Towards Self-Diagnosing Web Services

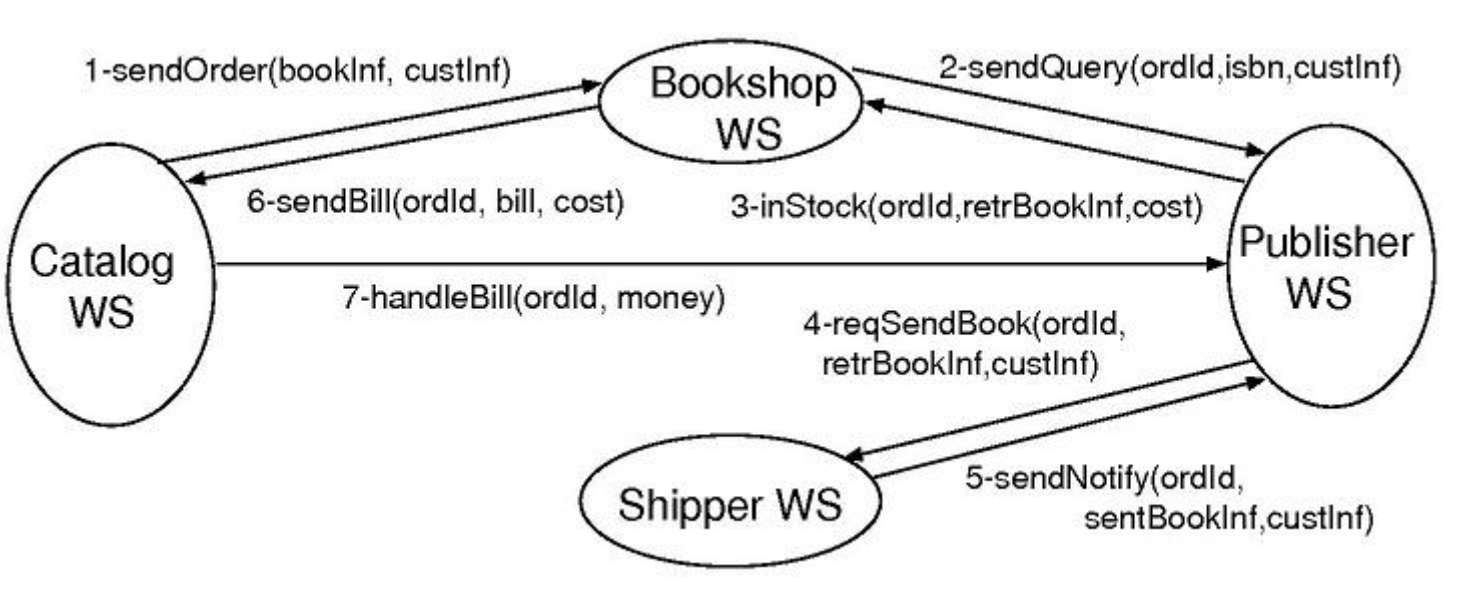
L. Ardissono • L. Console • A. Goy • G. Petrone
C. Picardi • M.Segnan • D. Theseider Dupré

Dipartimento di Informatica
Unversità di Torino (Italy)

Motivations

- Current practice for dealing with faults in distributed software systems:
  - exception handling
  - no attempt at identifying causes
- Aim: Advanced diagnostic capabilities for complex Web services (composed from individual services)
  - identifying the faulty service to apply the proper recovery action
  - Towards self-healing Web Servioces
- We propose a Model-based diagnosis approach for localizing the faulty service

Motivating example



If the customer receives the wrong book, which are the possible causes?

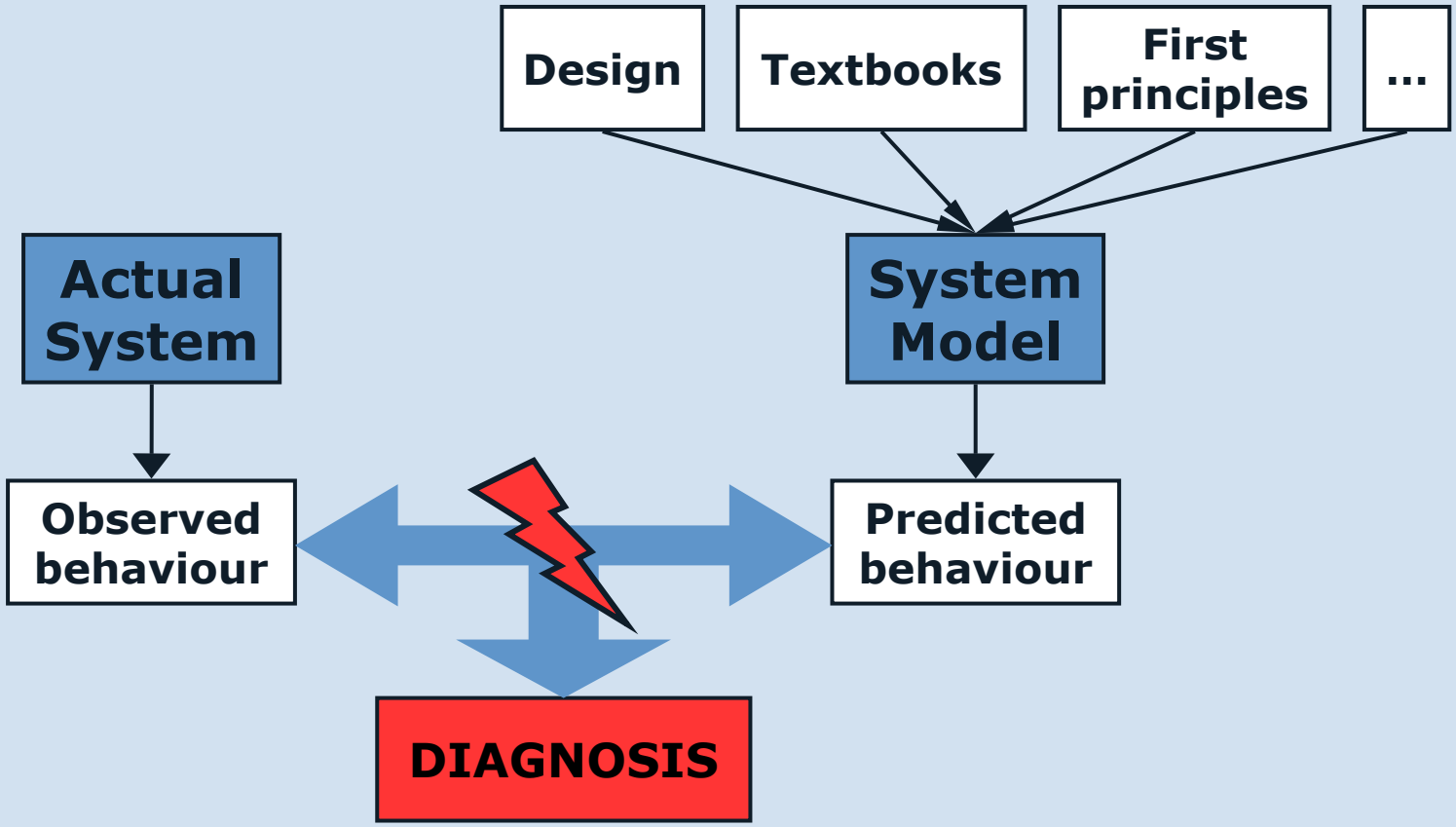
Model-Based Diagnosis (1)

- An approach to automated diagnosis
  - from AI (Artificial Intelligence) and Engineering
- Diagnosis:
  - finding the cause(s) of an unexpected behavior
  - determining the most appropriate repair/recovery action
  - Detection VS Identification VS Recovery (Repair)
- Main application
  - artefacts
- Basic assumption
  - a Model of the artefact is available

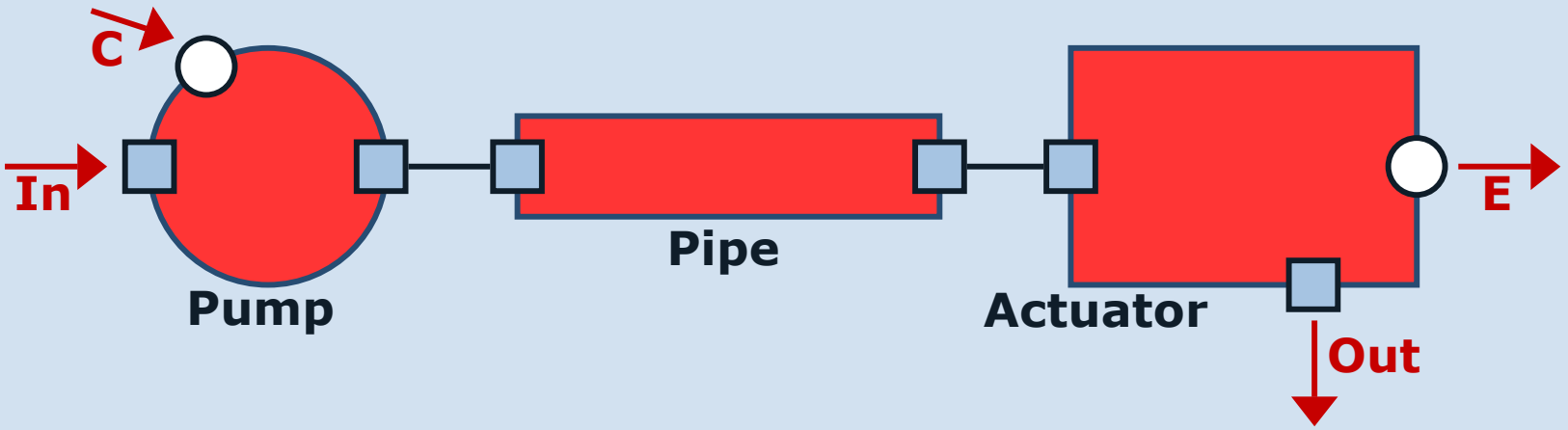
Model-Based Diagnosis (2)

- Different approaches to modelling.
- We focus on component-oriented modelling:
  - **Structure** of the artefact (the Complex Service):
    - components (services) and their connections to define super-components (component hierarchy)
  - **Function** or **Behaviour** of its component types (individual or elementary) services:
    - Nominal behavior
    - Behavior in presence of faults
  - **Qualitative** Models
    - Variables express qualitative properties of the system
    - e.g: **low**/**high** or **present**/**absent** or **correct**/**incorrect**

Model-Based Diagnosis (3)



Example



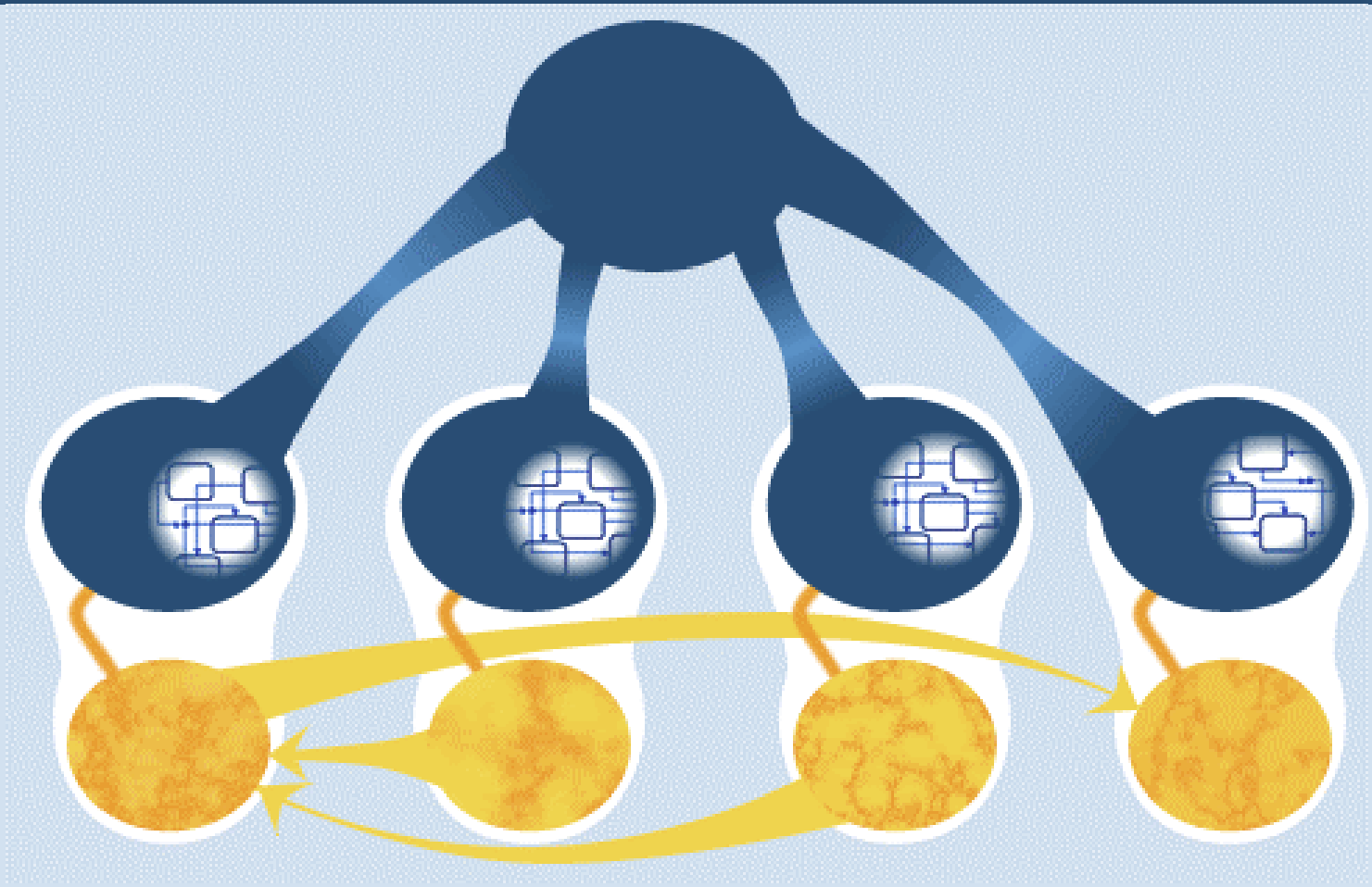
Component-oriented Models of WSs (1)

- Model of WS: abstraction of its computation
  - A set of **activities** with I/O variables
  - activity ≡ component (smallest diagnosable unit) with behaviour modes ***ok*** and ***fail***
- Model: Relation between such variables
  - Which variables are affected by each activity
  - Which variables may result as abnormal (***ab***) in case an activity fails
- Assumption:
  - for each activity in the ***ok*** mode, all inputs ***ok*** ⇒ all outputs ***ok***

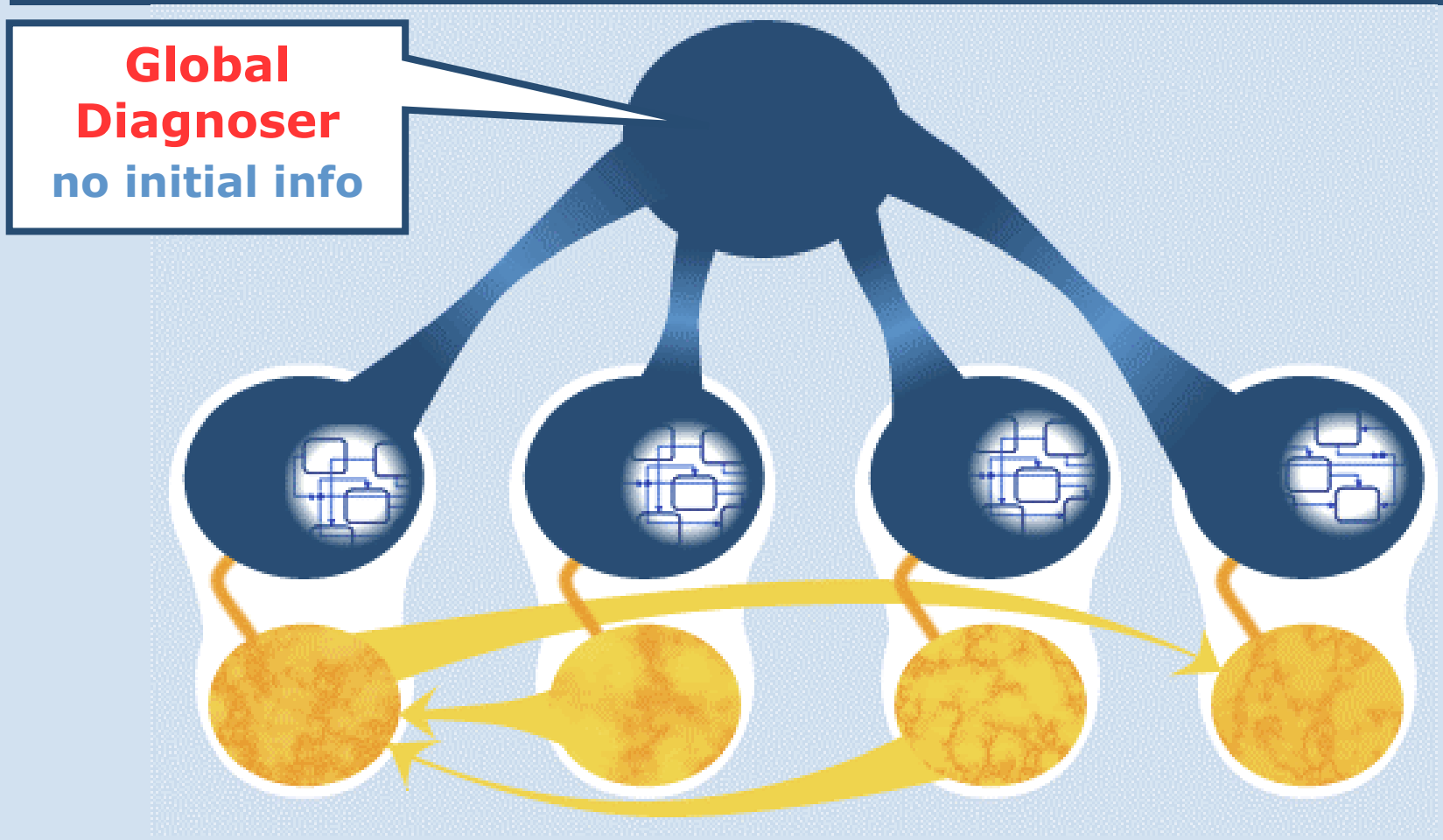
Component-oriented Models of WSs (2)

- Diagnosis is activated by **alarms** in the WS
- An alarm ***a***
  - typically corresponds to a **mismatch** of two variables ***x*** and ***y***
  - Or to an **unexpected** value of a variable
- The model contains also checkpoints:
  - analogous to alarms
  - evaluated **on demand**, not automatically.

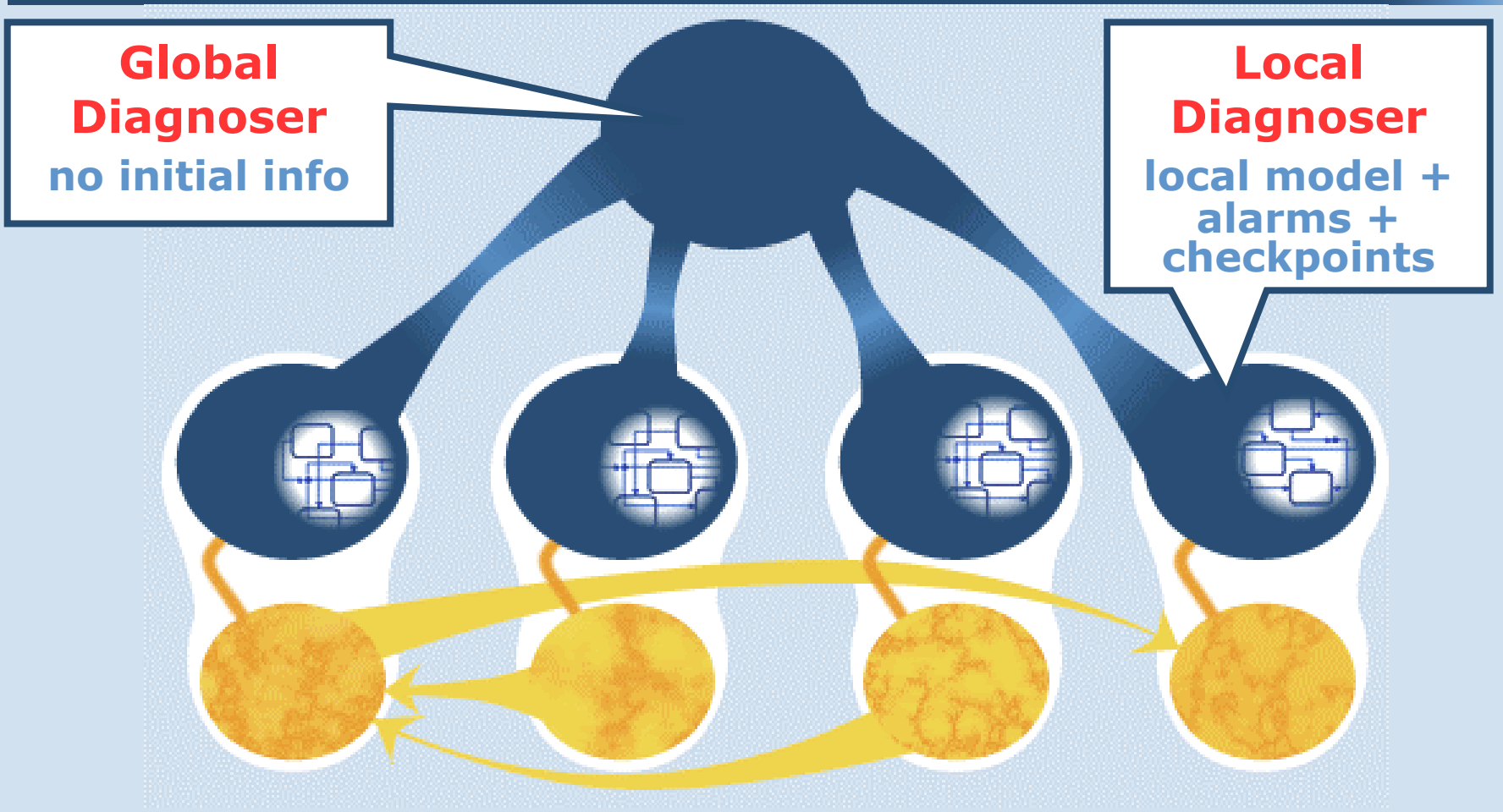
Decentralized Diagnosis



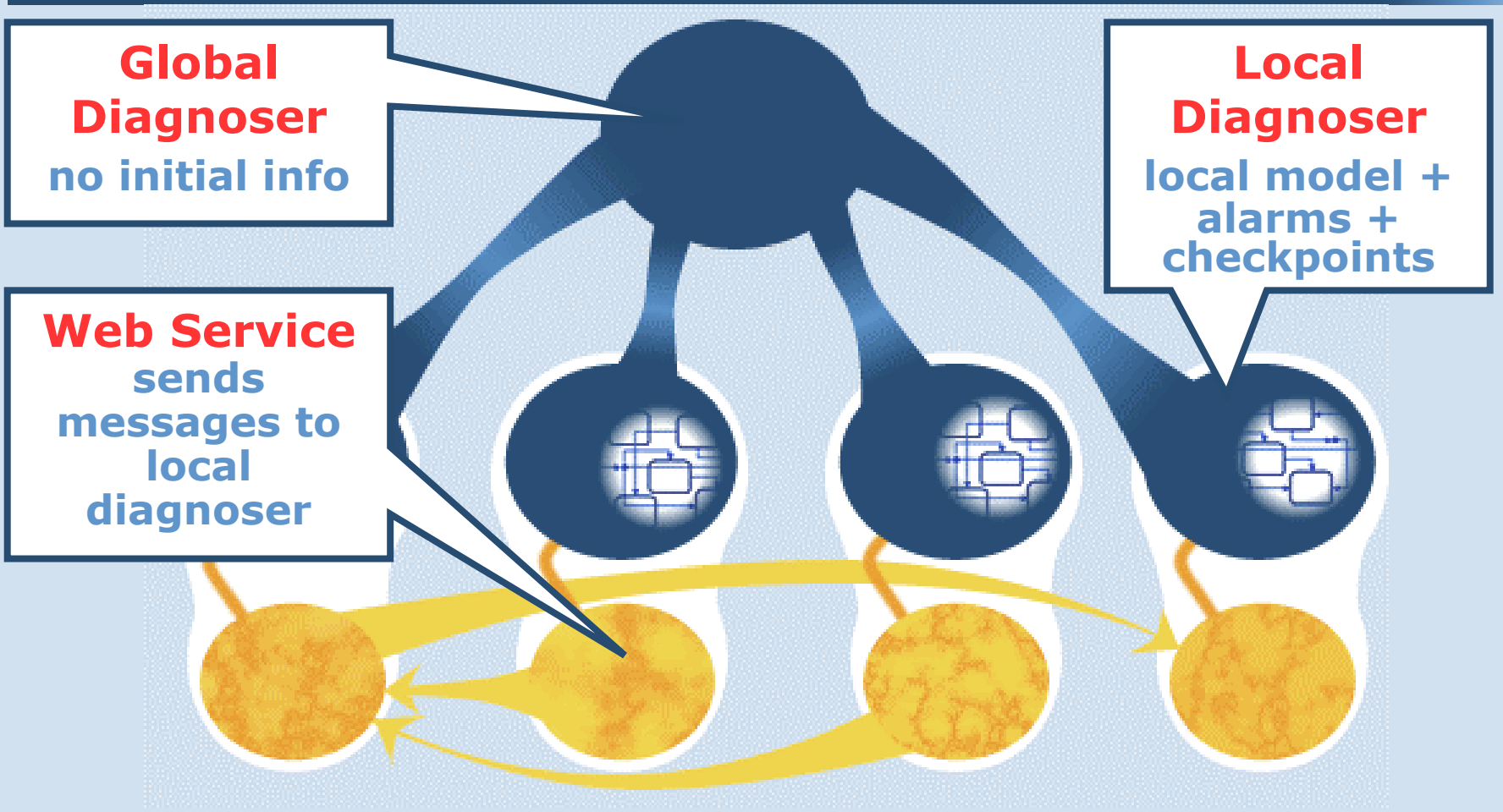
Decentralized Diagnosis



Decentralized Diagnosis



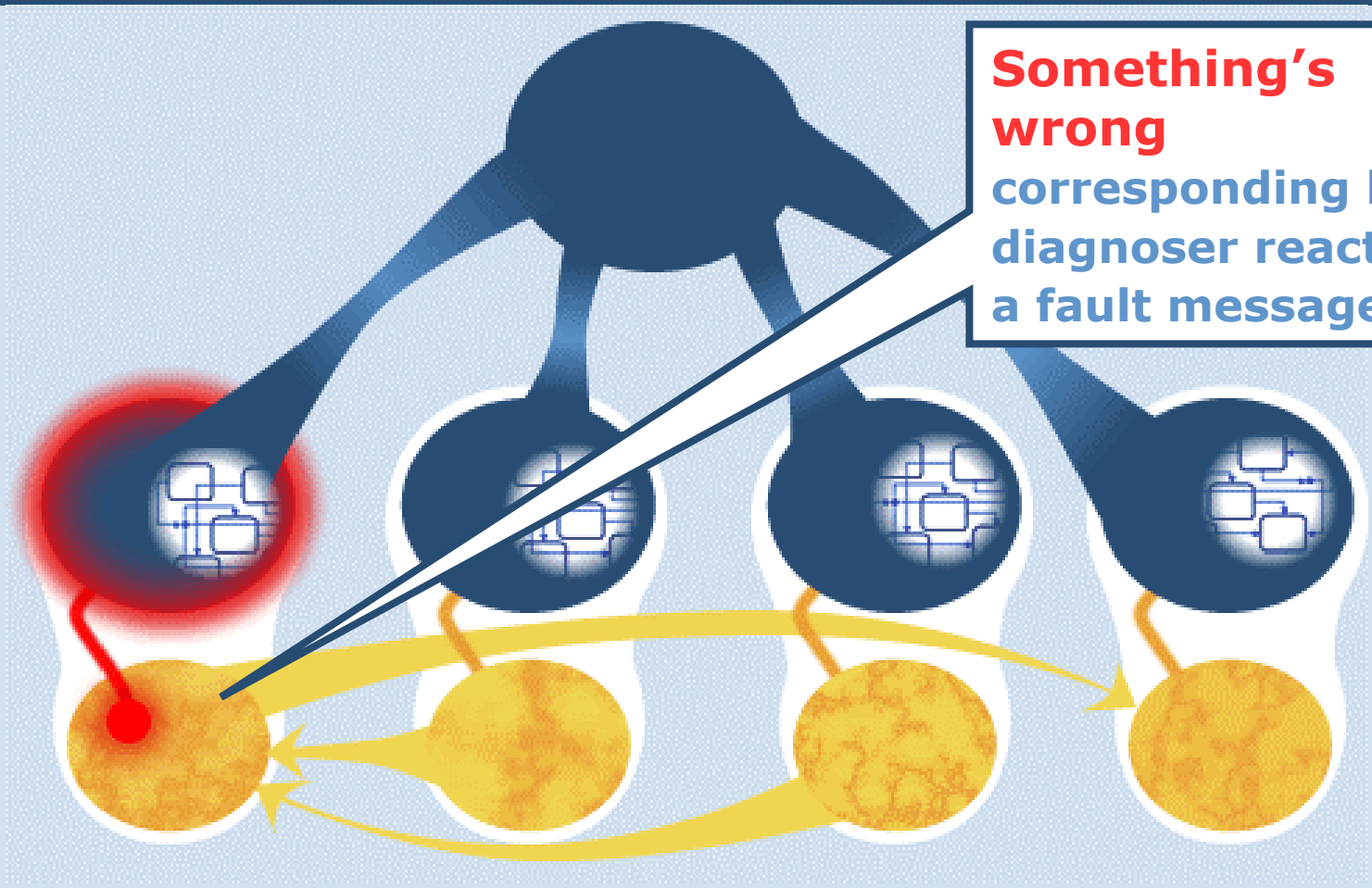
Decentralized Diagnosis



Our Approach

- We provide:
  - a **specification** of local diagnoser operations
  - a **formal characterization** of local diagnoser operations
  - A **communication protocol** between local and global diagnosers
  - an **algorithm** for the Global Diagnoser
    - starts with no information on local services
    - the algorithm only assumes that local diagnosers meet the specifications ofr their operations
    - the algorithm **merges** information from local diagnosers and **decides** which local diagnosers to contact.

Starting Diagnosis Upon Alarms



Starting Diagnosis Upon Alarms

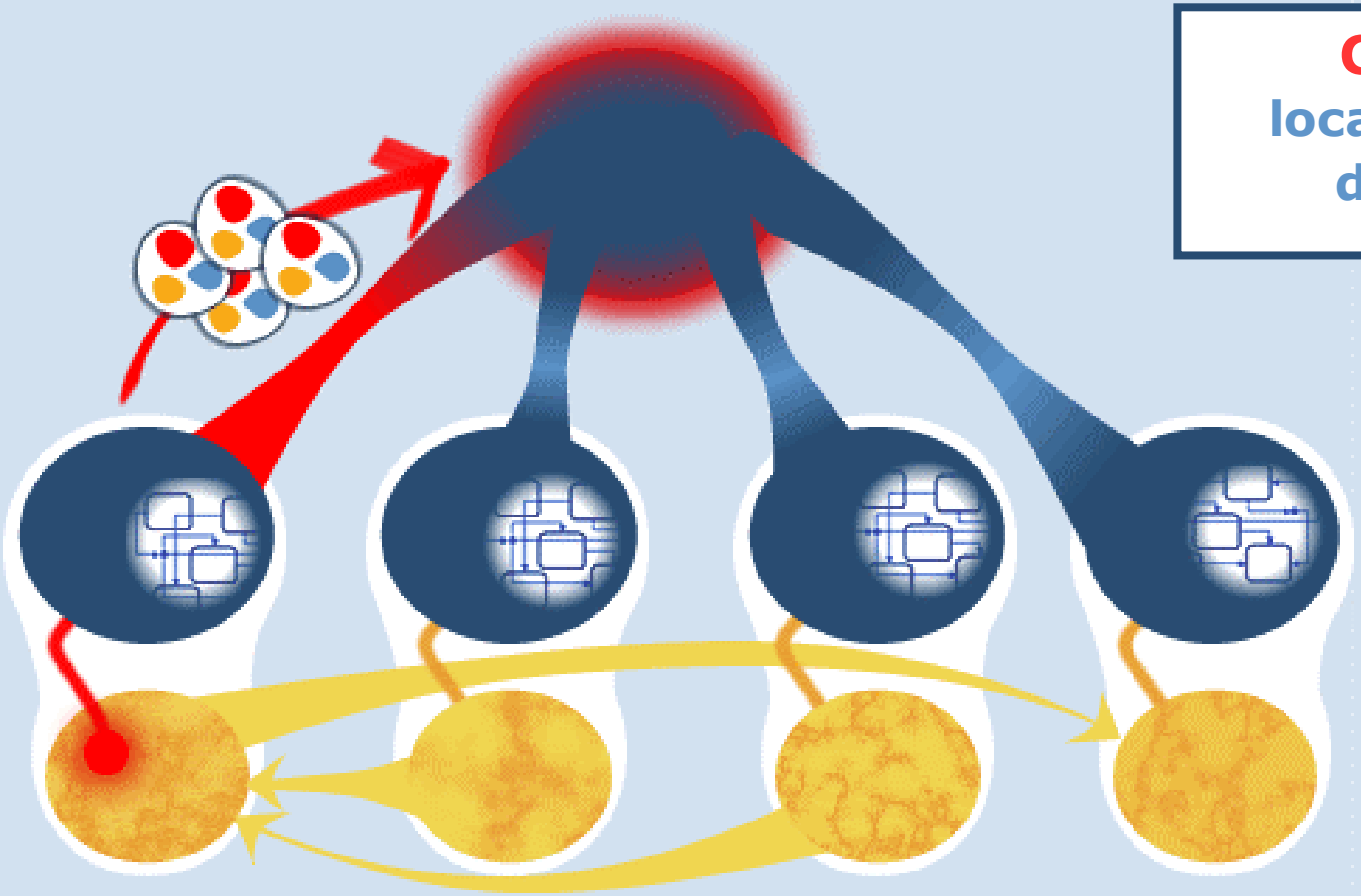
- **Initial info**:
  - local **observations** (alarms + checkpoints) **OBS**
- **Compute**:
  - a set of **candidate diagnoses** → hypotheses of misbehaviour that explain **OBS**
    - **internal misbehaviour**: errors occurred inside the WS
    - **external misbehaviour**: errors in inputs received from other WSs (**blame on other services**)
  - **consequences** of each hypothesis on service outputs
    - can be used to validate/discard a candidate diagnosis
- Standard **MBD techniques** can be applied.

Local Candidate Diagnosis

A **local candidate diagnosis** contains three elements:

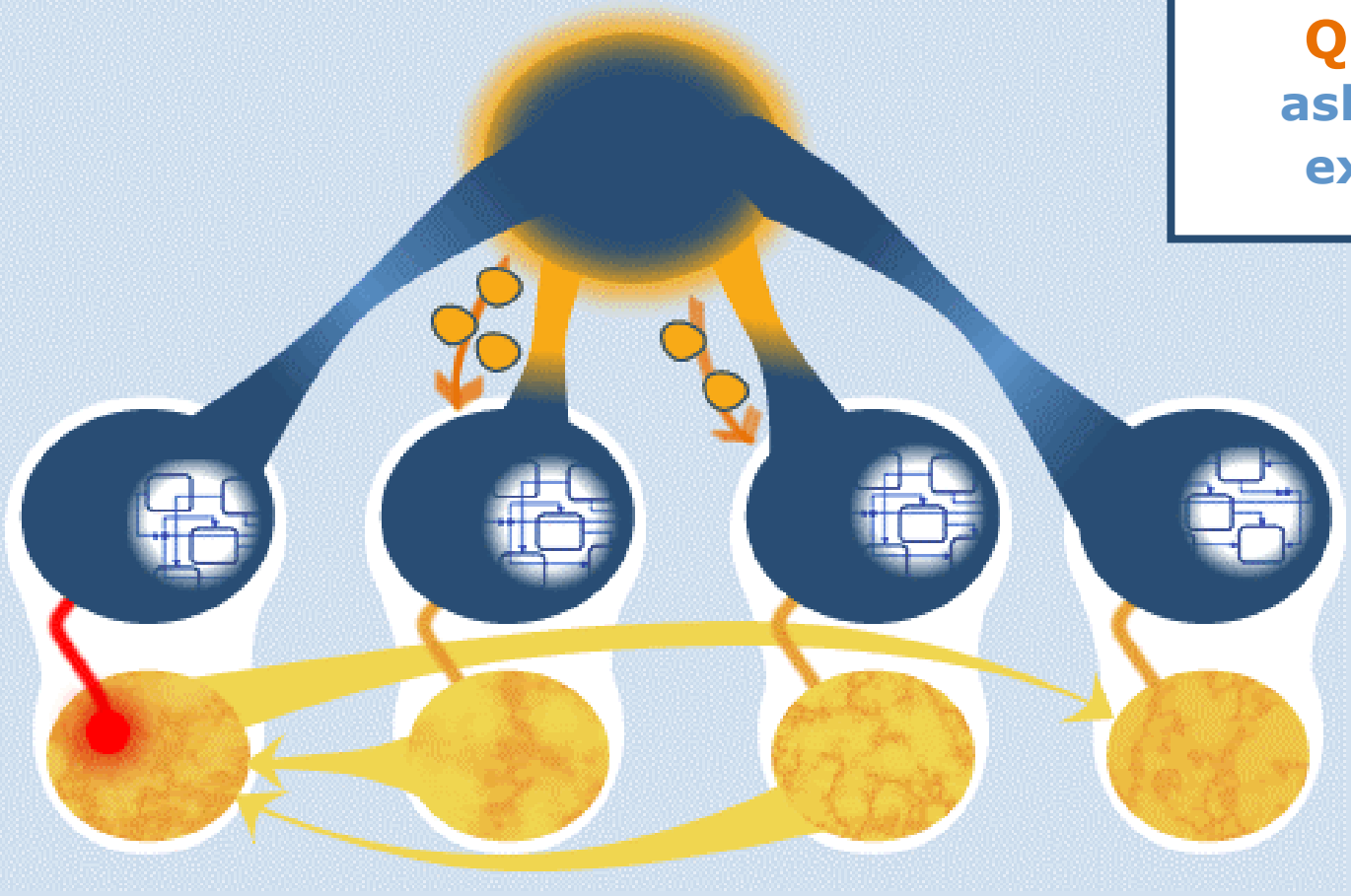
- hypotheses on local behaviour
- blames on other (input) services
- consequences of hypotheses on other (output) services

The Role of the Global Diagnoser



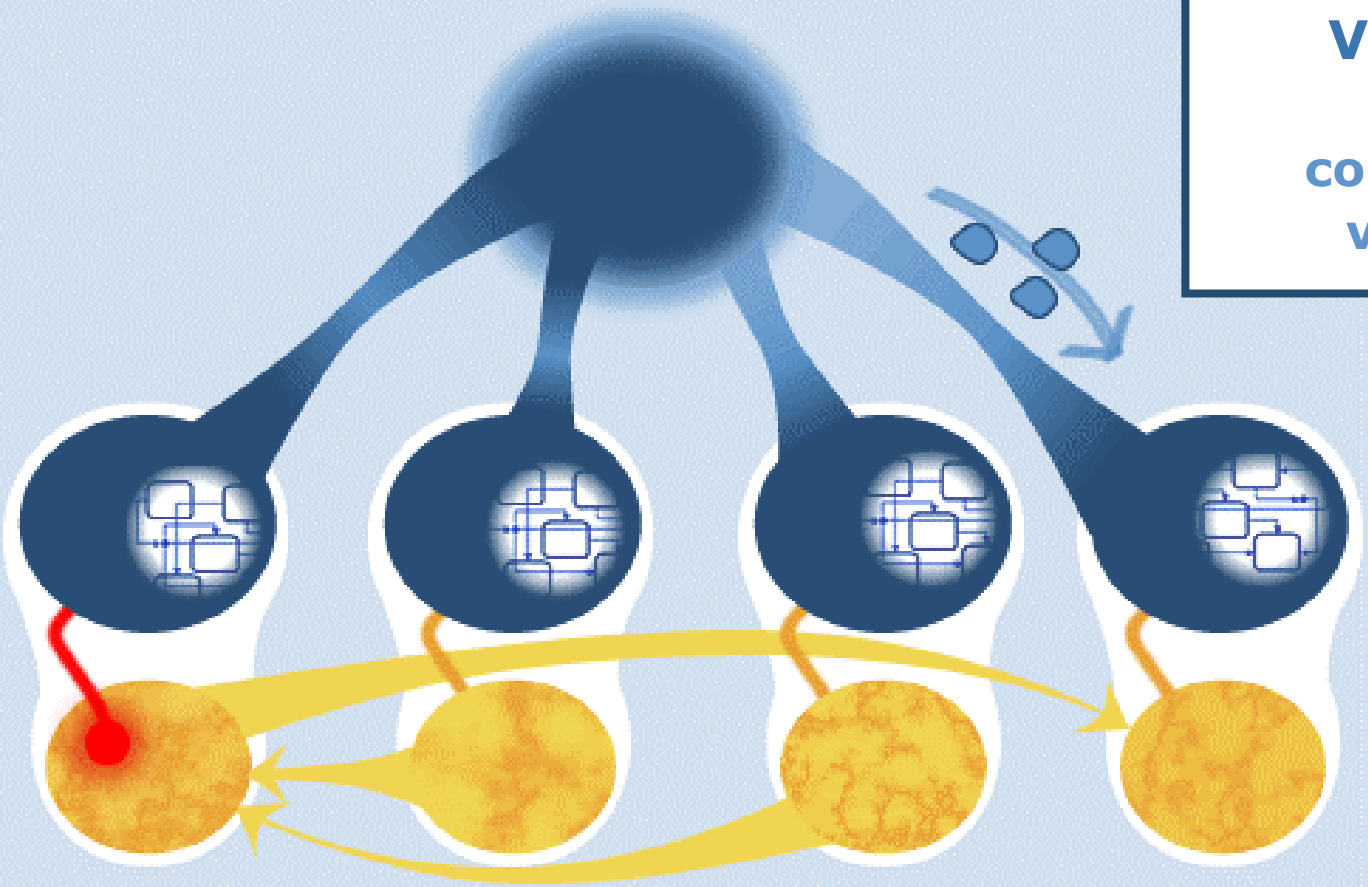
COLLECT
local candidate diagnoses

The Role of the Global Diagnoser



QUESTION
ask for blame explanation

The Role of the Global Diagnoser



VALIDATE
ask for consequence validation

LocalDiagnosers - Explanation

- Local diagnoser receives **blames**
- It produces local candidate diagnoses that explain **observations AND blames**.
  - additional hypotheses of internal misbehaviour
  - additional blames
  - additional consequences
- New **local candidate diagnoses**:
  - **merged** with the ones that originated the blame **by the global diagnoser**
- If no explanation:
  - the candidate diagnosis that originated the blame is **rejected by the global diagnoser**

LocalDiagnosers - Validation

- Local diagnoser receives **consequences**
- It **verifies** through local observations whether the consequences hold.
- Produces:
  - additional consequences on other services
- If **initial consequences** hold:
  - **the global diagnoser adds new consequences** to the local candidate diagnosis that originated them.
- If **initial consequences** do not hold:
  - the candidate diagnosis that originated them blame is **rejected by the global diagnoser**.

Characterization of Local Diagnosers

- Candidate diagnoses are represented by **partial assignments** to model variables
  - assignments of behaviour modes to internal activities
  - assignments of correctness status to model variables
- For both **explanation**/**validation**:
  - local diagnosers receive the parts of the assignments that concerns them
  - work by **completing** partial assignments
  - operation can be carried out by standard MBD techniques
- Both can be characterized in the same way
  - **one operation** that **explains** and **validates** at the same time.

The Global Diagnoser

- Each request for **explain**/**validate**
  - produces new **blames**
  - produces new **consequences**
- The Global Diagnoser:
  - repeatedly asks for explanations and validations
  - until there is nothing to explain/validate
- A local diagnoser may be invoked **multiple times**
  - the general case **does not assume a persistency** of local diagnosers
    - each invocation can be considered separately
  - however persistency improves efficiency.

An Intelligent Global Diagnoser

- The global diagnoser keeps track of candidate diagnoses
  - information from different local diagnoser mantained as a **set of partial assignments**
- Intelligent behaviour to **reduce overhead**:
  - depending on assigned/unassigned variables may avoid questioning some services
- May exploit (if present) information on **workflow**
  - in order to **focus diagnosis**
  - in order to select an **optimal questioning order**, to avoid multiple calls to the same local diagnoser

Conclusions and Future Work

- **Advantages** of the approach:
  - **reduction** of communication overhead
    - **decentralized** VS purely distributed
    - does not explore the **whole model** if not necessary
    - no **restrictive assumptions** on the models
  - abstract models of correctness propagation
    - could be at least partially **derived automatically** (to investigate)
- **Future** work:
  - exploit **coordination** mechanisms and coordination info
  - local diagnosers only characterized
    - propose **efficient algorithms** for local diagnosers