# Coordination with Uncoupled Objects
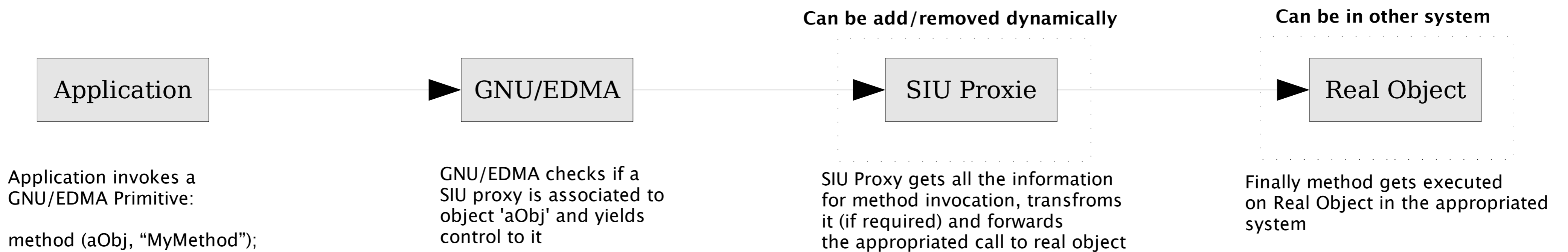## The GNU/EDMA Approach

David Martinez Oliveira, Camino Fernandez, Juan M. Dodero
University Carlos III Madrid. Department of Computer Science
Contact: {dmartin,camino,dodero}@dei.inf.uc3m.es

## GNU/EDMA Features

- **Uncoupled Component and Object Model**
- **Dynamic OO features at object & component level**
  - **Class/Object Oriented**
  - **Multiple Dynamic Inheritance (Delegation/Consultation)**
  - **Object Cloning and Shared Parents (as in Self)**
  - **Super or INNER inheritance (as in Java/Beta)**
  - **Dynamic Object Specialization (Dynamic Subclassing)**
- **Full System Reflection (Extended Meta-Object Protocol)**

- **Run-Time System Update and Evolution with Version Control**
- **Built-in Extension Systems (SIU Proxies)**
  - **Dynamic AOP infrastructures**
  - **System Integration: Java, .Net, Perl, Python,...**
  - **Solution Integration: Concurrent, Distributed, Mobile,...**

## SIU Proxies and Generic Method Interception

- SIU Proxies are normal GNU/EDMA Classes
- Allow to provide new implementation to system primitives
- Can be dynamically associated to objects at run-time or at load-time
- Designed as a general wrapper solution for system integration

**Can be add/removed dynamically**     **Can be in other system**

Application → GNU/EDMA → SIU Proxie → Real Object

Application invokes a
GNU/EDMA Primitive:

method (aObj, "MyMethod");

GNU/EDMA checks if a
SIU proxy is associated to
object 'aObj' and yields
control to it

SIU Proxy gets all the information
for method invocation, transforms
it (if required) and forwards
the appropriated call to real object

Finally method gets executed
on Real Object in the appropriated
system

## A threefold Coordination Scenario

- Study Case: Simple Local Object is dynamically transformed to other execution environments:
  - Transform Local Object to Shared Object (concurrent exectuion environment)
  - Transform Local Object to Remote Object (distributed execution environment
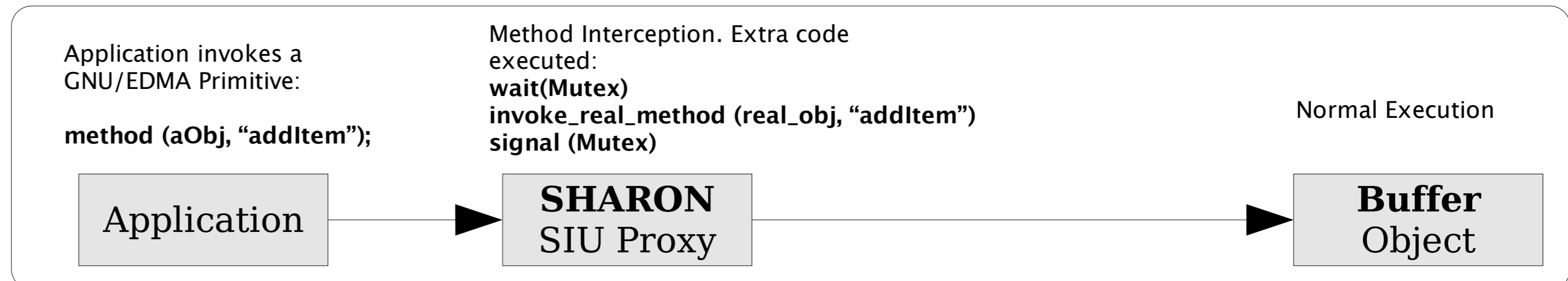  - Transform Local Object to Mobile Object (Mobile Computing execution environment)

**Buffer**
+ addItem
+ delItem
+ getItem

### Concurrent Components

Application creates a Buffer Object assigning
a Shared Object SIU Proxy:

**aObj = new_obj ("SHARON:Buffer");**

A SHARON SIU Proxy is automatically attached to the
new Buffer created instance

Application invokes a
GNU/EDMA Primitive:

**method (aObj, "addItem");**

Method Interception. Extra code
executed:
**wait(Mutex)**
**invoke_real_method (real_obj, "addItem")**
**signal (Mutex)**

Normal Execution

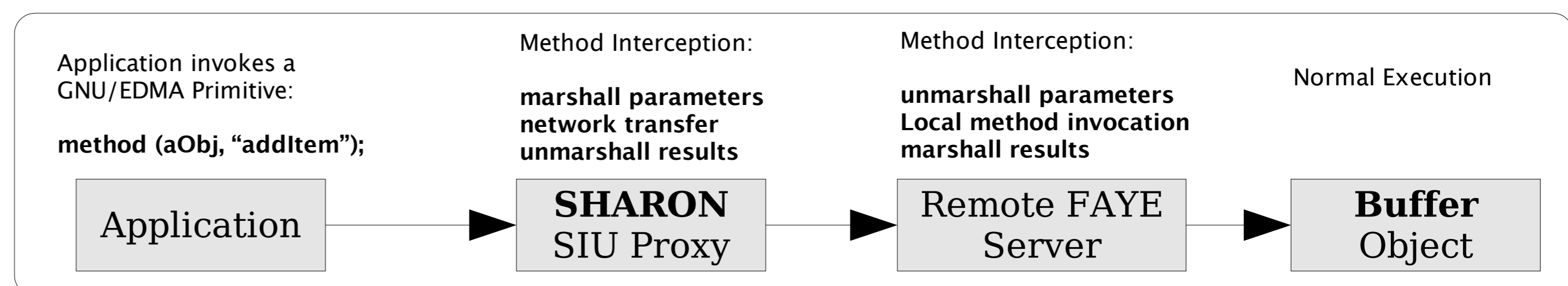Application → **SHARON** SIU Proxy → **Buffer** Object

### Distributed Components

Application creates a Buffer Object assigning
a Shared Object SIU Proxy:

**aObj = new_obj ("FAYE:(host)Buffer");**

A FAYE SIU Proxy is automatically attached to the
new Buffer created instance

Application invokes a
GNU/EDMA Primitive:

**method (aObj, "addItem");**

Method Interception:

**marshall parameters**
**network transfer**
**unmarshall results**

Method Interception:

**unmarshall parameters**
**Local method invocation**
**marshall results**

Normal Execution

Application → **SHARON** SIU Proxy → Remote FAYE Server → **Buffer** Object

### Mobile Computation

Application creates and specialises a Buffer Object to make it mobile

```
aObj = new_obj ("Buffer");
method (aObj, "AGNES_AGENT>setDefaultHome");
method (aObj, "AGNES_ITINERARY>addHost", "host1");
method (aObj, "AGNES_ITINERARY>addHost", "host2");
method (aObj, "AGNES_MAIN_LOOP@INNER|SUPER>process");;
method (aObj, "next_host");
```

For Mobile computing, GNU/EDMA dynamic specialization features
are used. Object is dynamically extended with the required behaviors
to move within the AGNES Mobile Platform

**Buffer Obj**

**AGNES_AGENT**
+ setDefaultHome
+ next_host

**AGNES_ITINERARY**
+ addHost

**AGNES_MAIN_LOOP**
+ process