JADE
An Autonomic Management System Experiment with Self-Healing

Fabienne Boyer, Sylvain Sicard

INRIA Rhne-Alpes
655, avenue de l'Europe, Montbonnot
38334 St Ismier Cedex, France
http://www.inria.fr

Sardes
System Architecture for Reflective Distributed Environments

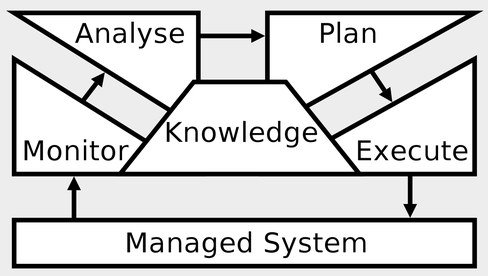
{Fabienne Boyer, Sylvain Sicard}@inrialpes.fr
http://sardes.inrialpes.fr/research/jade

1 Introduction

- Distributed systems
  - Many nodes
  - Heterogenous environment
  - Dynamic context
  - Failure probability increase with number of nodes
  - ...
  - ...
- Management
  - Complex task (configuration, deployment, failure management, etc)
  - Achieved by humans
- Consequence
  - Error prone (configuration files)
  - Low reactivity
  - Overcost (Hardware/Human)

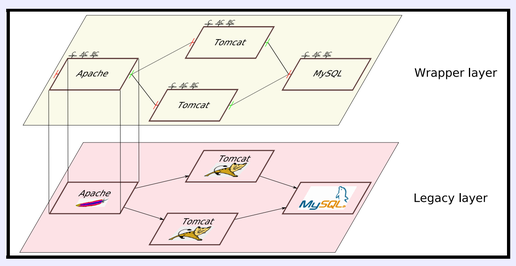
2 Approach

- Autonomous Systems
- Properties (Kephart et al. [2])
  - Self-Configuration
  - Self-Healing
  - Self-Optimizing
  - Self-Protect
- Improvements
  - Less errors
  - Higher reactivity
  - Better ressource usage

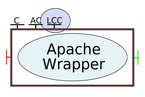


3 Legacy Wrapping

- Systematic legacy wrapping
  - Management of legacy entities is wrapped in FRACTAL components
  - Provides
    - Uniform view of management interfaces
    - Introspection capabilities
    - Architectural view of legacy software
  - Legacy and wrapper are collocated
- Gain
  - Architectural view
  - Introspection, Monitoring
  - Deployment
  - Reconfiguration
  - Uniform Management interface



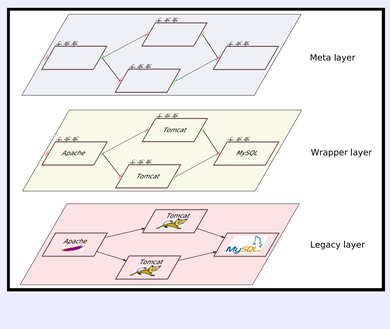

```
public Interface LifeCycleController {

   /** Starts the component to which this interface belongs. */
   public void startFc() ;

   /** Stops the component to which this interface belongs. */
   public void stopFc() ;
}
```




```
Public class ApacheWrapperImpl implements LifeCycleController, ... {

...
...

  public void startFc ( ) throws JadeException {
     ShellCommand.syncExec(dirInstall + "/bin/httpd -f " + dirLocal+ "/conf/httpd.conf");
  }

  public void stopFc( ) throws JadeException {
     BufferedReader br = new BufferedReader(new FileReader(dirLocal + "/logs/httpd.pid"));
     String pid = br.readLine();
     ShellCommand.asyncExec( "kill -TERM " + pid );
  }
}
```

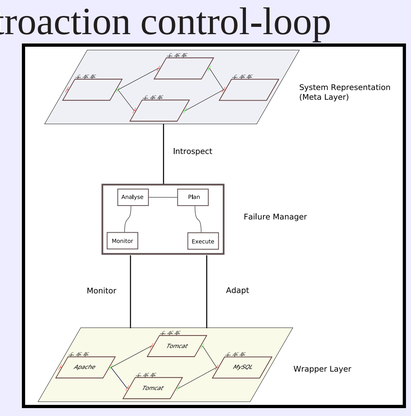
4 System Representation

- Provides a backup view of the systems architecture and configuration
- Principles
  - Isomorphic component structure
  - Is causally connected to the system
- ...
- ...
- ...



5 Self-Healing Control Loop

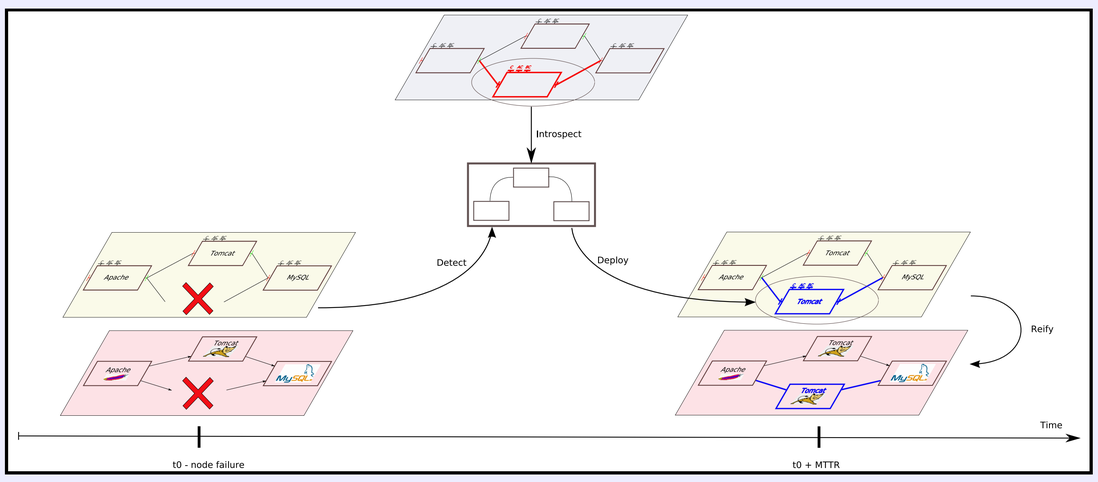
- Targets
  - Fail silent failure of nodes
  - Architectural Repair
- Autonomic Manager is built as a retroaction control-loop
  - **Monitor** : Ping
  - **Analyse** : Analyse failure and system architecture to insulate failure
  - **Plan** : Build a target architecture and a reconfiguration planification
  - **Execute** : Basic reconfiguration (binding and content)
  - **Knowledge** : System Representation



- Repair actions are triggered by node failure notifications.
- Repair algorithm steps
  1. **Analyze failure**
     - Identify Failed node
     - Identify components hosted by the failed node
  2. **Compute a target architecture**
     - Allocate a new node in the cluster
     - Build an equivalent architecture in a Repair Plan
  3. **Deploy the target architecture**
     - Compute a diff between Running system and Repair Plan
     - Patch (deploy) the diff on the running system

6 J2EE Repair Scenario

- J2EE architectures are challenging for self-management
  - Distributed
  - Heterogeneous
  - Very complex management (administrators need to be experts)
- Fail-Silent failure injected on Tomcat node



7 Conclusions

- Contributions
  - Architectural-Based Management
  - Legacy systems management
  - Uniform management interface
  - Architectural patterns
  - Reflexivity
  - Generic failure management
- Future work
  - Other applications (JORAM/JMS usecase)
  - Other environments (Grid, Peer-to-peer, etc)
  - Fiabilisation of singles points of failure (Meta layer and Failure Manager)

[1] C. Amza, E. Cecchet, A. Chanda, A. Cox, S. Elnikety, R. Gil, J. Marguerite, K. Rajamani and W. Zwaenepoel. Specification and Implementation of Dynamic Web Site Benchmarks In *IEEE 5th Annual Workshop on Workload Characterization (WWC-5)*, Austin, TX, Nov. 2002.

[2] J. O. Kephart and D. M. Chess. The Vision of Autonomic Computing In *IEEE Computer Magazine*, Volume 36, Number 1, 2003.

[3] E. Bruneton, T. Coupaye and J. B. Stefani. Recursive and Dynamic Software Composition with Sharing In *International Workshop on Component-Oriented Programming (WCOP-02)*, Malaga, Spain, June 2002, http://fractal.objectweb.org/.

[4] S. Bouchenak, F. Boyer, D. Hagimont, S. Krakowiak, A. Mos, N. de Palma, V. Quma and J. B. Stefani. Architecture-Based Autonomous Repair Management: An Application to J2EE Clusters In *24th IEEE Symposium on Reliable Distributed Systems (SRDS-2005)*, Orlando, FL, Oct. 2005.