A short story of Diagnosis
from passivity to on-line diagnosis of distributed systems

Xavier Le Guillou

DREAM project
IRISA – UR1
Rennes, France

2008/06/17
Definition

**Diagnosis** ['daɪəɡənəsɪs] *(diagnosis diagnoses)*

Diagnosis is the discovery and naming of what is wrong with someone who is ill or with something that is not working properly.

*(source Robert & Collins)*
Definition

**diagnosis** [,ˌdɛɪʒənəsɪs] (diagnosis diagnoses)

Diagnosis is the **discovery** and **naming** of what is wrong with someone who is ill or with something that is not working properly.

(source Robert&Collins)
Motivations

Diagnosis aims at:

- exhibiting faulty behaviours of a system
- identifying the underlying fault
Motivations

Diagnosis aims at:

- exhibiting faulty behaviours of a system
- identifying the underlying fault

Diagnosis is motivated by three-step logic:
Motivations

Diagnosis aims at:

- exhibiting faulty behaviours of a system
- identifying the underlying fault

Diagnosis is motivated by three-step logic:

1. every system is subject to faults
Motivations

Diagnosis aims at:

- exhibiting faulty behaviours of a system
- identifying the underlying fault

Diagnosis is motivated by three-step logic:

1. every system is subject to faults
2. faults are costly
Motivations

Diagnosis aims at:

- exhibiting faulty behaviours of a system
- identifying the underlying fault

Diagnosis is motivated by three-step logic:

1. every system is subject to faults
2. faults are costly
3. someone must pay
Diagnosis’ theory of evolution

Autonomous systems
Diagnosis’ theory of evolution

Autonomous systems

WNS policy
(wait and see)
Diagnosis’ theory of evolution

Autonomous systems

WNS policy (wait and see)       Off-line diagnosis

$t$
Diagnosis’ theory of evolution

Autonomous systems

WNS policy (wait and see)  Off-line diagnosis  On-line diagnosis

$t$
Off-line diagnosis

Role of forensics: no matter how long after a fault, determine what fault happened.

- sufficient for certain problems
  - predictive diagnosis
  - flaw discovery
  - determination of frequent faults
- inadequate for many dynamic systems...
Off-line diagnosis

Role of forensics: no matter how long after a fault, determine what fault happened.

- sufficient for certain problems
  - predictive diagnosis
  - flaw discovery
  - determination of frequent faults
- inadequate for many dynamic systems...

⇒ need for on-line diagnosis
On-line diagnosis

Role of monitor: permanently provide an explanation to an incomplete flow of ordered observations.

- need for a model of the system
- need for efficient algorithms

We consider the “diagnoser” approach.
The model

In this approach, an automaton represents the trajectories of the system

\[ 1 \xrightarrow{a} 2 \xrightarrow{c} 3 \xrightarrow{f1} 4 \xrightarrow{b} 5 \]
The model

In this approach, an automaton represents the trajectories of the system.

From this automaton we extract a deterministic “diagnoser”.
At run-time

- A flow of observable events is generated by the system
- The diagnoser is fed by this flow
- A (partial) diagnosis is always available
Diagnosis’ theory of evolution (r2)

A short story of Diagnosis
A first step: decentralized systems

The system:
- a set of components
- a single flow of observations

The diagnosis method:
- merging automata thanks to a shared alphabet
- building the diagnoser
- recognizing on-line
How to merge automata...
How to merge automata...
How to merge automata...
How to merge automata...
Limits of this methods

- Global knowledge of the system
- Single flow of events
- Complexity of the global automaton ($e^{|c|}$)
On-line diagnosis of distributed systems

The very idea:
- apply a monitoring algorithm locally
- merge local diagnoses on a global diagnoser

The very crucial thing:
- find a valid merging operation
Our method

At design time:

1. list all the possible behaviours of a component
2. “label” the status of variables exchanged between components for each path
3. decide whether this path can trigger a global diagnosis process
About status of variables

Considering different behaviours (diagnoses):

- **normal case**
  - both *param* and *return* are correct
- **local error**
  - both *param* and *return* are erroneous
- **external error**
  - *param* is correct but *return* is erroneous
Merging strategy

Local diagnoses can only merge if their variables have the same status:

Normal Case

1

2

3

4

5

(param:corr)

(return:corr)

Incoming Error

1

2

1

2

(in:err)

(out:err)

Normal Case

1

2

3

4

5

(param:corr)

(return:corr)

A short story of Diagnosis
Merging strategy

Local diagnoses can only merge if their variables have the same status:

- Normal Case
- Incoming Error

Diagram shows the process with nodes and arrows indicating the flow of information and variables (param:corr) and outcomes (return:corr, out:err).
Merging strategy

Local diagnoses can only merge if their variables have the same status:
Merging strategy

Local diagnoses can only merge if their variables have the same status:

Normal Case

1
2
3
4
5

(param:corr)
(return:corr)

2
1

(in:corr)
(out:corr)
Where is the interest?

Concurrence between local behaviours: refinement

Normal Case

1

\(\text{(param:corr)}\)

2

\(\text{(return:corr)}\)

3

\(\text{(return:corr)}\)

4

5

Normal Case

\(\text{(in:corr)}\)

1

\(\text{(out:corr)}\)

2

\(\text{(in:err)}\)

1

\(\text{(out:err)}\)

2

Incoming Error

A short story of Diagnosis
Where is the interest?

Concurrence between local behaviours: refinement

Normal Case

1

2

(param:corr)

3

(return:corr)

4

5

Normal Case

1

2

(in:corr)

(out:corr)

Incoming Error

1

2

(in:err)

(out:err)

A short story of Diagnosis
Conclusion

A decentralized approach to monitor distributed systems:
- respect of privacy (no intrusion)
- no need for global model

Prospects:
- include a model of interactions
- learn model from logs