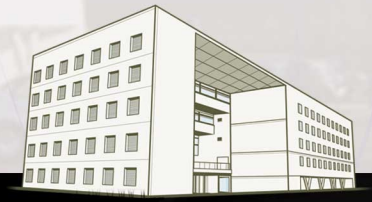
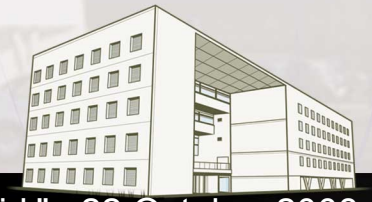
A Control of a mono and a multi scale measurement of a grid

Imene Elloumi elloumi@irit.fr
S. Ravelomanana raveloma@irit.fr
Manel Jelliti jelliti@irit.fr
Michelle Sibilla Sibilla@irit.fr
Thierry Desprats **desprats@irit.fr**



Outline

- **Motivation**
- **Monitoring Pattern**
  - -Control of a mono scale measurement
  - -Control of a multi-scale measurement
- **Illustration**
- **Results and Future Work**



Context : Grid

- A grid is a virtual infrastructure which consists of distributed and heterogeneous resources.
- Complex, composite and interdependent system.

Knowledge that never changes	***Knowledge that changes slowly***	***Knowledge that rapidly changes***
The Initial Speed of processors, the memory amount available,…	Operating or not of the machine: on / off	***Speed and Laod CPU, performance of network : bandwidth, latency, jitter,…*** **=> QoS !**

Problems

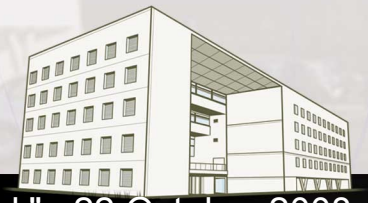
1. Need to mask heterogeneity of the format of management information and important number of metrics.

2. Among same limits of monitoring tools:
   - -Lack of aggregation of information for more relevance at all levels.
   - -A posteriori analysis of monitoring information.

3. Need for an inductive expertise.

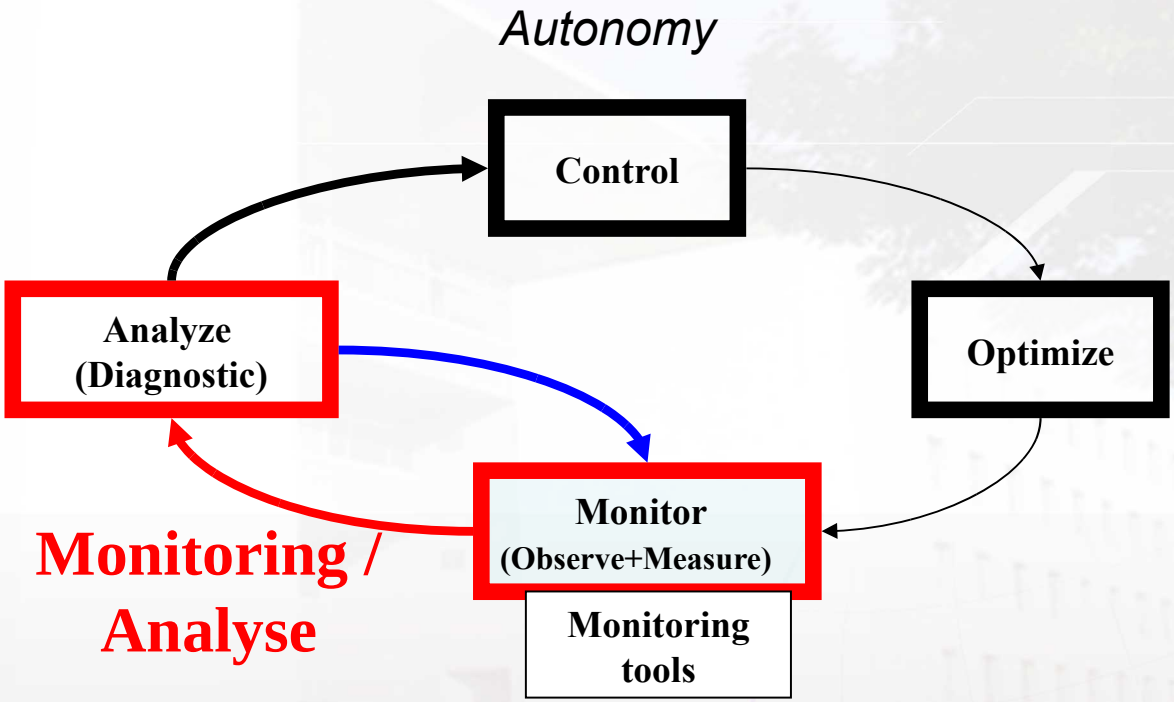
4. No accurate diagnostic in real time.

5. Absence of preventive monitoring.



Objective

Towards to guarantee of the availability of grid



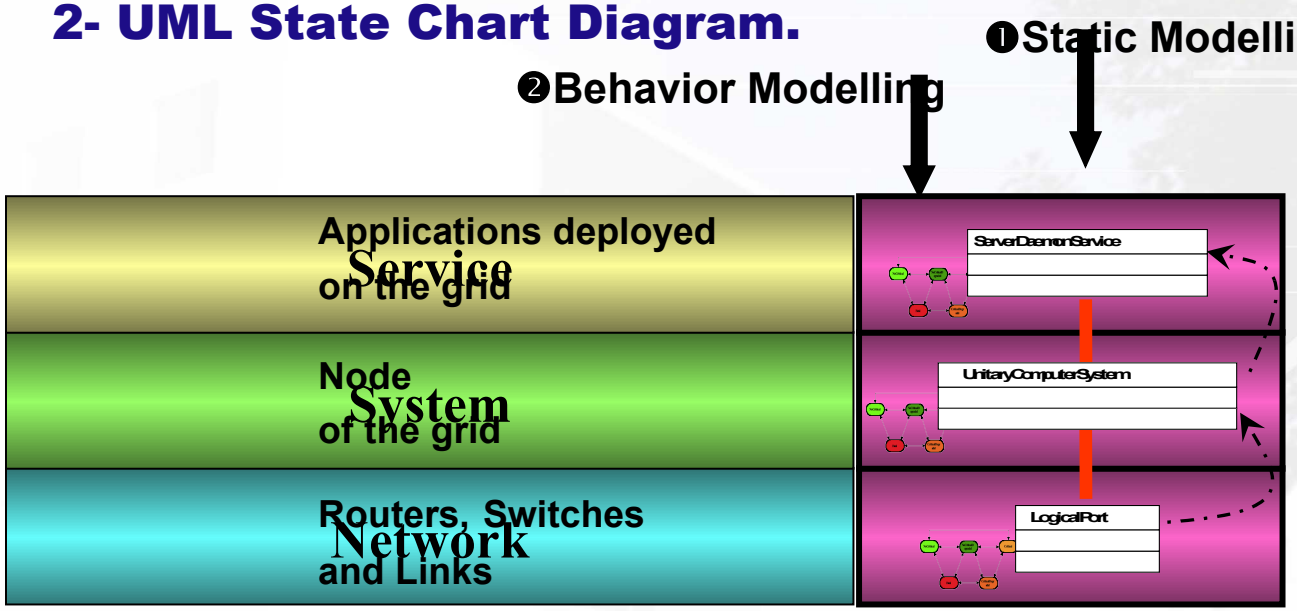
Inductive Monitoring of Network in Real Time

Proposed Solution

An unified and standard approach based on:

1- Common Information Model CIM/DMTF + OGF partner

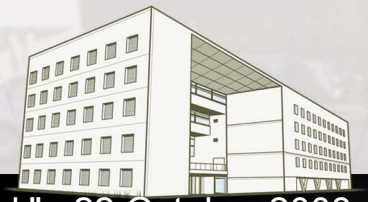
2- UML State Chart Diagram.



3- Interesting observation pattern/DMTF that can cover one or an aggregation of observable property :

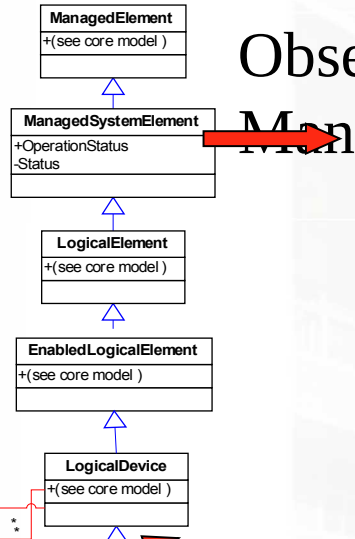
=Specifying the change of one critical state to other.

=Control and automate this observation pattern.



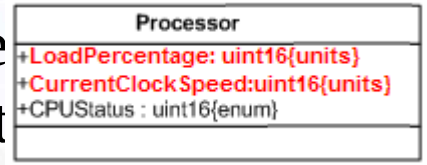
Monitoring pattern

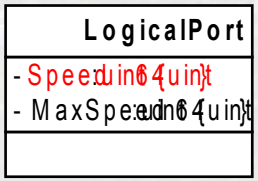
Modeling the thresholds for a property observed:



Observed Ele

ManagedSyst

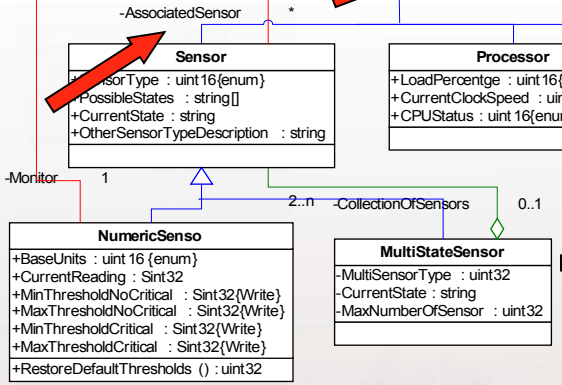


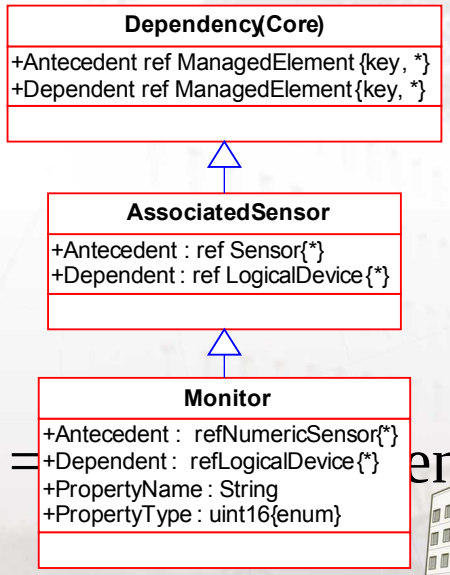


Observation => "Monitor" dependency

Sensor => NumericSensor

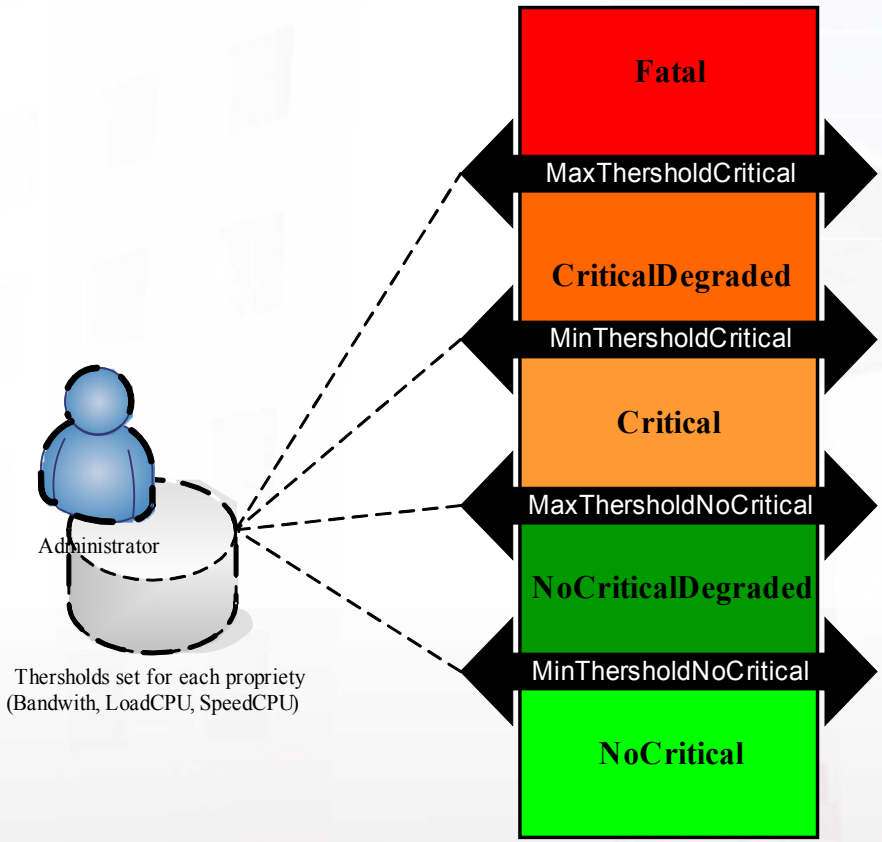
Controled Property to set in the "Monitor" dependency



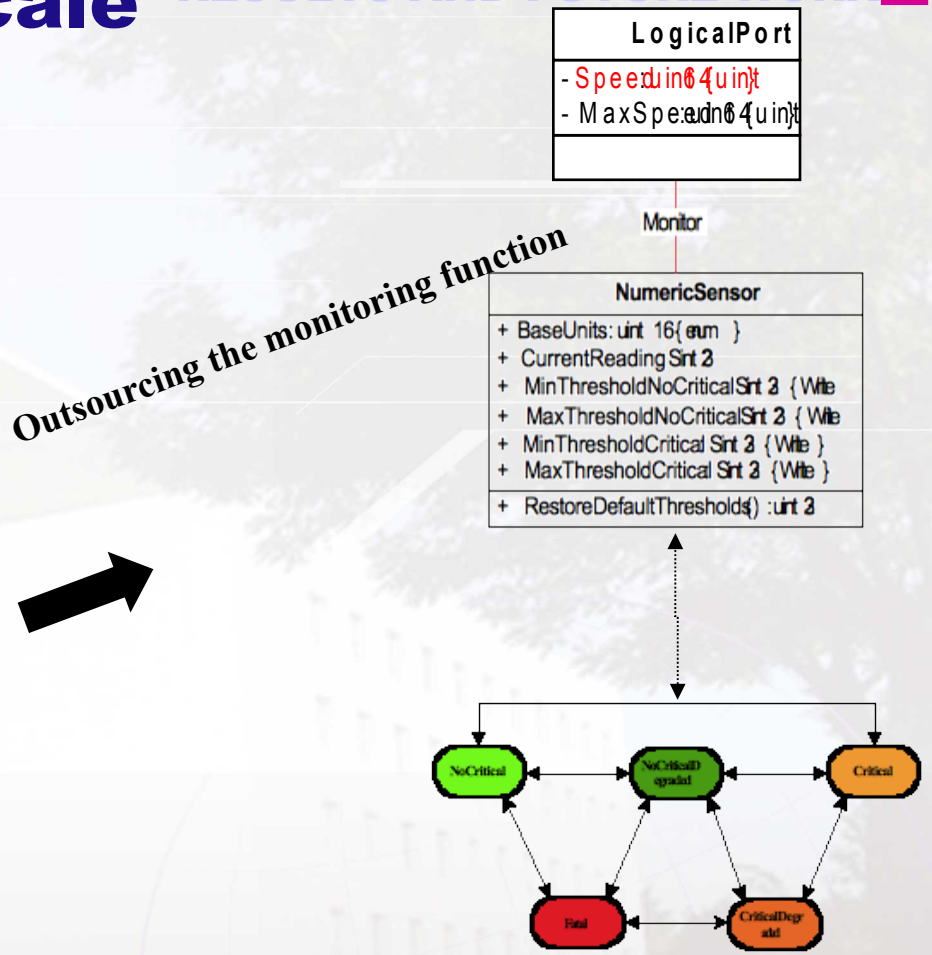


MultiSensor = ensor

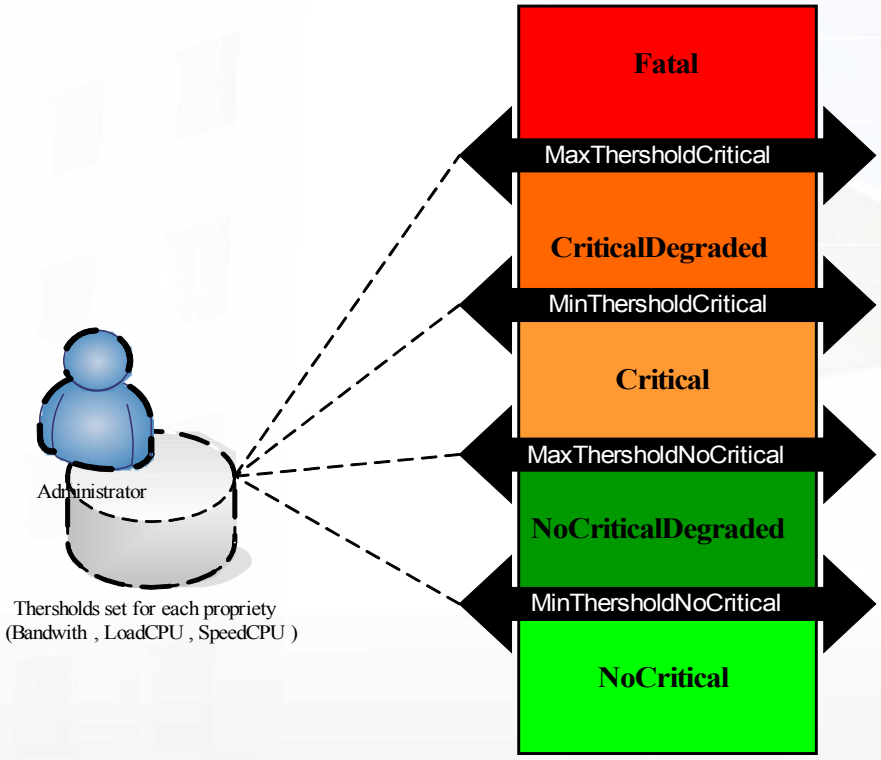
Control of a mono scale measurement

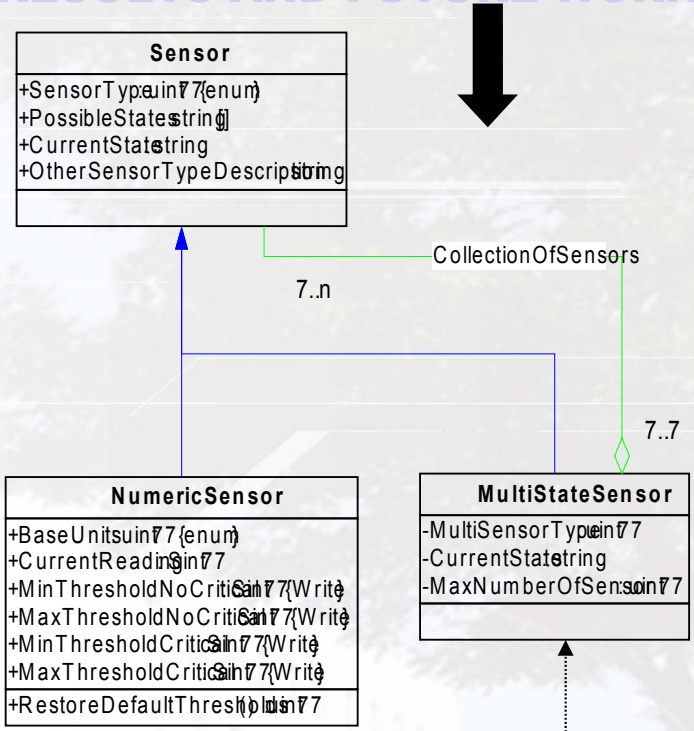


Outsourcing the monitoring function

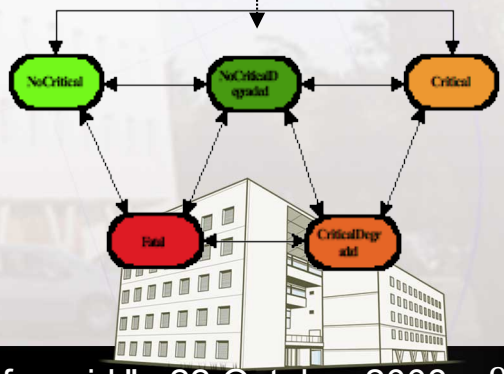


Control of a multi-scale measurement





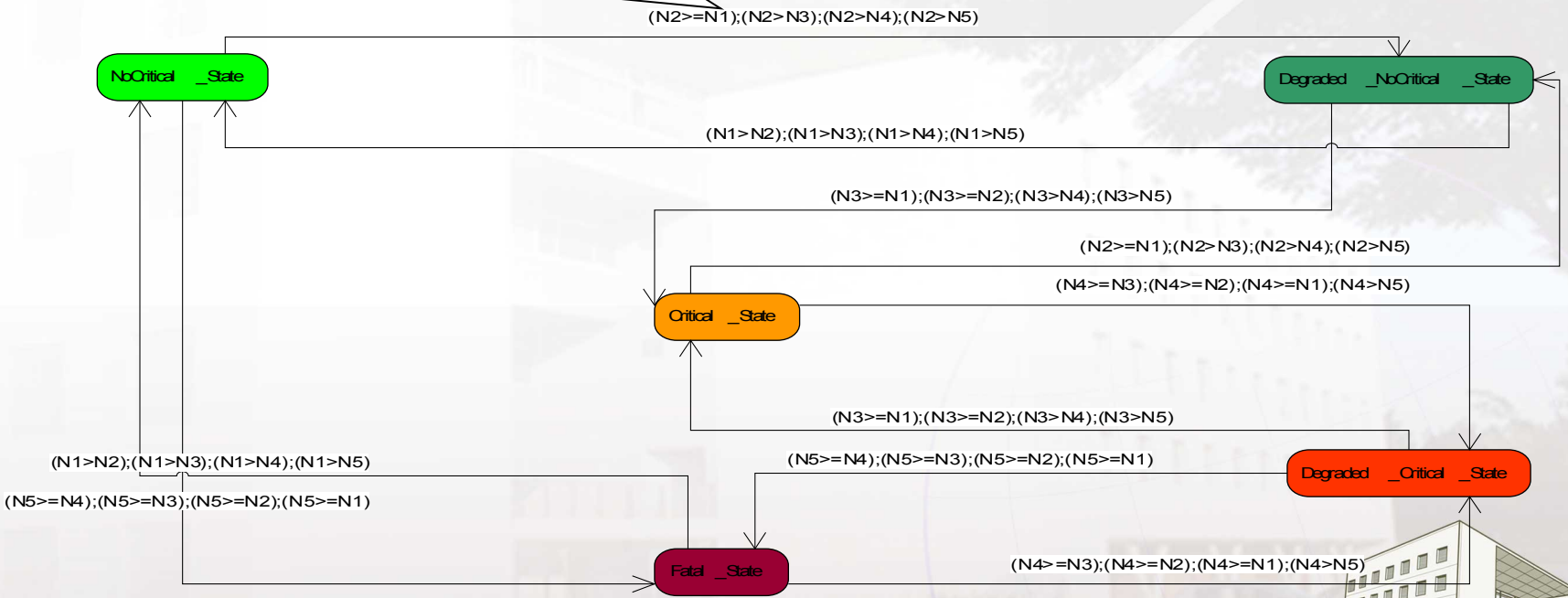
Need of Quantification



Aggregation of management Information

A Quantification of the criticality of states Example

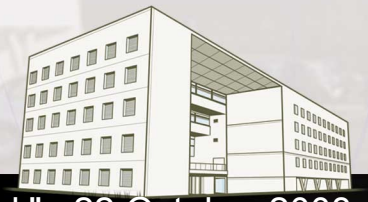
N1= f1(x)/F(x)=(([number of sensor in a NoCritical state]*10)*100)/([number of sensor in a NoCritical state]*10)+([number of sensor in a DegradeNoCritical state]*15)+([number of sensor in a Critical state]*20)+([number of sensor in a DegradeCritical state]*25)+([number of sensor in a Fatal state]*30))



Extract of the textual description of the StateChart diagram example

MOF (Management Object Format) notation of DMTF

```
statemachine IRIT_NumericSensorMachine CIM_NumericSensor{
    state NoCritical{
        transition ("DegradedNoCritical")
    {
            on signal_event( CIM_InstanceModification ), condition
     Condition1{
             action1
    };      }        }
…
}
```

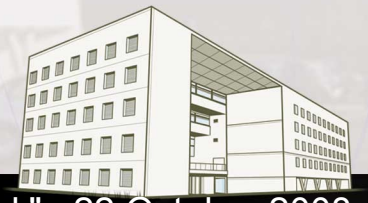


Condition1

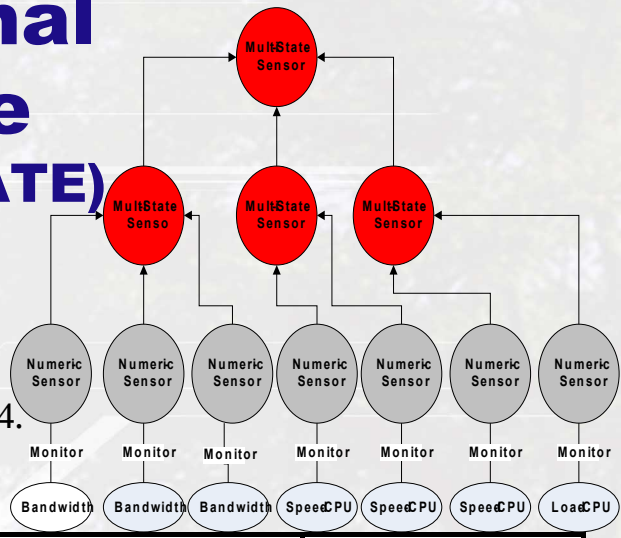
expressed in CQL (CIM Query Language) conditionnal language

```
$Instance$ //the instance to which is associated statechart
      diagram
SELECT*   // Specification of the evenement
FROM CIM_IndicationModification IM,
   ((SELECT
       (SELECT PropertyName
        FROM Monitor m
        WHERE       Antecedent = $Instance$
        )PN
    FROM LogicalDevice LD, Monitor m
    WHERE  m.Antecedent=$instance$
    AND       m.Dependant=LD
   ) VALUE,
// Specification of the condition from the NoCritical  state: C1
  (SELECT  MinThresholdNoCritical
   FROM NumericSensor
   WHERE
       $Instance$.System.CreationClassName=NumericSenso
       r.SystemCreationClassName
   AND
   $Instance$.SystemName= NumericSensor.SystemName

   AND
   $Instance$DeviceID= NumericSensor.DeviceID
  ) MinThresholdNoCritical
  ) C1,
  (SELECT SystemCreationClassName SCCN,
     SystemName SN, DeviceID DID
   FROM LogicalDevice LD, Monitor m
   WHERE m.Antecedent=$Instance$
   AND m.Dependent=LD
   )X1
WHERE IM.SourceInstance.SCCN=X1.SCCN
AND      IM.SourceInstance.SN=X1.SN
AND      IM.SourceInstance.DID=X1DID
AND      VALUE<C1.MinThresholdNoCritical
```



A pattern for multi-dimensional monitoring application example
(Figure table III. MUILTI-DIMENSIONAL STATE)



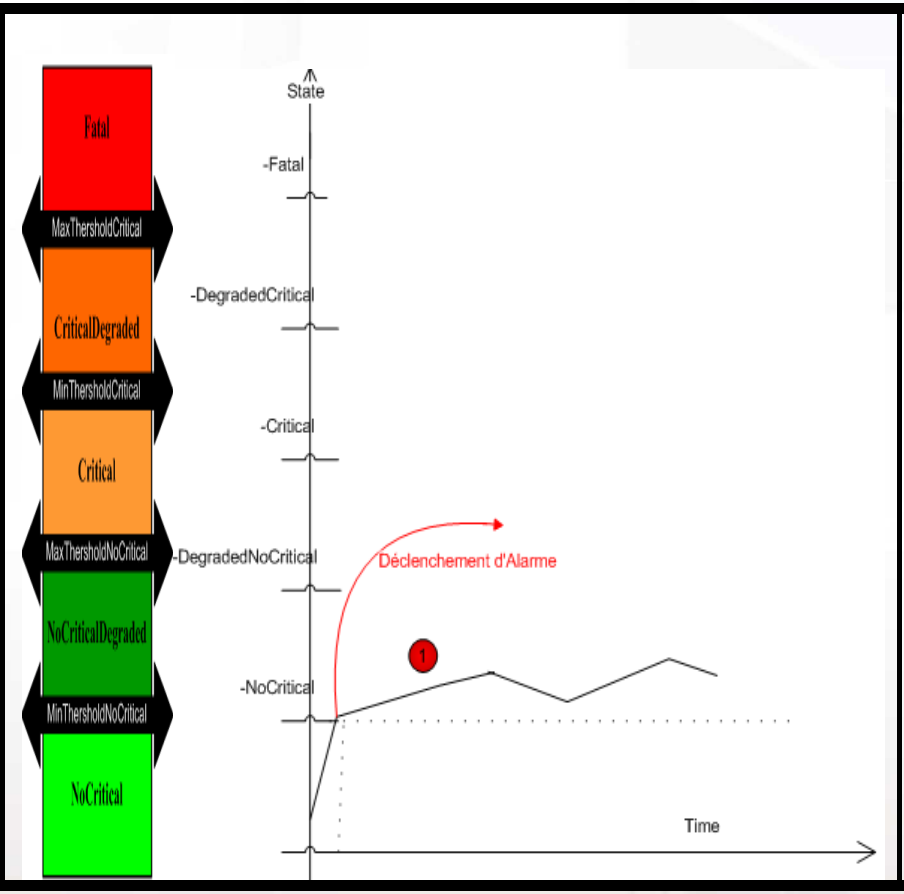
Hypothesis:
Current State of Sensors for Load CPU : Nnc=10, Nncd=7, Nc=14, Ncd=15, Nf=4.
Current State of Sensors for CPU Speed : Nnc=7, Nncd=11, Nc=16, Ncd=3, Nf=13.
Current State of Sensors for Bandwidth, : Nnc = 12, Nncd=27, Nc=32, Ncd=15, Nf=14.
And a=10, B=15, C=20, D=25, E=30.

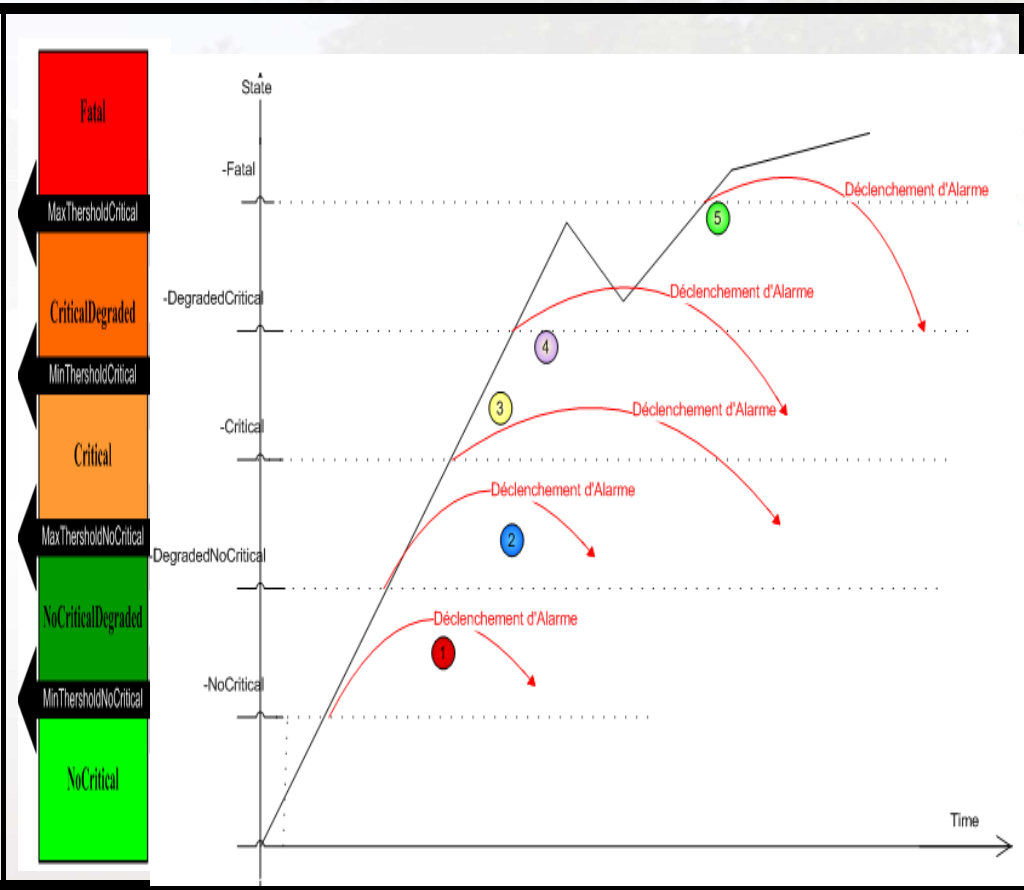
Pattern	MSS (Bandwidth)	MSS (Speed CPU)	MSS (Load CPU)	MSS (MMS Load and MMS Speed CPU)
Fi(x)=	f1(x)=120 f2(x)=405 f3(x)=640 f4(x)=375 f5(x)=420	f1(x)=70 f2(x) =165 f3 (x)=320 f4(x)=75 f5(x)=390	f1(x)=100 f2(x)=105 f3(x)=280 f4(x)=375 f5(x)=120	f1(x)=0 f2(x)=0 f3(x)=0 f4(x)=1 f5(x)=1
F(x)=[f1(x)+f2(x)+f3(x)+ f4(x) +F5(x)]/100	19.6	10.2	9.8	0,02
Valuation	N1<N4<N2<N5<N3	N4<N1<N3<N5	N1<N5<N3<N4	N4<N1<N4=N5
Multi-State Sensor	N2>Ni, (i !=2) Then his State is « **Degraded No-Critical** »	N5>Ni, (i !=5) Then his State is « **Fatal** »	N4>Ni, (i != 4) Then his State is « **Degraded Critical** »	N5=N4 Then his State is « **Fatal**»

Contributions

- A standard model (CIM).
- Generic observation pattern (for mono and multi scale measurement).
- Control of transition from one state to an another.
- Precision and personalization of control.
- Dignostic of a component monitored.
- The f ne scale with a same granularity that is instantiated for all properties monitored.
- Monitor of heterogeneous source.

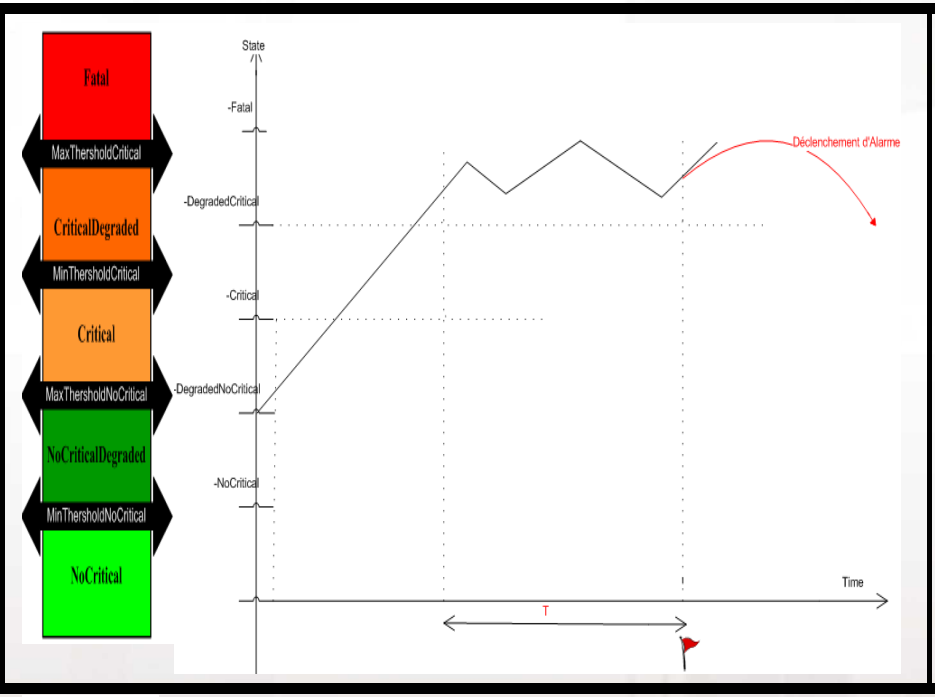
Alarms release problem

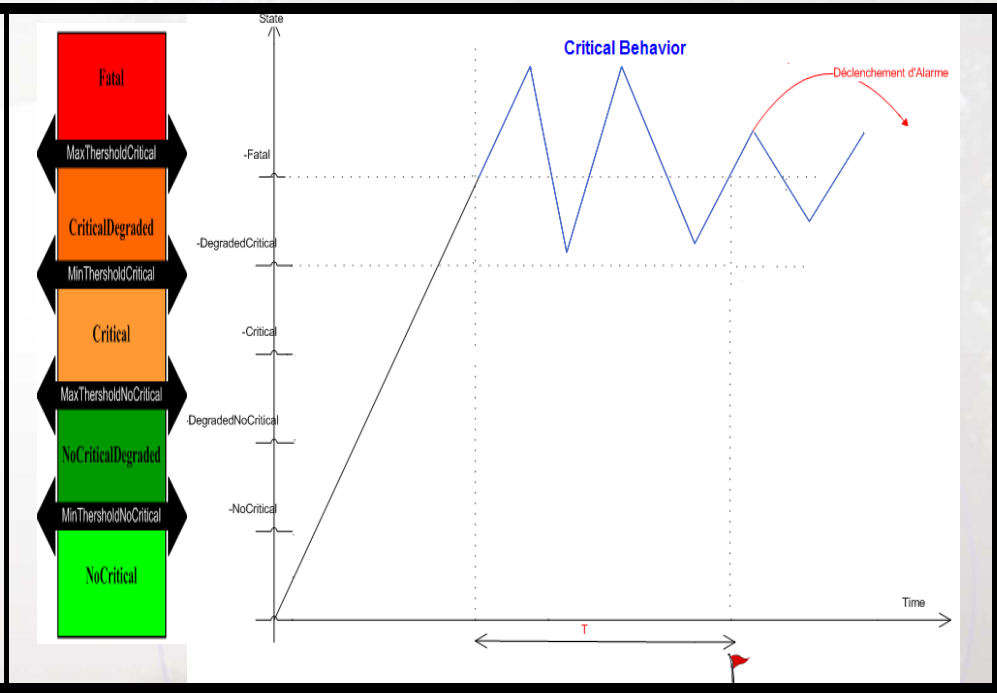




Future Work

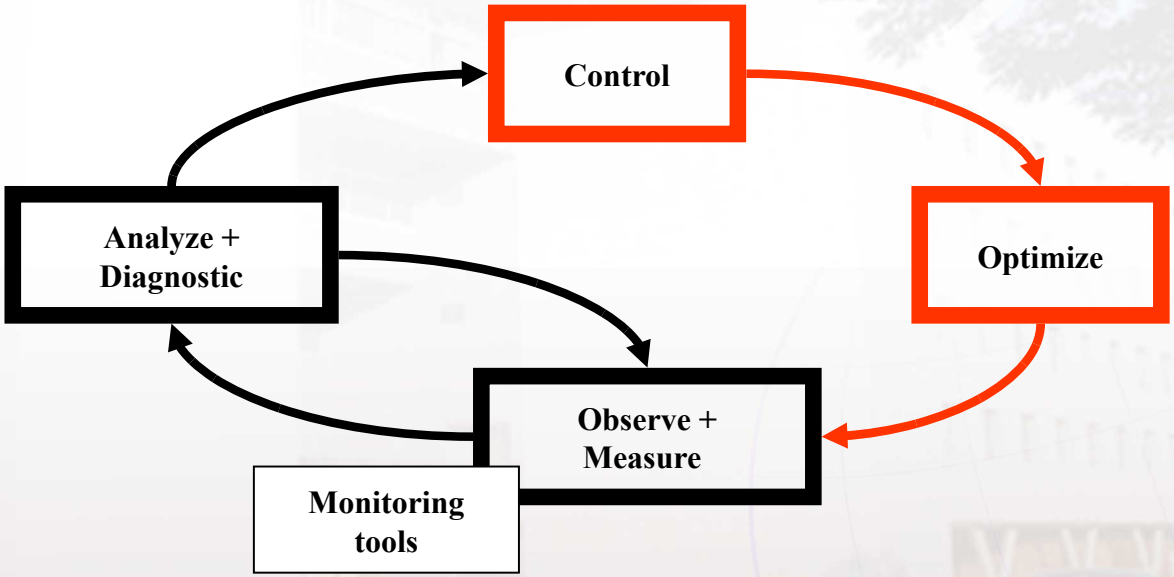
- Ref ning the analysis for more precision.
- Introduce the notion of time.
- Identify of a critical behavior: analysis improved of behavior in time + regular diagnostic of behavior –> statistical analysis –> detection of anomaly.





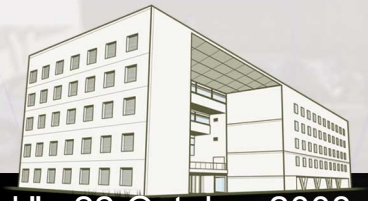
Future Work

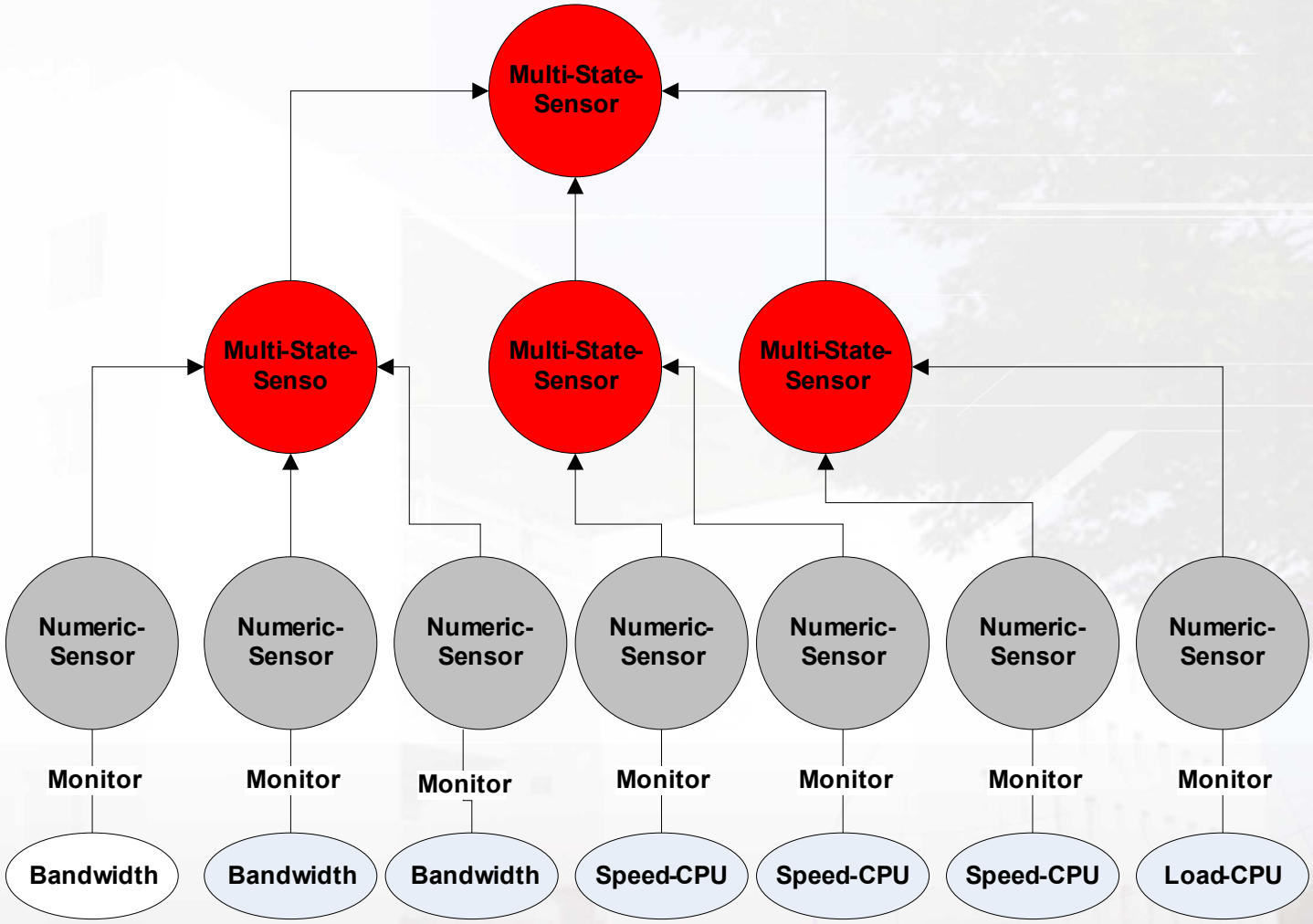
- Implementing the solution in Openpegasus.
- Work in other metric: jitter and latency.
- Improve the quantification which must be adjustable (for example: add a class for this).
- Integrate the solution into the autonomic loop



Thanks for your Listening...

QUESTIONS




Multi-State-Sensor
Multi-State-Senso
Multi-State-Sensor
Multi-State-Sensor
Numeric-Sensor
Numeric-Sensor
Numeric-Sensor
Numeric-Sensor
Numeric-Sensor
Numeric-Sensor
Numeric-Sensor
Monitor
Monitor
Monitor
Monitor
Monitor
Monitor
Monitor
Bandwidth
Bandwidth
Bandwidth
Speed-CPU
Speed-CPU
Speed-CPU
Load-CPU