Mobile and Context-aware Interactive Systems

Gaëlle Calvary, Joëlle Coutaz and James L. Crowley

Master of Science in Informatics at Grenoble
Université Joseph Fourier (Grenoble I)
ENSIMAG / INP Grenoble
Lesson Plan

1) Introduction: Context Aware Systems and Services
2) Software components for perception, action and interaction
3) Situation Models: a formal foundation for context modeling
4) Acquiring situation models
5) Autonomic methods for software components
Lesson Plan

1) Introduction: Context Aware Systems and Services
   • Evolutions of Computing
   • Intelligent Systems and Services and Social Common Sense
   • Service Composition
2) Software components for perception, action and interaction
3) Situation Models: a formal foundation for context modeling
4) Acquiring situation models
5) Autonomic software components
Evolution of Computing

**Moore’s Law for Transistor Density:**
Transistor Density on IC’s (per m²) **doubles every 18 months.**

**Law for Digital Device Density:**
The number of networked programmable digital devices per person **doubles every 3 years.**

**Epochs in computing:**
Main-Frame Computers: (1960-1980): 1 digital devices per 100 to 1000 persons
Mini-Computers (1970-1990): 1 digital devices per 10 to 100 person
Personal Computing (1980 - 2000): 1 digital devices per 1 to 10 persons
Mobile Computing (1990 - 2010) : 1 to 10 digital devices per person
Internet Computing (2000 - 2020) : 10 to 100 digital devices per person

**Next Big Thing:** Ambient Intelligence

Ambient Intelligence (2010 - 2030) : 100 to 1000 digital devices per person
Embedded Sensors and Actuators

Embedded Vision systems technologies in Mobile Phones
(ST MicroElectronics)
Embedded Systems

Technological Foundations:

- DSPs and Multi-core processors
- Low cost sensors and actuators
- Batteries and low-energy computing
- Low cost, High-bandwidth communications (wired and wireless).

Economic Drivers:

- Mobile Computing
- Mobile Telephones
- Mobile Devices: Ipods, GPS Systems, Electronic Books....

Challenge:

- Context aware services.
- Social Awareness and polite interaction
- Privacy and personal spaces
- Distraction and "Disruption"
Affective Communicating Objects

Examples: Nabastag, AIBO

- Small, autonomous, “cute” devices
- With embedded Perception, Action, Computation and Transmission.
- Using speech, vision, gesture, lights and other modes for interaction
- Wireless Net communications (devices are on the internet)
Problem:
Spontaneous action by "intelligent" systems disrupt human activity.
What do we mean by Intelligent?

Intelligence describes the interaction of an entity with its environment.*

What do we mean by Intelligent?

Intelligence describes the interaction of an entity with its environment.*

Intelligence is a description (an ascribed property)

What do we mean by Intelligent?

Intelligence describes the interaction of an entity with its environment.*

Intelligence is a description (an ascribed property)

Intelligence describes an entity that interacts.

What do we mean by Intelligent?

Intelligence describes the interaction of an entity with its environment.*
What do we mean by Intelligent?

Intelligence describes the interaction of an entity with its environment.*
What do we mean by Intelligent?

To be considered "intelligent", a system must be embodied, autonomous, and situated.

Embodied: Possessing a body (sensory/motor components)
Autonomous: Self-governing;
            Have independent existence
Situated: Behaviour determined by the environment

Embodied Systems

Embodied: Incarnated. Possessing a body.

Body: A sensori-motor system for tightly coupled interaction with an environment.

Examples of Bodies:
Natural: Human, mammal, insects, bacteria, plants,
Artificial: Humanoid Robot, AIBO, mobile robots, roomba?
Abstract: none.
Embodied Systems

Embodied: Incarnated. Possessing a body.

Body: A sensori-motor system for tightly coupled interaction with an environment.

Environment: A system composed of multiple interacting entities.

Examples of Environments:
- Natural: Jungle, desert, sea floor…
- Artificial: Office, home, family, social network, computer games…
- Abstract: Chess, mathematics, any academic discipline…
Ambient Intelligence = Ubiquitous Distraction

Problem: Spontaneous action by "intelligent" systems disrupt human activity.

Reason: Intelligent systems are autistic
- No awareness of context or social situation
- No abilities for polite interaction

Proposal: A theory context aware systems and services
- A framework for modeling context and situation
- A framework for building robust, autonomous systems and services
- A framework for learning polite socially aware interaction.
Common sense: The collection of shared concepts and ideas that are accepted as correct by a community of people.

Social Common Sense: shared rules for polite, social interaction that govern behavior within a group

Situated Social Common Sense: Social common sense conditioned on a model of situation
Situated Social Common Sense

Assertions:

• Politeness is a problem of Situated Social Common Sense
• Politeness requires understanding social situation

Social Common Sense varies over individuals and groups.

⇒ Social Common Sense must be learned
⇒ learning Social Common Sense requires a theory
Lesson Plan

1) Introduction: Context Aware Systems and Services
2) Software components for perception, action and interaction
3) Situation Models: a formal foundation for context modeling
4) Acquiring situation models
5) Autonomic methods for software components
Lesson Plan

1) Introduction: Context Aware Systems and Services
2) Software components for perception, action and interaction
   • A Layered Software Architecture
   • Modules, Components, Federations and Services
   • Streams, Events and Transactions
3) Situation Models: a formal foundation for context modeling
4) Acquiring situation models
5) Autonomic software components
CHIL Layered Software Model Reference Architecture
Software Architectural Reference Model

User Services

Situation Modeling

Logical Sensors, Logical Actuators

Sensors, Actuators, Communications

Ontology Server, Utilities
CHIL Meeting Services

Services (e.g. JADE Agents) access the Situation Model to acquire information about the situation of people and the environment. Services use the Roles to query and subscribe to information.

Connector
- Roles of people in the room (talker, audience)
- Meeting state (on, off, break, Q&A)

Memory Jog
- List of attendees
- ID of a person in the region of interest
Core Component: Situation Model

- Services
  - Situation Model
    - Perceptual Components
      - Function calls
      - Events
      - Streams
Software Architectural Reference Model

User Services

Situation Modeling

Perceptual Components

Logical Sensors, Logical Actuators

Sensors, Actuators, Communications

Ontology Server, Utilities
Components for Perception and Action

Perception - Action Layer:
Ad-hoc assembly of components to provide software services.
Sensory Motor Components

Process model (Finkelstein et al 94).
Data flow Software Architecture (Shaw-Garlan 96)
Process Federations (Estublier and Cunin 97)
Auto-Critical Software Modules

Perceptual Components are composed of modules.

Module: Synchronous Data Transformation

Modules transform data and returns a report on results
Report describes resources used (time, memory) and quality of result
Example: Skin detection

Transform RGB pixels into probability of skin
Theory: Bayes rule
Implementation: table lookup
Probabilistic Detection of Skin

Chrominance:
\[ r = \frac{R}{R + G + B} \quad g = \frac{G}{R + G + B} \]

Probability of all colors
\[ p(r, g) \approx \frac{1}{N_{\text{Tot}}} h_{\text{Tot}}(r, g) \]

Probability of skin
\[ p(r, g \mid \text{skin}) \approx \frac{1}{N_{\text{skin}}} h_{\text{skin}}(r, g) \]

\[ p(\text{skin} \mid r, g) = \frac{p(r, g \mid \text{skin})p(\text{skin})}{p(r, g)} \approx \frac{h_{\text{skin}}(r, g)}{h_{\text{Tot}}(r, g)} = h_{\text{ratio}}(r, g) \]
Skin Blob Tracking
Example:
Skin blob tracker can be composed of skin detection module, robust grouping module and tracking module.
Sensory Motor Components

Components are autonomous assemblies of modules
Components are cyclic and asynchronous.
Components communicate via data streams, events and transactions.
Multi-Cue Face Tracking

Blink Detection:
• Precise but infrequent

Correlation:
• Fast and Precise but fragile

Probabilistic Chrominance:
• Slower and less precise, but reliable.

Approach:
coordinate multiple redundant detection processes

Multi-Cue Face Tracking
Blue Eye Video
Entity Detection and Tracking Process

- Hardwired Control in C++
- Observation Modules:
  - Color Histogram Ratio, Background Difference, Motion History Image,
  - Local Appearance, Receptive Field Histograms
- Industrial Grade System
Lesson: Detect, Track then Recognize.

Tracking (constant time) focusses attention for recognition (Unbounded Time)

Tracking:
1) Conserves identity
2) Focusses Processing Resources
3) Provides Robustness to noise
4) Permits Temporal Fusion
Perceptual Components use three kinds of communications

- **Streams**: Synchronous data channels (should be time stamped)
- **Events**: Asynchronous Messages
- **Transactions**: Query and response between components

**Examples**

- **Streams**: audio microphone channel, video channel
- **Event**: Message that a person has spoken
- **Query**: How many people are observed?
Supervised Perceptual Component

Supervisor Provides:

- Execution Scheduler
- Parameter Regulator
- Command Interpreter
- Description of State and Capabilities
Perceptual Components

Observation Modules:
- Color Histogram Ratio
- Receptive Fields
- Background Difference
- Motion History Image
Supervised Perceptual Component

Process Phases:

While True Do

• Acquire next image
• Calculate ROI for targets
• Verify and update targets
• Detect new targets
• Regulate module parameters
• Interpret entities
• Process messages
Detection and Tracking of Entities

Entities: Correlated sets of blobs
- Blob Detectors: Background difference, motion, color, receptive field histograms
- Entity Grouper: Assigns roles to blobs as body, hands, face or eyes
CHIL Perceptual Component Catalog

Output components:
1. Multimodal Speech Synthesis

2D visual components:
1. Person localization and tracking
2. Body detection
3. Head orientation
4. Face detection and recognition

3D visual components:
1. Person tracking
2. Gesture/posture recognition
3. Head & hand tracking using stereo
4. 3D Pointing gesture recognition using stereo

62 Perceptual Components provided by 8 different partners

Audio-visual components:
1. A/V person tracking
2. Person identity tracking
3. Activity recognition
4. AVSR - mouth (lips) observation
5. Emotion recognition

Audio components:
1. Speech recognition (including far-field)
2. Source localization
3. Speech detection
4. Speaker identification
5. Acoustic emotion recognition
6. Acoustic event classification
7. Beamforming
Lesson Plan

1) Introduction: Context Aware Systems and Services
2) Software components for perception, action and interactive
3) Situation Models: a formal foundation for context modeling
4) Acquiring situation models
5) Autonomic software components
Lesson Plan

1) Introduction: Context Aware Systems and Services
2) Software components for perception, action and interactive
3) Situation Models: a formal foundation for context modeling
   • Situation Models for Interaction
   • Entities, Relations, and Situation Graphs
   • Roles and Situations
   • Context and Situation.
4) Acquiring situation models
5) Autonomic software components
Situation Models:
a formal foundation for context modeling

Context:
• The situation within which something exists or happens, and that can help explain it [Cup];
• Any information that can be used to characterize situation. [Dey01]

Situation:
• the set of things that are happening and the conditions that exist at a particular time and place. [Cup].

[Cup] Cambridge University on-line dictionary of the English language
Situation Models: An analytical tool for describing interactions

P. Johnson-Laird 1983 - Situation Model

An analytical tool to allow Human Psychologists to model human to human interaction.

Situation: Relations between entities

Entities: People and things;

Relations: An N-ary predicate (N=1,2,3 …)

Example: John is facing Mary. John is talking to Mary.
Proposal: Use situation models as a software framework for systems and services that interact with humans

**Situation:**

- A configuration of relations between entities, with
- The appropriateness of actions for the situation.

**Context:**

- A situation network composed from
- A set of entities, relations, actions, and situations
Situation Models for Interaction

**In Theatre:**
A script defines a linear sequence of scenes.
Actors use props to play roles
The roles define the space of action for an actor
   (movements, expressions, etc)
The Script defines the appropriate spoken phrases for each scene

**In human activity**
People play roles in shared interaction contexts
Roles define appropriate and inappropriate actions
Social interaction is not linear but includes alternatives and loops.
   (A network rather than a sequence.)

Social interaction is modeled as a Situation Graph
A situation graph describes a state space of situations

A Situation determines:
- System Attention: entities and relations for the system to observe
- System Behaviours: List of actions that are allowed or forbidden
Situation Models for Interaction

Each situation indicates:

- Transition probabilities for accessible situations
- The appropriateness or inappropriateness of actions.
Roles and Situations

Role: An abstract person or thing

A role predicts the actions that might be taken by an actor or the actions enabled by an object.

Entity: A correlated set of observed properties.

Two kinds of entities:

Actor: An entity that can spontaneously act to change a situation.

Prop: An entity that can not spontaneously act.
A **role** is a "variable" for entities.

Roles allow generalizations of situations.
Roles enable **learning by analogy**
Perceptual Components

Role, Relation Assignments

- Speaker Identification
  - Entity $E_A$
  - SAD
- Position Detection
  - Entity $E_T$
  - Tracker
- Person Recognition
  - $E_T$
  - (NewEntity $E_T$), (Lost $E_T$), (Enters RID $E_T$), (Exits RID $E_T$), (Person $E_T$ PID)

- (Speaking $E_A$), (Stop-Speaking $E_A$), (Person $E_A$ PID)
Context and Situation

- Situation Modeler for identifying context aware states written in Java
- High level perceptual components written in Java language
- Control Agent written in Java language
- Low level perceptual components written in C language

Diagram:
- Situation Modeler
- Whiteboard Watcher Agent
- Table Watcher Agent
- Perceptual Component Agent
- Face ID
- Body Tracker
- Speech Activity Detection
- Agenda Tracker

Arrows indicate the flow of data or events between components.
CHIL Services using Situation Model

Services (e.g. JADE Agents) access the Situation Model to acquire information about user and environment context.

Services use the Roles to query and subscribe to information.

Connector
- Roles of people in the room (talker, audience)
- Meeting state (on, off, break, Q&A)

Memory Jog
- List of attendees
- ID of a person in the region of interest
Core Component: Situation Model

Services

Situation Model

Perceptual Components

Function calls
Events
Streams
Example: Context Aware Video Acquisition System

**Situations:**
- $S_0$ Empty room $\implies A_1$
- $S_1$ Actor enters the room $\implies A_1$
- $S_2$ Speaker (actor) speaks $\implies A_2$
- $S_3$ Audience (actor) asks a question $\implies A_3$
Video Acquisition Service

Service Supervisor

Event Bus

Actor Tracker
Camera 1

Actor Tracker
Camera 1

Actor Tracker
Camera 1

Speech Detection
Microphone Array

Speech Location
Automatic Video Acquisition System
(version 1.0 - Jan 2005)
Video Acquisition System V2.0

- Process Supervisor
- Situation Modeling
- Event Bus
- Audio-Visual Composition
  - Vocal Activity Detector
  - New Slide Detection
  - New Person Detection
  - Face Detection
  - Speaker Tracker
  - Audience Camera
  - Face Detection Camera
  - Steerable Camera 1
  - Wide Angle Camera
  - Projector

- Streaming Video
- MPEG
Lesson Plan

1) Introduction: Context Aware Systems and Services
2) Software components for perception, action and interactive
3) Situation Models: a formal foundation for context modeling
4) Acquiring situation models
5) Autonomic software components