Because it is not only about connecting objects!

François Pacull
Francois.Pacull@bag-era.fr

27 octobre 2017
The horrible truth

Check the boxes that corresponds to your environment

- The network is reliable
- Latency is zero
- Bandwidth is infinite
- The network is secure
- Topology does not change
- There is a single administrator
- Transport costs zero
- Network is homogeneous
- There is no legacy component
- Failures are not an issue
10 open the eyes and try again

< 5 you start to see where are the problems, we have to discuss

0 welcome to the real world, good news we can help you
Complex systems

Hardware components

Software components
(including user interface)

Not designed to work together

Geographically distributed

Heterogeneous
The network is reliable
Latency is zero
Bandwidth is infinite
The network is secure
Topology does not change
There is a single administrator
Transport costs zero
Network is homogeneous
There is no legacy component
Failures are not an issue
Why are we unique?

Resource based approach vs. Service based approach

Because service base approach is done by computer scientist for computer scientist.

Not for you !!!!!

Two of the most complex issues to master natively available

Complex event processing

When a set of conditions are true at the same time at different geographical locations => we trigger set of actions

Distributed transactions

We enclosed set of actions in a transaction => all or nothing
We model the world as resources contained in bags
This is the establishment of a new era: the bag era*

A bag encapsulates a component
(hardware or software)

We interact with the components through consultation (read),
consumption (get) and insertion (put) of resources

*Thank you Remy for the name
$2H_2 + O_2 = 2H_2O$
1) reaction is triggered *as soon as* the needed resources are present
2) it occurs *as many time as* the resources allow
3) intermediate state are *not visible*
4) resources are consumed, created, just need to be *present.*
Transposition

1) reaction is triggered as soon as the needed resources are present
   Complex event processing
   Waiting for the occurrence of conditions distributed over our complex system

2) it occurs as many time as the resources allow
   Pro-activity
   rule is always active for further processing

3) intermediate state are not visible
   Distributed transaction
   We do not want to see incomplete processing because it leads to inconsistency
   And we do not want the processing to be stopped in the middle of the stream

4) resources are consumed, created, just need to be present.
   Expressiveness
   It is easier to express processing in term of resources manipulation.
Some examples

1) Smart Parking
   - Car detected at parking lot
   - Badge read at associated charging station
   - Sufficient credit
     - Update database
     - Set light to red
     - Send and sms to the owner

2) Field gateway
   - 3G communication is down
   - Sigfox is available
   - Read locally available storage
   - Storage is > critical limit
     - Switch to degraded mode
     - Compute remaining storage time
     - Send warning to admin via Sigfox

3) Biological analyze fly-case
   - Temperature of peltier reach 50 degrees
   - Ultraviolet light is on
   - Camera is ready
   - Remaining storage is sufficient
     - Loop index < number of iterations
       - Take a picture
       - Increment loop information
       - Update workflow status

Same spirit independently of
- the application domain
- embedded system or large network
- easily understandable by end user
- can be generated from domain specific language
- full process is the aggregation of individual scenario
Mapping examples

Database
- Names
- Profiles
- Activities

Robot

Sensors
- Types
- Values
- Locations

Motors
- Status

Data

Requests

AI / Processing

UI
Application domains

- Robotic
- Silver Economy
- Industrial IoT
- Building management
- Energy management
- Digital Health
- Complex systems
- Innovation
- Use case validation
- Industry 4.0
Benefits

Smaller code
Faster to debug
Easier to maintain
Simple to upgrade, extend

Native:
- Heterogeneity
- Scalability
- Dependability
- Distributed control
- Incremental approach
- Security

Local coordination
Edge computing
Cloud computing

« Tool box »
High level mechanisms
Small footprint
State of the art
François Pacull
CTO
Francois.Pacull@bag-era.fr
http://www.bag-era.fr