



# **Wendelin Big Data**

*Industrial Monitoring Platform*

2014-04-03 – Paris

# Who are we?

- **Jean-Paul Smets**
- **Nexedi CEO**
- **Author of ERP5**
- **jp@nexedi.com**
- **Ivan Tyagov**
- **Senior Developer**
- **Wendelin project lead**
- **ivan@nexedi.com**

# Who is missing?

- **Kirill Smelkov**
- **Senior Developer**
- **wendelin.core**
- **Sebastien Robin**
- **Project Director**
- **Author of POC**

# Agenda

- **Where do we come from**
- **Wendelin Architecture**
- **Detailed Example**
- **Future Roadmap**

# Where do we come from?

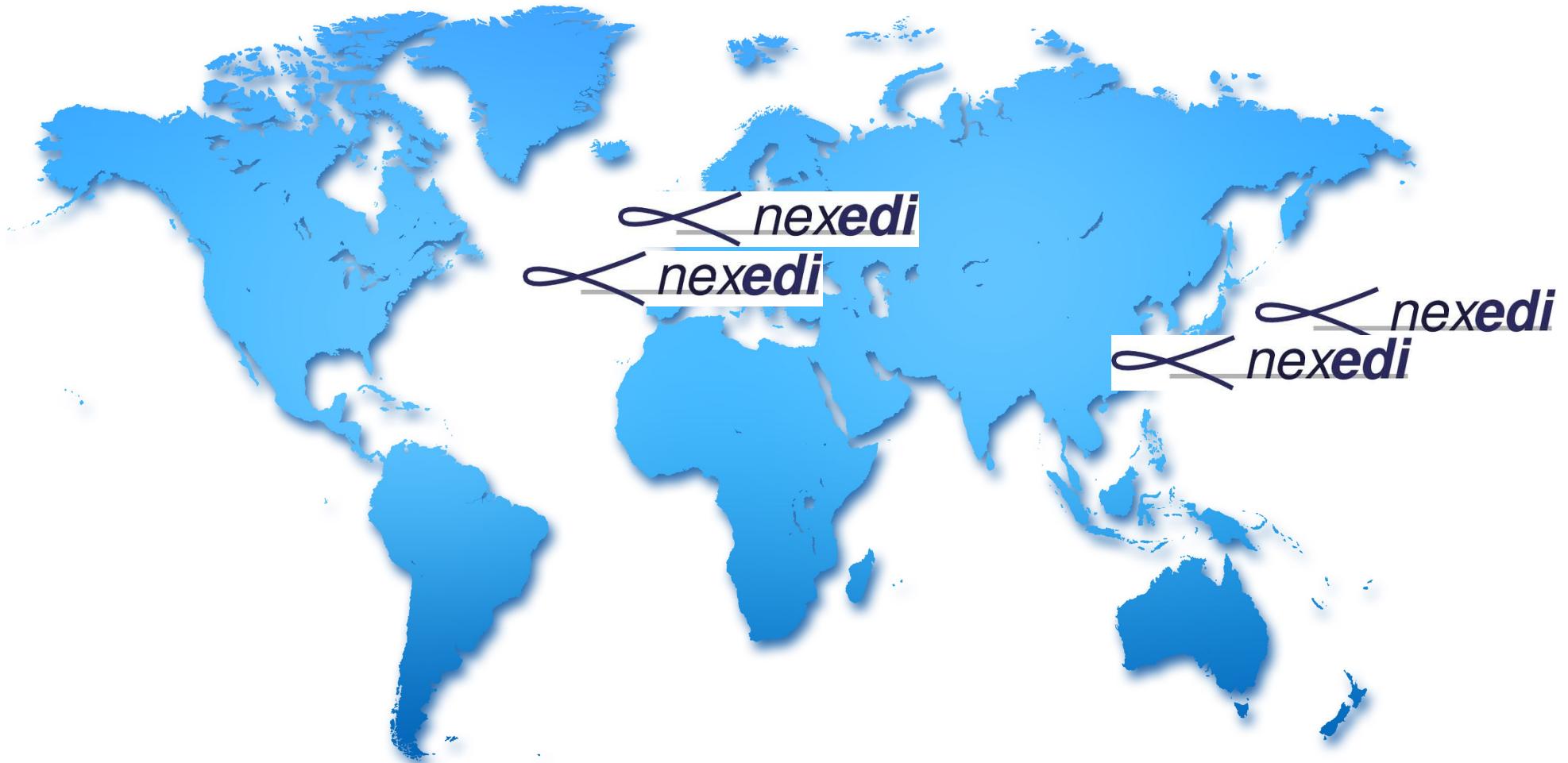


# Nexedi

- **Possibly Largest OSS Publisher in Europe**

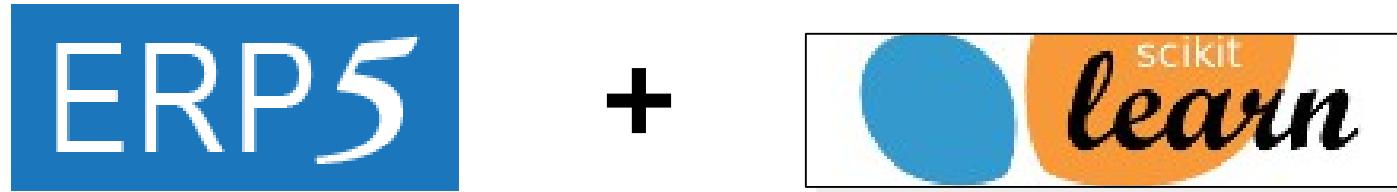
- **ERP5: ERP, CRM, ECM, e-business framework**
- **SlapOS: distributed mesh cloud operation system**
- **NEO: distributed transactional NoSQL database**
- **Wendelin: out-of-core big data based on NumPy**
- **re6st: resilient IPv6 mesh overlay network**
- **RenderJS: javascript component system**
- **JIO: javascript virtual database and virtual filesystem**
- **cloudooo: multimedia conversion server**
- **Web Runner: web based Platform-as-a-Service (PaaS) and IDE**
- **OfficeJS: web office suite based on RenderJS and JIO**







# Application Convergence



?

# Case 1: Wind Turbines



- Collect logs
- Collect records
- Predict failure
- Plan maintenance
- Reduce downtime  
→ add X% profits

ERP5

scikit  
learn

ERP5

# Case 2: Cars



- Collect logs
- Collect records
- Predict failure
- Plan maintenance
- Reduce downtime  
→ increase loyalty



# Case 3: Solar Energy



- Collect logs
- Collect records
- Predict degradation
- Plan maintenance
- Increase efficiency  
→ add X% to profits

ERP5

scikit  
learn

ERP5

# Wendelin Architecture



# Standard Hardware no router / no SAN



x 160

- 2 x 10 Gbps
- 2 x 6 core Xeon CPU
- 512 GB RAM
- 4 x 1 TB SSD
- 1 x M2090 GPU

+



x 32

- 10 Gbps
- Unmanaged

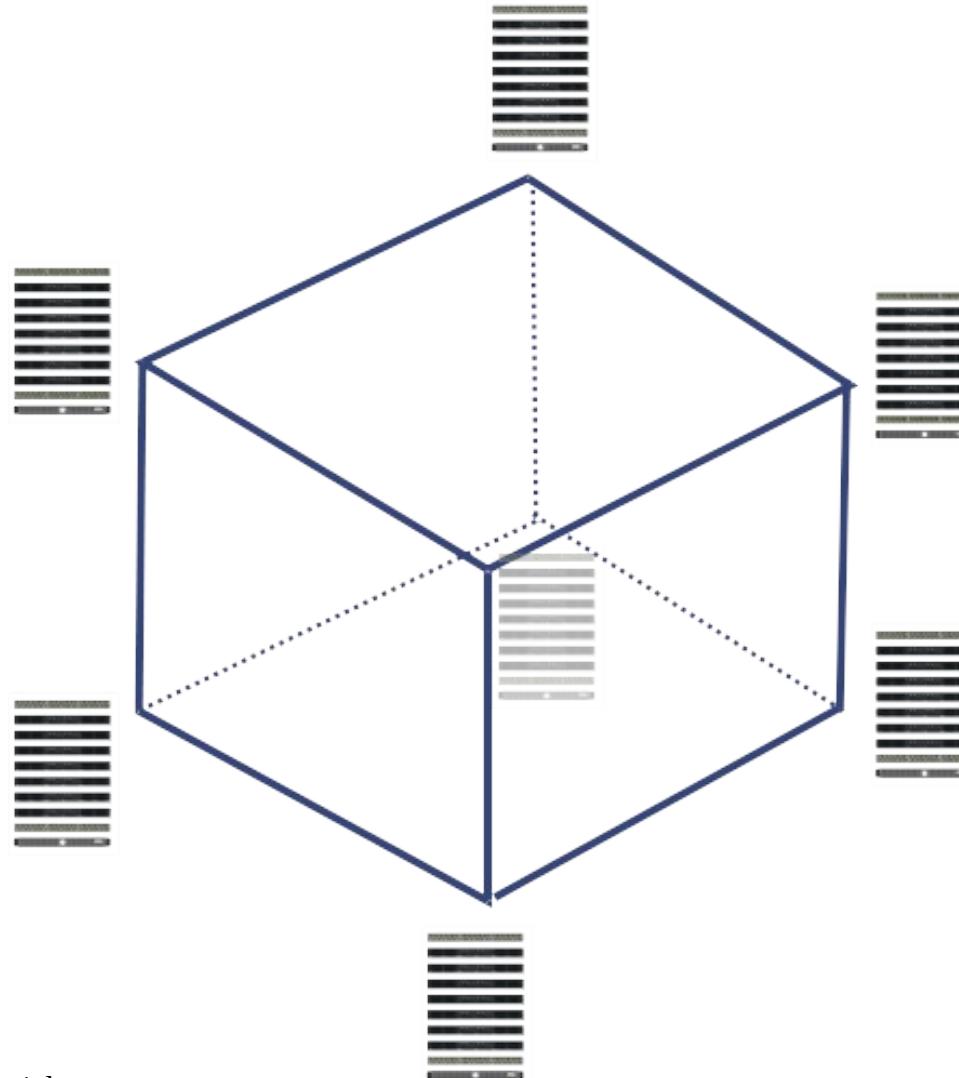
+



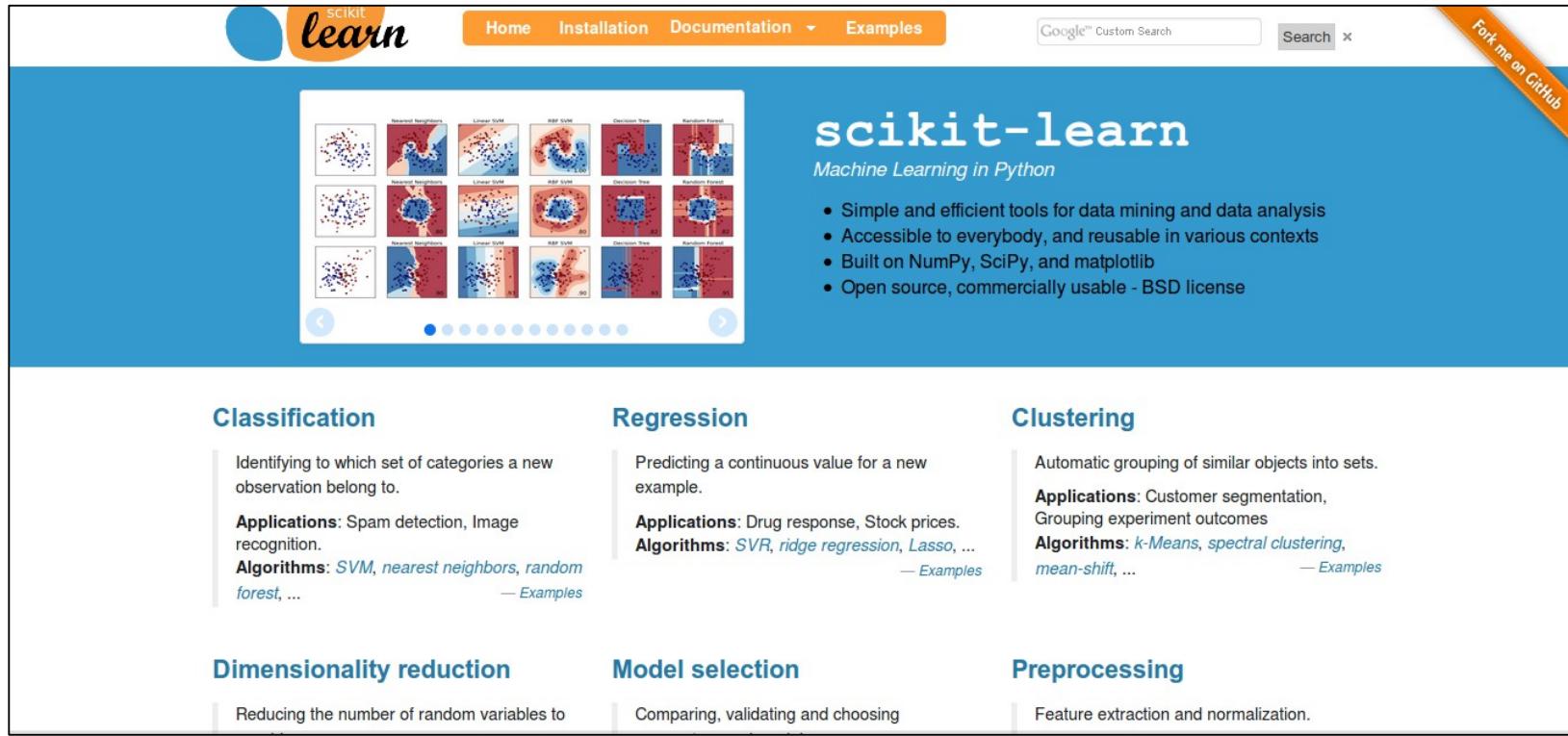
x 320

 nexedi

# Wendelin Hypercube Datacenter



# Take the Best Analytics [scikit-learn.org](http://scikit-learn.org)



The screenshot shows the official scikit-learn website. At the top, there's a navigation bar with links for Home, Installation, Documentation, Examples, a Google Custom Search bar, and a search button. A "Fork me on GitHub" badge is in the top right. The main header features the scikit-learn logo and the text "Machine Learning in Python". Below the header, there's a grid of nine small plots illustrating different machine learning models like K-Means, Linear SVM, and Random Forest. To the right of these plots is a bulleted list of features: Simple and efficient tools for data mining and data analysis; Accessible to everybody, and reusable in various contexts; Built on NumPy, SciPy, and matplotlib; and Open source, commercially usable - BSD license.

**Classification**  
Identifying to which set of categories a new observation belongs to.  
**Applications:** Spam detection, Image recognition.  
**Algorithms:** *SVM, nearest neighbors, random forest, ...*

[— Examples](#)

**Regression**  
Predicting a continuous value for a new example.  
**Applications:** Drug response, Stock prices.  
**Algorithms:** *SVR, ridge regression, Lasso, ...*

[— Examples](#)

**Clustering**  
Automatic grouping of similar objects into sets.  
**Applications:** Customer segmentation, Grouping experiment outcomes  
**Algorithms:** *k-Means, spectral clustering, mean-shift, ...*

[— Examples](#)

**Dimensionality reduction**  
Reducing the number of random variables to

**Model selection**  
Comparing, validating and choosing

**Preprocessing**  
Feature extraction and normalization.



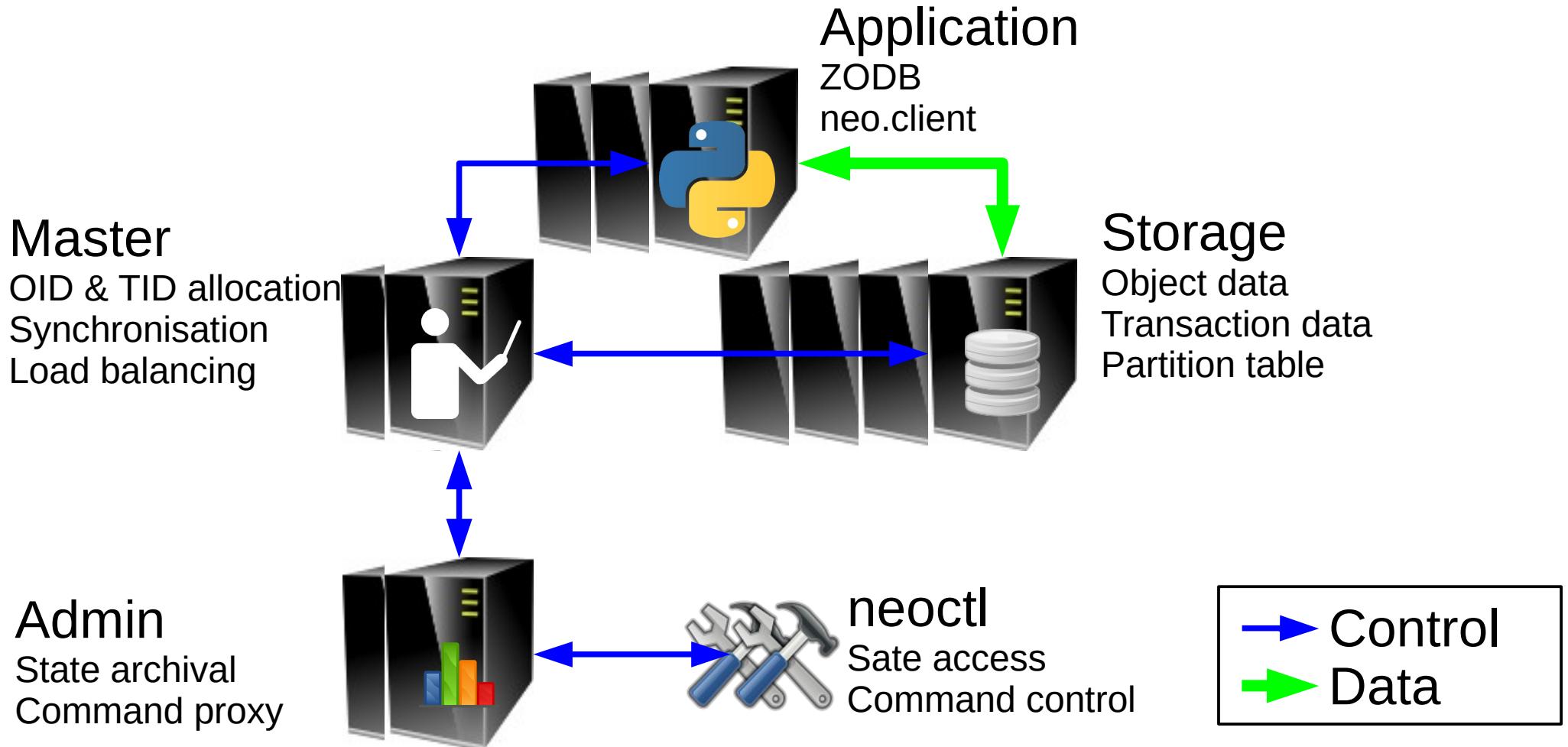
Spotify®

Google

cloudera

nexedi

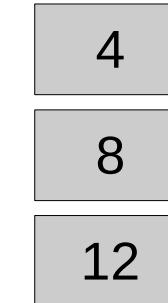
# Add Distributed Storage [neoppod.org](http://neoppod.org)



# “Magic” out-of-core for NumPy

PyData Paris 2015 – 16h45 Kirill Smelkov

ZBigArray



# Add Elastic PaaS

erp5.com

```
# Initialize data
data_size = 1000000
server_count = 1000
chunk_size = data_size / server_count
data = array(data_size)

# Process data in parallel on each server (Map Reduce, Batch, etc.)
for server in server_count:
    data.activate().process(server*chunk_size, chunk_size)
```

ERP5 PaaS

# And Multicloud Deployment [slapos.org](http://slapos.org)

Update software profile - helloworld (workspace/my\_repository/software)

Profiles Workspace Run software Software log SR result Run instance Instance log Inspect instance Destroy instance

Edit software profile [More]:

```
0 [buildout]
1
2 extends =
3 # "slapos" stack describes basic things needed for 99.9% of SlapOS Software
4 # Releases
5     ../../stack/slapos.cfg
6 # Extend here component profiles, like openssl, apache, mariadb, curl...
7 # Or/and extend a stack (lamp, tomcat) that does most of the work for you
8 # In this example we only need the dash binary to run a simple "hello world"
9 # shell script.
10    ../../component/dash/buildout.cfg
11
12 parts =
13 # Call installation of slapos.cookbook egg defined in stack/slapos.cfg (needed
14 # in 99.9% of SlapOS Software Releases)
15     slapos-cookbook
16 # Call creation of instance.cfg file that will be called for deployment of
17 # instance
18     template
19
20 # Download instance.cfg.in (buildout profile used to deployment of instance),
21 # replace all ${foo:bar} parameters by real values, and change $$${foo:bar} to
22 # ${${foo:bar}}
23     [template]
24     recipe = slapos.recipe.template
25     url = ${:_profile_base_location_}/instance.cfg.in
26     output = ${buildout:directory}/instance.cfg
27 # MD5 checksum can be skipped for development (easier to develop), but must be filled for production
28     md5sum = 1fc461c00e86485bee77a942f39e3c43
29     mode = 0644
30
```

Save

The SLAPOS logo features a stylized green 'S' shape above the word 'SLAPOS' in a bold, sans-serif font.

# Wendelin Platform 100% open source

100% Python

Scikit Learn

NEO

ERP5

SlapOS



Data Analytics

Distributed Storage



Elastic PaaS

Multicloud Deployment

Multi Data Center

# Wendelin Options 100% open source

100% Python

OpenCV-Python

NLTK

Blaze

Numba / Parakeet

Pandas

Scikit Learn

NEO

**Video Processing**

Intel Russia / Willow / Itseez

**Natural Language Toolkit**

U. Texas / Chalmers

**Full out-of-core arrays**

Continuum / DARPA

**JIT compiler / type inference**

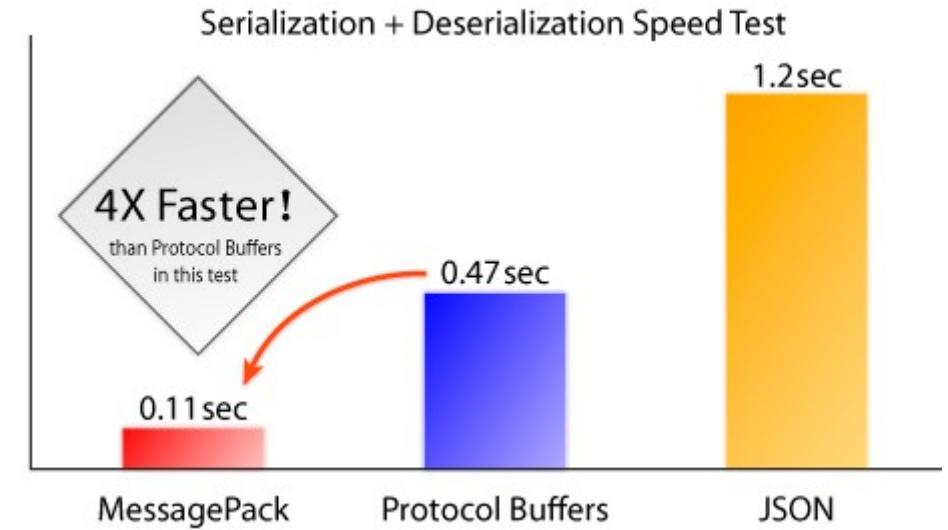
Continuum / DARPA

**Time sequence processing**

DataPad / JP Morgan

# Data Ingestion: fluentd

- Based on MsgPack middleware
- Created by TreasureData (BaaS pioneers)
- Used by Amazon
- Numerous plugins
- Scalable and resilient
- Bandwidth saver



# Wendelin UI



- **HTML5 Render** [RenderJS](#)
- **Data vizualisation**
- **Offline support** [JIO](#)
- **Data access** [REST API](#)
- **Batch processing**

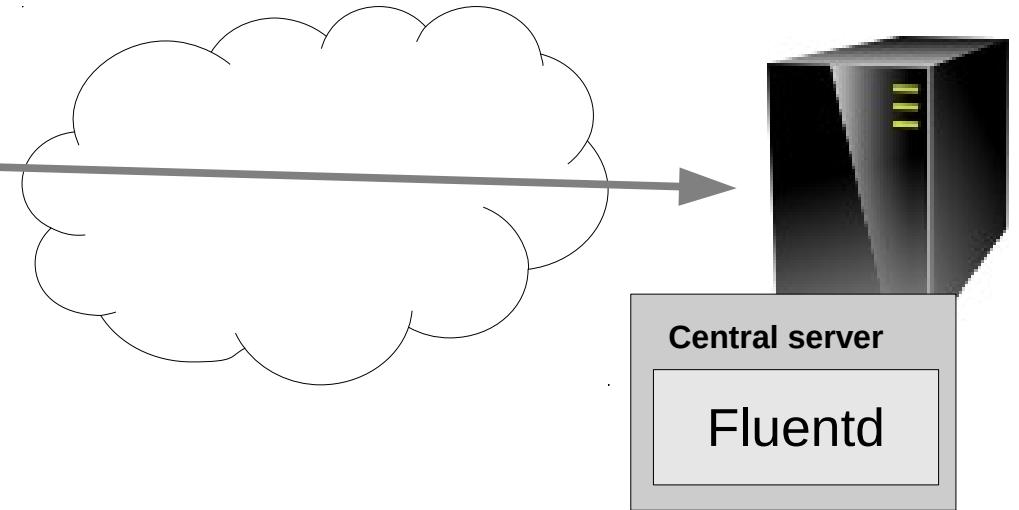
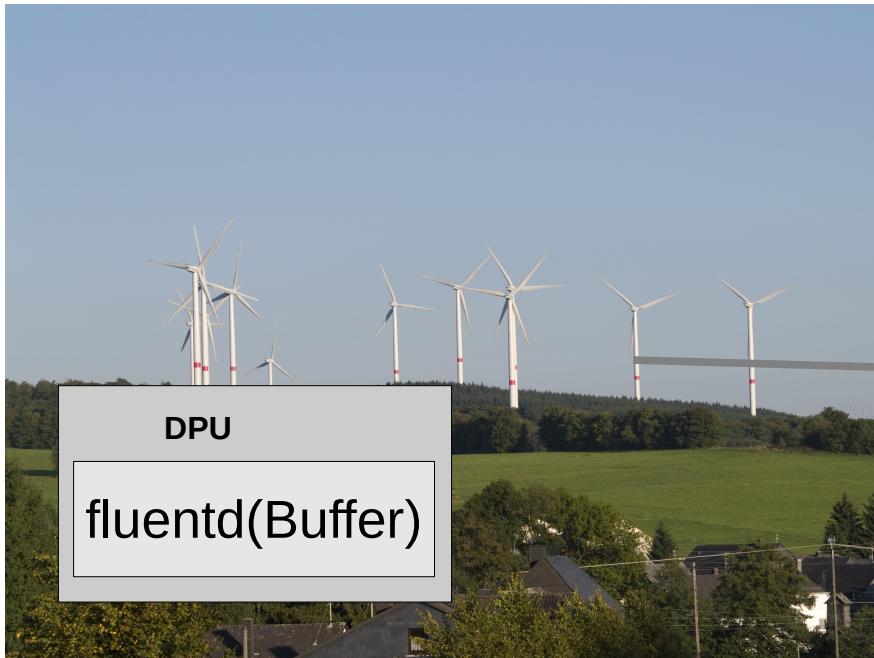
# Wendelin Distinctive Advantages

- Native out-of-core NumPy (scikit-learn, pydata)
- Native parallel processing
- Bare metal performance (GPU, FORTRAN)
- Transactions (ingestion, processing)
- NewSQL queries
- Built-in PaaS
- Lower deployment cost (10x less than...)

# Detailed Example



# Data Transportation **fluentd**



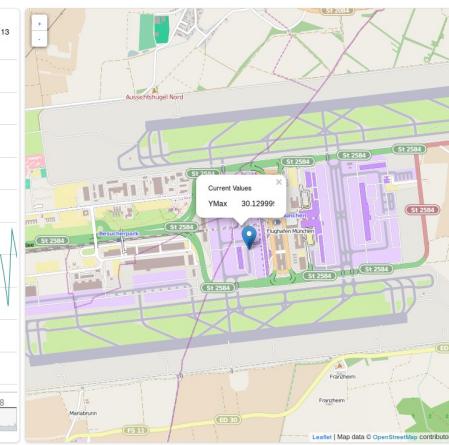
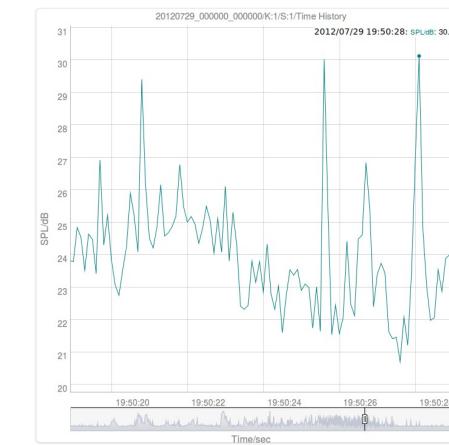
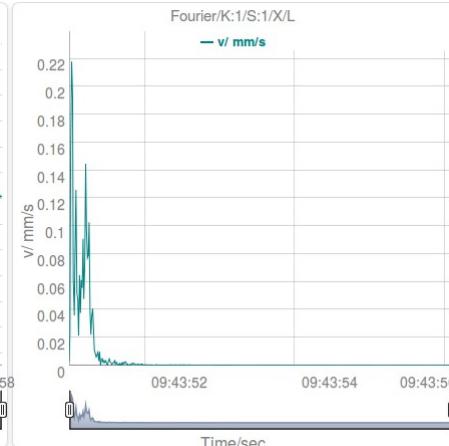
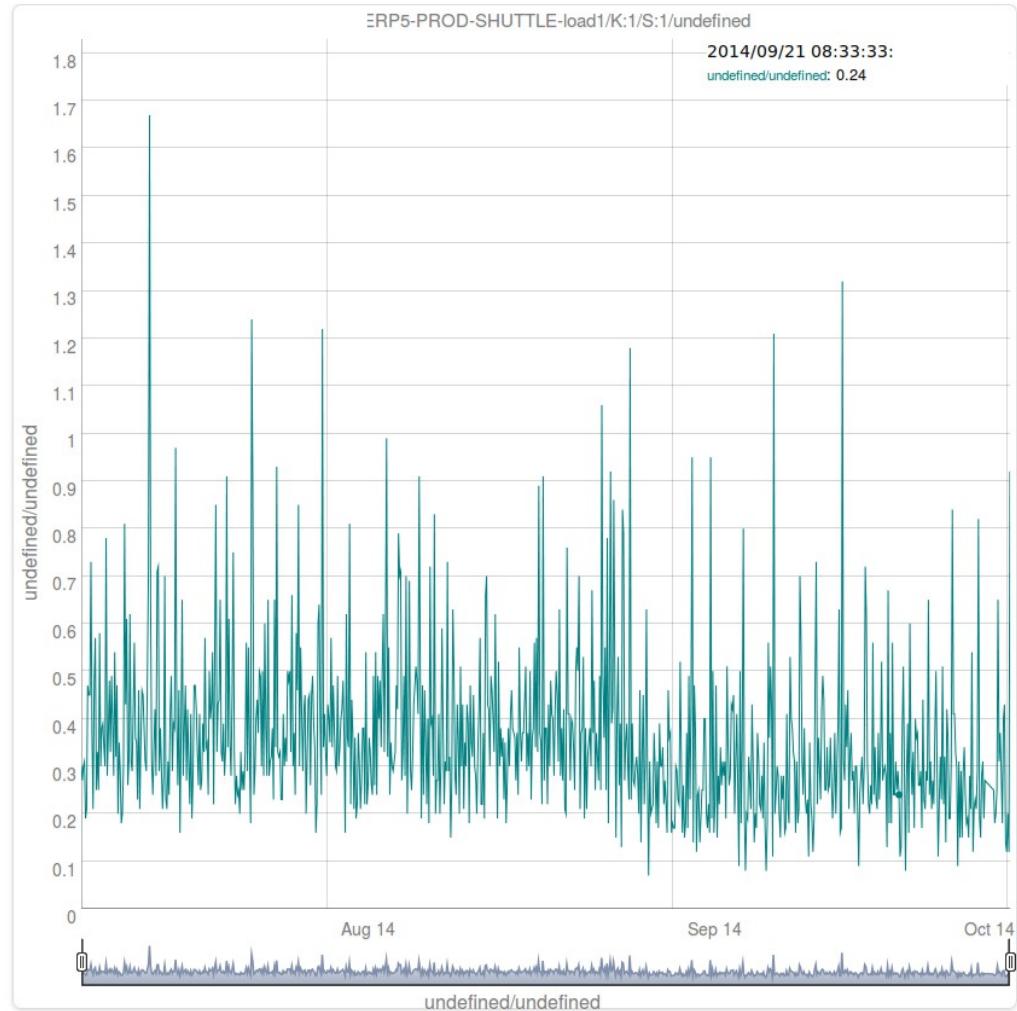
3 months benchmark  
Frequent downtime (server, network)  
Very poor networking (ADSL, 3G)

< 0.001% loss

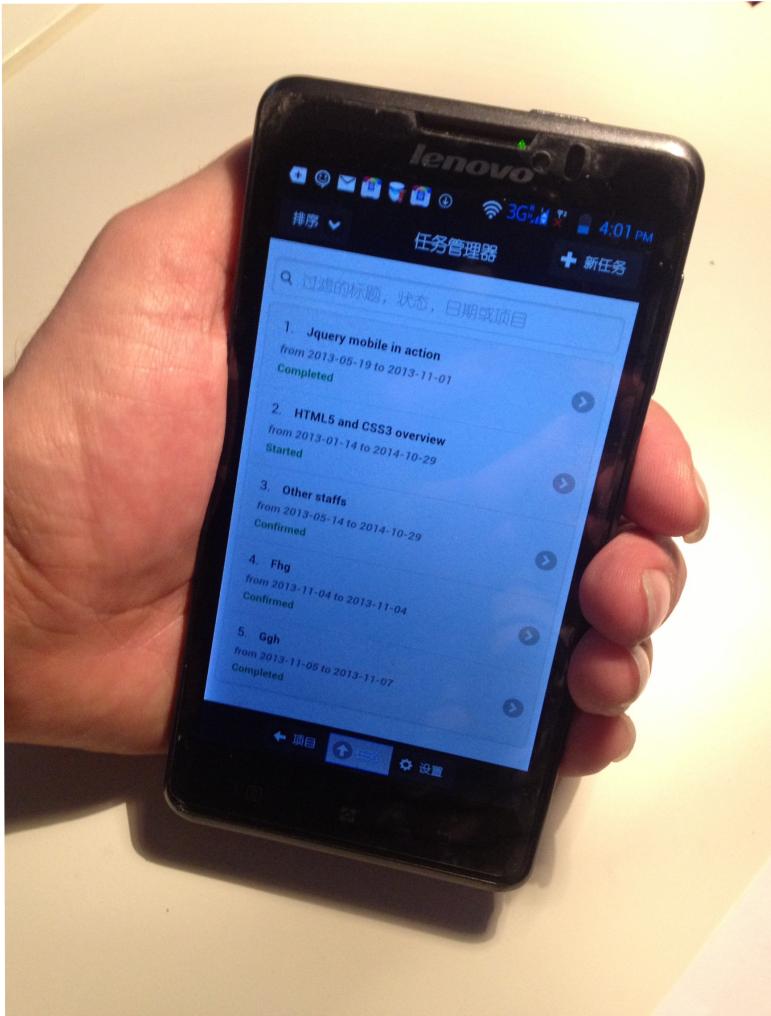
# UI: HTML5 Components RenderJS



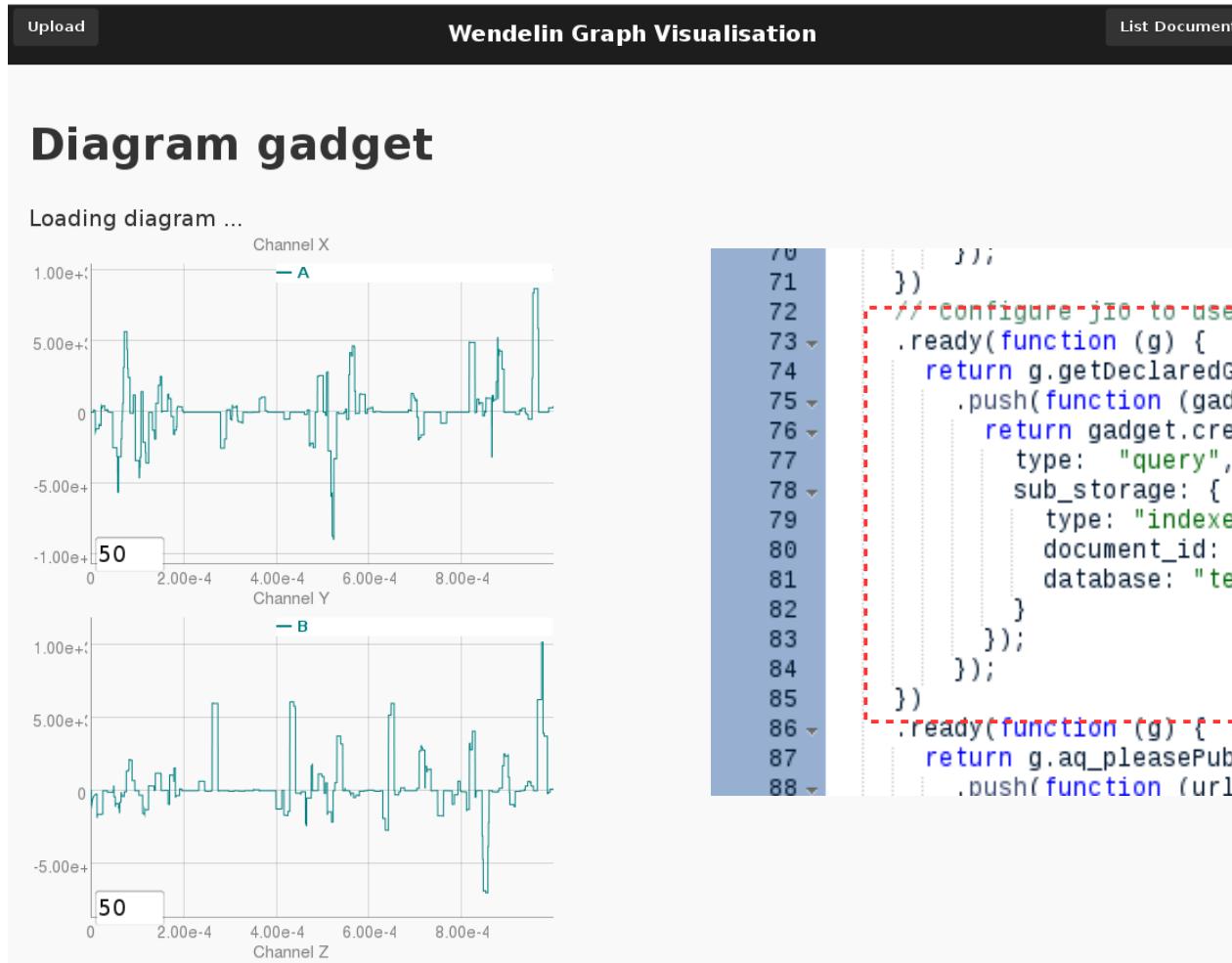
# Extend UI Components RenderJS



# UI : Responsive RenderJS

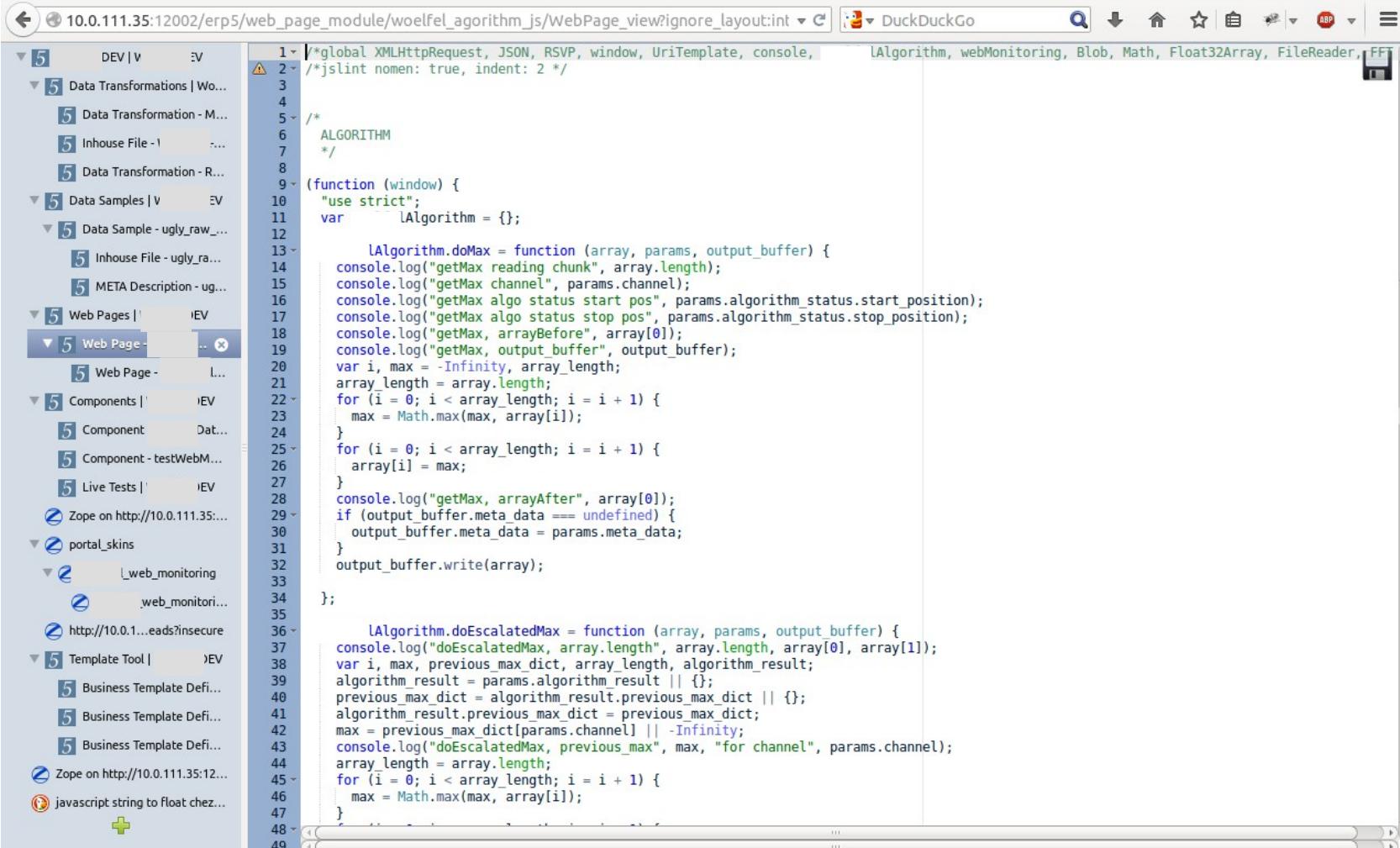


# UI: Offline / Other Backends JIO



```
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
    });
    //configure jio to use localstorage
    .ready(function (g) {
        return g.getDeclaredGadget("JIO")
            .push(function (gadget) {
                return gadget.createJio({
                    type: "query",
                    sub_storage: {
                        type: "indexeddb",
                        document_id: "/",
                        database: "test_ivan"
                    }
                });
            });
    });
    .ready(function (g) {
        return g.aq_pleasePublishMyState({page: 'listbox'})
            .push(function (url) {
```

# Data Science in Javascript vs. Python ?



The screenshot shows a web browser window with two main panes. The left pane is a file tree navigation interface, likely from a Zope or Plone application. The right pane is a code editor displaying a JavaScript file.

**File Tree (Left):**

- DEV | V EV
- 5 Data Transformations | Wo...
- 5 Data Transformation - M...
- 5 Inhouse File - I ...
- 5 Data Transformation - R...
- 5 Data Samples | V EV
- 5 Data Sample - ugly\_raw\_...
- 5 Inhouse File - ugly\_ra...
- 5 META Description - ug...
- 5 Web Pages | ! IEV
- 5 Web Page - L...
- 5 Components | ! EV
- 5 Component Dat...
- 5 Component - testWebM...
- 5 Live Tests | ! EV
- Zope on http://10.0.111.35...
- portal\_skins
- \_web\_monitoring
- .web\_monitori...
- http://10.0.1...eads?insecure
- Template Tool | ! EV
- 5 Business Template Defi...
- 5 Business Template Defi...
- 5 Business Template Defi...
- Zope on http://10.0.111.35:12...
- javascript string to float chez...

**Code Editor (Right):**

```
/*global XMLHttpRequest, JSON, RSVP, window, UriTemplate, console, lAlgorithm, webMonitoring, Blob, Math, Float32Array, FileReader, FFT */
/*jslint nomen: true, indent: 2 */

/*
ALGORITHM
*/
(function (window) {
    "use strict";
    var _lAlgorithm = {};

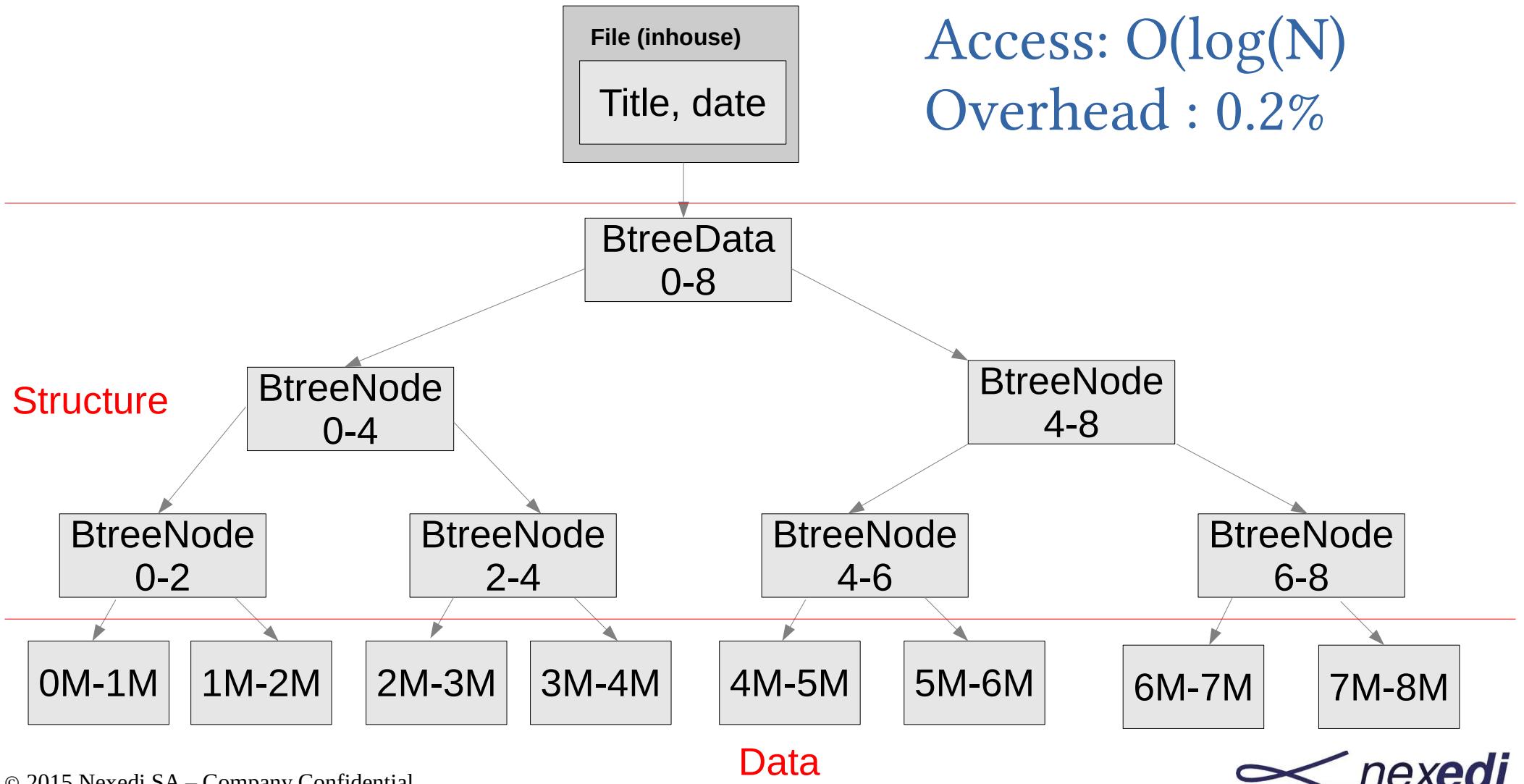
    _lAlgorithm.doMax = function (array, params, output_buffer) {
        console.log("getMax reading chunk", array.length);
        console.log("getMax channel", params.channel);
        console.log("getMax algo status start pos", params.algorithm_status.start_position);
        console.log("getMax algo status stop pos", params.algorithm_status.stop_position);
        console.log("getMax, arrayBefore", array[0]);
        console.log("getMax, output_buffer", output_buffer);
        var i, max = -Infinity, array_length;
        array_length = array.length;
        for (i = 0; i < array_length; i = i + 1) {
            max = Math.max(max, array[i]);
        }
        for (i = 0; i < array_length; i = i + 1) {
            array[i] = max;
        }
        console.log("getMax, arrayAfter", array[0]);
        if (output_buffer.meta_data === undefined) {
            output_buffer.meta_data = params.meta_data;
        }
        output_buffer.write(array);
    };

    _lAlgorithm.doEscalatedMax = function (array, params, output_buffer) {
        console.log("doEscalatedMax, array.length", array.length, array[0], array[1]);
        var i, max, previous_max_dict, array_length, algorithm_result;
        algorithm_result = params.algorithm_result || {};
        previous_max_dict = algorithm_result.previous_max_dict || {};
        algorithm_result.previous_max_dict = previous_max_dict;
        max = previous_max_dict[params.channel] || -Infinity;
        console.log("doEscalatedMax, previous_max", max, "for channel", params.channel);
        array_length = array.length;
        for (i = 0; i < array_length; i = i + 1) {
            max = Math.max(max, array[i]);
        }
    };
});
```

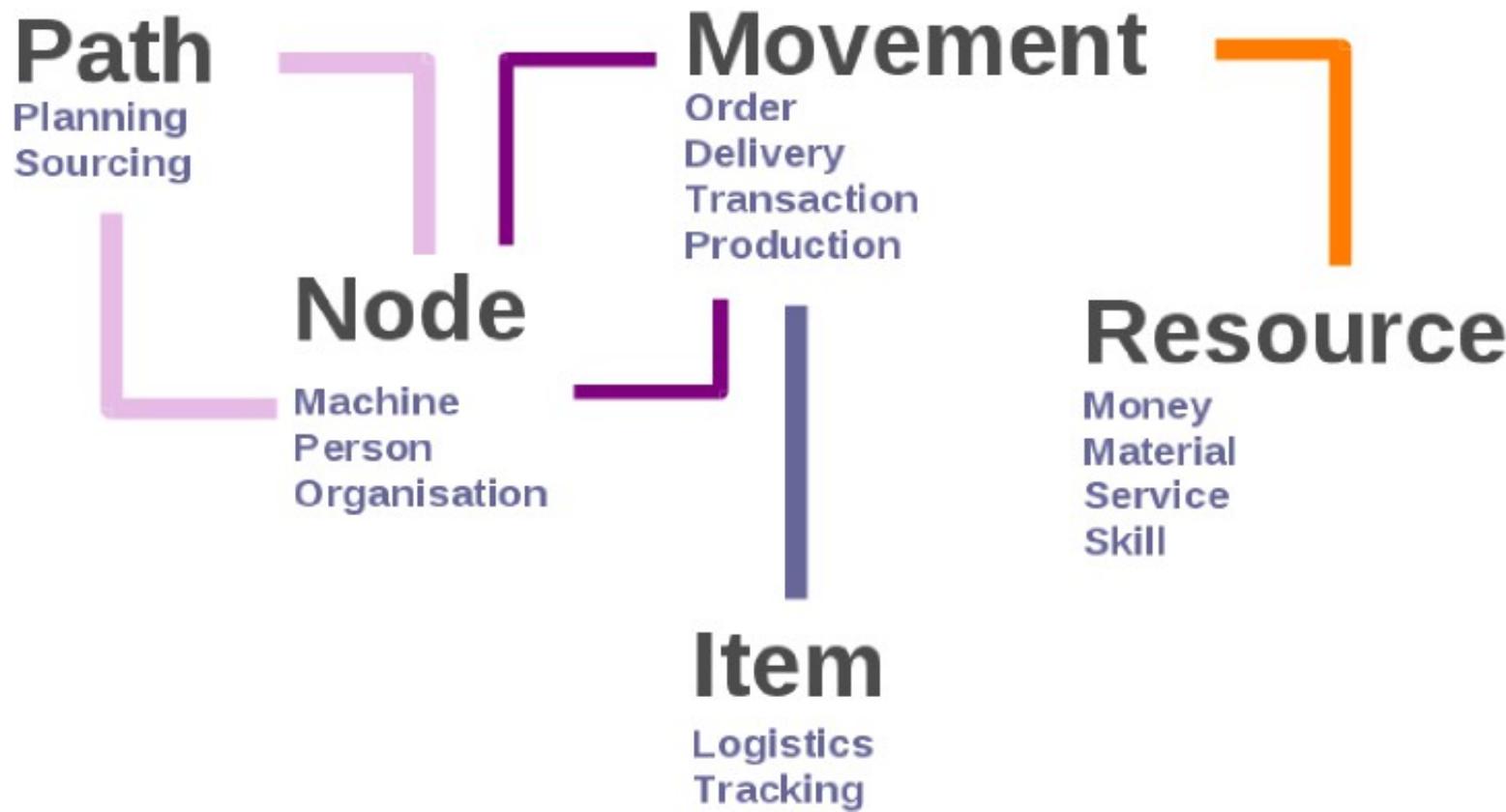
# Data Sciences in Javascript ? [phantomjs](#)

- **Small data on client side** 
- **Small data on server side** 
- **Medium data (> 1 GB) in JS** 
- **Out-of-core data in JS** 
- **PyData compiled in JS** 
- **PyData in NaCl / PNaCl** 

# Storing large streams in NEO



# UBM Monitoring Model?

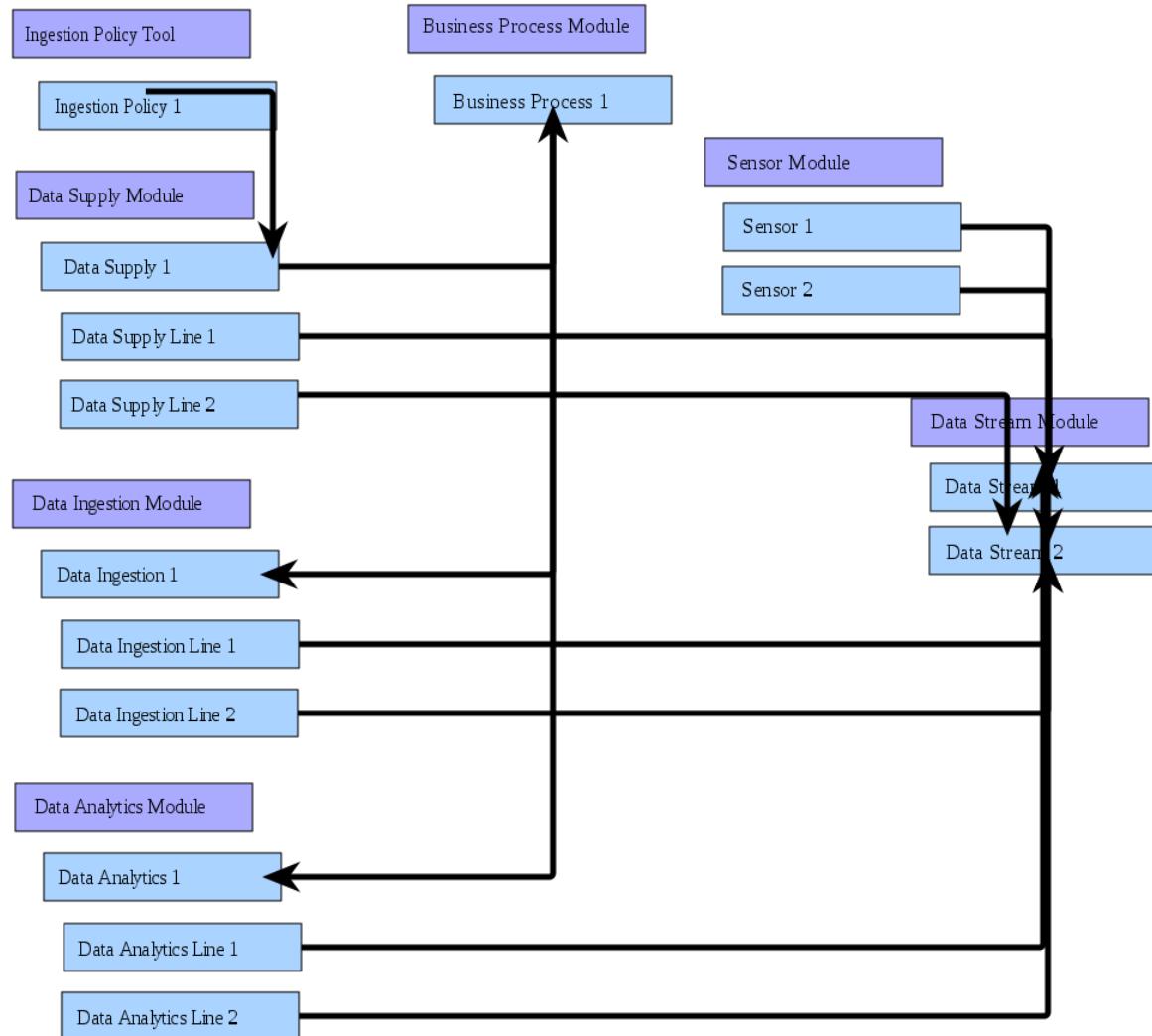


# UBM Business Model



- **Movement** – ingestion of data
- **Resource** – type of data (ex. memory log)
- **Node** – data source, data owner
- **Path** – data source registration
- **Item** – sensor, data itself, license, data set

# UBM Business Model



# What UBM gets us for free

- Accounting, billing and payment
- User registration and management
- Rule based security model
- Customer relationship management
- Web Content Management

→ save 12+ months and > 200 K€ on any Big Data project

# Future Roadmap



WENDELIN

# Roadmap

[www.wendelin.io](http://www.wendelin.io)

- **Mainly accelerate learning curve**

- Universal packaging
- Ready to use examples
- Act as a backend to ipython notebook
- Port joblib to CMFActivity

- **Yet, you can start using part of Wendelin now!**

- wendelin.core out-of-core for NumPy** PyData Paris 2015 – 16h45 Kirill Smelkov
- JIO abstract data access library**
- RenderJS components** <http://learn.renderjs.org>
- UI sample application** <https://lab.nexedi.cn/Tyagov/wendelin/>
- Open Source**

# R&D Partners

[www.wendelin.io](http://www.wendelin.io)

- Wendelin-IA (FSN)
  - Nexedi
  - Abilian
  - 2<sup>nd</sup> Quadrant
  - Paris 13
  - IMT
  - INRIA / ENS
  - MMC Rus (Ru)
  - X Corp
- Windelin (Eurostars)
  - Nexedi (FR)
  - MariaDB (FI)
  - Y Corp (DE)





# **Wendelin Big Data**

*Industrial Monitoring Platform*

2014-04-03 – Paris



**Wendelin Big Data**  
*Industrial Monitoring Platform*

2014-04-03 – Paris

© 2015 Nexedi SA – Company Confidential



# Who are we?

- **Jean-Paul Smets**
- **Nexedi CEO**
- **Author of ERP5**
- **[jp@nexedi.com](mailto:jp@nexedi.com)**
- **Ivan Tyagov**
- **Senior Developer**
- **Wendelin project lead**
- **[ivan@nexedi.com](mailto:ivan@nexedi.com)**

# Who is missing?

- **Kirill Smelkov**
- **Senior Developer**
- **wendelin.core**
- **Sebastien Robin**
- **Project Director**
- **Author of POC**

# Agenda

- **Where do we come from**
- **Wendelin Architecture**
- **Detailed Example**
- **Future Roadmap**

## Where do we come from?



© 2015 Nexedi SA – Company Confidential



The solution that was deployed at the Lightning Protection Center complies is based on open source software – with full access to source code – and does not use software made by IBM, Oracle or EMC. It is thus a “No IOE” compliant solution, in line with directives published by Chinese governments for certain markets.

# Nexedi

- Possibly Largest OSS Publisher in Europe

- ERP5: ERP, CRM, ECM, e-business framework
- SlapOS: distributed mesh cloud operation system
- NEO: distributed transactional NoSQL database
-  - Wendelin: out-of-core big data based on NumPy
- re6st: resilient IPv6 mesh overlay network
- RenderJS: javascript component system
- JIO: javascript virtual database and virtual filesystem
- cloudoooo: multimedia conversion server
- Web Runner: web based Platform-as-a-Service (PaaS) and IDE
- OfficeJS: web office suite based on RenderJS and JIO



© 2015 Nexedi SA – Company Confidential

 **nexedi**



# Application Convergence



?

## Case 1: Wind Turbines



- Collect logs
  - Collect records
  - Predict failure
  - Plan maintenance
  - Reduce downtime
- add X% profits

ERP5



ERP5

## Case 2: Cars



© 2015 Nexedi SA – Company Confidential

- Collect logs
- Collect records
- Predict failure
- Plan maintenance
- Reduce downtime  
→ increase loyalty

ERP5



ERP5

 *nexedi*

## Case 3: Solar Energy



- Collect logs
- Collect records
- Predict degradation
- Plan maintenance
- Increase efficiency  
→ add X% to profits

ERP5



ERP5

# Wendelin Architecture



© 2015 Nexedi SA – Company Confidential



The solution that was deployed at the Lightning Protection Center complies is based on open source software – with full access to source code – and does not use software made by IBM, Oracle or EMC. It is thus a “No IOE” compliant solution, in line with directives published by Chinese governments for certain markets.

## Standard Hardware no router / no SAN



x 160

- 2 x 10 Gbps
- 2 x 6 core Xeon CPU
- 512 GB RAM
- 4 x 1 TB SSD
- 1 x M2090 GPU

+



x 32

- 10 Gbps
- Unmanaged

+

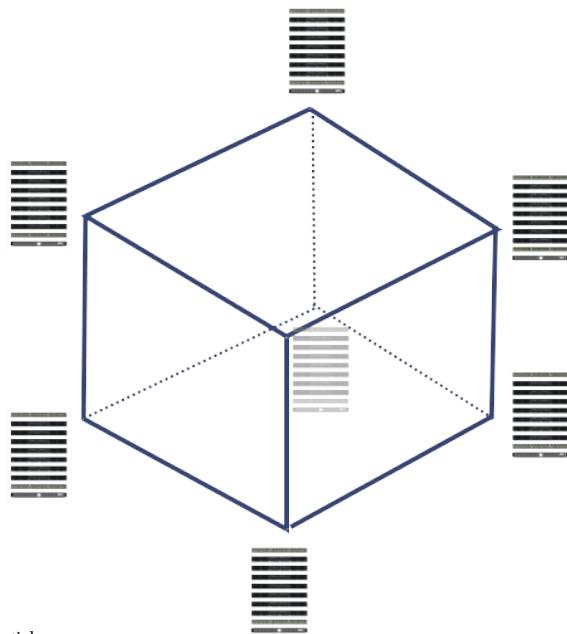


x 320

 nexedi

© 2015 Nexedi SA – Company

# Wendelin Hypercube Datacenter



© 2015 Nexedi SA – Company Confidential

 *nexedi*

# Take the Best Analytics [scikit-learn.org](http://scikit-learn.org)

The screenshot shows the official website for scikit-learn, a Python library for machine learning. The top navigation bar includes links for Home, Installation, Documentation, Examples, Google Custom Search, and a Search bar. A prominent orange banner on the right side features the text "Fork me on GitHub". The main content area is titled "scikit-learn Machine Learning in Python" and lists several categories of machine learning:

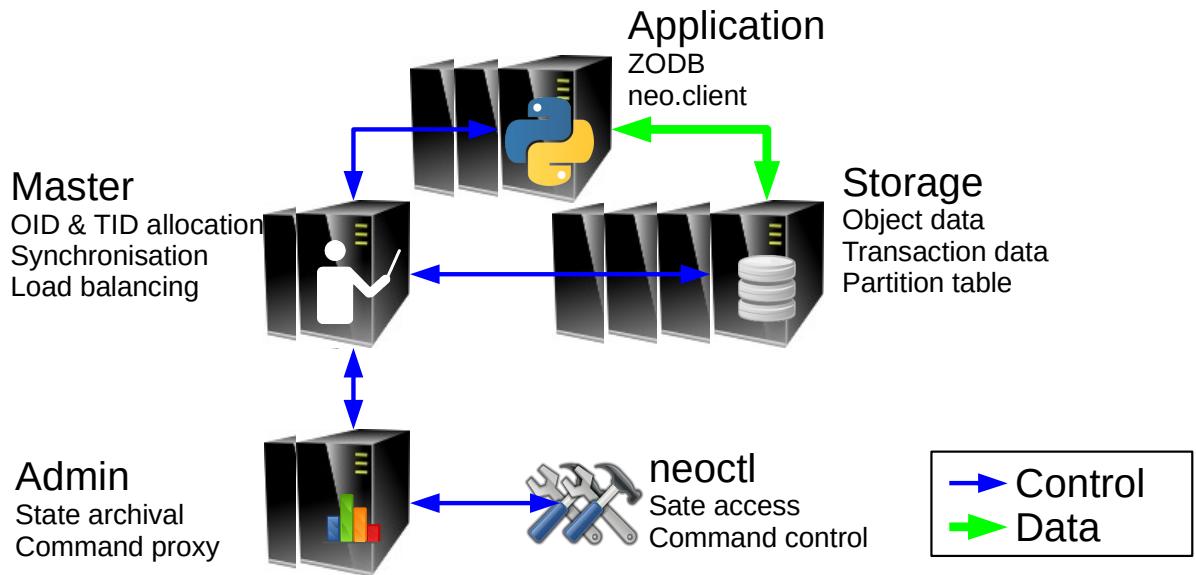
- Classification**: Identifying which set of categories a new observation belongs to. Applications include Spam detection, Image recognition. Algorithms listed are SVM, nearest neighbors, random forest, etc.
- Regression**: Predicting a continuous value for a new example. Applications include Drug response, Stock prices. Algorithms listed are SVR, ridge regression, Lasso, etc.
- Clustering**: Automatic grouping of similar objects into sets. Applications include Customer segmentation, Grouping experiment outcomes. Algorithms listed are k-Means, spectral clustering, mean-shift, etc.
- Dimensionality reduction**: Reducing the number of random variables to
- Model selection**: Comparing, validating and choosing
- Preprocessing**: Feature extraction and normalization.

Sponsor logos for Inria (Inventors for the Digital World) and TELECOM ParisTech are visible on the right side of the page.

© 2015 Nxedli SA – Company Confidential



# Add Distributed Storage [neoppod.org](http://neoppod.org)



© 2015 Nexedi SA – Company Confidential

 *nexedi*

# “Magic” out-of-core for NumPy

PyData Paris 2015 – 16h45 Kirill Smelkov

ZBigArray

1	2	3	4	5	6	7	8	9	10	11	12
---	---	---	---	---	---	---	---	---	----	----	----



1

5

9



2

6

10



3

7

11



4

8

12



© 2015 Nexedi SA – Company Confidential

 *nexedi*

## Add Elastic PaaS [erp5.com](http://erp5.com)

```
# Initialize data
data_size = 1000000
server_count = 1000
chunk_size = data_size / server_count
data = array(data_size)

# Process data in parallel on each server (Map Reduce, Batch, etc.)
for server in server_count:
    data.activate().process(server*chunk_size, chunk_size)
```

ERP5 PaaS

© 2015 Nexedi SA – Company Confidential



# And Multicloud Deployment [slapos.org](http://slapos.org)

Update software profile - helloworld (workspace/my\_repository/software)

Profiles Workspace Run software Software log SR result Run instance Instance log Inspect instance Destroy instance

Edit software profile [More]:

```

0 [buildout]
1
2 extends =
3 # "slapos" stack describes basic things needed for 99.9% of SlapOS Software
4 # Releases
5 # /stack/slapos.cfg
6 # Extend here component profiles, like openssl, apache, mariadb, curl...
7 # Or/and extend a stack (lamp, tomcat) that does most of the work for you
8 # In this example we only need the dash binary to run a simple "hello world"
9 # shell script.
10 .../component/dash/buildout.cfg
11
12 parts =
13 # Call installation of slapos.cookbook egg defined in stack/slapos.cfg (need
14 # in 99.9% of SlapOS Software Releases)
15 slapos-cookbook
16 # Call creation of instance.cfg file that will be called for deployment of
17 # instance
18 template
19
20 # Download instance.cfg.in (buildout profile used to deployment of instance),
21 # replace all ${foo:bar} parameters by real values, and change $$${foo:bar} to
22 # ${foo:bar}
23 [template]
24 slapos.recipe.template
25 url = ${profile_base_location_}/instance.cfg.in
26 output = ${buildout_directory}/instance.cfg
27 # MD5 checksum can be skipped for development (easier to develop), but must be filled for production
28 md5sum = 1fc461c00e86485bee77a942f39e3c43
29 mode = 0644
30

```

**Save**

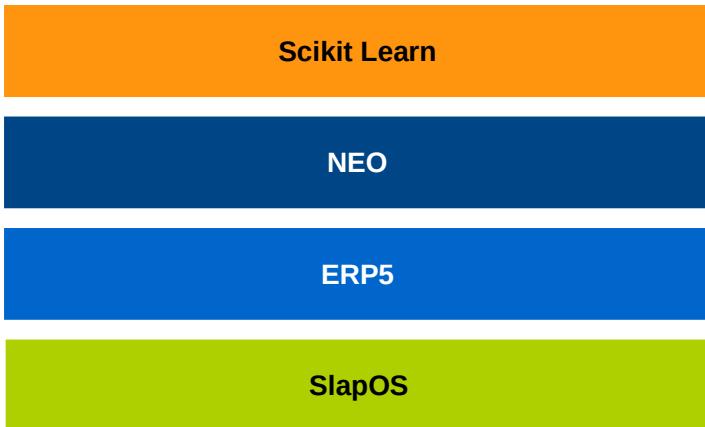


© 2015 Nexedi SA – Company Confidential



# Wendelin Platform 100% open source

100% Python



Data Analytics

Distributed Storage



Elastic PaaS

Multicloud Deployment

Multi Data Center



© 2015 Nexedi SA – Company Confidential

 *nexedi*

# Wendelin Options 100% open source

100% Python

OpenCV-Python

NLTK

Blaze

Numba / Parakeet

Pandas

Scikit Learn

NEO

**Video Processing**  
Intel Russia / Willow / Itseez

**Natural Language Toolkit**  
U. Texas / Chalmers

**Full out-of-core arrays**  
Continuum / DARPA

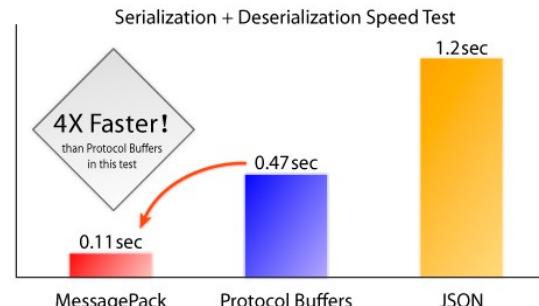
**JIT compiler / type inference**  
Continuum / DARPA

**Time sequence processing**  
DataPad / JP Morgan

# Data Ingestion: fluentd



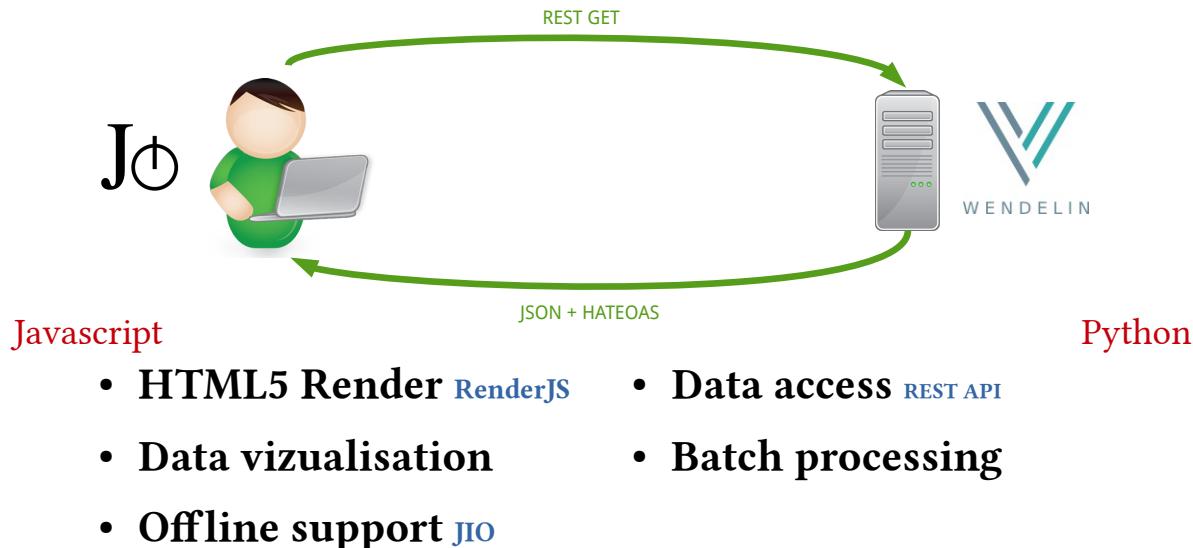
- Based on MsgPack middleware
- Created by TreasureData (BaaS pioneers)
- Used by Amazon
- Numerous plugins
- Scalable and resilient
- Bandwidth saver



© 2015 Nexedi SA – Company Confidential

 nexedi

# Wendelin UI



© 2015 Nexedi SA – Company Confidential



The experimental HTML5 UI of ERP5 uses a library called JIO to abstract the relation between the browser and the server.

The browser sends REST requests over HTTP.

The server returns JSON data over HTTP with a self-discoverable format, something called HATEOAS.

The user interface is implemented as a javascript application that runs on the browser side. HTML is generated on the browser side by Javascript code. Form data is prepared by Javascript code and sent as JSON to the server. Many features of the application are still available even offline.

The role of the server in this architecture is only to provide access to the data and to validate the data before updating records in the database, using global consistency rules.

# Wendelin Distinctive Advantages

- Native out-of-core NumPy (scikit-learn, pydata)
- Native parallel processing
- Bare metal performance (GPU, FORTRAN)
- Transactions (ingestion, processing)
- NewSQL queries
- Built-in PaaS
- Lower deployment cost (10x less than...)

## Detailed Example

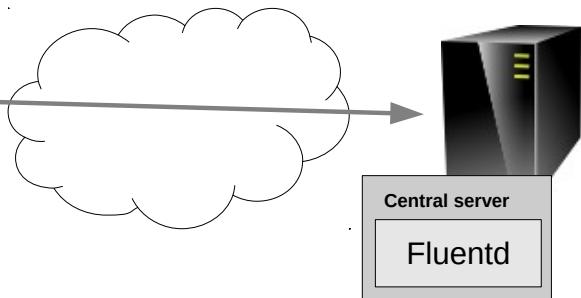
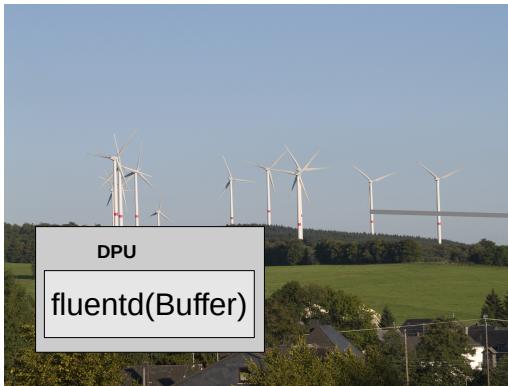


© 2015 Nexedi SA – Company Confidential



The solution that was deployed at the Lightning Protection Center complies is based on open source software – with full access to source code – and does not use software made by IBM, Oracle or EMC. It is thus a “No IOE” compliant solution, in line with directives published by Chinese governments for certain markets.

# Data Transportation fluentd



3 months benchmark

Frequent downtime (server, network)

< 0.001% loss

Very poor networking (ADSL, 3G)

© 2015 Nexedi SA – Company Confidential

 *nexedi*

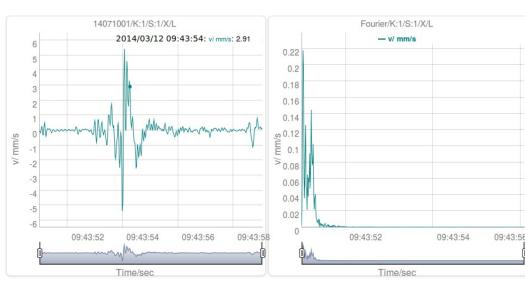
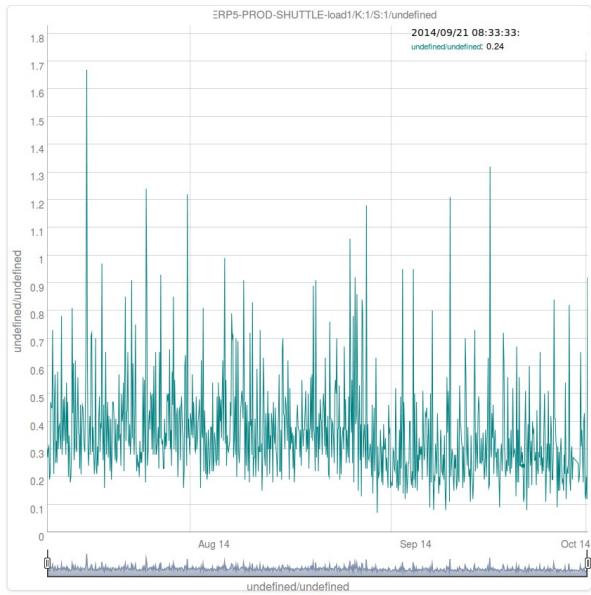
## UI: HTML5 Components [RenderJS](#)



© 2015 Nexedi SA – Company Confidential

 *nexedi*

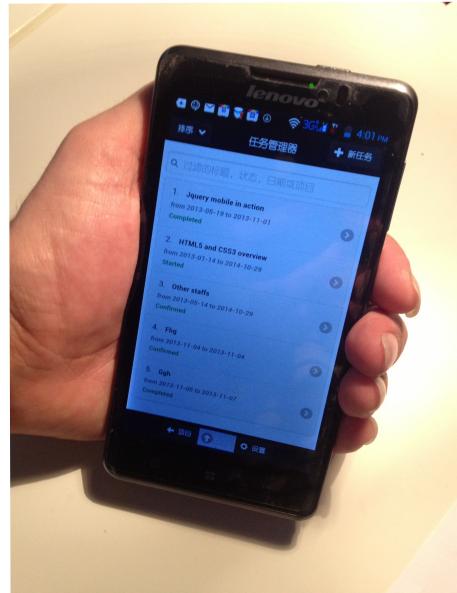
# Extend UI Components [RenderJS](#)



© 2015 Nexedi SA – Company Confidential

 *nexedi*

## UI : Responsive RenderJS



© 2015 Nexedi SA – Company Confidential

 *nexedi*

## UI: Offline / Other Backends JIO

Diagram gadget

Loading diagram ...

Wendelin Graph Visualisation

Channel X

Channel Y

Channel Z

Upload List Document

JIO

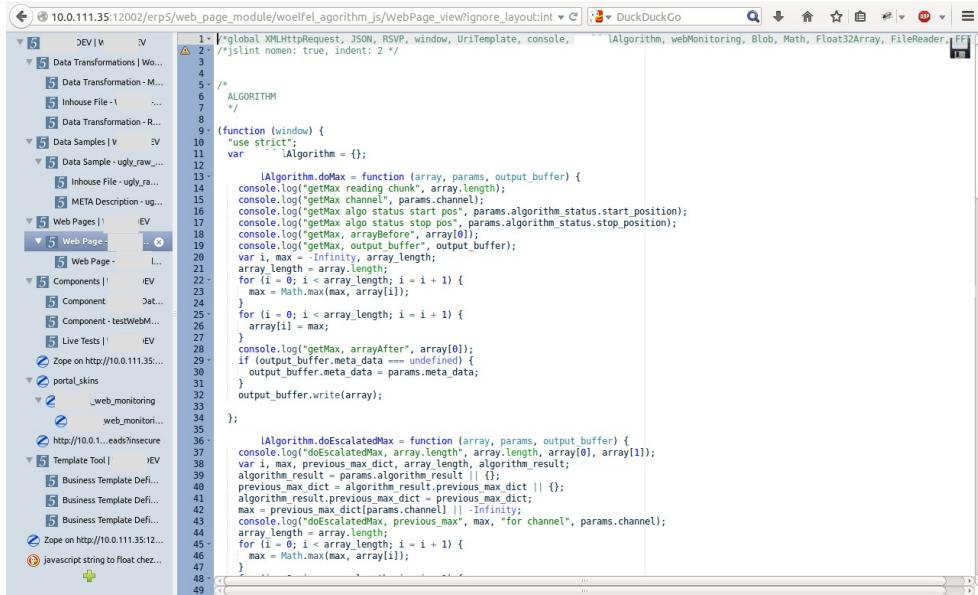
```
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
```

```
    JIO
  })
//configure JIO to use localstorage
.ready(function (g) {
  return g.getdeclaredGadget("JIO")
  .push(function (gadget) {
    return gadget.createJio({
      type: "query",
      sub_storage: {
        type: "indexeddb",
        document_id: "/",
        database: "test_ivan"
      }
    });
})
.ready(function (g) {
  return g.aq_pleasePublishMyState({page: 'listbox'})
  .push(function (url) {
```

© 2015 Nexedi SA – Company Confidential

 nexedi

## Data Science in Javascript vs. Python ?



```

1- /*global XMLHttpRequest, JSON, RSVP, window, UriTemplate, console, */
2- /*jslint nomen: true, indent: 2 */
3-
4-
5- /*
6-  * ALGORITHM
7-  */
8-
9- (function (window) {
10-   "use strict";
11-   var Algorithm = {};
12-
13-   [Algorithm].doMax = function (array, params, output_buffer) {
14-     console.log("getMax reading chunk", array.length);
15-     console.log("getMax channel", params.channel);
16-     console.log("getMax algo status start pos", params.algorithm_status.start_position);
17-     console.log("getMax algo status stop pos", params.algorithm_status.stop_position);
18-     console.log("getMax arraybefore", array[0]);
19-     console.log("getMax output buffer", output_buffer);
20-     var i, max = -Infinity, array_length;
21-     array_length = array.length;
22-     for (i = 0; i < array_length; i = i + 1) {
23-       max = Math.max(max, array[i]);
24-     }
25-     for (i = 0; i < array_length; i = i + 1) {
26-       array[i] = max;
27-     }
28-     console.log("getMax arrayAfter", array[0]);
29-     if (output_buffer.meta_data === undefined) {
30-       output_buffer.meta_data = params.meta_data;
31-     }
32-     output_buffer.write(array);
33-   };
34-
35-   [Algorithm].doEscalatedMax = function (array, params, output_buffer) {
36-     console.log("doEscalatedMax, array.length", array.length, array[0], array[1]);
37-     var i, max, previous_max_dict, array_length, algorithm_result;
38-     algorithm_result = params.algorithm_result || {};
39-     previous_max_dict = algorithm_result.previous_max_dict || {};
40-     previous_max_dict[params.channel] = previous_max_dict[params.channel] || {};
41-     algorithm_result.previous_max_dict = previous_max_dict;
42-     max = previous_max_dict[params.channel] || -Infinity;
43-     console.log("doEscalatedMax, previous_max", max, "for channel", params.channel);
44-     array_length = array.length;
45-     for (i = 0; i < array_length; i = i + 1) {
46-       max = Math.max(max, array[i]);
47-     }
48-   };
49- });

```

© 2015 NEXEDI SA – Company Confidential

 nexedi

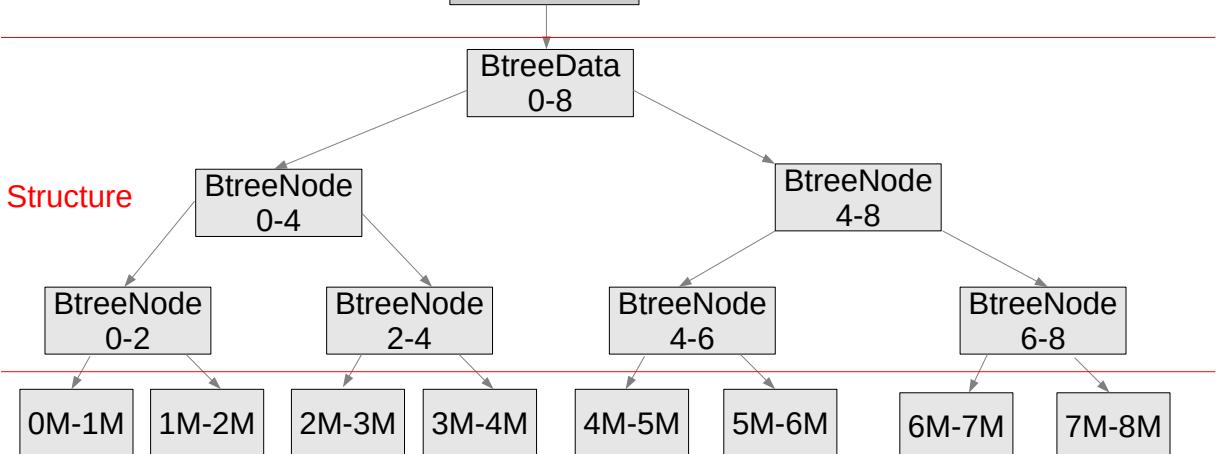
## Data Sciences in Javascript ? [phantomjs](#)

- Small data on client side
- Small data on server side
- Medium data (> 1 GB) in JS
- Out-of-core data in JS
- PyData compiled in JS
- PyData in NaCl / PNaCl

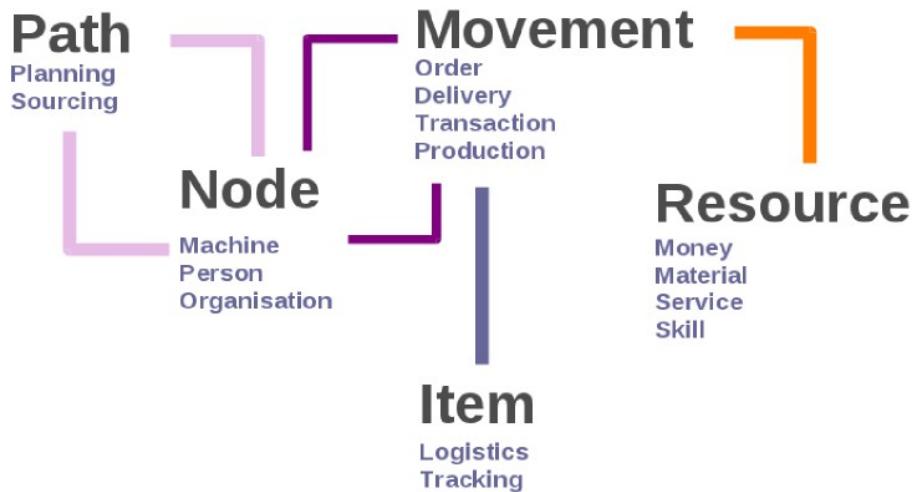
# Storing large streams in NEO



Access:  $O(\log(N))$   
Overhead : 0.2%



# UBM Monitoring Model?



© 2015 Nexedi SA – Company Confidential

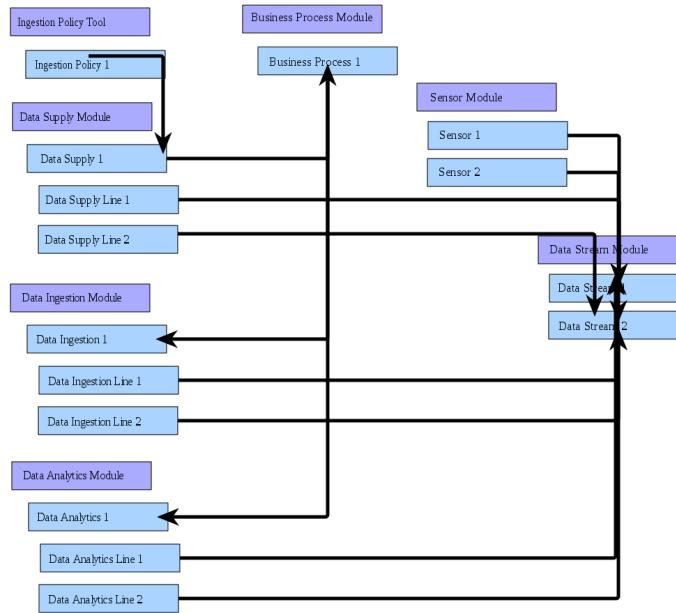
 *nexedi*

# UBM Business Model



- **Movement** – ingestion of data
- **Resource** – type of data (ex. memory log)
- **Node** – data source, data owner
- **Path** – data source registration
- **Item** – sensor, data itself, license, data set

# UBM Business Model



© 2015 Nexedi SA – Company Confidential

 *nexedi*

# What UBM gets us for free

- Accounting, billing and payment
- User registration and management
- Rule based security model
- Customer relationship management
- Web Content Management

→ save 12+ months and > 200 K€ on any Big Data project

# Future Roadmap



© 2015 Nexedi SA – Company Confidential



The solution that was deployed at the Lightning Protection Center complies is based on open source software – with full access to source code – and does not use software made by IBM, Oracle or EMC. It is thus a “No IOE” compliant solution, in line with directives published by Chinese governments for certain markets.

# Roadmap

[www.wendelin.io](http://www.wendelin.io)

- **Mainly accelerate learning curve**

- Universal packaging
- Ready to use examples
- Act as a backend to ipython notebook
- Port joblib to CMFActivity

- **Yet, you can start using part of Wendelin now!**

- wendelin.core out-of-core for NumPy
- JIO abstract data access library
- RenderJS components
- UI sample application
- Open Source

PyData Paris 2015 – 16h45 Kirill Smelkov

<http://learn.renderjs.org>

<https://lab.nexedi.cn/Tyagov/wendelin/>

© 2015 Nexedi SA – Company Confidential



# R&D Partners

[www.wendelin.io](http://www.wendelin.io)

- **Wendelin-IA (FSN)**

- Nxeddi
- Abilian
- 2<sup>nd</sup> Quadrant
- Paris 13
- IMT
- INRIA / ENS
- MMC Rus (Ru)
- X Corp

- **Windelin (Eurostars)**

- Nxeddi (FR)
- MariaDB (FI)
- Y Corp (DE)



© 2015 Nxeddi SA – Company Confidential

 *nxeddi*



**Wendelin Big Data**  
*Industrial Monitoring Platform*

2014-04-03 – Paris

© 2015 Nexedi SA – Company Confidential

