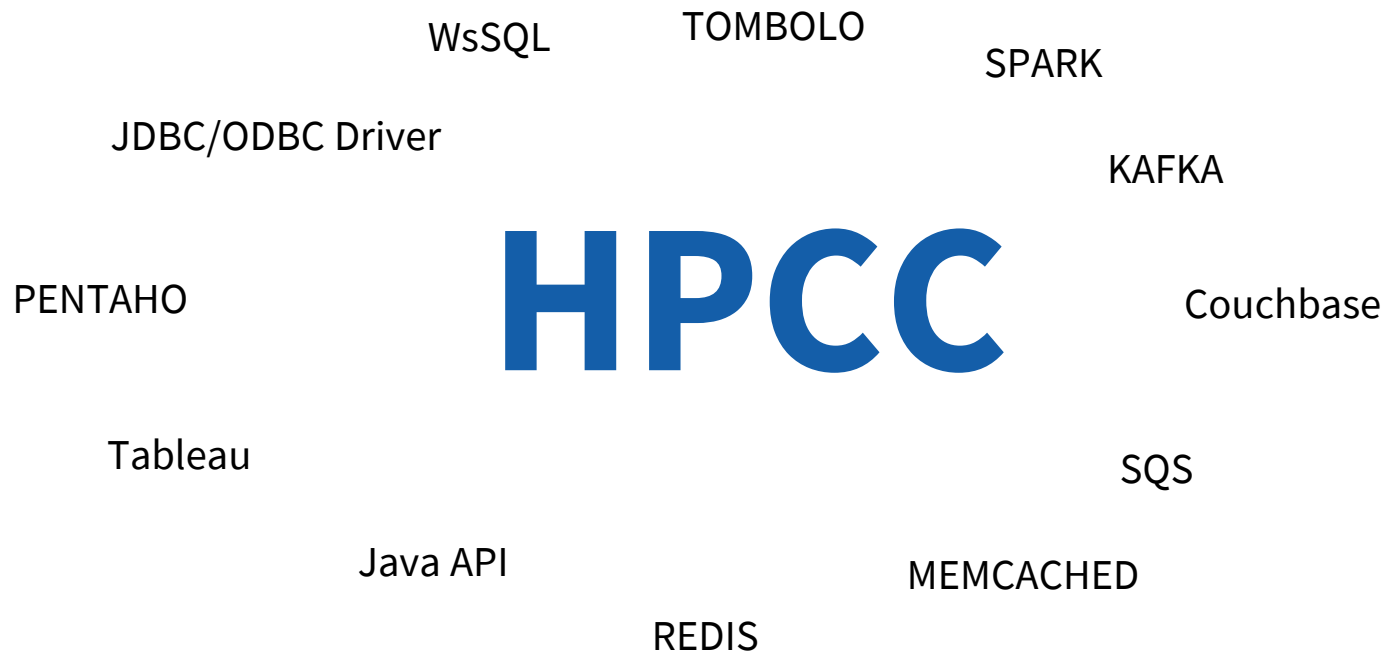
Métodos ensemble

Random Forest

Gradient Boosted
Forest

Gradient Boosted
Trees

Plugins para conectividade



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

```
1 // *****
2 // Elementos constituintes basicos da ECL
3 // Uma definicao
4 Mydef := 'Olá mundo'; // definicao do tipo "value"
5
6 // Uma acao
7 OUTPUT('Olá mundo');
8 OUTPUT(mydef);
9
10 // *****
11 // Estruturas de dados basicas em ECL
12 // Estrutura RECORD
13 rec := RECORD
14   STRING10 Firstname;
15   STRING   Lastname;
16   STRING1  Gender;
17   UNSIGNED1 Age;
18   INTEGER  Balance;
19   DECIMAL7_2 Income;
20 END;
21
22 // Declaracao DATASET
23 ds := DATASET([{'Alysson','Oliveira','M',26,100,1000.50},
24               {'Bruno','Camargo','',22,-100,500.00},
25               {'Elaine','Silva','F',19,-50,750.60},
26               {'Julia','Caetano','F',45,500,5000},
27               {'Orlando','Silva','U',67,300,4000}],rec);
28 OUTPUT(ds);
```

```
35 //Filtragem de datasets
36 recset := ds(Age<65);
37 recset := ds(Age<65,Gender='M');
38 IsSeniorMale := ds.Age>65 AND ds.Gender='M'; //definição do tipo "boolean"
39
40 SetGender := ['M','F']; //definicao do tipo "set"
41 recset := ds(Gender IN SetGender);
42 recset; // definicao do tipo "recordset"
43 COUNT(recset); //Equivale a: OUTPUT(COUNT(recset));
44
45 *****
46 //Transformacoes basicas em ECL
47 //Eliminacao de campos desnecessarios
48 tbl := TABLE(ds,{Firstname,LastName,Income});
49 tbl;
50
51 // Ordenacao de valores
52 sorttbl := SORT(tbl,LastName);
53 sorttbl;
54
55 // Remocao de duplicidades
56 dedptbl := DEDUP(sorttbl,LastName);
57 dedptbl;
58
59 // Adicao de campo no dataset
60 rec2 := RECORD
61   UNSIGNED   recid;
62   STRING10   Firstname;
63   STRING     Lastname;
64   STRING1    Gender;
65   UNSIGNED1  Age;
66   INTEGER    Balance;
67   DECIMAL7_2 Income;
68 END;
69 rec2 MyTransf(rec Le, UNSIGNED cnt) := TRANSFORM
70   SELF.recid:=cnt;
71   SELF := Le;
72 END;
73 newds := PROJECT(ds,MyTransf(LEFT,COUNTER));
```

Macros nativas

- Processo de ingestão para qualquer conjunto de registros

```
1  FM_Upper(Ds):= FUNCTIONMACRO
2      #EXPORTXML(Data, RECORDOF(Ds));
3      #DECLARE(newrecord)
4      #SET(newrecord, 'newrecord := RECORD \n')
5      #FOR(Data)
6          #FOR(Field)
7              #IF('%@type%' = 'string')
8                  #APPEND(newrecord, '%@ecltype%' + ' '+'@label%' + ':=' std.Str
9              #ELSE
10                 #APPEND(newrecord, #TEXT(Ds) + ' . '+'@label%' + ';\n')
11             #END
12         #END
13     #END
14     #APPEND(newrecord, '\nEND;')
15     #APPEND(newrecord, '\nsaida := TABLE('+#TEXT(Ds)+', newrecord); \n')
16 RETURN '%newrecord%';
17 ENDMACRO;
18
19 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



DOCUMENTATION 

BEST PRACTICES 

R&D BLOG 

TOUR 

SETTINGS 

CONTACT US 

DISCLAIMER 

WHILE THERE ARE SAFEGUARDS IN PLACE, THE SYSTEM MAY OCCASIONALLY GENERATE INCORRECT OR MISLEADING INFORMATION AND PRODUCE OFFENSIVE OR BIASED CONTENT. IT IS NOT INTENDED TO GIVE ADVICE.



Write an initial code for a data ingestion action for a dataset with 12 different definitions, being each one of them a different month of the year



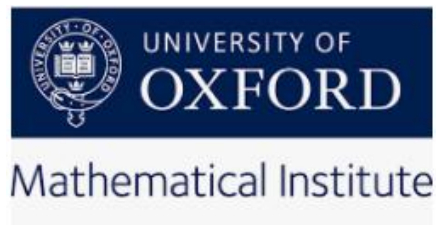
```
EXPORT monthsIngestion := MODULE
  EXPORT months := RECORD
    STRING Jan;
    STRING Feb;
    STRING Mar;
    STRING Apr;
    STRING May;
    STRING Jun;
    STRING Jul;
    STRING Aug;
    STRING Sep;
    STRING Oct;
    STRING Nov;
    STRING Dec;
  END;
  EXPORT ds := DATASET(yourFile,months,CSV);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

Relacionamento com Academia

Universidade de São Paulo
Brasil



Imperial College
London



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



- Co-Orientação de IC's
- Co-Orientação de TCC's e Metrados
 - Artigos internacionais (IoTSMMS-2025 - Lyon, France)
 - Artigos publicados (ERAD/RS, CotB, Fusion, etc)
 - Apresentações no HPCC Summit
- Compra de equipamentos



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

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 9] (Oferecido por LexisNexis) Processamento e Análise de Big Data e Aplicação de Algoritmos de Machine Learning através da utilização da Plataforma HPCC Systems

Resumo: O minicurso apresenta os conceitos essenciais de processamento e análise de grandes volumes de dados (Big Data) e do desenvolvimento de serviços de consulta utilizando a plataforma open-source HPCC Systems, composta por um cluster computacional de alto desempenho. Os participantes também aprenderão a aplicar algoritmos de Aprendizado de Máquina nesse ambiente e terão a oportunidade de praticar os conhecimentos adquiridos em um ambiente de treinamento disponibilizado em sala de aula.

Formato: Teórico e Prático (1h30)

Pré-requisitos: Curso básico e introdutório. Necessário uma conta no GitHub.

Horário e Local: 13/11 (8:30 - 10:00) — LAB 102/Prédio 43413

Vagas: 40

Ministrante



Alysson Oliveira (LexisNexis Risk Solutions)

Alysson Oliveira é engenheiro de software na LexisNexis Risk Solutions, formado pela USP e UTFPR. Atua principalmente no suporte e desenvolvimento de programas de treinamento para a plataforma HPCC Systems no Brasil, atendendo públicos acadêmicos, pesquisadores, profissionais de computação e dados, além de pessoas interessadas em pivotar suas carreiras. Também busca estabelecer parcerias com universidades para oferecer aos alunos de graduação e mestrado oportunidades de desenvolver projetos científicos aplicados.



Considerações Finais & Perguntas



- Alysson.Oliveira@lexisnexisrisk.com



- Mauro.marques@lexisnexisrisk.com

