Model-driven Analytics with Models@run.time:The Case of Cyber-Physical Systems

SnT

uni.lu
UNIVERSITÉ DU LUXEMBOURG

Serval
securityandtrust.lu

> What this talk is and is not

- It is about
  - data collection, processing
  - Analytics
  - What-if simulation
  - For complex IoT/CPS
- It is not
  - About security
- BUT could be used for risk management, prediction, simulation

> Predicting and prescribing

Forbes / Tech

APR 21, 2015 @ 10:50 AM 41,294 VIEWS

How Big Data Is Changing Healthcar

ANNALS OF SCIENCE | NOVEMBER 11, 2013 ISSUE

CLIMATE BY NUMBERS

Can a tech firm help farmers survive global warming?

BY MICHAEL SPECTER

Germany to win FIFA World Cup 2014; predicts Google, Microsoft and Baidu !

149

"I would love to have Paul the octopus to help me, but he already died, poor thing. So I cannot predict anything for this Final."

This was the reaction of Shakira, the Colombian musical mega-star when asked about predicting the world cup final winners. Baidu trends shows that Germany has 58.6% chances of lifting the trophy as compared to 41.4% of that of Argentina.

Israeli 'web prophet' maps the past to predict the future

Dr. Kira Radinsky, 26, who started studying at the Technion at 15, wins recognition from MIT for pioneering software that finds historical patterns to point the way ahead

Big Data Will Effectively Fight Terrorism In The World

> Data, information, knowledge

- Data are raw, unpolished
- Formatted and aggregated to be manipulated: information
- Knowledge: what the human being can learn from information

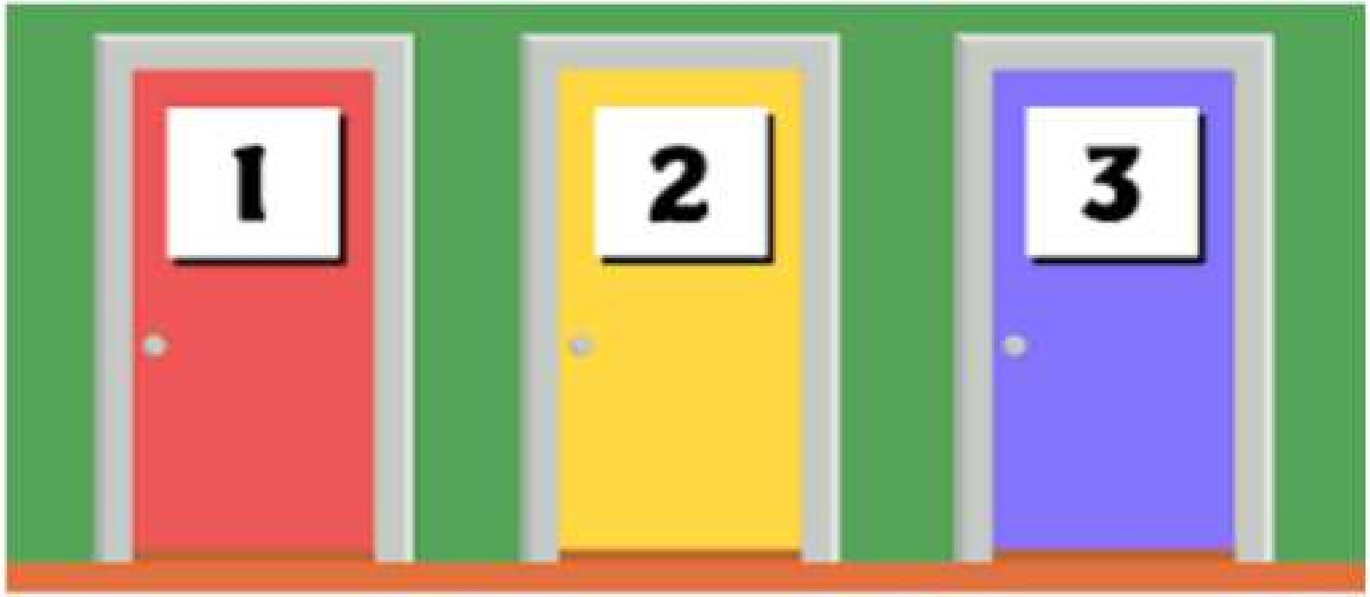
…hopefully for becoming wiser, reaching wisdom

A kind of magic – decision support services

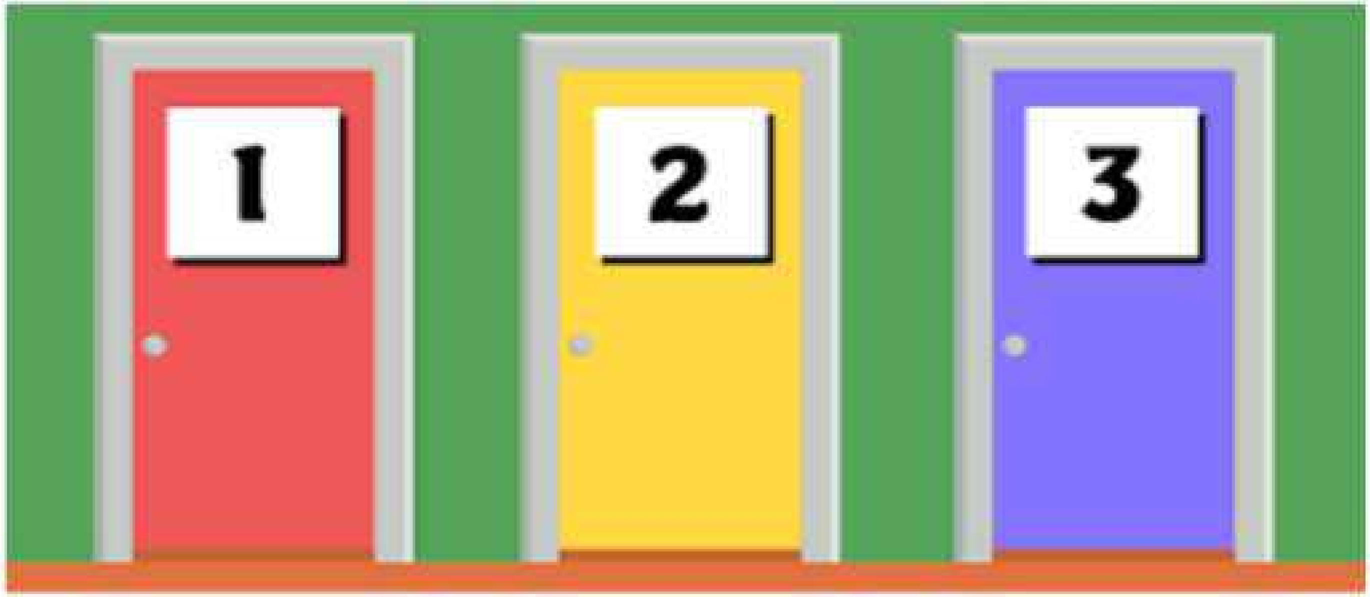
> Next slide is a test: make a choice, take decision

- Be silent
- If you know this example, please keep it for you

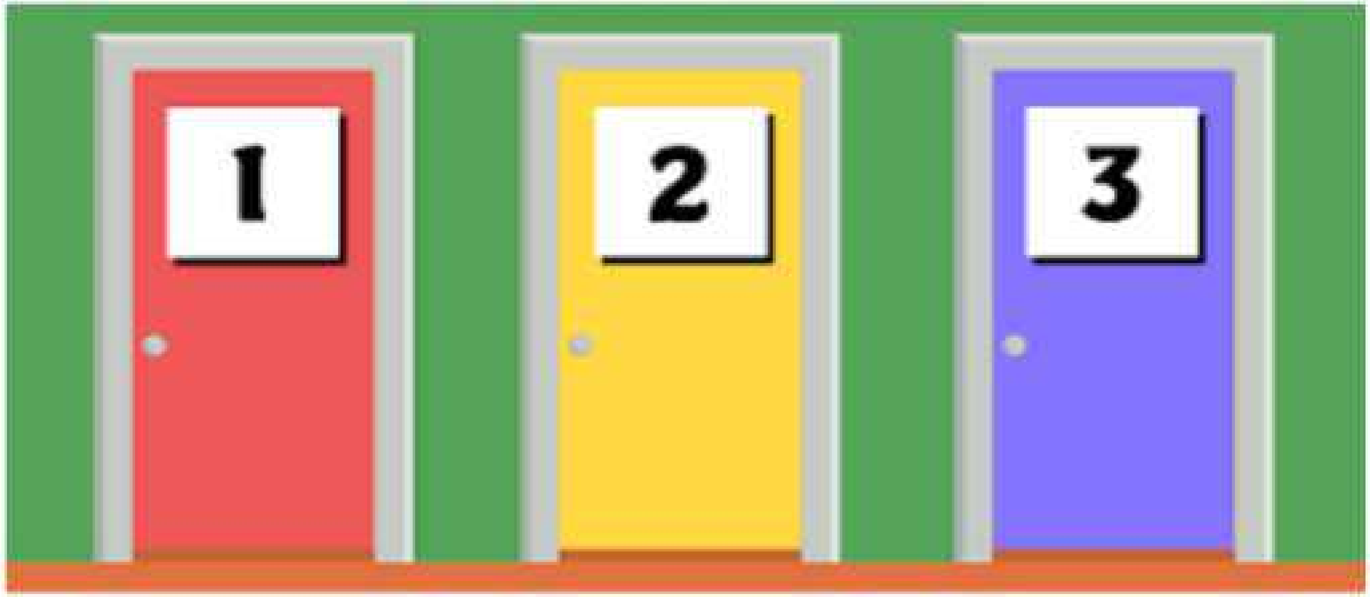
A kind of magic – decision support services



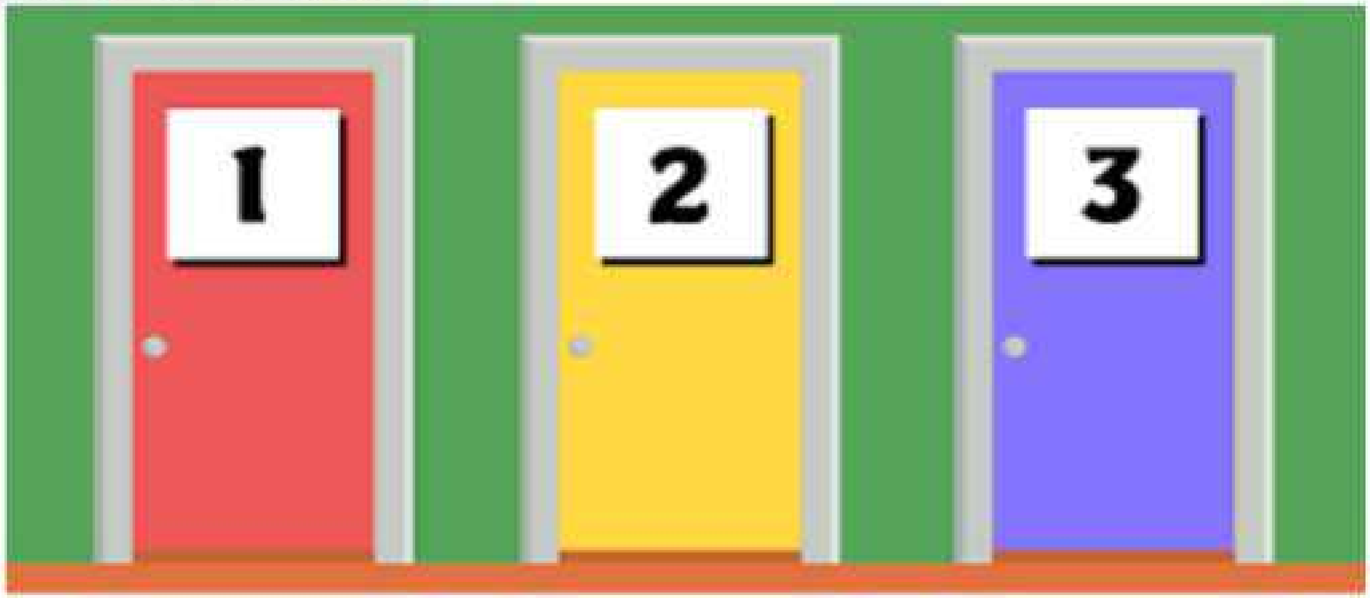
A kind of magic – decision support services



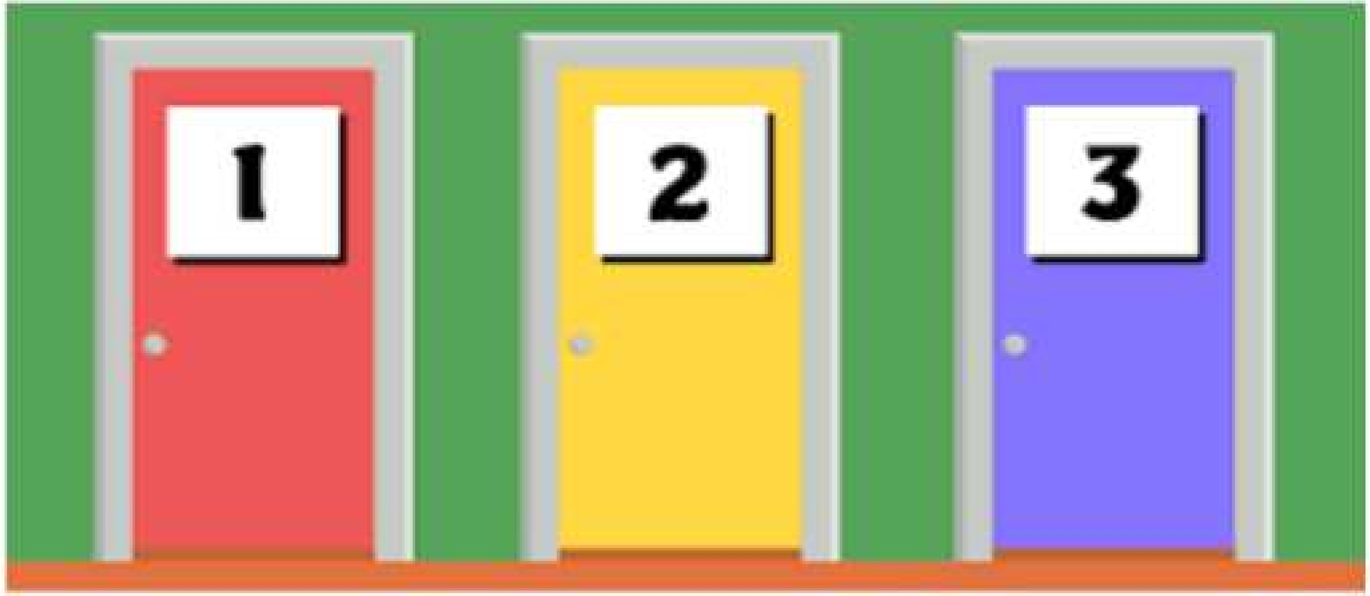
A kind of magic – decision support services



A kind of magic – decision support services



A kind of magic – decision support services



10 seconds to answer

- Would you swap to the other door?
- Would you stick to your choice?

- Change the door: twices the chances to win

Follow the good star and find the best itinerary

Non-intuitive decision

- Based on

  Something that is

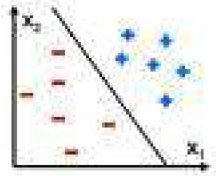
  Surprisingly

  A new information

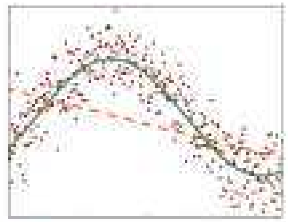
- No magic
  - Science, maths and …
  - Sofware to make it efficient

> Ingredients for analytics

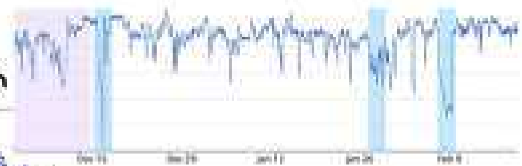
$$P(A|B) = \frac{P(A \text{ and } B)}{P(B)}$$



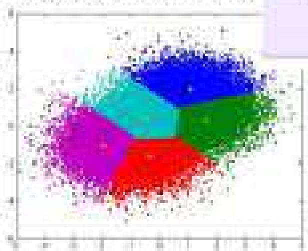
The core: data science



- Probabilities and statistics
- IA and Machine learning
  - Supervised,
    - Classification, regression, anomaly-detection



  - Non-supervised
    - Clustering: association rules



  - Feature select

$$\text{Rule: } X \Rightarrow Y \quad \begin{cases} Support = \dfrac{frq(X,Y)}{N} \\ Confidence = \dfrac{frq(X,Y)}{frq(X)} \\ Lift = \dfrac{Support}{Supp(X) \times Supp(Y)} \end{cases}$$

Software

> Ingredients for analytics

High Level

- Customizing
- Expert-friendly
- Vizualization
- Validation/veracity
- Security and privacy

Software everywhere

Low-level

- Sustainable, performant
- Storage,
- online processing
- Streaming
- Data retrieval

> Today's talk is all about software enablers for analytics

> About us

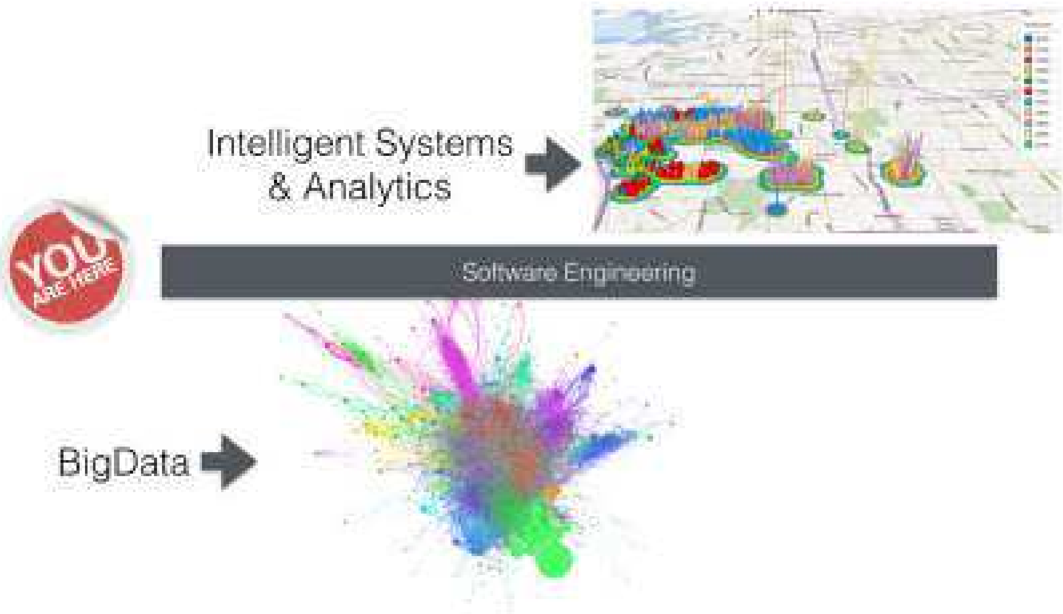
- Research from University of Luxembourg:
  - Interdisciplinary Centre for Security, Reliability and Trust *(SnT)*
  - **SerVal Team (SEcurity, Reasoning and VALidation)**
- Authors:
  - **Thomas Hartmann**: *PhD student*
  - **Francois Fouquet**: *Research Associate*
  - **Assaad Moawad**: *PhD student*
  - **Gregory Nain**: *Research Associate*
  - **Jacques Klein**: *Senior Research Scientist*
  - **Yves Le Traon**: *Professor, Head of the SerVal research group*

Serval

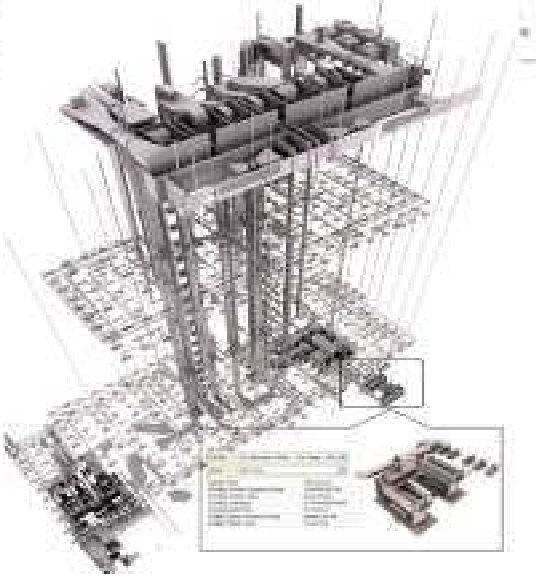
securityandtrust.lu

One of our research field

Software Engineering for smart things: **smart cities, grids,…**



Smarties



> Some research collaborations with industry

- CREOS – grid operator
  - Smartmeters/smart grid modelling and monitoring
  - Managing security incidents
- POST (Telecom)
  - IoT and SmartHome
  - Big Data for Smarthome
  - Model-driven and middleware

+ EU project bIoTope on SmartCities

Paul Wurth

Big Data for SmartBuilding

Recommendation systems

Ville de Luxembourg

Smart Building

Itrust

Security risk analysis– application to smart meters

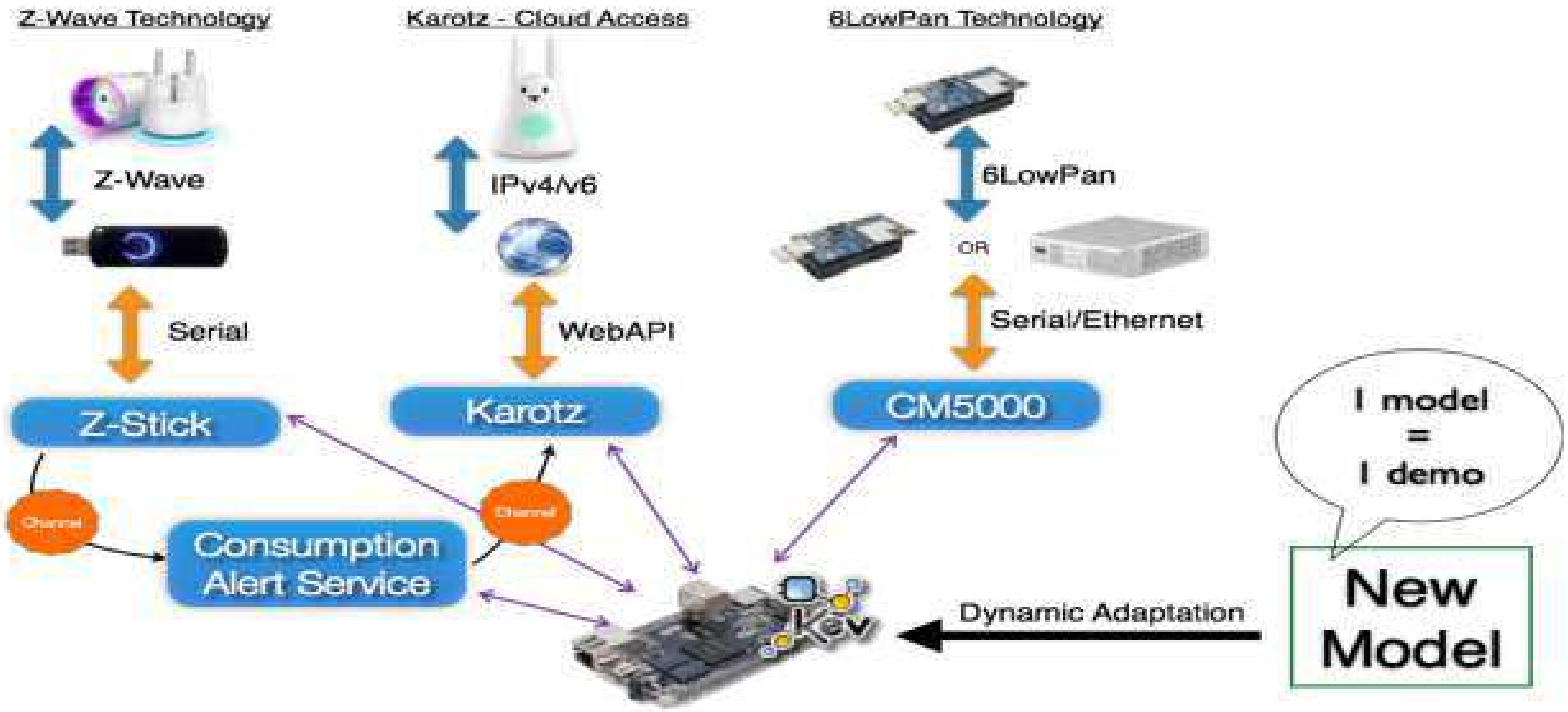
CETREL – credit card transaction authorizations

Analytics for testing

The Internet of Things Lab

- Internet of Things to support Smart Environments
  - Homes, Offices, Buildings, Cities
- Tests and Experimentations
  - Flexible
  - Adaptable
  - Scale 1:1
- Showroom
  - Demonstrations
  - Projects

> First work: Kevoree platform



> Cyber-physical systems

Examples

internet of things

industry 4.0

smart devices

> Cyber-physical systems

What are cyber-physical systems?

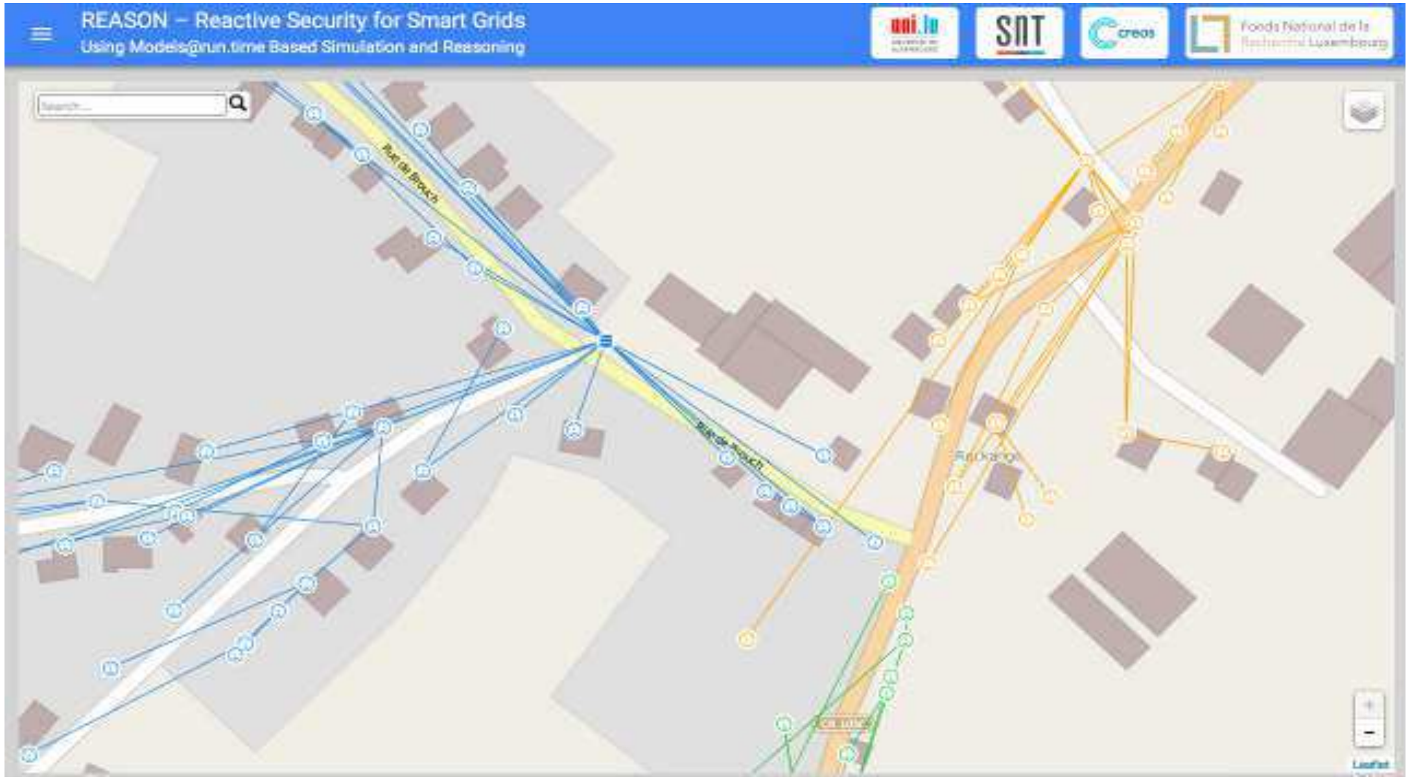
- Interacting networks of **physical and computational** components
- Provide the foundation of **critical infrastructures**
- Form the basis of emerging and future **smart services**
- Will bring advances in personalized health care, emergency response, traffic management, **electric power generation and delivery**, …

http://www.nist.gov/cps/

> Cyber-physical systems

Need to autonomously take sustainable decisions…

> Case study: smart grids



> Case study: smart grids

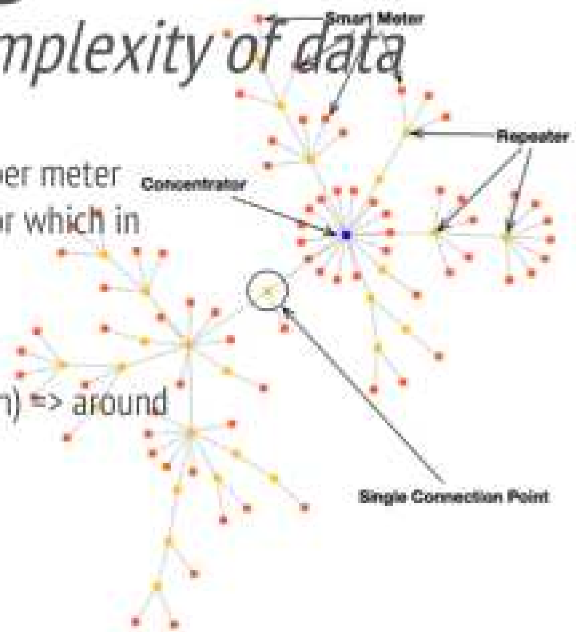
- To continuously analyze (in **near real-time**) the data collected nowadays in smart grids (e.g., metering data, topology data, …)

> **Make "smart" decisions to autonomously stabilize and improve the state of the grid**

> Case study: smart grids

The problem is not the volume but the complexity of data

- Every **15 minutes one consumption value per smart meter** => 96 values per day per meter
- The full grid is divided in *n* regions, every region is managed by a data concentrator which in turn manages 100 smart meters => **9600 consumption values per day**
- Around 10 cables in every region; cables are connected in cabinets
- Each smart meter is physically connected to one cable
- Logical/communication topology changes frequently (depending on signal strength) => around **30 changes per hour**
- **Reactions** need to be computed in **milliseconds to seconds**



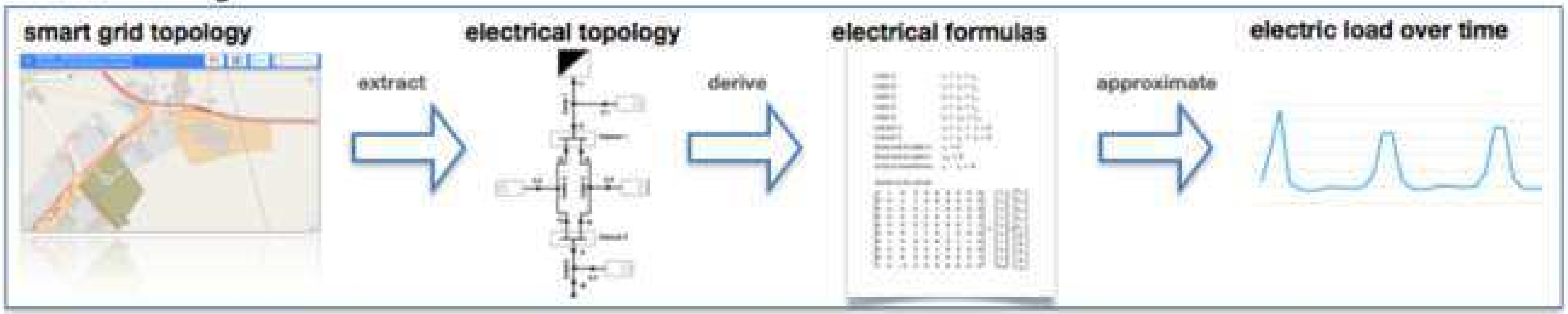
⇒ **a lot of small data sets which are semantically interconnected**
⇒ **Heterogeneous**

> Case study: smart grids

Example: electric load prediction

Question: can an electric car be charged without danger of overloading?

Decision making:

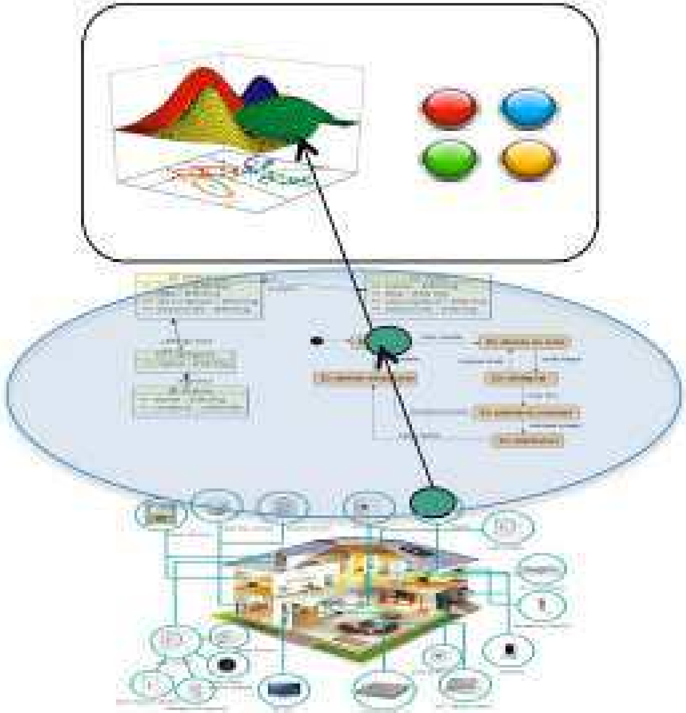


> For CPS and smart systems

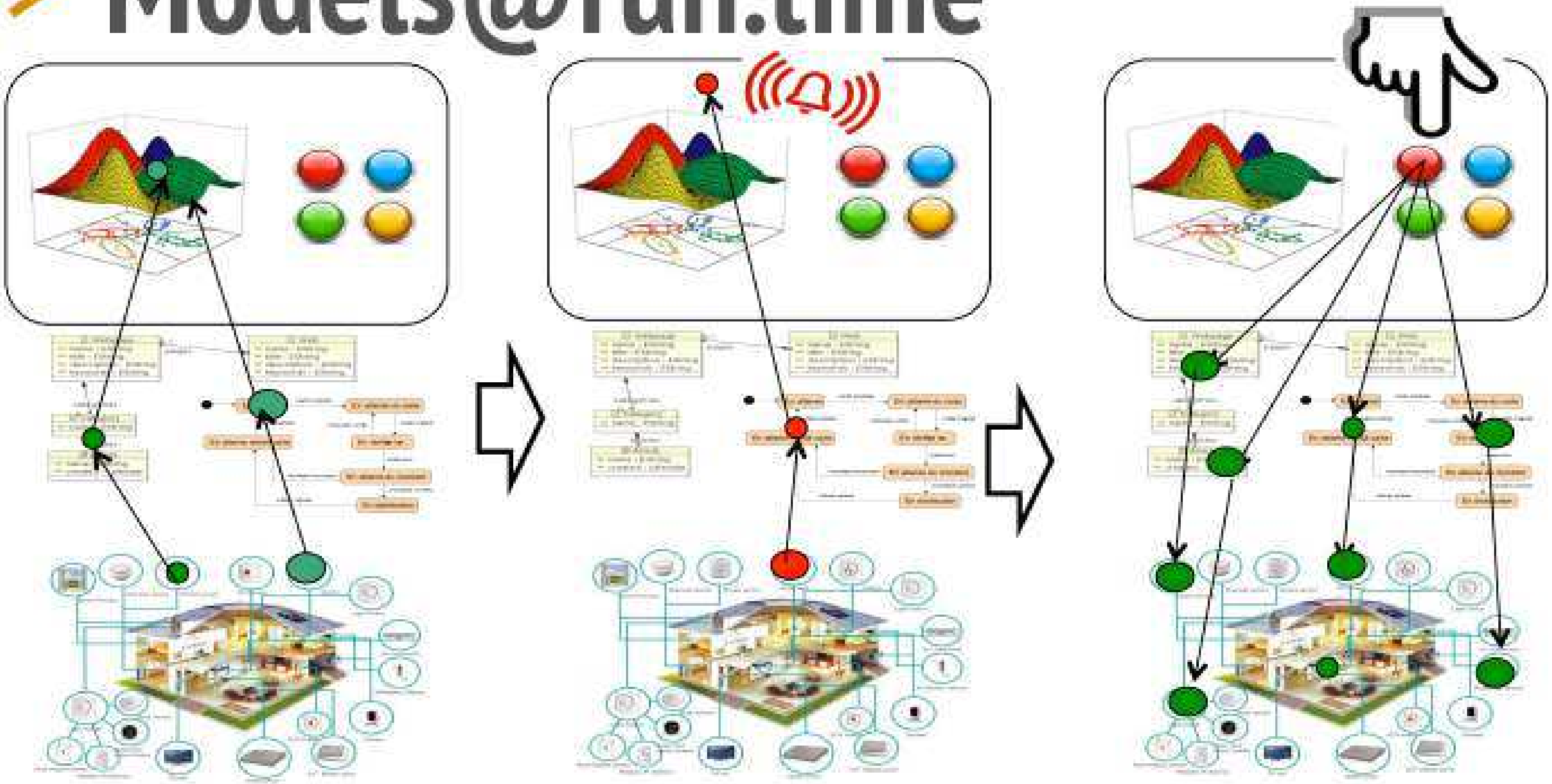
- We need to

Explore past, reason about present, predict futures, and prescribe what to do… now

- Micro analytics
- Stream processing
  - Near-real time
- Navigate into past
  - Fast navigation
- Aggregate heterogeneous data
  - Models + semantics
- Manage distribution



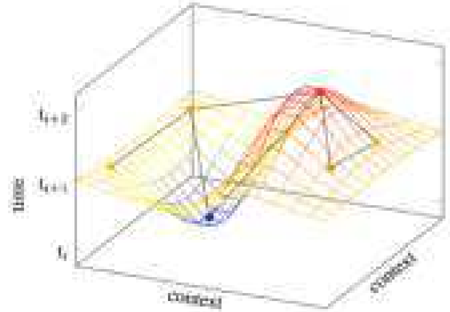
> Models@run.time

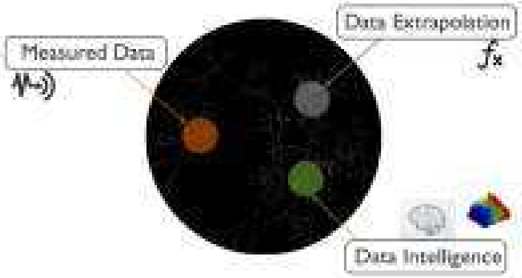


Model: Bridging the gap between data and abstraction

Storage

Live analytics





Model

> Open-issues and enablers

- Models are good for managing complex data: heterogeneous
- Models/DSL are more than a database schema
  - Embed semantics, reasoning, operations
- But not meant for
  - streaming
  - near-real time processing
  - efficient storage
  - distributed software

> From Big-data analytics...

Raw data
e.g., collected from sensors

→

Data storage,
e.g., data warehouses

→

Data extraction and transformation,
e.g., aggregate, cube, filter, roll up, ...

→

Analytics
mostly in batch

↑ Machine learning, mathematical models, business intelligence, ...

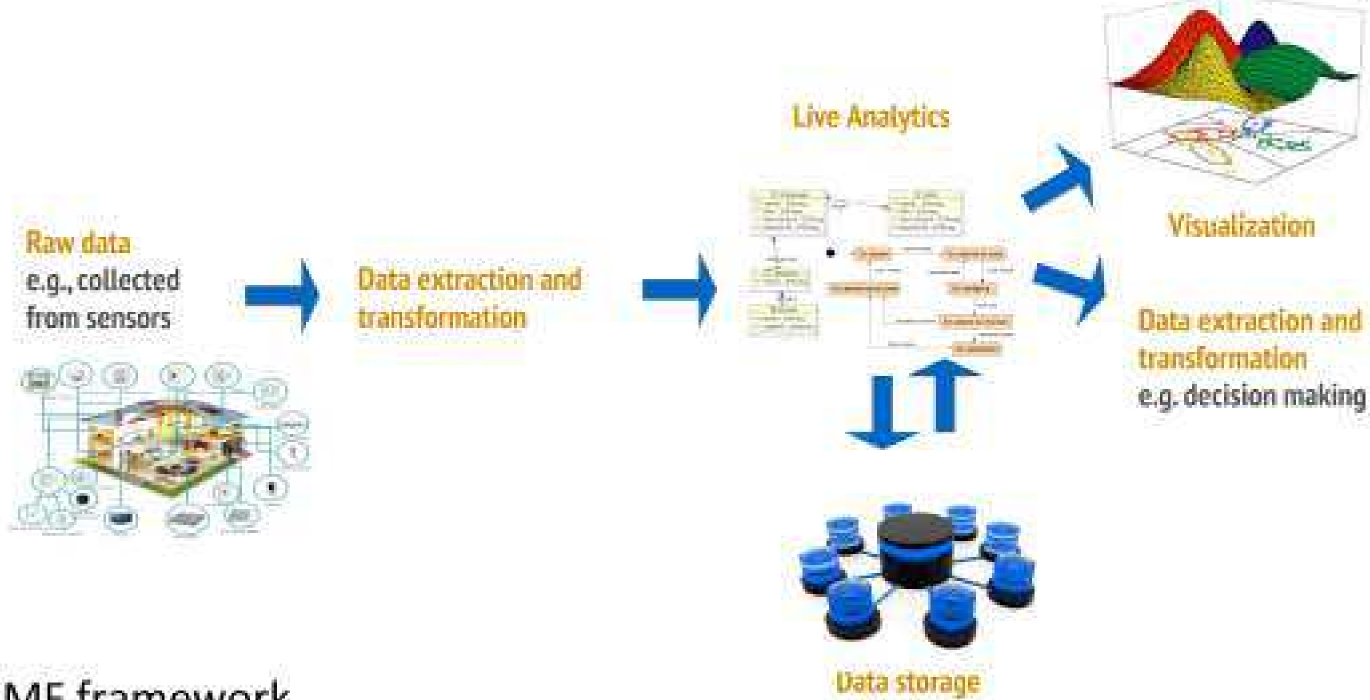
→

Data extraction and transformation,
e.g., for visualization

→

Visualization

> ...to model centric analytics



KMF framework

> All is about enablers

Learn from present not only from past → Timed data exploration

Real-world is usually continuous → Smart data structures (processing and storage)

Scaling with heterogeneous distributed data → Model instance distribution for scaling

From descriptive to prescriptive → Near real time machine learning

Proposed Solution: Models@run.time based Analytics…

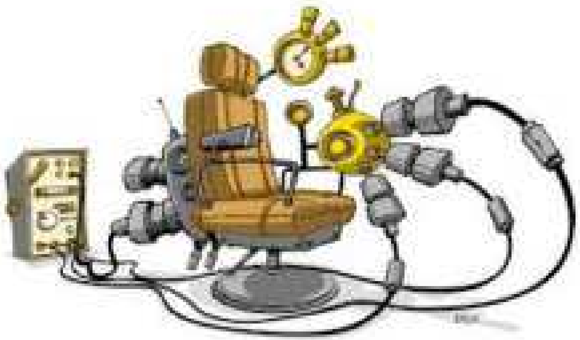
> Important enablers for model-driven data analytics

- Enabler 1. Modeling time-aware systems
- Enabler 2. Making models@run.time continuous
- Enabler 3. Distributed models@run.time

=> These enablers will be presented in more detail

> Modeling time-aware systems

First Enabler: Time machine



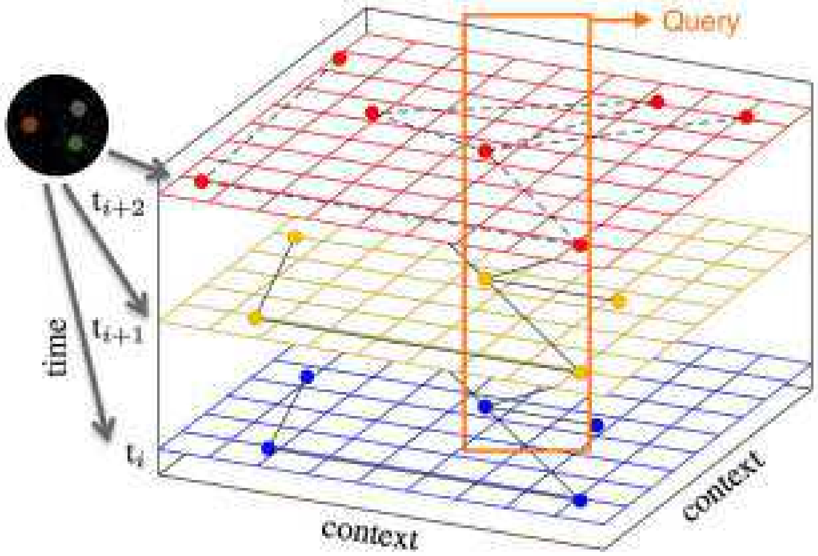
> Time machine for temporal models

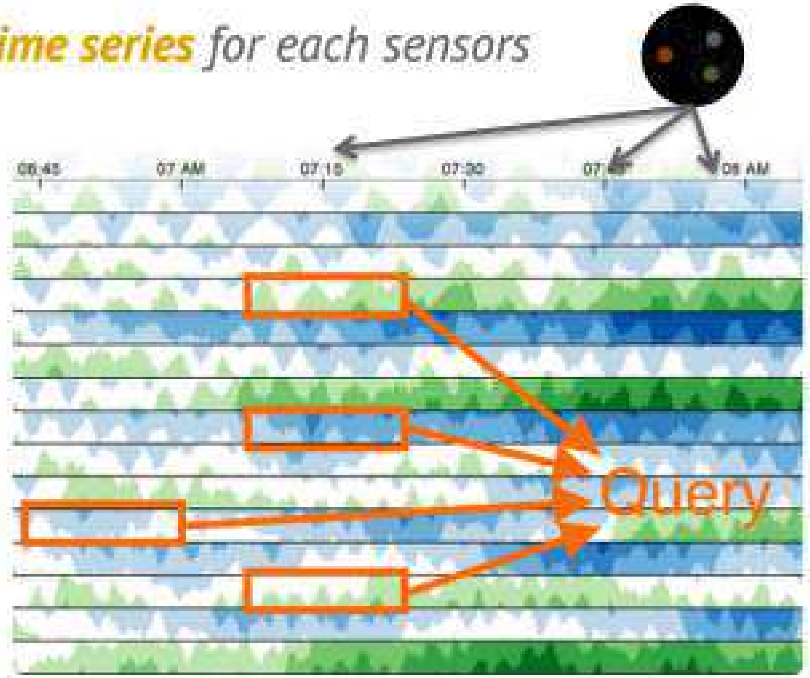
- Storing time stamped objects is costly

  build a "Time-machine" for free visit of past observations

> How to represent this context for different times?

*Regularly **sample** and store the context, or **time series** for each sensors*

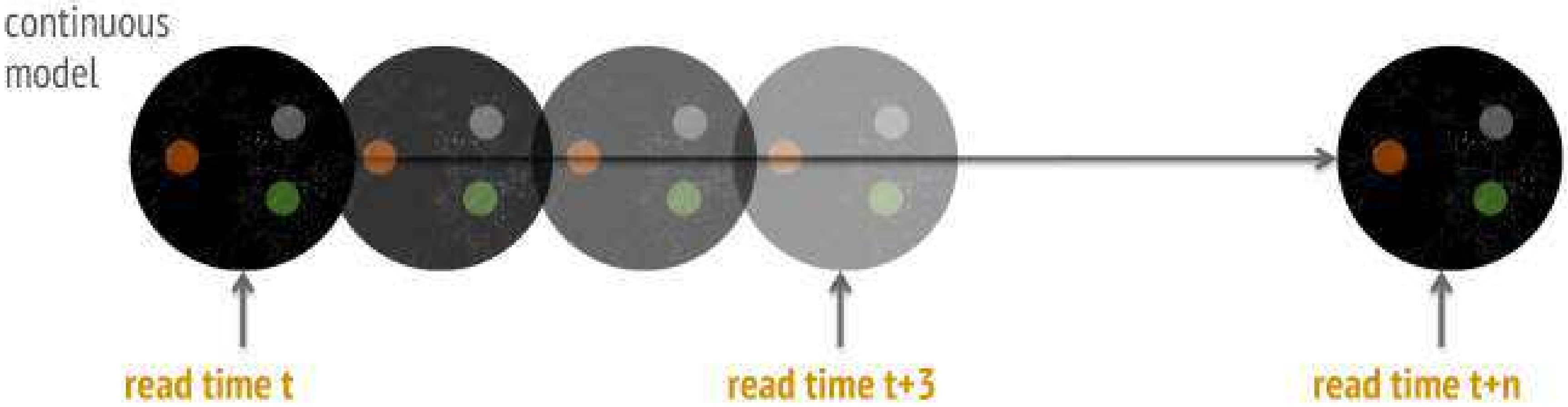




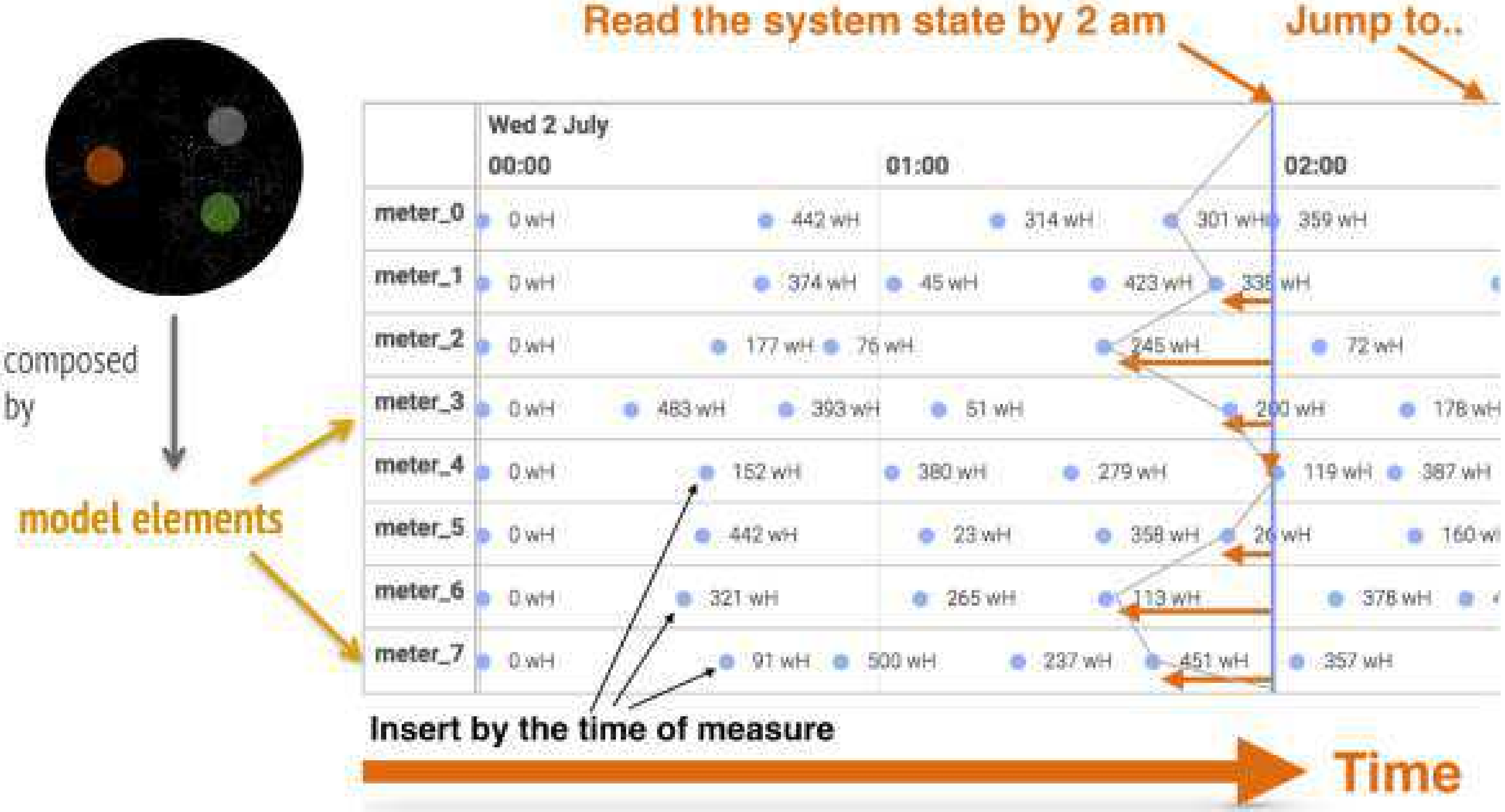
> Continuous models@run.time

(Hartmann et al., SEKE'14)

- Rather than querying a database, let's consider a model as a **virtually continuous** structure
  - i.e. should be readable for **any time**, by **extrapolating** all of its values **when READed**



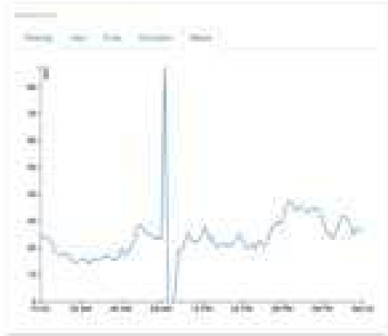
> Behind the scenes



> Performance impact?

> Experiment

- Case study is taken from a **real-world problem** from **Creos S.A.**
- Goal: **predict electric load** in a region based on current load and a range of historical data
  - data are retrieved at different times
  - high percentage of reading errors
  - predict if the load in a certain region will likely exceed or surpass a critical value.
- We compare snapshotting with time-distorted approach *(insert and read ability)*
- We vary the **size** of the **history** for the **extrapolation** (the bigger the more accurate)
  - small: 10 hours history (30 time units)
  - large: 2 month history (4800 time units)



> Measured impact

Evaluation on SmartGrid exploration, Classic NoSQL versus Model+NoSQL

Google LevelDB

- Snapshotting compared to time-distorted contexts

Scenario	*Snapshotting (Reasoning)*	*Time-distorted (Reasoning)*	*Snapshotting (Insertion)*	*Time-distorted (Insertion)*
SDP	1075.6 ms	1.8 ms	**267 ms**	17 ms
SWP	1088.4 ms	0.8 ms		
LDP	180109.0 ms	187.0 ms		
LWP	181596.1 ms	157.6 ms		

- ⊕ Reasoning improvement (factor): *598 (SD), 1361 (SW), 963 (LD), 1152 (LW)*
- ⊕ Insertion improvement (factor): *17*

> Enabler 2. Continuous models@run.time

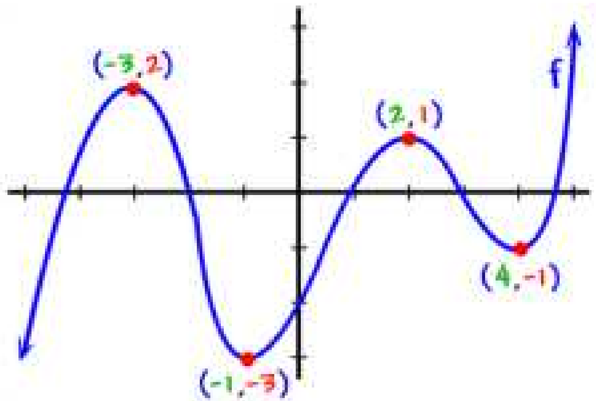
> Building continuous models

- **Idea:** Using mathematical **polynomials** for continuous model attributes
- Inspired by **signal processing** techniques
- Polynomials are able to describe and store a **continuous set of values**
- Extend modeling techniques with **continuous data types**

=> Robustness and storage and quick manipulation

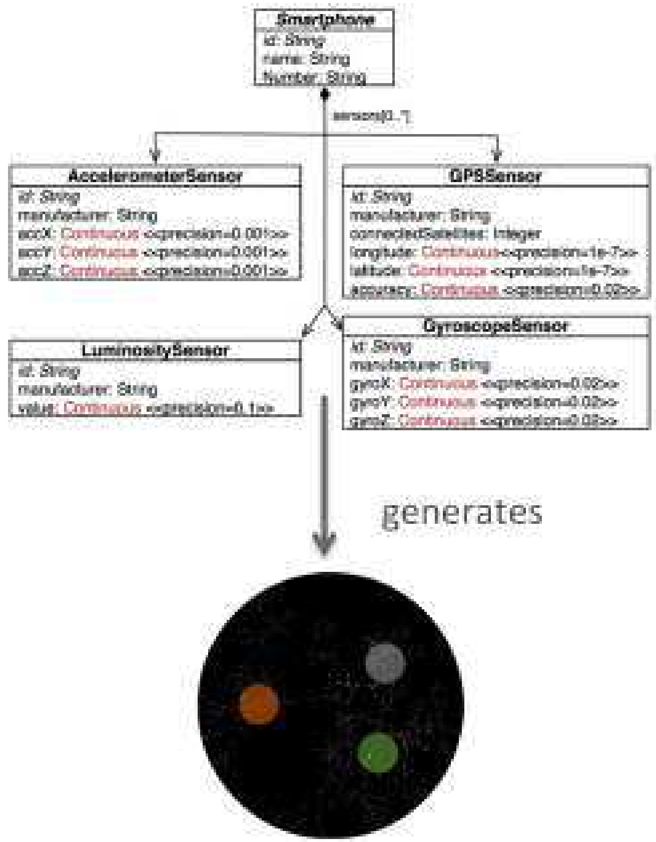
$$3x^2(x+5)$$

$$\begin{aligned} 3x^2(x+5) &= 3x^2(x) + 3x^2(5) \\ &= 3x^2x^1 + 3 \cdot 5x^2 \\ &= 3x^3 + 15x^2 \end{aligned}$$

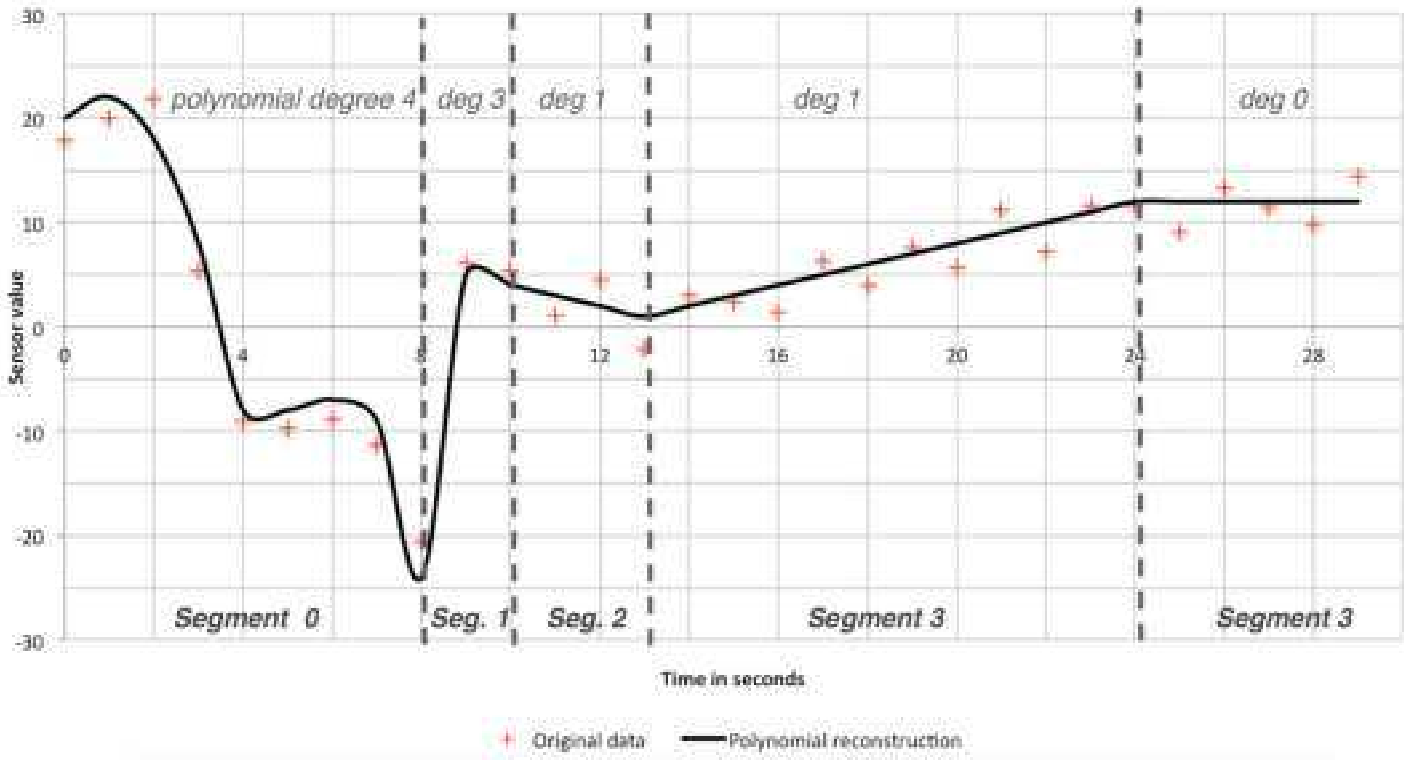


> How to do in modeling techniques?

- We add a new **meta-attribute** type for meta models with an precision definition
- The precision depicts the maximum **tolerated error** for the model representation **diverging** from the reality *(measures)*
- The **transparent polynomial management is generated in the runtime models**
- Continous and non-continous data can be **mixed** in the same meta-model and resulting models



We segment polynomials according to the tolerated error...



> Performance impact?

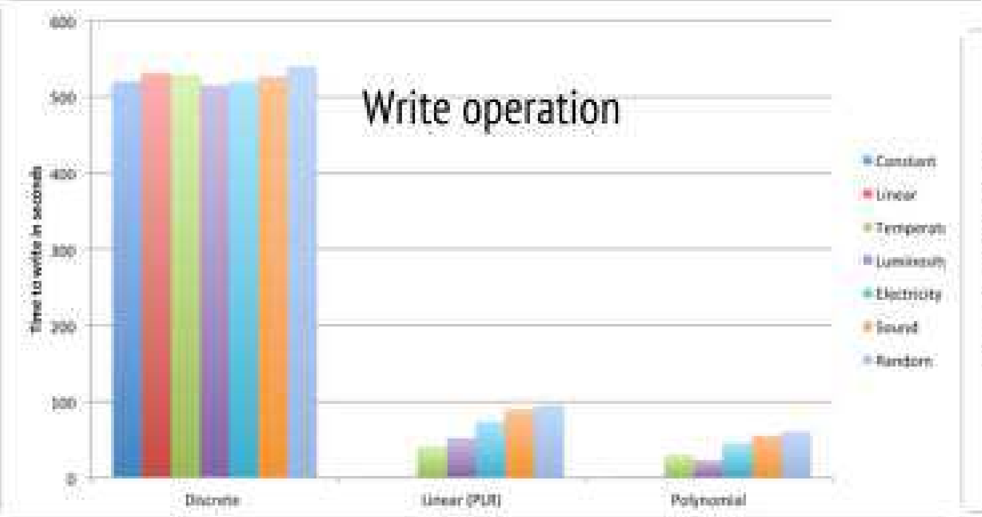
> Experiments

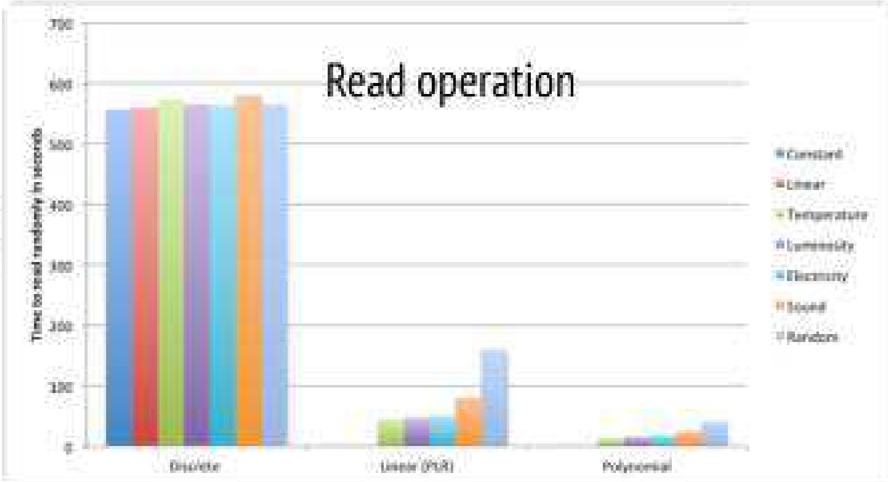
- We evaluate our continuous models on 7 different CPS datasets *(from best to worst in term of signal complexity)*
- We evaluate performance for read/write operations and for **continuity reconstruction ability** (extrapolation of missing measures)
- **5 Millions points** for each datasets

Database	Sensor
DS1: Constant	c=42
DS2: Linear function	y=5x
DS3: Temperature	DHT11 (0 50'C +/- 2'C)
DS4: Luminosity	SEN-09088 (10 lux precision)
DS5: Electricity load	from Creos SmartMeters data
DS6: Music file	2 minutes samples from wav file
DS7: Pure random	in [0;100] from random.org

> Storage: Read/Write operation results

- Divide by 100 the needed storage (compression)
- Continuous models are faster for all datasets, mainly because we drastically reduce the number of managed points in the time index
- We use Google's LevelDB NoSQL database for storage





> Robustness: Continuity reconstruction

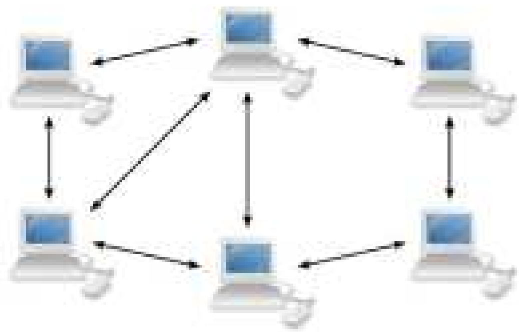
- To simulate **measurement losses** we randomly drop one value among ten, then we evaluate the ability of the continuous model to rebuild the signal after
- **Continuous models are significantly better in all cases**

Database	Discrete	Linear	Polynomial
DS1: Constant	0%	0%	0%
DS2: Linear function	5 %	0%	0%
DS3: Temperature	8.5%	3%	3%
DS4: Luminosity	9.9%	3.6%	3.5%
DS5: Electricity	17 %	7%	6%
DS6: Sound sensor	21%	15%	13%
DS7: Random	31.8%	31.1%	30.8%

> Enabler 3. Distribution
Distributed models@run.time

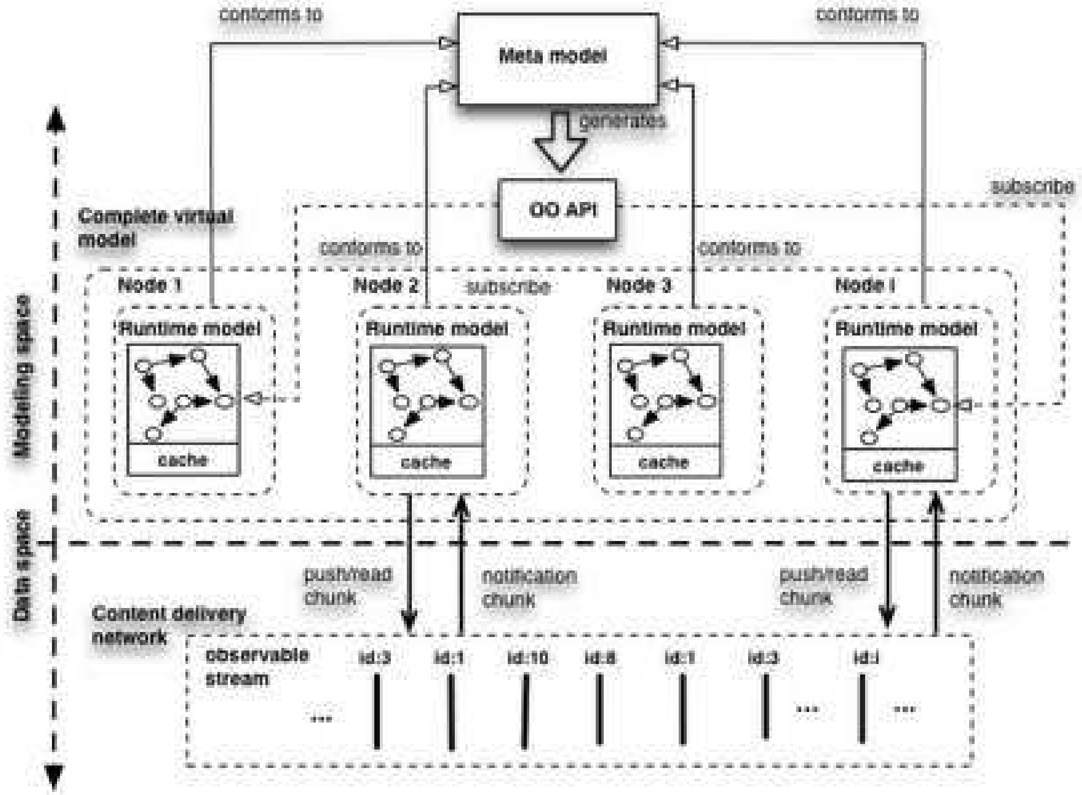
> Peer-to-peer distributed models

- CPSs often rely on the **collaboration of multiple devices** for smart decision making
- Models@run.time have to **scale to a "Big Data scale"** and must be accessible from everywhere
- We defined models as **observable streams** of chunks (a chunk contains one model element) exchanged in P2P manner
- We enable a transparent **lazy loading** *(only retrieve mandatory chunks)* mechanism
- **Virtually the model is now complete and accessible from every node. Data will be loaded asynchronously on when needed.**



> Distributed Models@run.time architecture schema

Peer-to-peer

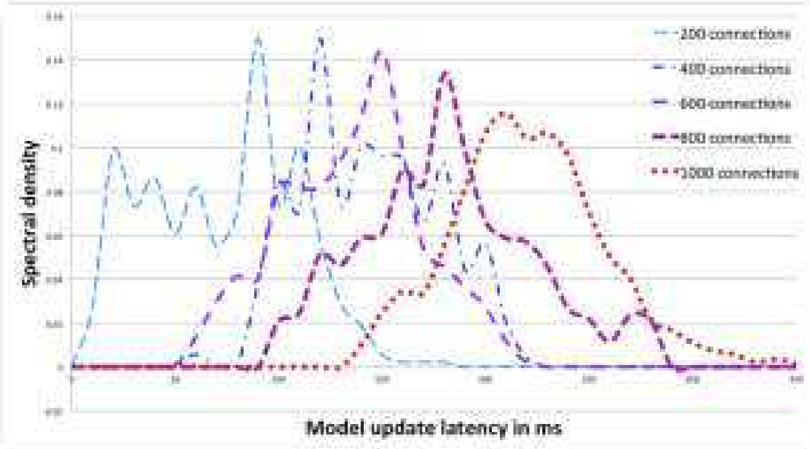


> Evaluation results

- We scale to models with **millions** of elements and **thousands** of connected, distributed nodes *(configuration of the smart grid Luxembourg for concentrator and number of smart meters)*
- Around **200 ms latency** in the worst case (in order to create an alert for a smart meter)

Nodes Nb.	Min(ms)	Max(ms)	Avg(ms)
200	11	188	88.01
400	63	220	128.75
600	87	253	169.52
800	102	289	185.62
1000	141	355	224.66

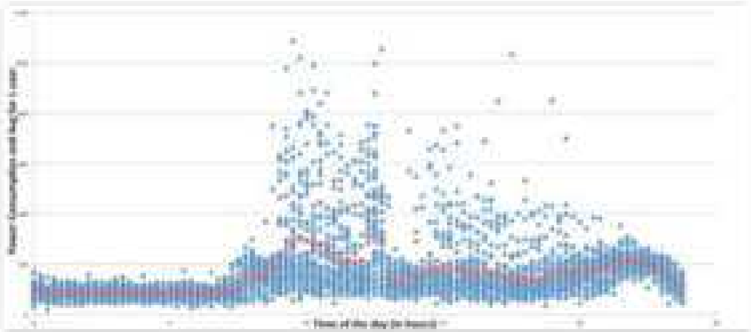
TABLE I
MEASURED LATENCY (IN MS) TO PROPAGATE CHANGES

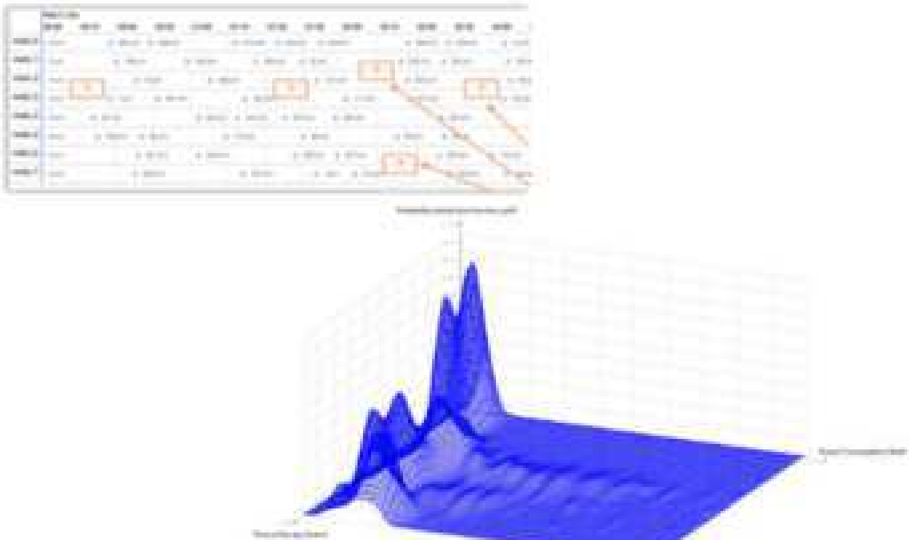


Concrete application: The Luxembourg smart grid

> Concrete application: smart grid

Probability of consumption data





Power consumption measures (in blue)

and average (in red)

Probability distribution function (pdf)

built by online live machine learning

➔ **Detection** and warning if consumption **values are suspicious** (based on Gaussian mixture algorithms)

> Concrete application: smart grid

Our multi-profile, directly integrated into the model out-performed standard error alarm system
context= weather, day, kind of customer

Attribute	Single Profiler	Multi-context profiler
Precision	0.602	0.808
Recall	0.99	0.99
Accuracy	0.779	0.918
F1 score	0.749	0.890

> Electric load prediction on grid cables

- **Goal**: approximating the electrical load in cables in near real-time
- **Results**: only 5% derivation compared to a full calculation with powerful power flow calculation tools
- **Novelty**: leveraging our model abstraction, data analytic capabilities and simplified electrical formulas
- Joint work with Yves Reckinger from Creos
- Is **integrated** in our **prototype** implementation

> Electric load prediction on grid cables

- We demonstrated the precision of these extrapolations within a derivation of 5%
- We also demonstrated the ability to fulfill near-real time requirements
- **This is now fast enough to be embedded in an on-field tablet for decision support systems**

Scenario	Overall	Creating	Solving
Transformer Substation 1 (103 meters, 12 cables)	191 ms	190 ms (99.95%)	$\leq$ 1 ms (0.05%)
Transformer Substation 2 (71 meters, 10 cables)	157 ms	156 ms (99.94%)	$\leq$ 1 ms (0.06%)
Transformer Substation 3 (56 meters, 8 cables)	143 ms	142 ms (99.93%)	$\leq$ 1 ms (0.07%)

TABLE I. PERFORMANCE EVALUATION

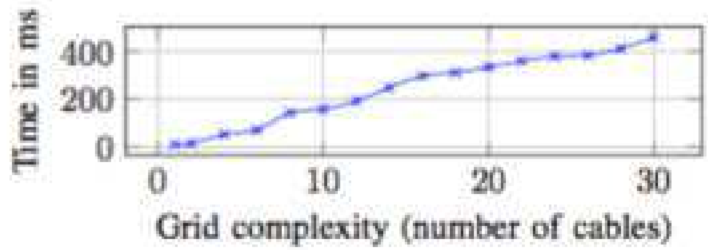


Fig. 5. Scalability of the electric load approximation

> Conclusion

> Where are we?

- Proposed an approach to enable what we call **model-driven analytics** (for CPSs) with models@run.time
- Developed a **data framework** called **KMF** based on this approach (https://github.com/kevoree-modeling/framework)
- Developed a data analytics tool for the smart grid of Creos in Luxembourg
- Ported the data analytics tool to fully run on an Android tablet

> What's next? Enable more

- Integrating machine learning approaches into this model-based approach
- Combining learned (virtual) and real data seamlessly in the same model
- Learning for detecting failure patterns and anomalies in data
- Application to security-related analytics

> Thank you…
> Questions?

« intelligently react to abnormal situations and ensure the quality of the information » (P1 conclusion)

> It's raining again!

Global / micro analytics

> It's raining again!

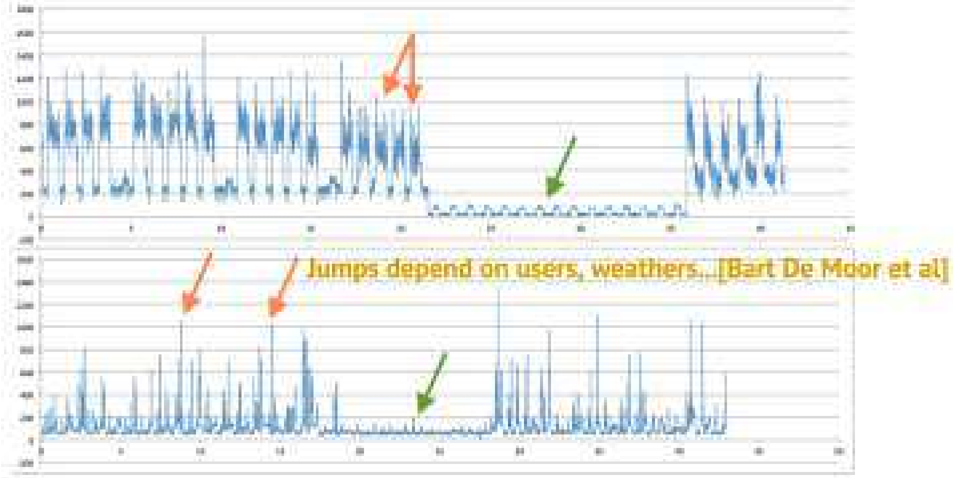
- Global analytics
  - Predict flood
- Micro analytics
  - Prediction: will this particular street be flooded
  - Prescription: Can you find an itinerary now for going there?

> Analytics for CPS: Smart Grid

Global analytics is looking for trends (*e.g.*, commonalities between all smart meters)
Micro smart analytics is contextual (*e.g.*, predict a particular sensor behavior..)

All customer consumption values are different..

Global analytics alone isn't what we want to do



> Data is dead… without what-if

- Data is temporal
- We can **look** at it, **add** it up, **roll** it up, **cube** it, **summarize** it, **compare** it, **filter** it, **join** it, ..
- We can even **find and learn** useful patterns and detect trends (machine learning)
- However, … data is a **record**, not a **conclusion** or an **insight** or a **solution**
- **What-if: the useful information**

> **To make sustainable decisions?**

> It's raining again!

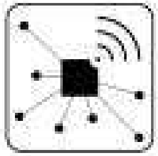
Rain is a real time stream

> Big Data or stream processing?

Big Data and **Stream processing**

Both offer nice features… but smart systems are in the middle…
Reasoning need **history**, and must **react in near real-time**

> Classical data analytics



Raw data
e.g., collected from sensors

Data storage,
e.g., data warehouses

Data extraction and transformation,
e.g., aggregate, cube, filter, roll up, ...

Analytics
mostly in batch

Machine learning, mathematical models, business intelligence, ...

Data extraction and transformation,
e.g., for visualization

Visualization

> And again!

Rain is not only about raindrops: heterogeneous and distributed data

> It's raining again

- Many raindrops!
- Falling all the time
- Distributed everywhere
- Depend on wind, temperature, topology … heterogeneous data

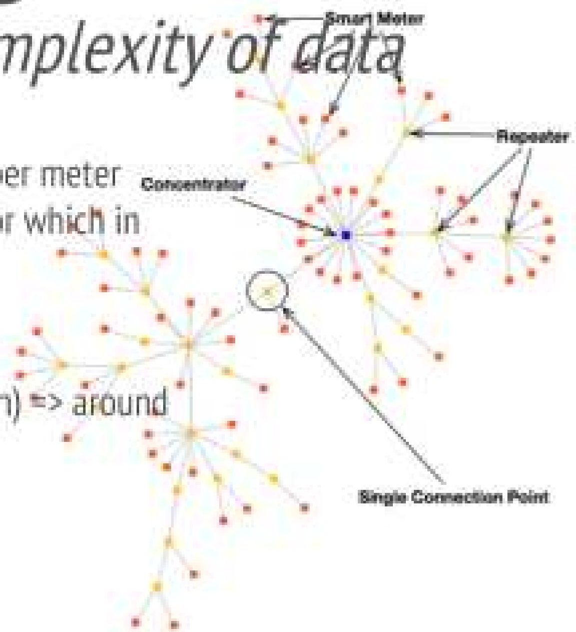
⇒ Shall we store every falling drop, when and where they fall?

⇒ Shall we instead model drops, wind and represent them in a simplified way (mathematical model) ?

Toward Model centric CPS

> Case study: smart grids

The problem is not the volume but the complexity of data



- Every **15 minutes one consumption value per smart meter** => 96 values per day per meter
- The full grid is divided in *n* regions, every region is managed by a data concentrator which in turn manages 100 smart meters => **9600 consumption values per day**
- Around 10 cables in every region; cables are connected in cabinets
- Each smart meter is physically connected to one cable
- Logical/communication topology changes frequently (depending on signal strength) => around **30 changes per hour**
- **Reactions** need to be computed in **milliseconds to seconds**

⇒ **a lot of small data sets which are semantically interconnected**

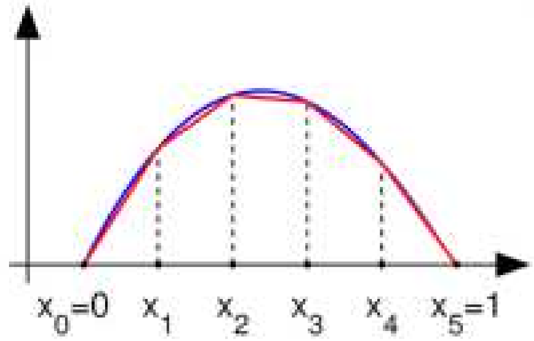
⇒ **Heterogeneous**

> Again and again!

Real-world is a mix of continuous and discrete phenomena: a drop has a continuous trajectory

> Models for CPS data...

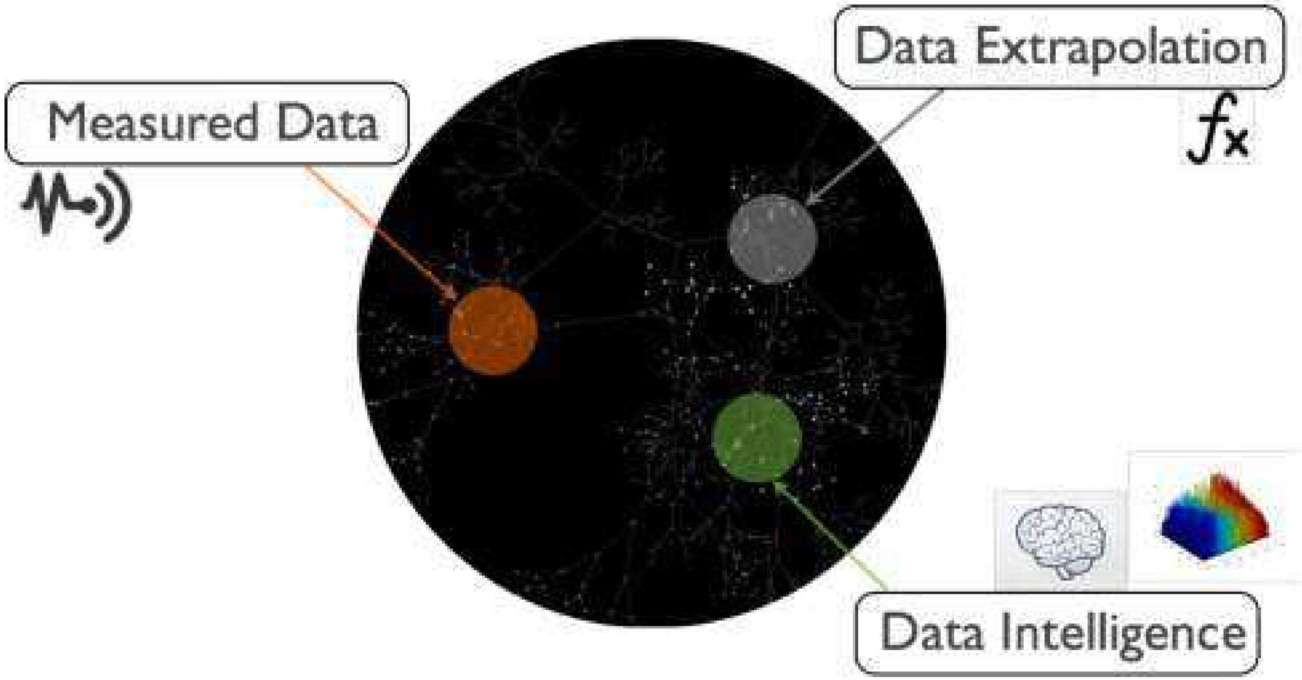
- Physical measurements are **continuous values**
  e.g., temperature, weather, time, consumption data, ...
- To **process** these measures in computer systems
  we **discretize** them
- Can easily lead to **millions of values**
- This is challenging for storage and computation power
- However, these values often don't change or only change **insignificantly**
- This wastes storage and computation power



=> The model is an abstraction

=> Knowing the domain definition, can we perfom better than just storing raw data in a database?

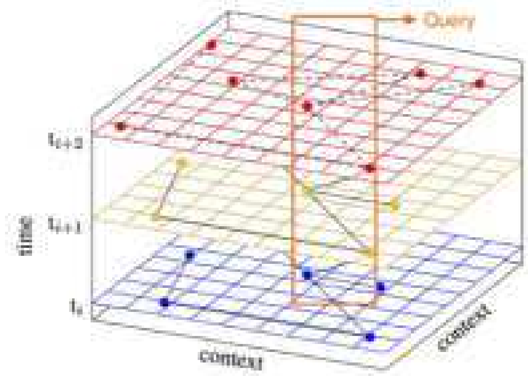
> Models as smart system brains

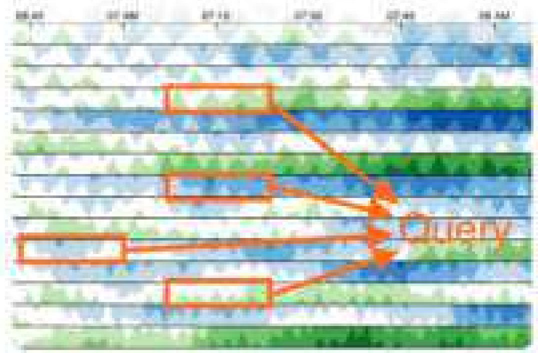


> However...

- Sampling at a very high rate leads to a massive stack of samples *(deep queries)*
- Time series per model element leads to very **wide queries** to extract a context

=> find, extract, and analyze a relevant context view is very hard to do within near real-time requirements

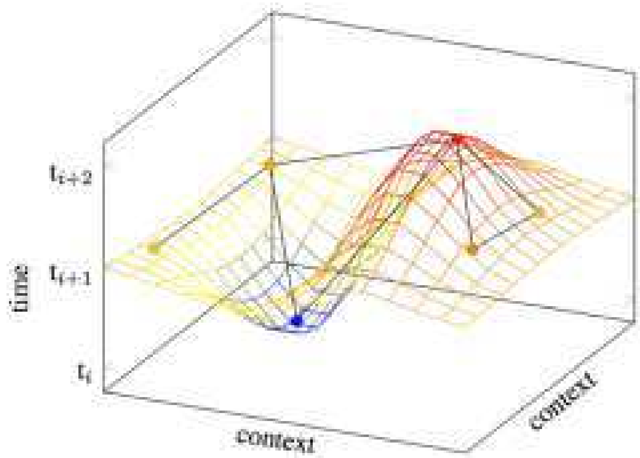




> Time-distorted contexts

How we see the time now?
An on-demand *(lazy loading)* view in a continuous model…

- Based on three pillars
  - Temporal validity for model elements
  - Navigating through time
  - Time-relative navigation



> Detection of important sections of signals…