

GoTM

Feedback from Fractal users

Fractal Workshop

ObjectWeb Architecture Meeting

March 25th 2004, Jussieu - France

Romain Rouvoy - Philippe Merle

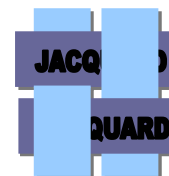
Projet INRIA Jacquard

LIFL / USTL

{rouvoy, merle}@lifl.fr



www.objectweb.org





Outline

→ **GoTM Introduction**

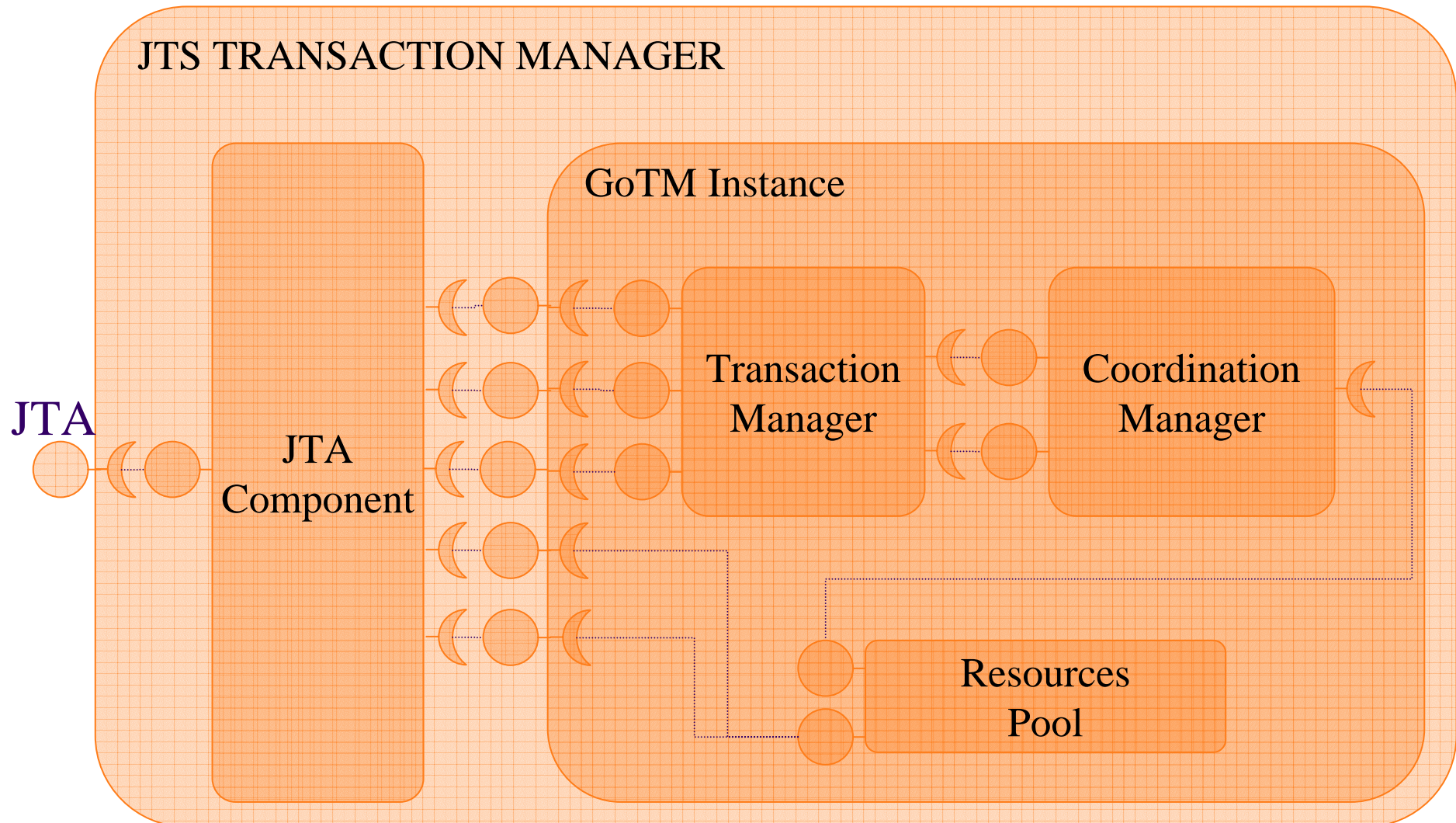
→ **Controller Mixin**

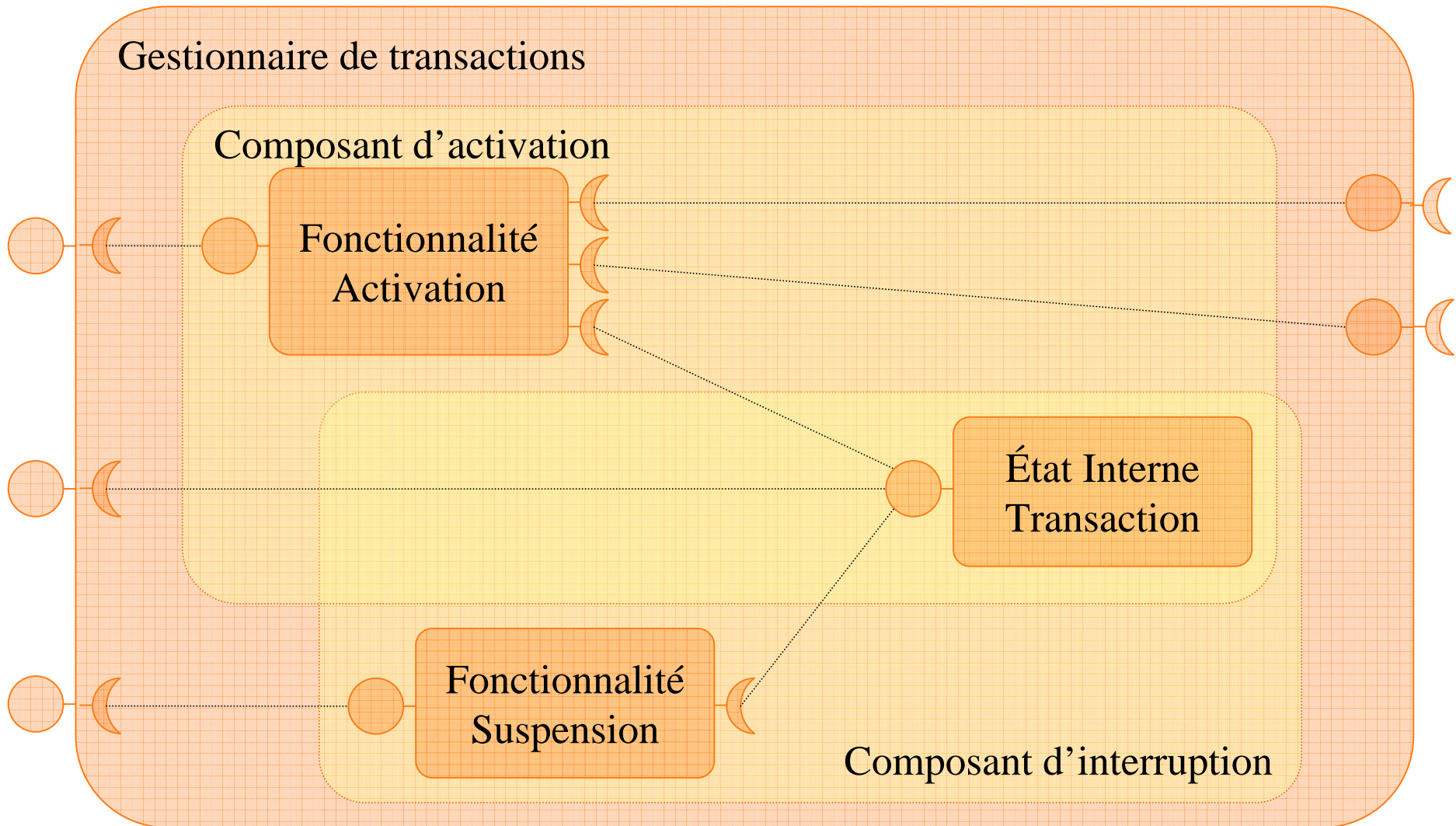
→ **Fractal Monolog**

→ **Fractal ADL v2**

→ **Fractal RMI**

→ **Fractal Browser**





→ Objective

- Binding of components and template components
- Parametric factories

→ Implementation

- Defining an additional TemplateController mixin
- First return the template component itself
- Specified as a particular template controller

→ Quite easy when you are familiar !

➔ Automatic binding of Monolog framework

- Log component bound to GoTM component
- Available log: Default Logger, LoggerFactory

➔ It's easy and transparent !

- Just implements `ow.monolog.Loggable` interface
- Logger prefix defined by component name

- **Fractal-oriented definition of the OW Transaction API**
- **Using multiple inheritance of component type**
 - `<definition name="A" extends="B,C" />`
- **Using template-controller tag for describing parametric factories**
- **Simplified/Leveraged XML syntax**

→ **Implicit Distributed Personality**

- Based on OW Transaction API

→ **Remote administration**

- Remote and dynamic administration of GoTM
- Browser on a different host

→ **Distributed Infrastructure**

- A transaction manager composed of distributed components
- Optimization/Replication solution

→ Fractal controllers as Fractal components

- Configuration of controllers through the ADL
 - Lightweight Fractal components
 - JOTDF and Jironde frameworks

→ Fractal factories

- ADL description tagged as factories
 - Propagation of the tag to sub components

→ Propagation of controller interfaces

- Attribute controller



Current Status

→ GoTM is an open Transaction Monitor

- Free software (LGPL)
- <http://gotm.objectweb.org/>



→ Library of Transactional Components (Fractal 2.0)

- Generic components (transaction, coordination, resource, ...)
- Personality components (JTA, ...)
- Integration components (JOTDF, Jironde, ...)

→ Tools for

- Designing (Fractal GUI)
- Assembling (Fractal ADL)
- Administrating (Fractal Browser)
- Distributing (Fractal RMI)

→ Powerful component model

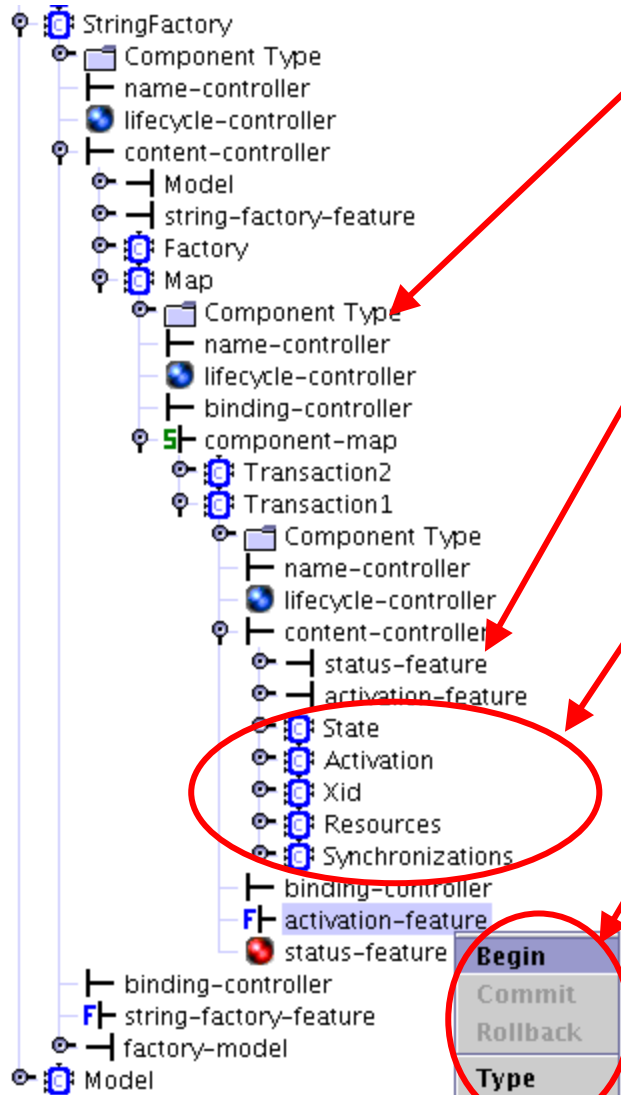
- Really extensible !
- Application beneficiate of the evolution of the component model
- A solution for integration of systems
 - ... <==> Browser <==> GoTM <==> Monolog <==> ...

→ Need more and more documentation

- Technologies inside (mixins, ...)
- Component methodology

→ More and more dynamic !

- Controllers
- Components



→ **Fractal Component Model**

→ **OW Transaction API**

→ **GoTM components composition**

→ **Managed by the Browser framework (with the basic Fractal plugin)**