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 [ˌdaɪ.əgˈnəʊ.sɪs] *(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 [ˌdaɪ.əgˈnəʊ.sɪs] *(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:

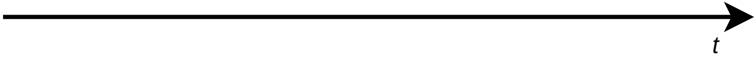
- exhibiting faulty behaviours of a system
- identifying the underlying fault

Diagnosis is motivated by three-step logic:

1. every system is subject to faults
2. faults are costly
3. someone must pay

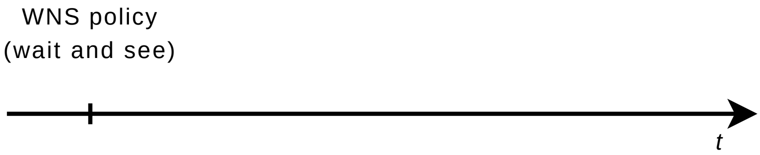
Diagnosis' theory of evolution

Autonomous systems



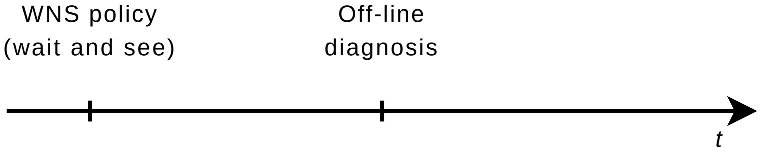
Diagnosis' theory of evolution

Autonomous
systems



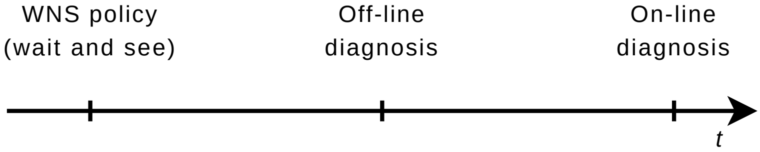
Diagnosis' theory of evolution

Autonomous systems



Diagnosis' theory of evolution

Autonomous
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...

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...

$\Rightarrow$ need for on-line diagnosis

On-line diagnosis

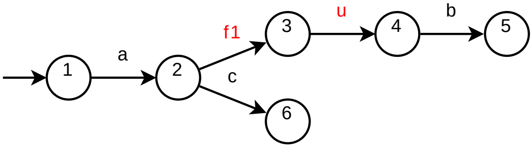
Role of monitor: permanently provide an explanation to an incomplete flow of ordered observations.

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

We consider the “diagnoser” approach.

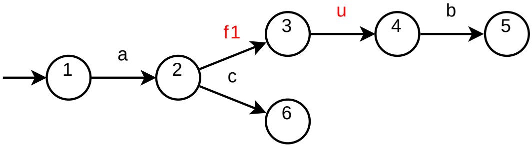
The model

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

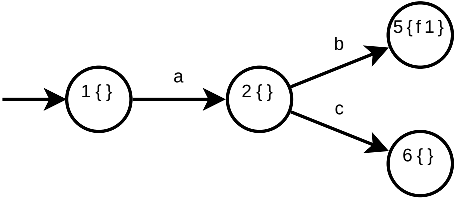


The model

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



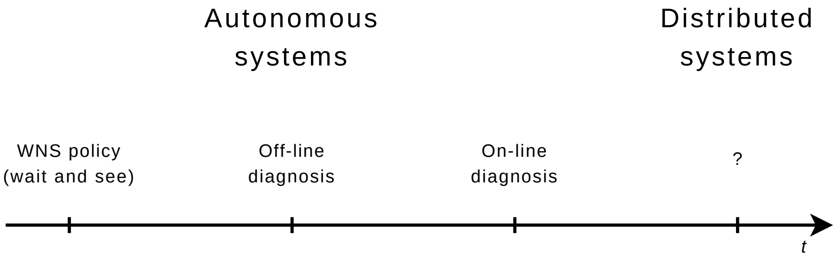
From this automaton we extract a deterministic "diagnoser"



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 first step: decentralized systems

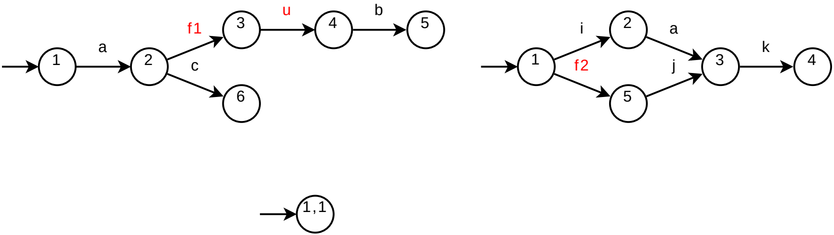
The system:

- a set of components
- a single flow of observations

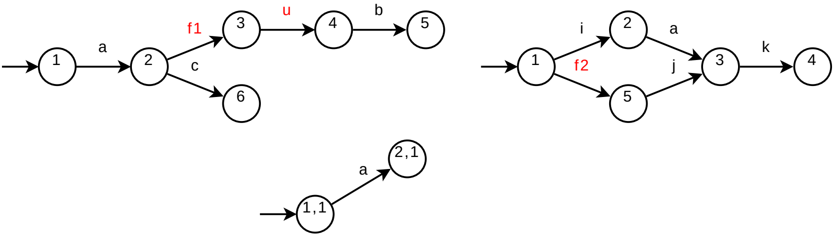
The diagnosis method:

- merging automata thanks to a shared alphabet
- building the diagnoser
- recognizing on-line

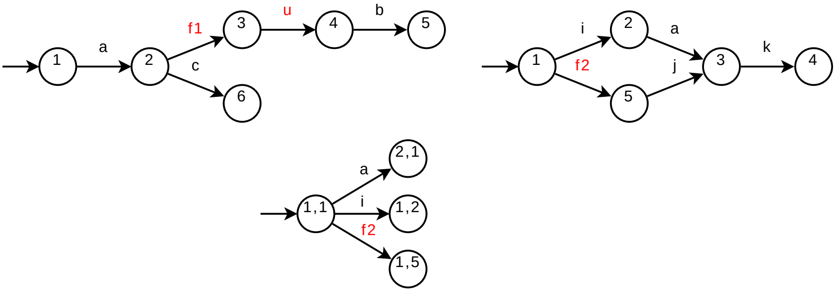
How to merge automata. . .



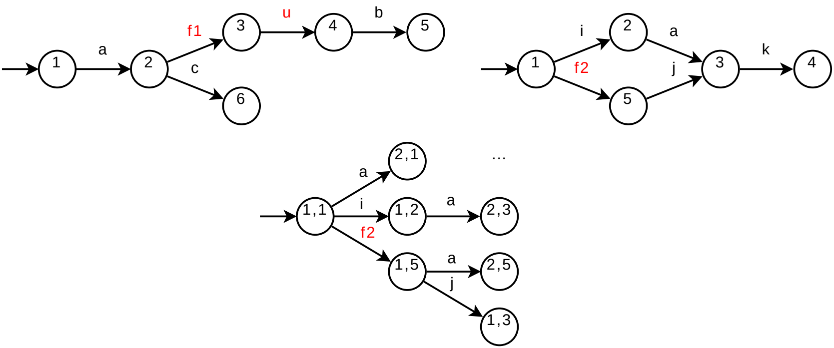
How to merge automata. . .



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:

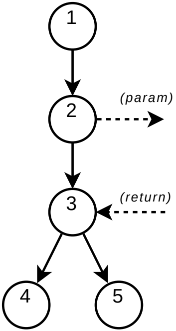
- find a valid merging operation

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

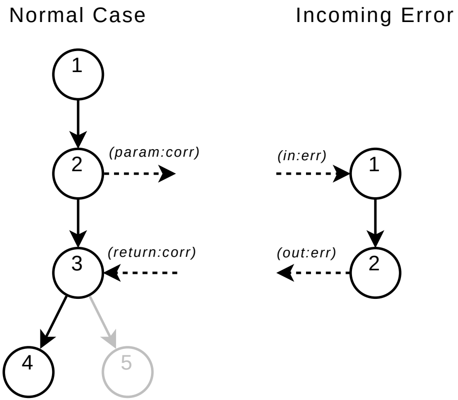


Considering different behaviours (diagnoses):

- 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

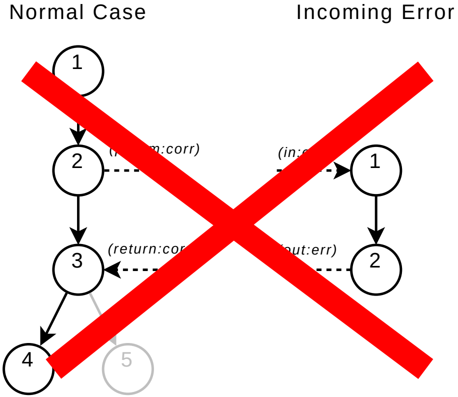
Merging strategy

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



Merging strategy

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

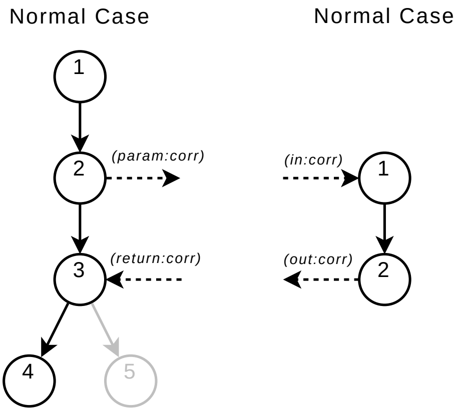


Merging strategy

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

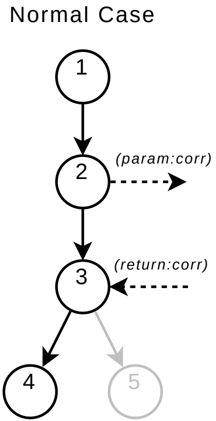
Merging strategy

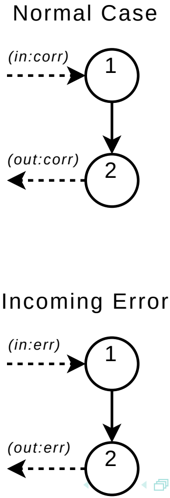
Local diagnoses can only merge if their variables have the same status:



Where is the interest?

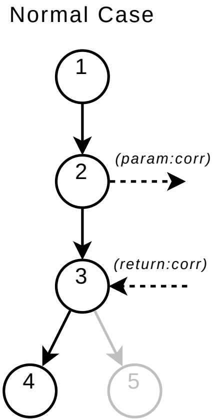
Concurrence between local behaviours: refinement

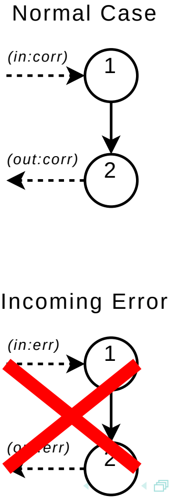




Where is the interest?

Concurrence between local behaviours: refinement





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