Making Ubiquitous Computing Reality

Diego Lz. de Ipiña Gz. de Artaza
Cátedra de Telefónica Móviles – Universidad de Deusto
dipina@eside.deusto.es
http://paginaspesonales.deusto.es/dipina
http://www.ipina-dorsman.org/trip

Most of the work presented was carried out with the support of:

Laboratory for Communications Engineering (LCE)
Cambridge University Engineering Department
England, UK

AT&T Laboratories
Cambridge

Basque Government
Education Department

Introduction

Goals:

- build **Sentient Spaces** = *computerised* environments that *sense* & *react*
- close gap between user and computer by using **context**
- make **ubiquitous computing** reality through **Sentient Computing**

Sentient Computing = **computers + sensors + rules**:

- distributed sensors capture context, e.g. temperature, identity, **location**, etc
- rules model how computers **react** to the stimuli provided by sensors
- 3 phases: (1) context capture, (2) context interpretation and (3) action triggering

To make viable widespread adoption of Sentient Computing, we propose:

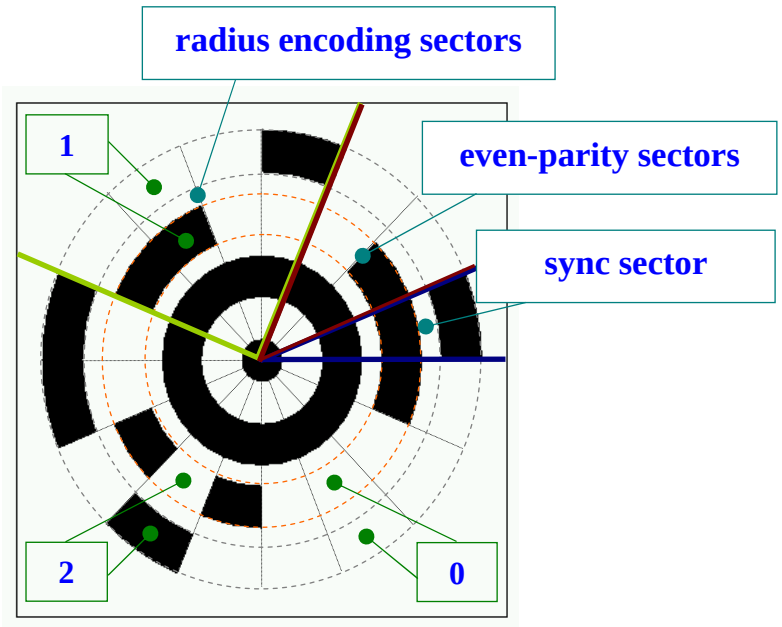
- location sensor deployable everywhere and for everyone
- middleware support for easier sentient application development:
  - rule-based monitoring of contextual events and associated reactions
  - user-bound service lifecycle control to assist in action triggering

TRIP: a Vision-based Location Sensor

"Develop an easily-deployable location sensor technology with minimum hardware requirements and a low price"

- TRIP (Target Recognition using Image Processing):
  - identifies and locates tagged objects in the field of view of a camera
- Requires:
  - off-the-shelf technology: **cameras+PC+printer**
  - specially designed 2-D circular markers
  - use of well-known **Image Processing** and **Computer Vision** algorithms
- Cheap, easily deployable → can **tag *everything*:**
  - e.g. people, computers, books, stapler, etc
- Provides accurate 3-D pose of objects within 3 cm and 2° error

TRIPcode 2-D Marker



* 10 2011 221210001

TRIPcode of radius 58mm and ID 18,795

- 2-D barcode with ternary code
- Easy to identify ***bull's-eye***:
  - invariant with respect to:
    - rotation
    - perspective
  - high contrast
- 2 16 bit **code encoding rings**:
  - 1 sector synchronisation
  - 2 for even parity checking
  - 4 for bull's-eye radius encoding
  - 3^9 = 19,683 valid codes

Target Recognition Process

Stage 0: Grab Frame

Stage 1: Binarization

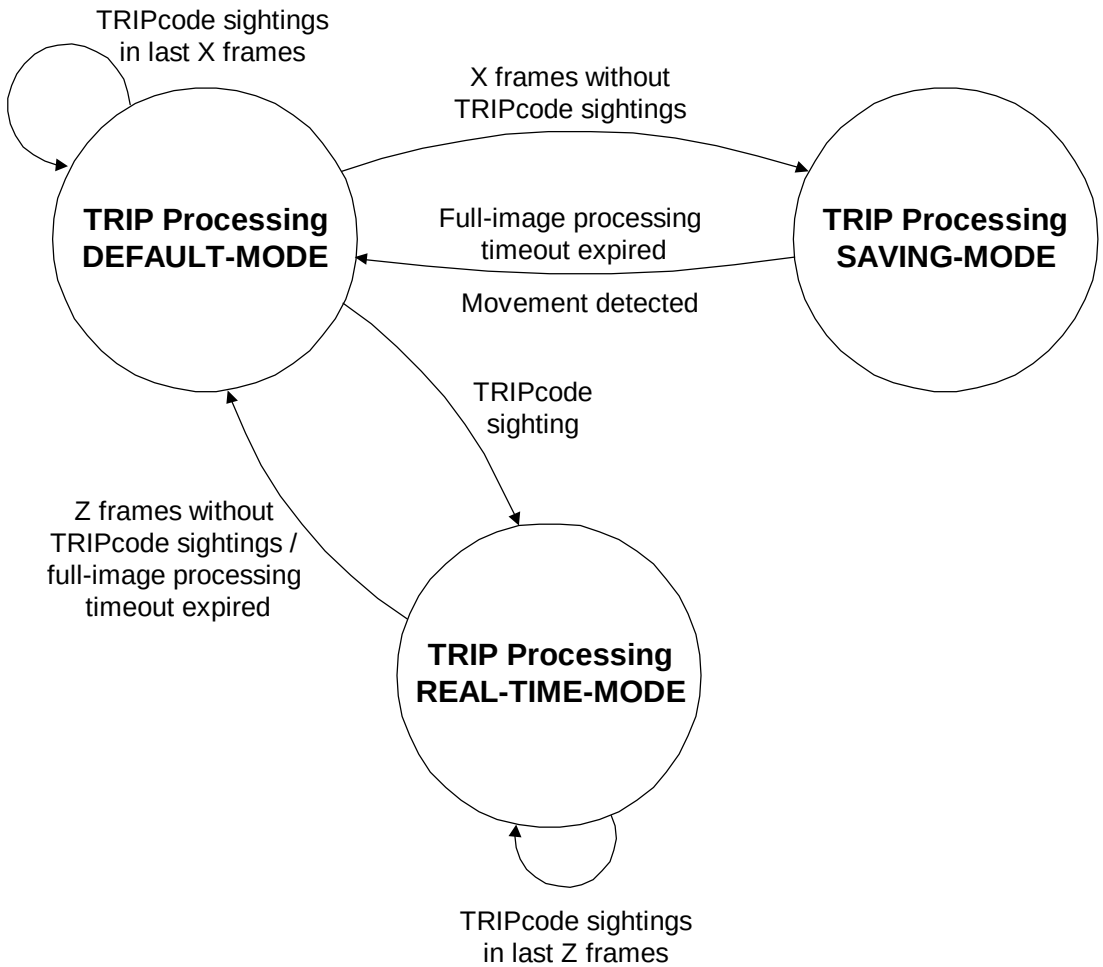
Stage 2: Binary Edge Detection

Stage 3: Edge Following & Filtering

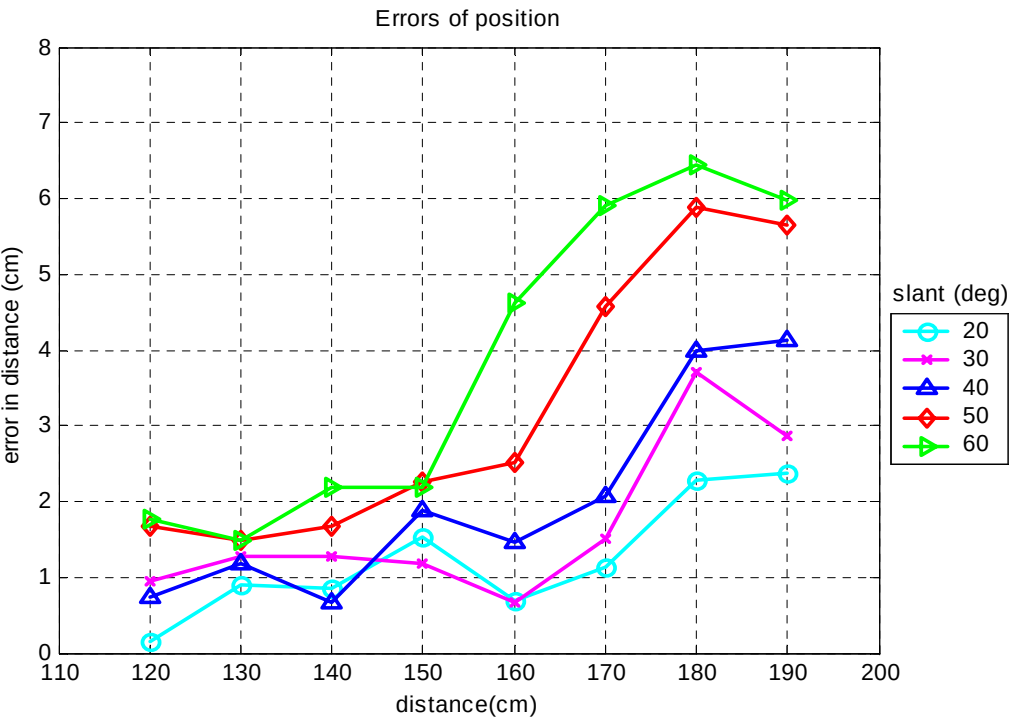
Ellipse params:
x (335.432), y (416.361) pixel coords
a (8.9977), b (7.47734) pixel coords
θ (15.91) degrees
Bull's-eye radius: 0120 (15 mm)
TRIPcode: 002200000 (1,944)
Translation Vector (meters):
(T_x=0.0329608, T_y=0.043217, T_z=3.06935)
Target Plane Orientation angles (degrees):
(α=-7.9175, β=-32.1995, γ=-8.45592)
d2Target: 3.06983 meters

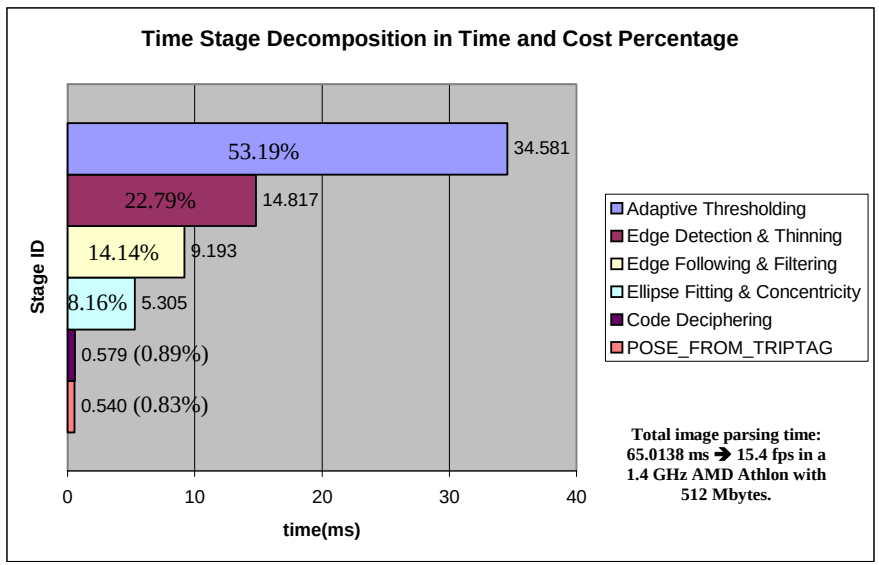
Stages 4-7: Ellipse Fitting, Ellipse Concentricity Test, Code Deciphering and POSE_FROM_TRIPTAG method

TRIP Sensor Adaptive Operation

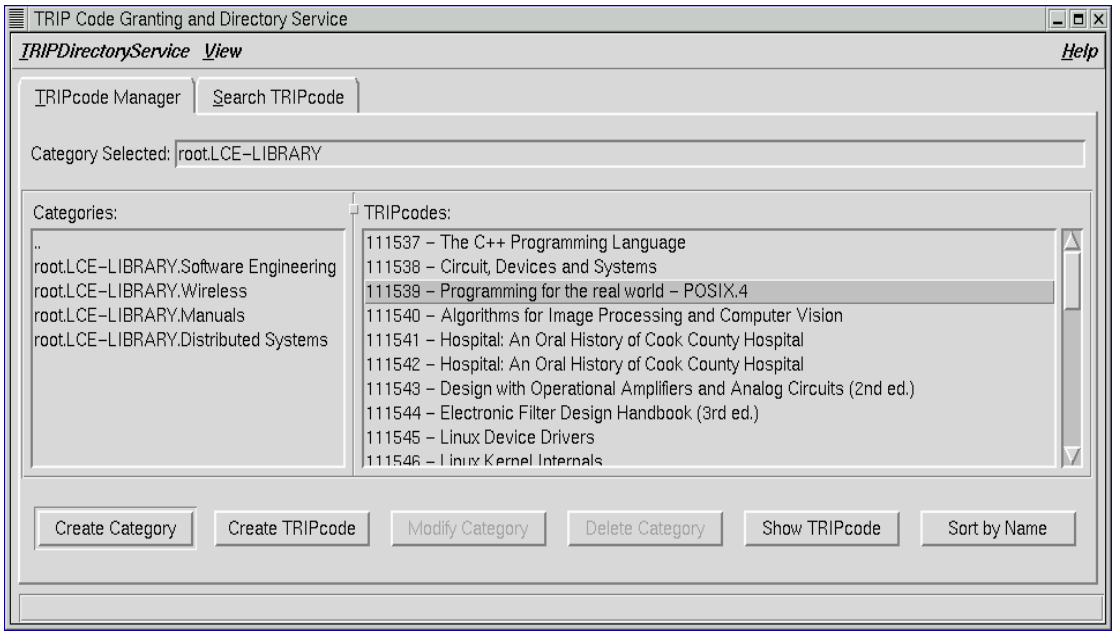


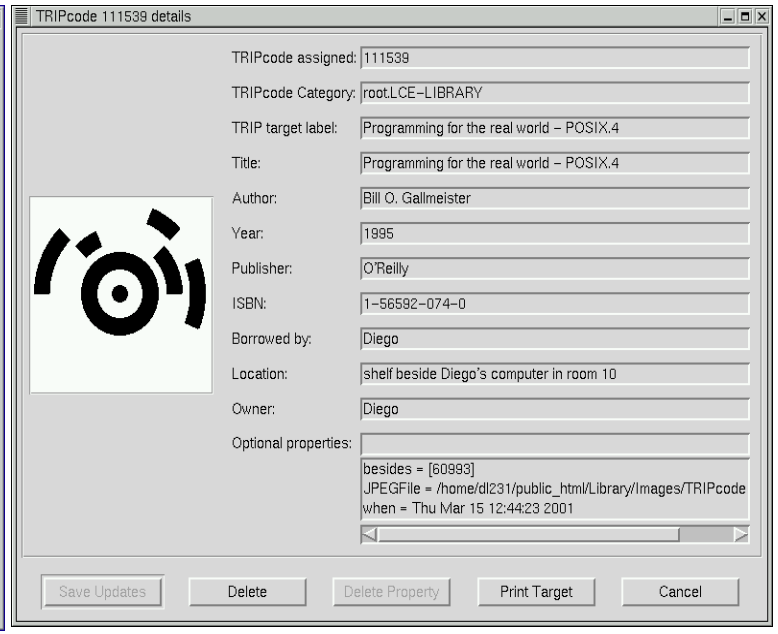
TRIP Performance and Accuracy Results





TRIP Directory Service



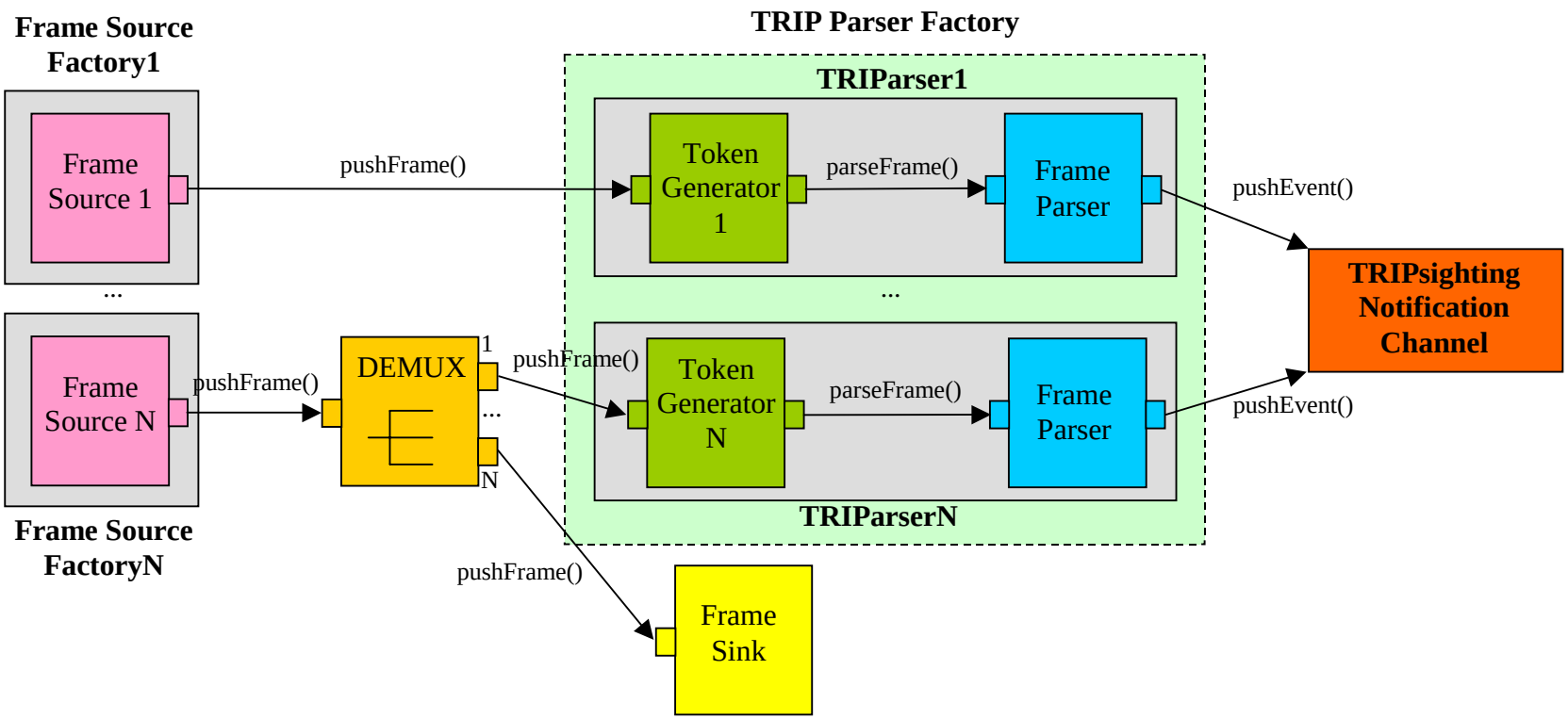


TRIP: a Distributed Sensor System

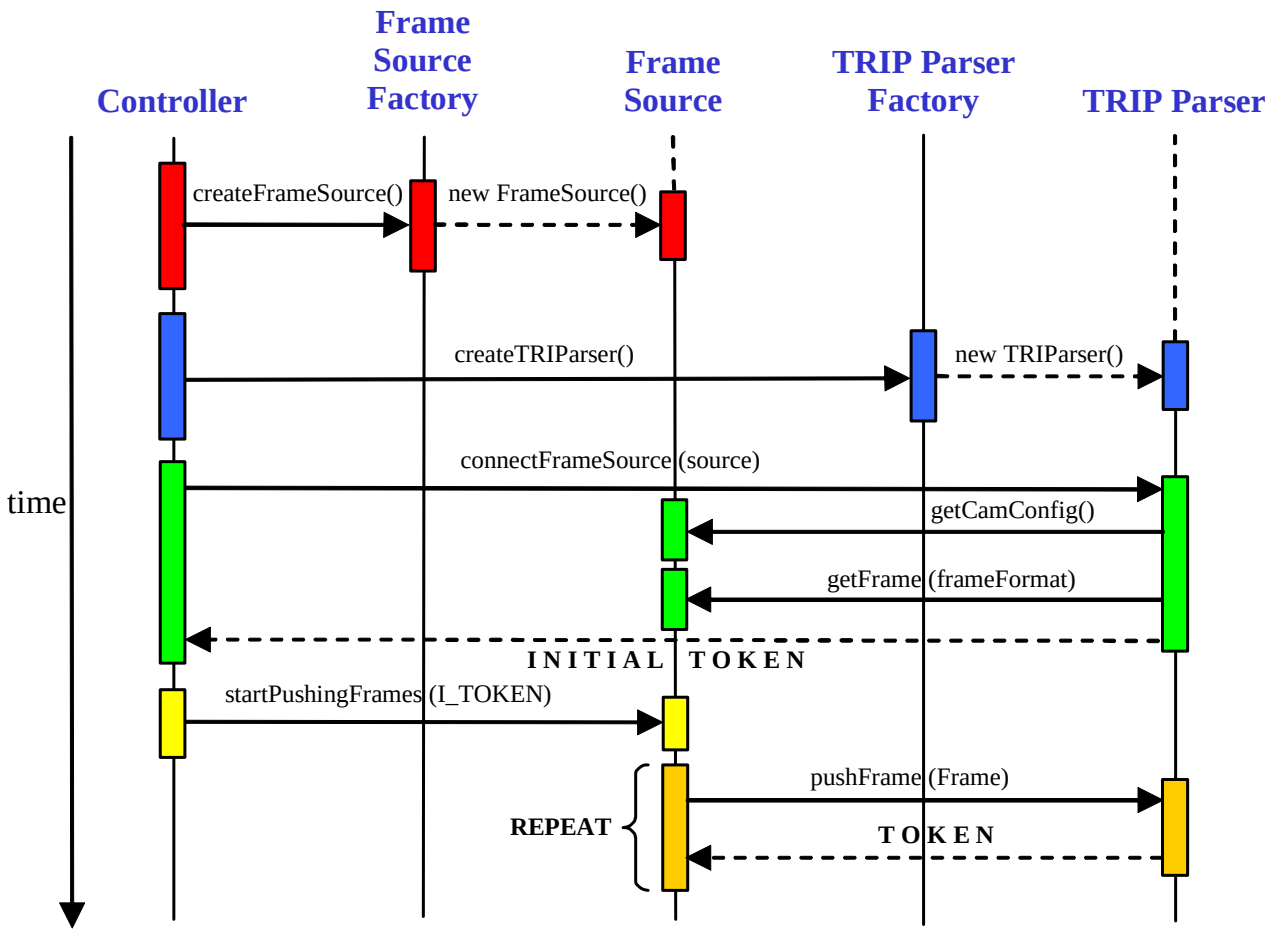
- TRIP C++ library and TRIP Directory Service
- Java package wrapping the TRIP C++ library
- CORBA-based **TRIParser** component:
  - accepts video frames from distributed frame sources
  - provides *synchronous* and *asynchronous* interfaces for video parsing
  - frame sources push frames using a token-based protocol for image flow control
  - pushes a **TRIPevent** per target sighting into a CORBA Notification Channel:

```
struct TRIPevent {
   double timestamp;
   unsigned long cameraID;
   string TRIPcode; // code ternary representation
   TargetPosition position; // (xpos, ypos, zpos) vector
   TargetOrientation orientation; // (α, β, γ) angles
};
```

A Token Based Flow Control Mechanism I



A Token Based Flow Control Mechanism II

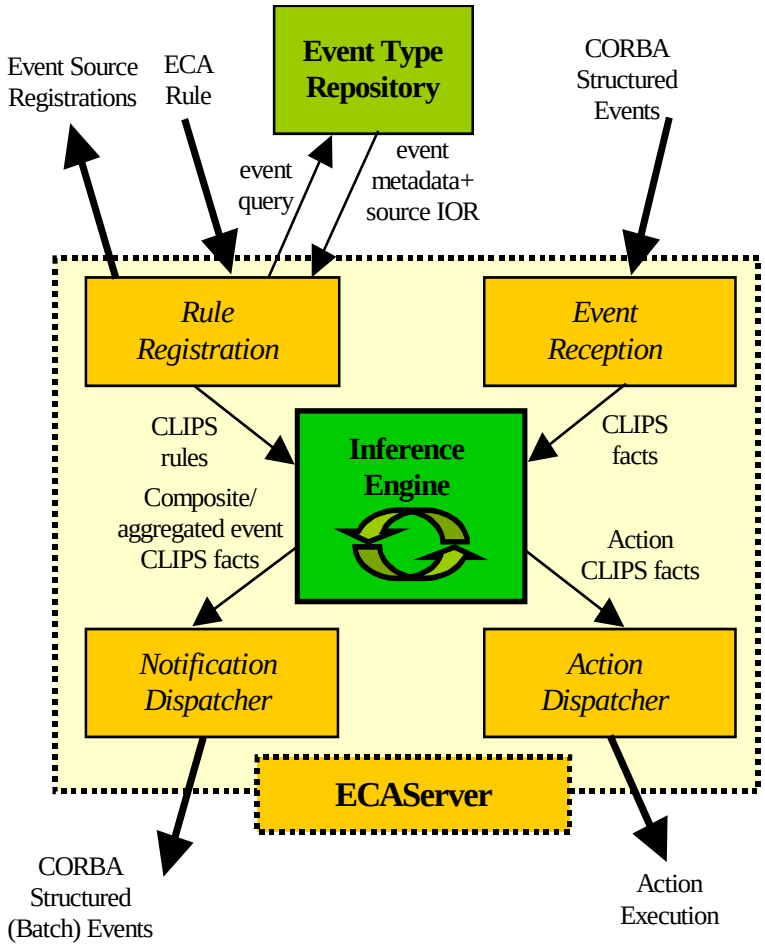


ECA Rule Matching Service

- Sentient Applications respond to an **Event-Condition-Action** (ECA) model:
  - monitor contextual events coming from diverse sources
  - correlate events to determine when a contextual **situation** occurs:
    - e.g. IF two or more people in meeting room + sound level high THEN meeting on
  - ineffective to force every app to handle same behaviour separately

- Solution → **ECA Rule Matching Service:**
  - accepts rules specified by the user in the ECA language
    - `<rule> ::= {<event-pattern-list> => <action-list> }`
  - automatically registers with the necessary event sources
  - notifies clients with **aggregated** or **composite** events or executes actions when rules fire:
    - **aggregated event** = new event summarizing a situation
    - **composite event** = batch of events corresponding to a situation

ECA Service Architecture



Building a Sentient Jukebox with ECA Service

"If it is Monday, a lab member is logged in and either he is working or it is raining outside, then play some cheerful music to raise the user's spirits"

```
within 15000 {/* Enforce events occur in 15 secs time span*/
  query PCMonitor$logged_in(user ?userID, host ?hostID) and
  test(dayofweek = "Monday") and
  Location$presence(user ?userID) before
  /* a presence event must occur before following events */
  ((PCMonitor$keyboard_activity(host ?hostID, intensity ?i) and
    test(?i > 0.3)) or
   (query WeatherMonitor$report(raining ?rainIntensity) and
    test(?rainIntensity > 0.2)))
=>
  notifyEvent(Jukebox$play_music(?userID, ?hostID, "ROCK"));
}
```

Mapping from ECA language to CLIPS

```
(assert (rule (ruleID 0) (ruleRegTime 1005472984621)))
(defrule rule0
  (PCMonitor$logged_in (user ?userID) (host ?hostID)
                       (timestamp ?time0#))
  (test (eq (dayofweek) "Monday"))
  (Location$presence (user ?userID) (timestamp ?time1#))
  (test (> ?time1# 1005472984621))
  (test (> ?time1# (- (curtime) 15000)))
  (or (and (and (PCMonitor$keyboard_activity (host ?hostID)
                        (intensity ?i) (timestamp ?time2#))
                (test (> ?time2# 1005472984621))
                (test (> ?time2# (- (curtime) 15000)))
                (test (> ?time2# ?time1#)))
           (test (> ?i 0.3)))
      (and (WeatherMonitor$report (raining ?rainIntensity)
                                  (timestamp ?time3#))
           (test (> ?rainIntensity 0.2))))
=>
  (bind ?currentTime# (curtime))
  (bind ?factID0# (assert (Jukebox$play_music# 0 ?currentTime#
                                          ?userID ?hostID "ROCK")))
  (notify-event ?factID0#))
```

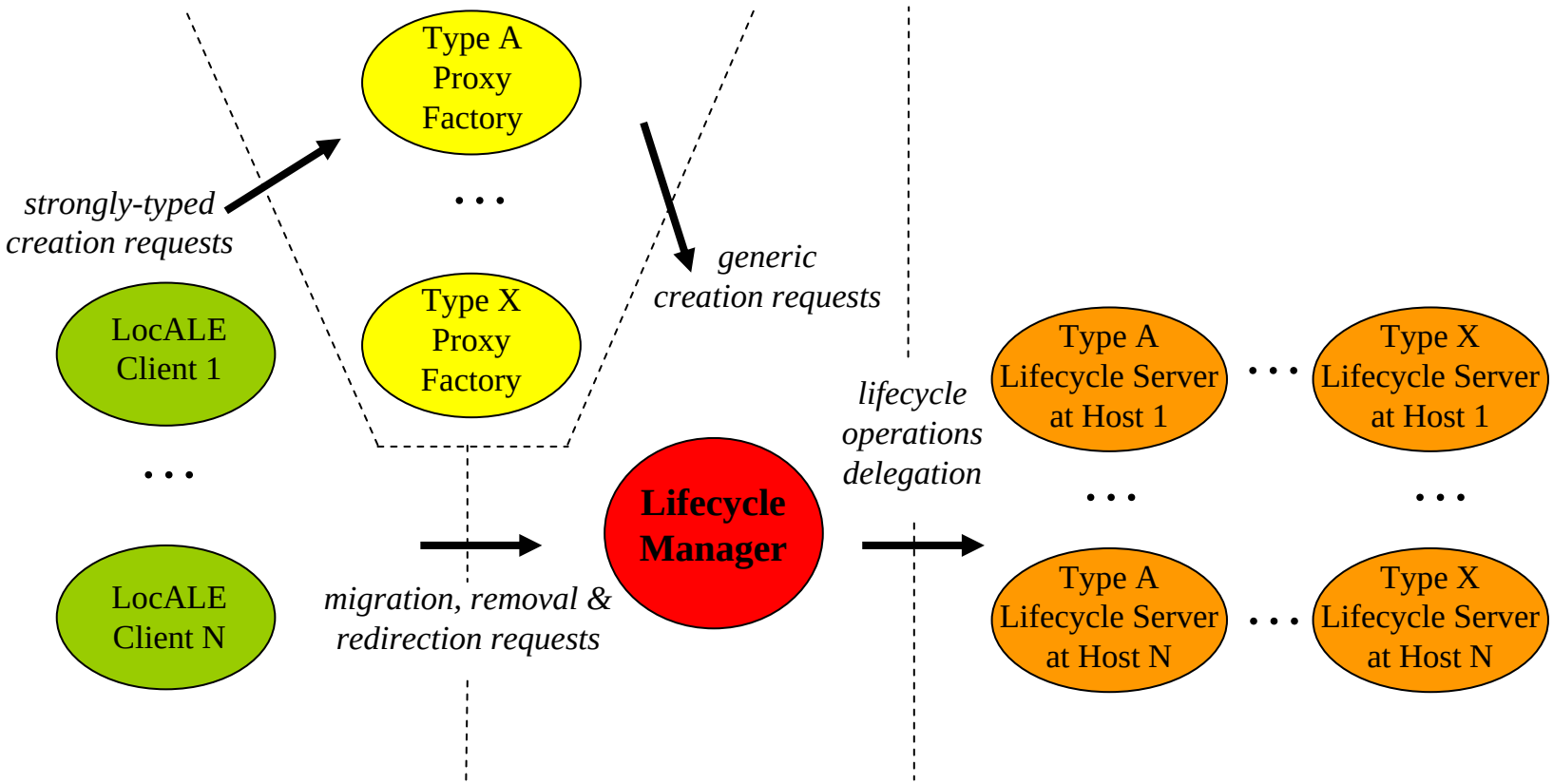
LocALE Framework

- Need to provide support for reactive behaviour of sentient systems:
  - e.g. user-bound service activation after aggregated event arrival
- LocALE = CORBA-based solution to **object lifecycle & location control:**
  - hybrid of CORBA's Object LifeCycle Service and Implementation Repository
  - addresses *location-constrained* service activation, deactivation and migration
  - adds mobility, fault-tolerance and load-balancing to objects in a **location domain**
  - generates permanent object references (independent of object network location)
  - undertakes transparent client request redirection upon object's location change
  - useful for third-party object location controllers:
    - e.g. "migrate the TRIP parser to another host when the owner of used host logs in"

Location constrained Object Lifecycle Control

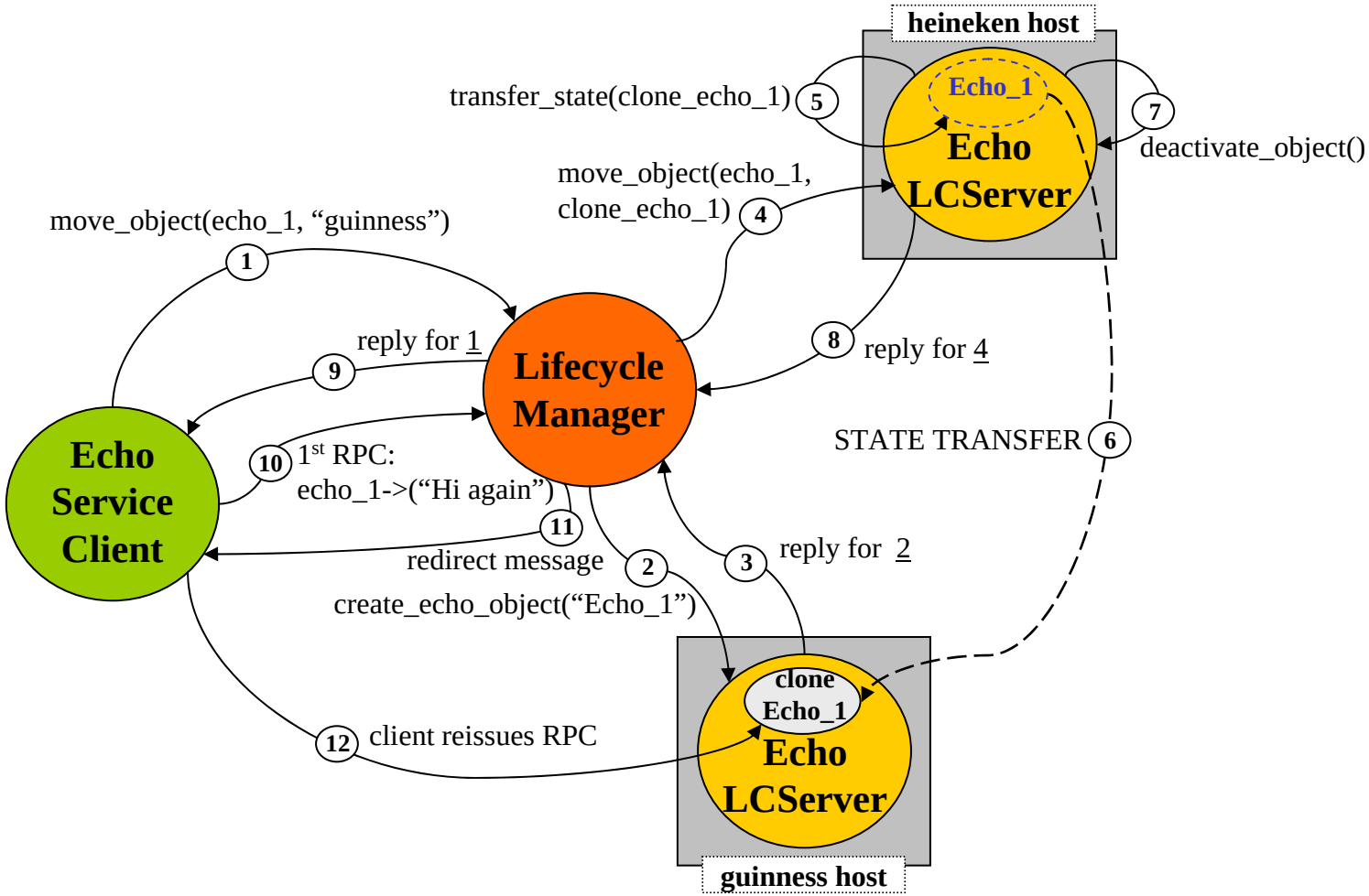
- Why is CORBA location transparency not always desirable?
  - sometimes want to control **where objects are first located and then relocated**
    - e.g. load-balancing or follow-me applications
- LocALE provides apps with location-constrained object **lifecycle-control:**
  - apps specify on distributed object creation their **initial location**:
    - within a host, e.g. `hostDN("guinness")`
    - any host in an spatial container (room), e.g. `roomID("Room_1")`
    - in any location domain's host, e.g. `hostDN("ANY")` or
    - in one of a given set of hosts, e.g. `hostGroup("heineken", "guinness")`
  - ... and **restrictions** under which an object can later be moved and/or recovered:
    - `LC_CONSTRAINT(RECOVERABLE | MOVABLE)` → any host of location domain
    - `LC_CONSTRAINT(RECOVERABLE_WITHIN_ROOM | MOVABLE_WITHIN_ROOM)`

LocALE Architecture



LocALE 3½-tier architecture

LocALE Lifecycle Control Flow



LCE Sentient Library I

- Augments a conventional library system with context-awareness
  - Not only can we know where a book is, but also what other books are next to it and images of them
  - It works best for locations where not all books lie in the same location but they can be scattered through a building, e.g. research lab
- Enables the automatic cataloguing process of books in a library
  - Through video processing the library´s database is updated, without user intervention

LCE Sentient Library II

Bookmarks Location: http://www-lce.eng.cam.ac.uk/Library/ What's Related

LCE Sentient Library Catalog Functions:

- Browse LCE Library Catalog
- Search for book
- Create new book category
- Enter new book details
- Edit book details and/or Print TRIPtag
- Enter new shelf details
- Edit shelf details and/or Print TRIPtag

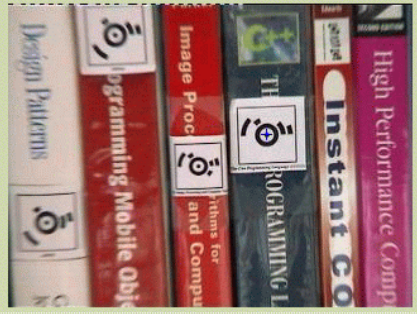
Created by Diego López de Ipiña (dl231@eng.cam.ac.uk)

Bookmarks Location: http://www-lce.eng.cam.ac.uk/Library/cgi-bin/showBookDetails.py What's Relate

Book ('111537') belonging to category root.LCE-LIBRARY details:

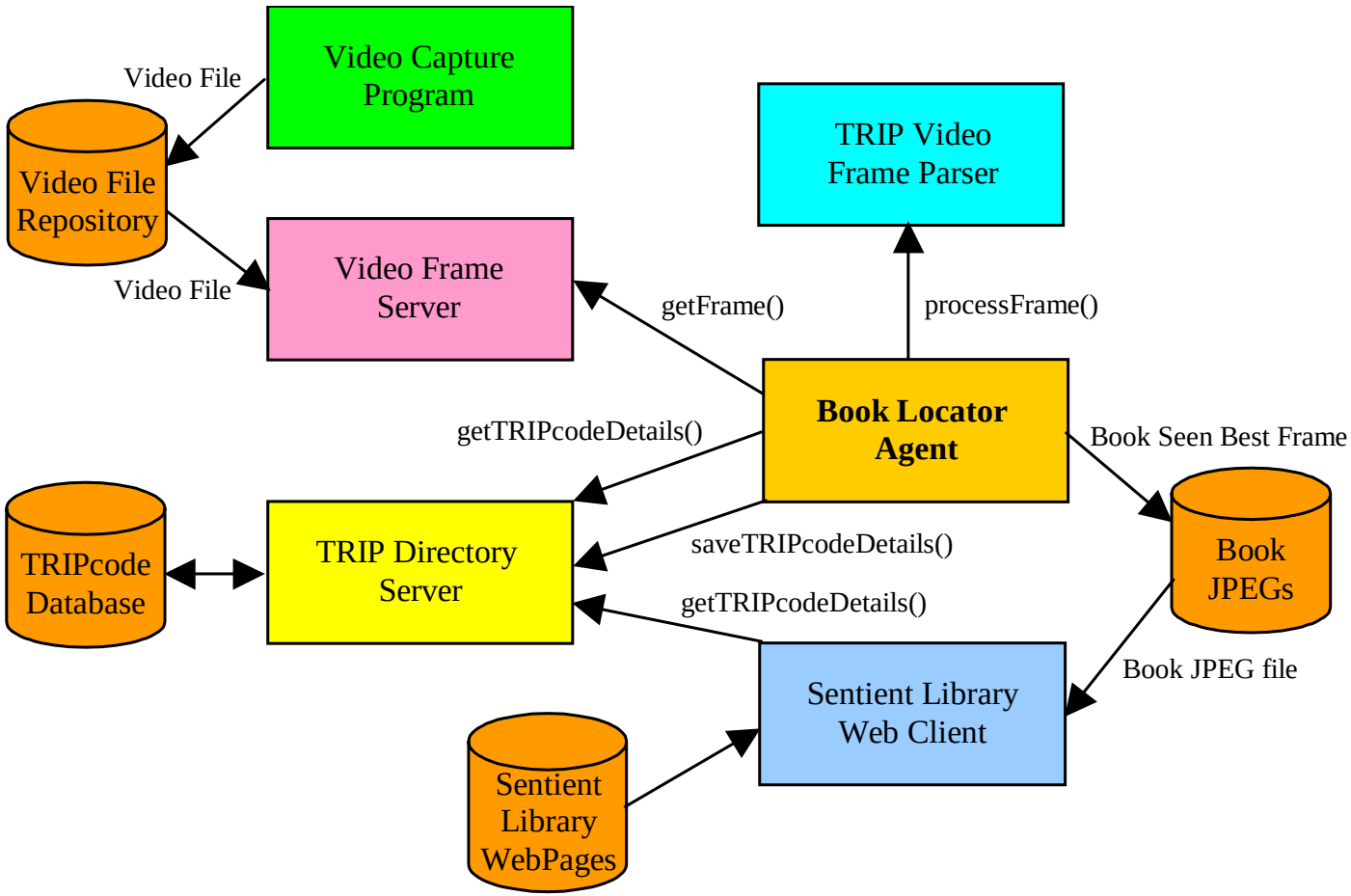
Title:	The C++ Programming Language
Author:	Bjarne Stroustrup
Year:	1997
Publisher:	Addison Wesley
ISBN:	0-201-88954-4
Borrowed by:	Diego
Owner:	Diego

Book ('111537') LOCATION details:



The book was last seen at shelf beside Diego's computer in room 10 beside books: [Programming Mobile Objects with Java, Design Patterns – Elements of Reusable Object-Oriented Software] on Wed Nov 7 11:17:39 2001

LCE Sentient Library III



Bookmarks Go To: http://www-lce.eng.cam.ac.uk/Library/cgi-bin/TRIPboardMonitorCtrler.py What'

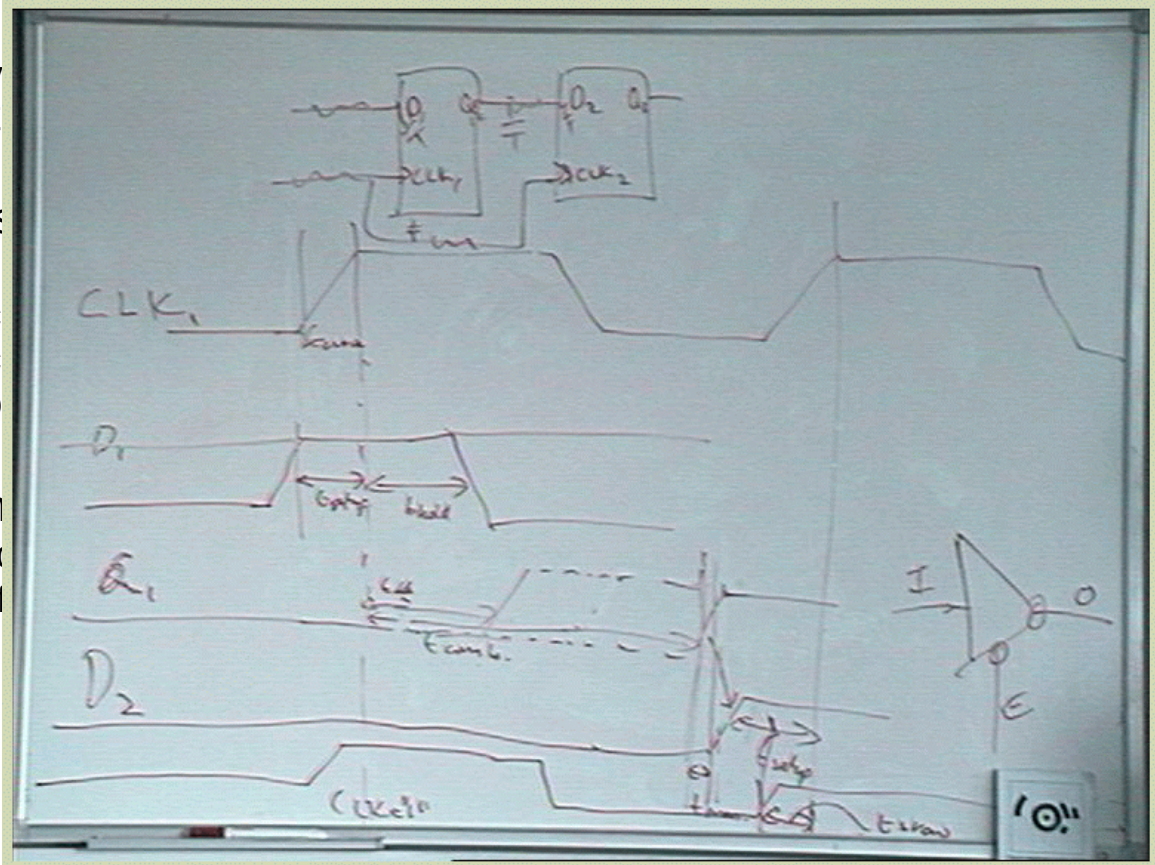
The LCE Meeting Room's Active TRIPboard

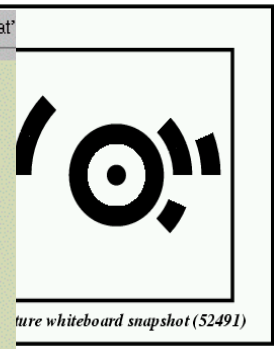
The LCE Meeting Room's Active TRIPboard Monitor current status is: **ACTIVE**.
Click on the buttons below to activate/deactivate the service.

The TRIP frame parser is currently running at: **tetleys.eng.cam.ac.uk** host.

Activate Monitor | Deactivate Monitor

The last snapshot taken (Jan 21 08:53) of the LCE Meeting Room's whiteboard was:





LCE

- Augm ng special *ringc*
- Activ iteboard is locat
- Regis

```
Loc                                    mCam") and
Loc                                    ) =>
  no                                   erID))
```

- By m
  - (
  - f

```
                                       stGroup
```

Follow Me Jukebox

- Provides mobile users with music from the nearest set of speakers
- MP3 decoder and player follow the user to his new location.
- Uses TRIP as a real-time location and music selection device
- Uses ECA Service to register contextual situations to be monitored
- Uses LocALE's migration support

TRIP enabled Teleporting I

- Monitors when a TRIPtag-wearing user gets closer than one metre to a web-cam placed on top of a computer, not currently used.
  - The service automatically displays the desktop associated with that user through Virtual Network Computing (VNC)
  - When the user later moves in front of another terminal, the VNC desktop is automatically deactivated from the previous terminal and activated on the new one.

TRIP enabled Teleporting II

TRIP enabled Teleporting III

```
/* RULE: notify when a user is spotted within 1 metre distance
   of camera 0 and nobody is typing at host "cruzcampo" where
   the camera is attached. */
{
 Location$TRIPevent(cameraID 0, TRIPcode ?userID,
                    d2Target ?distance) and
 test("002" ~ ?userID) and
 test(?distance < 1.0) and
 ((not query PCMonitor$keyboard_activity(host "cruzcampo",
                        intensity ?i, timestamp ?time0)) or
  (query PCMonitor$keyboard_activity(host "cruzcampo",
                    intensity ?i, timestamp ?time0) and
   test (?i < 1.0) and
   test ((curtime - ?time0) > 180000))) /* > 3 min */
=>
 notifyEvent(Sentient$Teleport(?userID, "cruzcampo"));
}
```

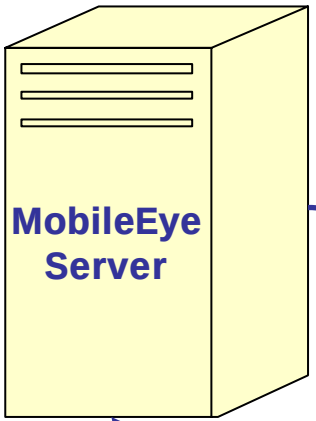
MobileEye

- Application presented by the University of Deusto CTME (Cátedra de Telefónica Móviles) at MovilForum 2003
- Aims to show how mobile communications and services can be enriched by adding context awareness to them
- As mobile phones (camera phones) are always with us, they can act as an eye through which we can obtain an enriched view of the world
- Thanks to their increasing computational and sensorial capabilities, mobiles can sense and react to the environment stimuli (images, bluetooth broadcasts) and enable us simpler and more natural interactions with the object that surround us
  - We could buy a coke through the mobile, switch on the lights in our house, open the front-door, receive information about a painting in a museum, etc.
- This application lies within a bigger ongoing project at the CTME named EMI^2 (Environment to Mobile Intelligent Interaction)

MobileEye Concept

Fifteen Sunflowers by Van Gogh (512311)

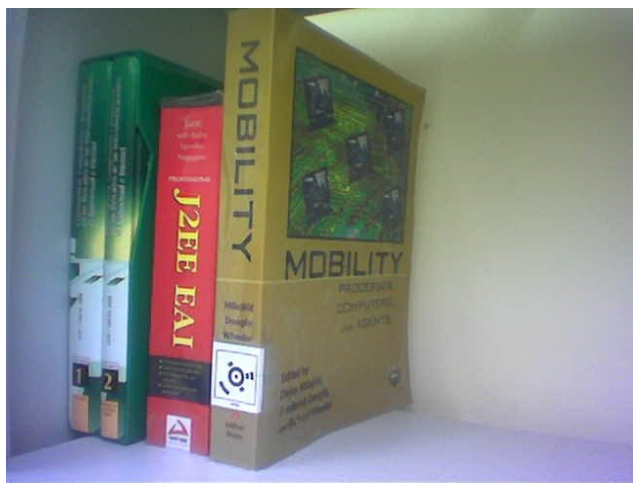
MobileEye Implementation



Parsed image response in XML

```
<?xml version="1.0"?>
<mobileeye>
  <state code="200"/>
  <code>61002</code>
  <descShort>Mobility</descShort>
  <descLong>Book about Mobile Agents</descLong>
  <url>http://www.deusto.es/library?book=123</url>
</mobileeye>
```

captured image



Reference

- For more details about the work presented:
  - Check my PhD dissertation: “Visual Sensing and Middleware Support for Sentient Computing”
  - The dissertation PDF and source code of the TRIP and MobileEye systems are available at:

    http://www.ipina-dorsman.org/trip

Conclusions

- Assortment of technologies to make Sentient Computing available to everyone:
  - TRIP 3-D location distributed sensor
  - rule-based programming paradigm for sentient applications
  - LocALE object lifecycle- and location-control middleware
  - sentient applications developed as ‘proof of concept’

Diego Lz. de Ipiña Gz. de Artaza
dipina@eside.deusto.es
http://paginaspesonales.deusto.es/dipina
http://www.ipina-dorsman.org/trip