



A new approach toward a modular multimodal interface  
for PDAs and smartphones

**Funzionalità e Architettura HW/SW**

Giovanni Frattini

*Salerno, September 11° 2008*

- » Giovanni Frattini, Federico Ceccarini, Fabio Corvino, Ivano De Furio, Francesco Gaudino, Pierpaolo Petriccione, Roberto Russo, Vladimiro Scotto di Carlo, Gianluca Supino
- » **ENGINEERING.IT S.p.A. ,Via Antiniana 2/A, 80078, Pozzuoli (NA), Italy**
- » All the authors are working to a co-funded project called CHAT at the time being
- » **Several experiences in R&D projects: from a multichannel platform called SemaPortal (at that time the Naples office was SEMA, to new approach based on Web Services (ServiceWare), to multimodality**

## CHAT project: highlight ...

**Cultural Heritage** fruition & e-learning applications of new **Advanced** (multimodal) **Technologies**

CHAT has been co-funded by Italian Ministry of Research

### Main goals

- A Platform to create Mobile multimodal services
- Supporting as much as possible mobile clients
- Synergic multimodality (multiple inputs concurring in defining the user will)
- Supporting a Speech driven interaction
- Supporting many interaction mode at the same time

### Partners

- Engineering.it
- University of Bari
- University of Salerno
- IRPSS (CNR Rome)

# Mobile Clients for multimodal applications

- » Rationale: millions of new smartphones and PDAs on the market: relatively few opportunities for exploiting their capabilities.
- » Main ideas underneath our research:
  - » A mobile client supporting synergic multimodality (simultaneous input signal acquisition)
    - Thin client supporting multimodal I/O capabilities
    - A thin client for streaming continuously signals over the net (upstream and downstream)
  - » Telecommunication protocols for locating (signaling) phones (notably SIP) and streaming contents (RTP)
  - » Web protocols (HTTP) for integrating signaling capabilities and content retrieval

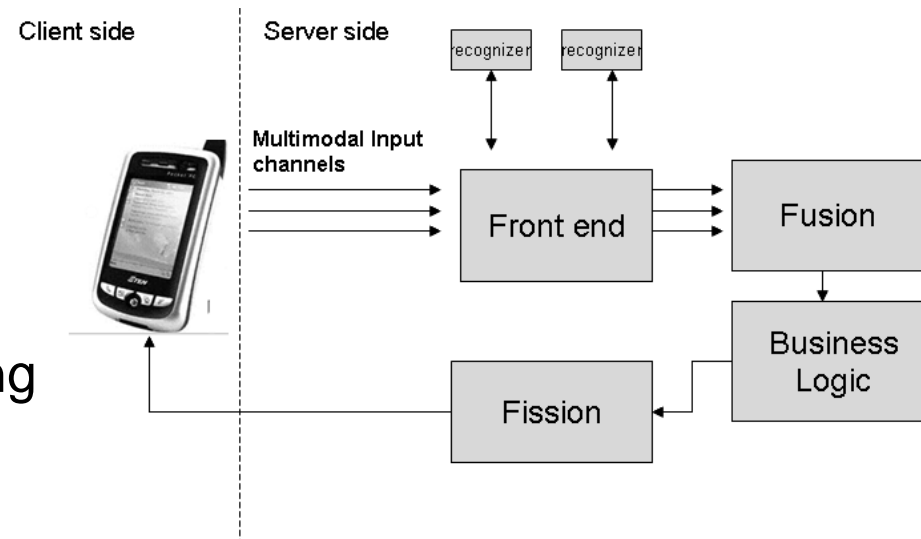
# Platform Architecture: highlight...

## Client side

- MMUIManager (MultiMedia User Interface Manager)

## Server side

- Front end: to collect the input coming from the clients and send them to specific recognizers (speech, sketch, handwriting recognizers);
- Fusion module: to merge inputs coming from different channels;
- Business logic module: to understand the user will and select the contents to send back to the client;
- Fission module: to 'push' content to final users.



# MMUIManager: dynamic composition of multimodal objects

- » We have considered as starting point a well known framework: **Piccolo**, University of Maryland.
  - **What is Piccolo:** “Piccolo is a toolkit that supports the development of 2D structured graphics programs”. In other words a Piccolo application is a graph of graphic objects.
  
- » **IDEA:** a mobile multimodal interface can be a graph of multimodal objects
  
- » The MMUIManager is able to dynamically compose “multimodal objects”
  - A language called LIDIM tell the client engine how to aggregate objects
  - At the moment objects are shown in a pre-configured template (e.g. the screen separated in 4 sub-frames)

**Multimodal Objects**

**LIDIM Engine**

**Native Platform**

**Thin client approach: light application on the client side, heavy weight processes on the server side. The client is able to send and receive contents streaming them from the server.**

**MMUIManager (MultiModal User Interface): a composition of multimodal objects (MMO). Main categories implemented**

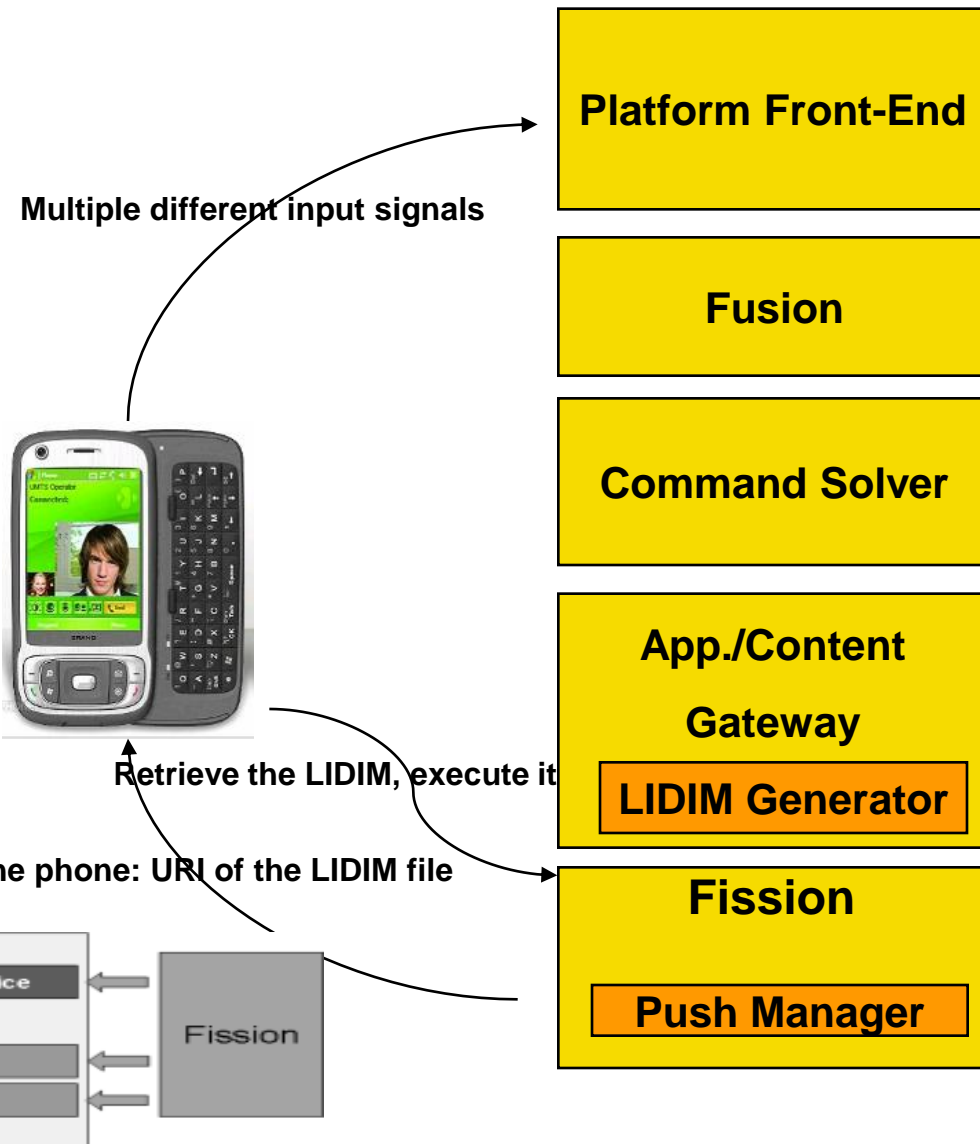
- » **Decorative MMO: just graphics**
- » **Input MMO: MMO that are able to acquire input commands. Eventually able to open stream to the server**
- » **Output MMO: objects that are able to show/play contents to the final user (eventually opening an output stream)**
- » **I/O MMO: These are object that are able to acquire and play multimedia contents**

**We have developed a framework based on J2ME (Java Micro-Edition). Java phones are far more diffused than any other enabling technology. Nevertheless, we have discontinued the development because of limitations (e.g. streaming not supported natively). New incumbent technologies (Google Android)**

**Best results using .NET Compact Framework Technology. Planned for future industrial deployment a porting on Symbian.**



- » The processing pipeline start from the acquisition of one or many input
- » Once interpreted the input are “undestood”
- » A gateway generate the most appropriate LIDIM
- » The push manager inform the Client that a new content is ready. The client retrieve it
- » The LIDIM is intepreted and processed.

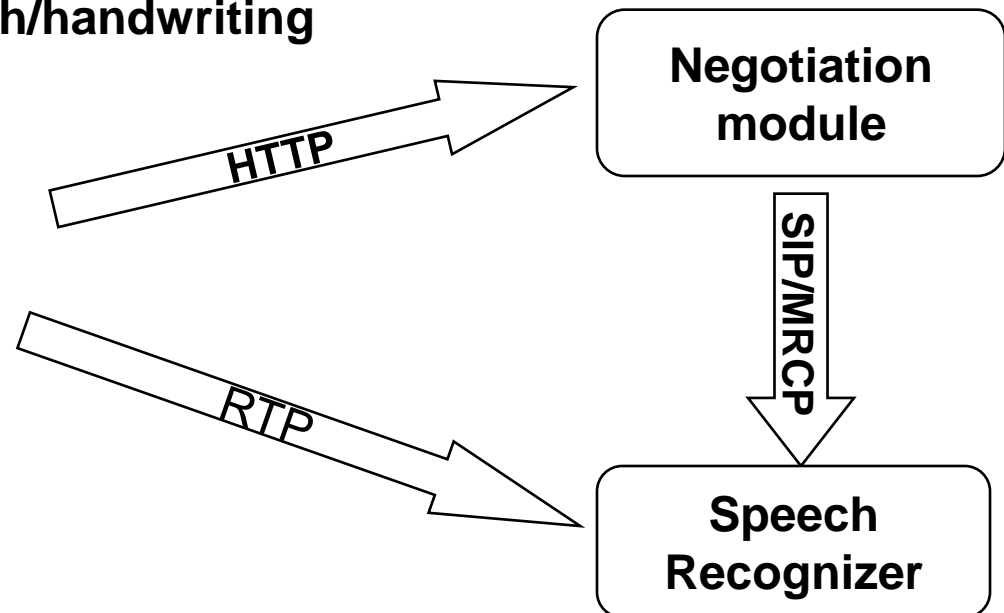


We are able to upstream signals using Telecommunication protocols  
(SIP/RTP)

## Applications

### Continous speech recognition

- » Negotiation protocols to open channel
- » Use of streaming protocols to transmit on the channel (RTP)
- » Audio record format (PCM 8kHz mono16 bit)
- » Commercial Speech recognition software (ASR Loquendo)
- » Other research : extentions of the RTP protocols for supporting the upstream to the server of skecth/handwriting



## Other relevant modes: DSGR, DHR

### Touch screen allows the acquisition of freehand sketch/writing



Traces are saved in InkML Format (W3C)

```
<ink>
<trace id='0'
start='1195474270(
duration='1000'> 1;
128 104 129 99 134
92 148 92 152 92 1
165 99 165 105 165
164 119 161 124 155
152 125 146 124 144
137 121 132 118 124
126 111 124 105<\
<trace id='1'
start='1195474271(
duration='0'> 126
```

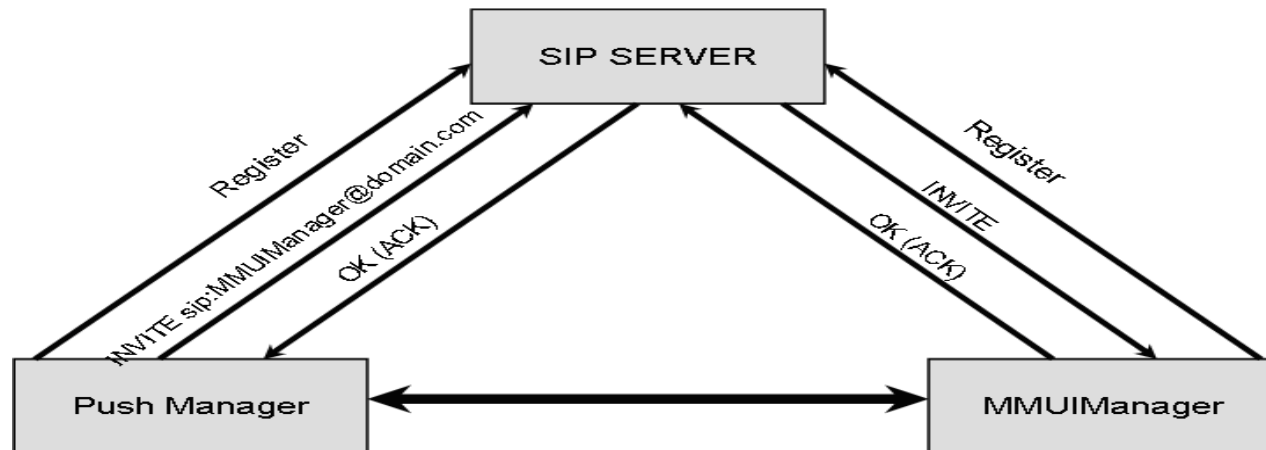
Jarnal for handwriting recognition;

New software and new algorithms developed in collaboration with the IRPPS (Istituto di Ricerche sulla Popolazione e le Politiche Sociali, Rome) for sketch recognition.

# Telecommunication in action: push towards device

A promising solution for interacting with the client could be based on SIP/RTP. The MMUIManager act as a SIP User Agent. It is possible to customize the signaling Phase to instructing the client on the next action.

The server notifies the MMUIManager module using the INVITE SIP command (Session Initiation Protocol). This notification contains the URI of the LIDIM file that must be downloaded.



**Logic can push contents towards devices without an explicit request from final users. For example, if the user activates Bluetooth on his mobile Device, the server can push contents depending on its position ( context)**

- 1. One of our problems: mobile technologies are continuously evolving. Is there any space for multimodality for mass market applications?**
- 2. At the moment the combination of positioning and content push seems more immediate and useful than pure multimodality. Any suggestion?**