# PAPRIKA

## A FRAMEWORK FOR PAPER-BASED INTERACTION

Carlos Sanchez Witt
Master Thesis Project
CHILI, EPFL, 2014

# What is Paprika

An **abstraction model** for:

- **Paper card** controllers,
- **Properties** and **Interactions** of said controllers

A **practical tool** for:

- **Easy implementation** of paper-based applications
- That requires **no deep knowledge** of the technology and mathematics under the hood

**A solution for Design and Development**

# Outline

- **Context and Motivation**
  - TUIs and Paper Interfaces
  - Paper Cards in Video Games
  - Modeling Interaction
- **Paprika**
  - Framework for Paper-based Interaction
  - JavaScript Implementation
- **Discussion**
  - Assumptions and Outstanding Challenges
  - Conclusion and Future Work

# Tangible User Interfaces

- An interface in which the user interacts with **digital information** through the **physical environment**.
- Examples: mouse, gamepad, touchscreen, etc.

**Why tangible?**

- Physical, natural, intuitive, direct manipulation

**Why digital?**

- Rich, dynamic, multi-media applications, simulations

# PAPER INTERFACES

## Paper is ubiquitous and affordable

- Notebooks, newspapers, maps, banknotes, playing cards, etc.

## Paper as a powerful interaction device

- Through **content** (writing, reading, annotating, drawing, etc.)
- Through **tangibility** (folding, wrapping, orienting, flipping, etc.)

## Research at CHILI


- Bonnard, Q. (2012). Paper Interfaces: an HCI Approach to Geometry Education. Ph.D. Thesis, École Polytechnique Fédérale de Lausanne (EPFL). doi:10.5075/epfl-thesis-5579
- Cuendet, S., Bonnard, Q., Kaplan, F., & Dillenbourg, P. (2011, May). Paper interface design for classroom orchestration. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (pp. 1993-1998). ACM.

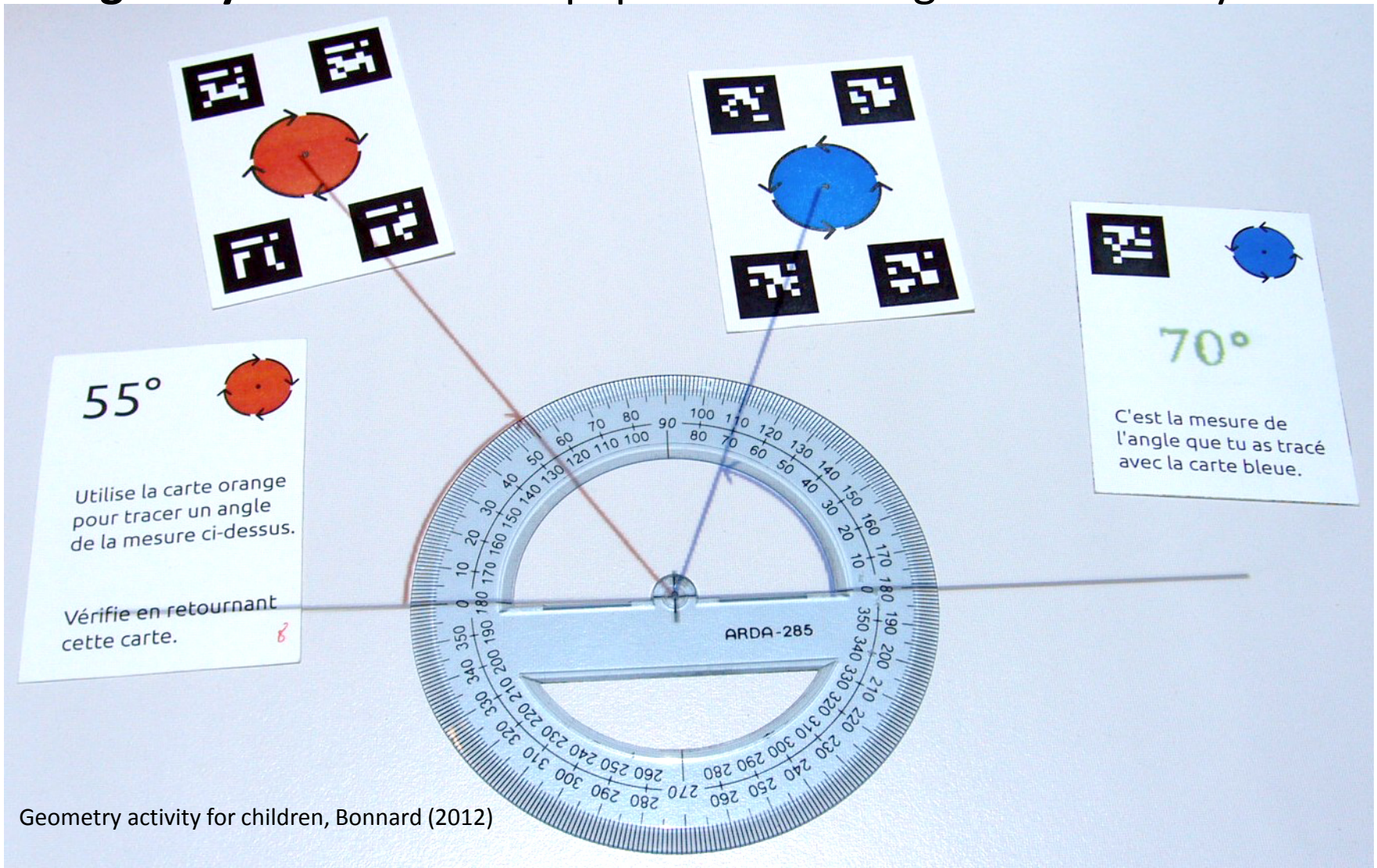
# Paper Cards in Video Games


The Eye of Judgment,
Sony PlayStation 3 (2006)


Computer Graphics
Hardware
Computer Vision & Tracking

# PAPER CARDS IN RESEARCH

**Tangibility** and **content** of paper cards in augmented reality




Geometry activity for children, Bonnard (2012)

# Fiducial Markers

2D pictograms placed on objects for the purpose of **identification** and **tracking**.

- Used to detect the **presence** of a marker,
- And extract its 4x4 **transformation** in 3D space.

## Chilitags

- Developed at CHILI
- Multiplatform
- Illumination tolerant
- Precise and reliable at low resolutions

https://github.com/chili-epfl/chilitags

# Modeling Interaction

**Tracking technologies ✔**

**Proven potential of paper interfaces ✔**

**Creating paper-based applications remains hard**

- **Different** technologies, **different** APIs, **difficult** to interface
- High abstraction manipulations **unavailable** (flipping, orienting)
- (Re)implementation requires **deep technical knowledge**

**DART**: Toolkit for Adobe Director, a model for AR environments.

**TUIO**: Protocol + API for tabletop interfaces.

Kaltenbrunner, M., Bovermann, T., Bencina, R., & Costanza, E. (2005, May). TUIO: A protocol for table-top tangible user interfaces. In *Proc. of the 6th Int'l Workshop on Gesture in Hu-man-Computer Interaction and Simulation*.

MacIntyre, B., Gandy, M., Dow, S., & Bolter, J. D. (2004, October). DART: a toolkit for rapid design exploration of augmented reality experiences. In *Proceedings of the 17th annual ACM symposium on User interface software and technology* (pp. 197-206). ACM.

# Outline

- **Context and Motivation**
  - TUIs and Paper Interfaces
  - Paper Cards in Video Games
  - Modeling Interaction
- **Paprika**
  - Framework for Paper-based Interaction
  - JavaScript Implementation
- **Discussion**
  - Assumptions and Outstanding Challenges
  - Conclusion and Future Work

# What is Paprika

An **abstraction model** for:

- **Paper card** controllers,
- **Properties** and **Interactions** of said controllers

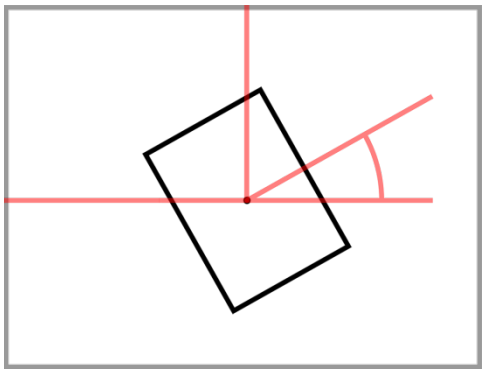
A **practical tool** for:

- **Easy implementation** of paper-based applications
- That requires **no deep knowledge** of the technology and mathematics under the hood

**A solution for Design and Development**

# Framework Elements

- **Playfield**
  - Reference **plane of action**
  - Two-dimensional
- **Controller**
  - **Paper card**, flat rectangle
  - Height and width, no depth



## Basic Properties

- **Presence**
  - Detection in playfield
- **Position**
  - Coordinates w.r.t. playfield
- **Rotation**
  - Angles of **orientation** and **inclination** w.r.t. playfield

# Abstract Interactions

Manipulations as **atomic events** derived from properties.

- Presence
  - **Appear/Disappear:** card has entered the playfield
- Position
  - **Approach/Retreat:** card has entered region of playfield
  - **Stack/Unstack:** card has been placed on top of another
- Rotation
  - **Orient/Disorient:** card has been oriented at a specific angle
  - **Tilt:** card has been inclined a specific amount
  - **Flip:** card has been flipped entirely (inclined past 180°)

# Tree + use cards

# e1 – orientation + flip

# e2 – position + stack

# JavaScript Implementation

**Target platform: The Web**

- Rich interactive applications and games

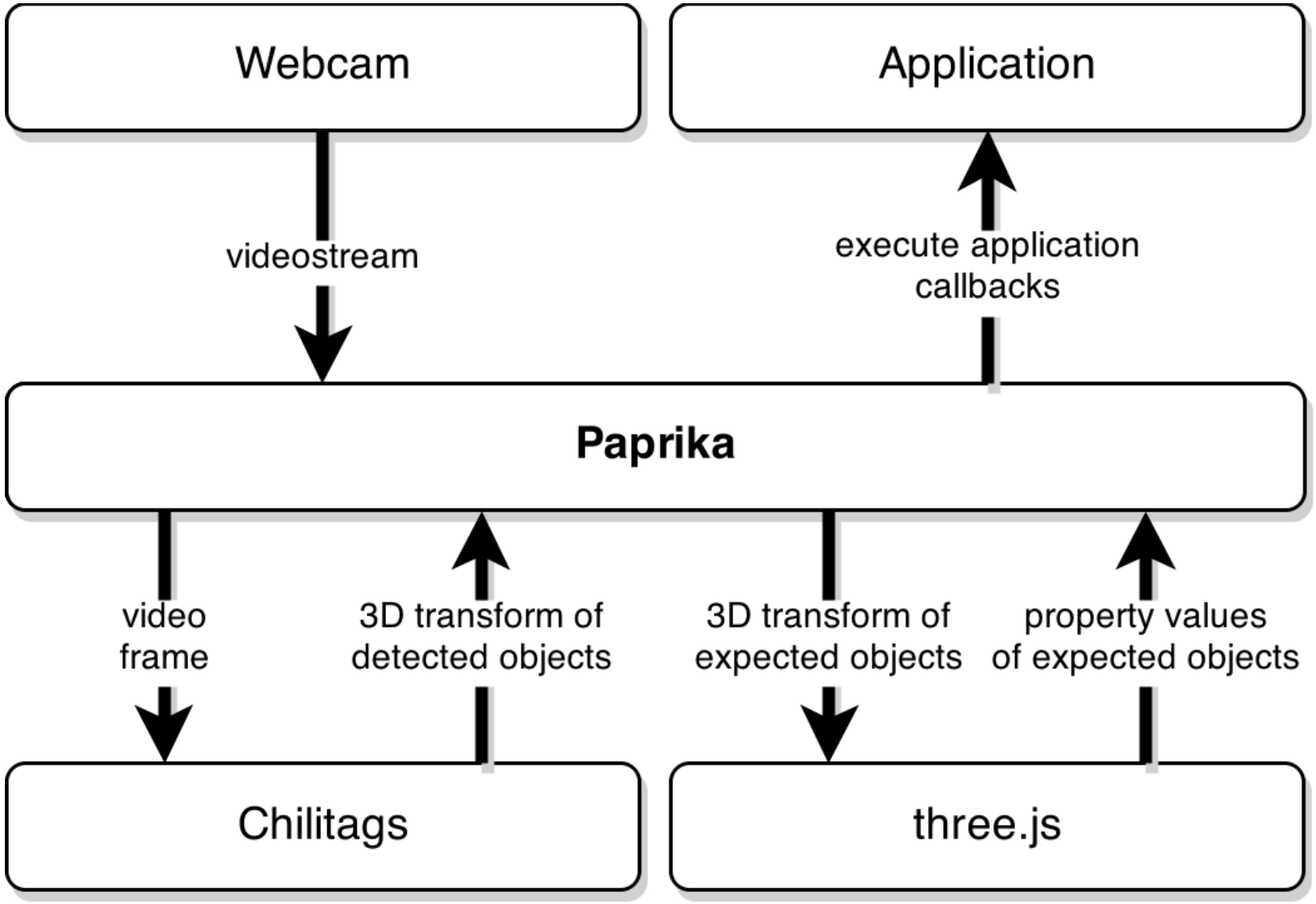
**Dependencies**

- **Chilitags**: computer vision solution for tracking
- **three.js**: computer graphics solution for mathematics

**Informal Evaluation**

- Implementation of HTML5 games using Paprika input.

# PAPRIKA EXECUTION FLOW



# RESULTS

## Abstract Framework

- A considerable set of **properties** and **interactions.**
- A **base language** for describing paper-card interfaces that **helps in communication** and **design** of paper-based applications.

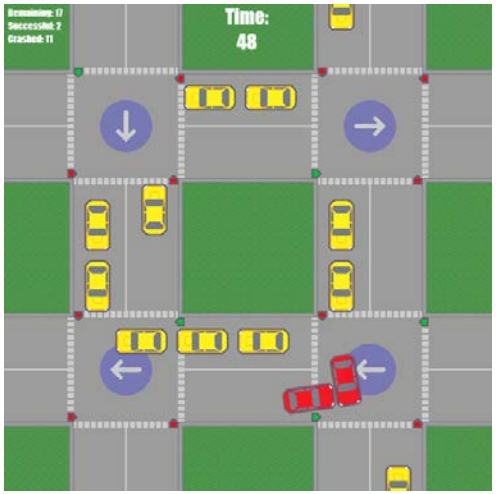
## JavaScript Implementation

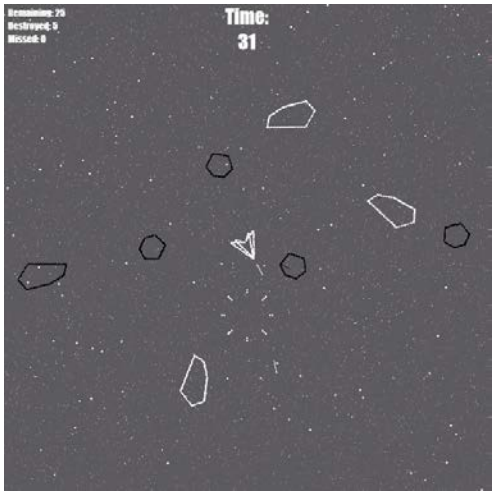
- A simple tool that **does not require technical knowledge** of computer vision, computer graphics, or AR.
- Accessible solution that **simplifies development** by providing continuous access to properties and defining **atomic events for high abstraction interactions.**

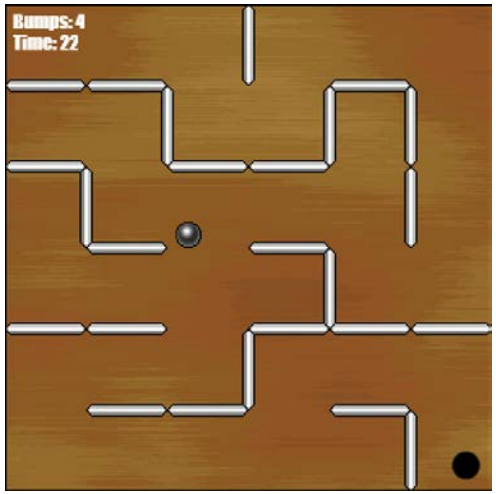
# RESULTS

## Informal Evaluation

- **Three browser games** developed using Paprika
- Put the framework **to the test** while in development
- Mutual influence between game design and **abstract concepts**
- **Phaser:** HTML5 JavaScript framework for 2D games



# Traffic demo

# Outline

- **Context and Motivation**
  - TUIs and Paper Interfaces
  - Paper Cards in Video Games
  - Modeling Interaction
- **Paprika**
  - Framework for Paper-based Interaction
  - JavaScript Implementation
- **Discussion**
  - Assumptions and Outstanding Challenges
  - Conclusion and Future Work

# CAMERA-CENTRIC IMPLEMENTATION

The **camera view** is the **playfield**.

- The **physical camera setup** must be taken into account by the **developer** using Paprika and communicated to the **player**.
- Properties are relative to the view, there is no notion of **physical scale** or **distances**.

**Need for a Referential System**

# Physical Playfield Setup

- **Developers** must **decide on intended setup.**
- This must be **communicated** to the **player**.

- Icons symbolizing the **top-down** and **mirror** setups used in our games.

# Non-exhaustive Set of Interactions

**Multi-card actions**

- Align, group, spread

**From defined properties**

- Movement, speed

**Unused properties of paper**

- Malleability: fold, deform, tear ?
- Ink: write, draw ?

**Expandable set of interactions**

# Implementation and Evaluation

- Fiducial markers: **visibility** is essential
- **Co-located** and **occluded** actions are hard
- Is our model portable to other technologies? (RFID)
- Abstraction model **untested** by other developers
- Abstract concepts and communicative potential **unevaluated**

**Involve the web developer community**

# SUMMARY

## Abstract Framework

- A considerable set of **properties** and **interactions.**
- A **base language** for describing paper-card interfaces that **helps in communication** and **design** of paper-based applications.

## JavaScript Implementation

- A simple tool that **does not require technical knowledge.**
- Accessible solution that **simplifies development** by providing access to properties and defining **atomic events for high abstraction interactions.**
- **Three browser games** developed using Paprika.

## Future Work

- Referential system development and expandable set of interactions
- Ongoing open source development

# Games