

Patterns of Musical Interaction with Computing Devices



www.inf.ufrgs.br/lcm

Luciano V. Flores,
Marcelo S. Pimenta (UFRGS),
Damián Keller (UFAC)



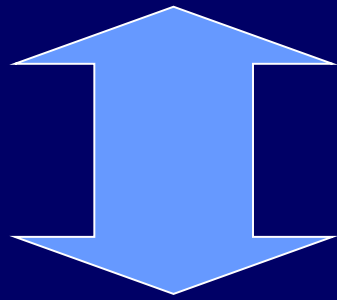
III UbiMus – 2012 – São Paulo



Overview

- Usual forms of musically interacting with computing devices.
- A way to organize/document/formalize these alternatives, suitable for interdisciplinary design.
- Discussion about the proposed design patterns.

- **HCI – Human-Computer Interaction**



- **CM – Computer Music**



- **⇒ MUSICAL INTERACTION**

Context

■ **Ubiquitous Music project**

- Cooperation: UFRGS, UFAC, Plymouth, USP, UFU, UNICAMP, FAESA, NUI-Maynooth
- Computer Music, HCI, Ubiquitous Computing, Music, Music Education, Musicology



■ **Thesis (a subproblem)**

- Music-making with ordinary, **everyday mobile devices**
- **Infrastructure for the design of musical interaction** (with such devices): principles/concepts, **patterns**, processes, tools

Ubiquitous Music (ubimus)

- Broad, interdisciplinary definition:
 - Ubiquitous systems of human agents and material resources that afford musical activities through creativity support tools [G-Ubimus 2012].
- Computer Science perspective:
 - Music (musical activities) supported by Ubiquitous Computing (ubicomp) technology [Weiser 1991] and applying its concepts.
 - Resources, tools ⇒ **COMPUTING DEVICES**

Questions (from the thesis)

- “How to play a mobile phone?”
 - Non-specific, not made for music
 - UI limitations (but... they have sensors 😊)

- “How to design musical interaction which involves non-specific interaction devices?”
 - Specifications change
 - Ubiquitous music: device independence

How to *play* computing devices?

- Possible ways of manipulating music
- Common solutions in CM

⇒ **PATTERNS** of musical interaction

How to *design* for ubimus?

- Ubicomp or new digital contexts
 - Abstract the device (device independence)
 - Focus on interaction, not interfaces

⇒ ***Interaction design patterns***

- Borchers, 2001; Tidwell, 2005.
- Abstract/encapsulate design solutions
- Encapsulate design/domain knowledge
[Flores et al. 2010]

Patterns

- Patterns are “repeating things”
- “A pattern is the abstraction from a concrete form which keeps recurring in specific non-arbitrary contexts”
[Riehle and Züllighoven 1996]

Patterns in design fields

- Common, high-quality solutions to also common design problems, which have been systematically collected and documented
- “A design pattern is a structured textual and graphical description of a proven solution to a recurring design problem”
[Borchers 2001]

The *design patterns* idea

- “A proven solution to a commonly recurring design problem” [Borchers 2000]
 - Carefully documented – “portable”/compact description
 - Not created, but collected – from observing/detecting/noticing common solutions in some design domain
 - May be organized in a “pattern language”, with hierarchical levels of abstraction and relationships
 - May be combined into more complex solutions
 - Works as a common terminology in design teams, and captures design knowledge

In the world, and as language

- “As an element of language, a [design] pattern is an instruction, which shows how this [...] configuration can be used, over and over again, to resolve the given system of forces, wherever the context makes it relevant. [...] The pattern is, in short, at the same time a thing, which happens in the world, and the rule which tells us how to create that thing, and when we must create it.”
[Alexander et al. 1977]

Interaction Design Patterns

Carousel

Applies to:

[edit]

User Interface: *Scroll & select interfaces > 4- and 5-way keypads, Screen-input interfaces, 2-D scroll*

Hardware: *Any*

A carousel allows the user to visually browse through content. It's especially useful for highly visual content, such as photos, videos, PDFs and so on. Non-visual items such as music can be browsed by using a well-understood individual graphic such as the album cover.

Some desktop systems (OSX column view) have explored iconic methods of displaying other content types, but these are still in their infancy so no usable standard exists for items without good, individual iconic representations of text, email, voicemail and other such common mobile content.

Slideshows are not carousels.

Design

[edit]

Individual files are displayed as thumbnails or icons within the carousel browser. A sense of motion should be designed into it. The example to the right implies the files revolve around the edge of the screen; the iPhone implies the files are off the screen to the right and left. Not all files available must be visible on the screen at once.

The selected item must be centered, and is generally indicated as being highlighted in some other manner. Generally, carousels make the selected item larger than other items on the screen. If this is much larger, all items may be of varying size, with larger items closer to the currently viewed item. This size implies proximity, and makes the item more visible for immediate scanning of several items, without selecting each item individually.

Scrolling or gesturing will move between selected items, and must reinforce the directionality of the motion paradigm.

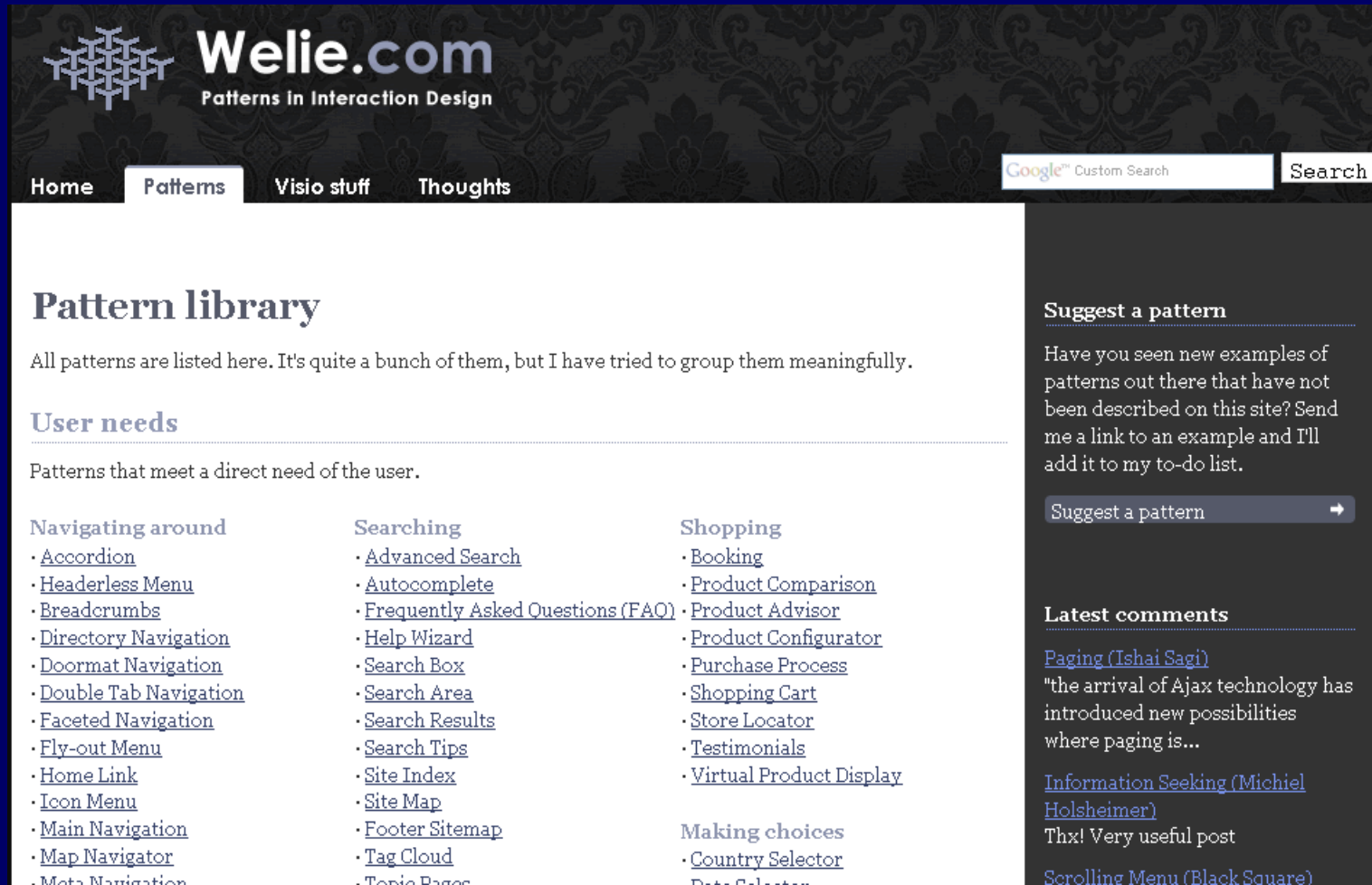
A method to select the item for viewing, editing or other actions must be made available.

Note that the carousel can scroll vertically, instead of horizontally. This may be preferable if your primary scrolling mechanism implies vertical-only scrolling.



The carousel of the Symbian S60 media browser, with both motion and still video (photo) clips in one interface.

Interaction Design Patterns



The screenshot shows the Welie.com website, which is a resource for interaction design patterns. The page has a dark background with a light-colored pattern. The header includes the Welie.com logo and navigation links for Home, Patterns, Visio stuff, and Thoughts. A search bar is located in the top right corner. The main content area is titled "Pattern library" and contains a list of patterns categorized into three groups: Navigating around, Searching, and Shopping. A sidebar on the right contains a "Suggest a pattern" form and a "Latest comments" section.

Welie.com
Patterns in Interaction Design

Home **Patterns** Visio stuff Thoughts

Google™ Custom Search Search

Pattern library

All patterns are listed here. It's quite a bunch of them, but I have tried to group them meaningfully.

User needs

Patterns that meet a direct need of the user.

Navigating around <ul style="list-style-type: none">• Accordion• Headerless Menu• Breadcrumbs• Directory Navigation• Doormat Navigation• Double Tab Navigation• Faceted Navigation• Fly-out Menu• Home Link• Icon Menu• Main Navigation• Map Navigator• Meta Navigation	Searching <ul style="list-style-type: none">• Advanced Search• Autocomplete• Frequently Asked Questions (FAQ)• Help Wizard• Search Box• Search Area• Search Results• Search Tips• Site Index• Site Map• Footer Sitemap• Tag Cloud• Topic Pages	Shopping <ul style="list-style-type: none">• Booking• Product Comparison• Product Advisor• Product Configurator• Purchase Process• Shopping Cart• Store Locator• Testimonials• Virtual Product Display
--	---	---

Making choices

- [Country Selector](#)
- [Date Selector](#)

Suggest a pattern

Have you seen new examples of patterns out there that have not been described on this site? Send me a link to an example and I'll add it to my to-do list.

Suggest a pattern →

Latest comments

[Paging \(Ishai Sagi\)](#)
"the arrival of Ajax technology has introduced new possibilities where paging is..."

[Information Seeking \(Michiel Holsheimer\)](#)
Thx! Very useful post

[Scrolling Menu \(Black Square\)](#)

Problems / needs

- An interdisciplinary project
 - A multidisciplinary research (and design) team
 - We have to work together, to cooperate
 - We “know” different “things” (perspectives) and “talk” different “languages” (vocabulary, terminology)
⇒ need for a common vocabulary
- Designing for the new digital technologies
 - Ever-changing technologies, contexts, uses
⇒ need for abstraction



The four collected patterns

- Natural Interaction / Natural Behavior**
- Event Sequencing**
- Process Control**
- Mixing**

Problem and principles

- How to manipulate music and musical information using computing devices?
 - Music manipulation, multimedia manipulation
- Principles
 - Musical-activity-independence
 - Combinations, to generate more complex designs

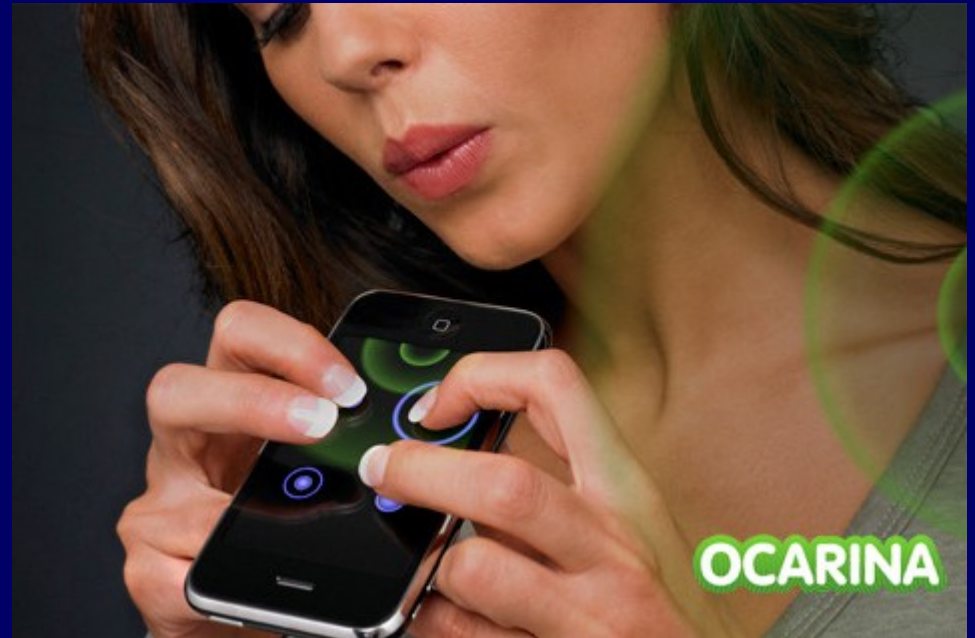
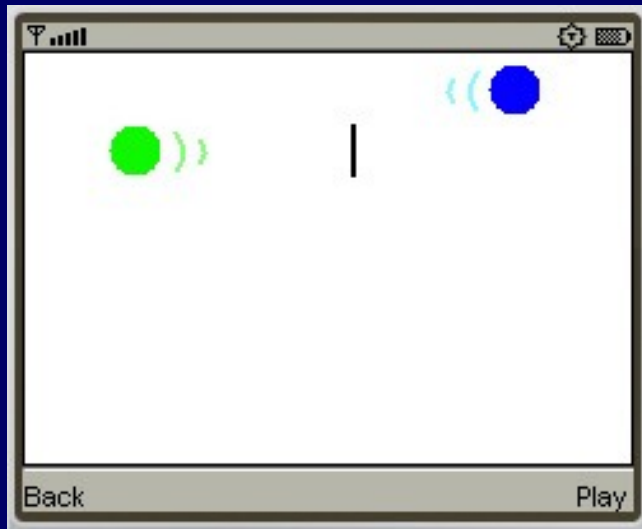
■ Natural Interaction / Natural Behavior

- Imitate real-world, natural interaction.
- Musical interaction which imitates real interaction with a sound-producing object. Thus, all musical gestures that we might regard as “natural” may be explored herein: striking, scrubbing, shaking, plucking, bowing, blowing, etc. It is related to the metaphor of “musical instrument manipulation” [Wanderley and Orio 2002], and to the “one-gesture-to-one-acoustic-result” paradigm [Wessel and Wright 2002] – hence its alternative label, “Natural Behavior”.

■ Drum! (Natural Interaction and Event Sequencing)



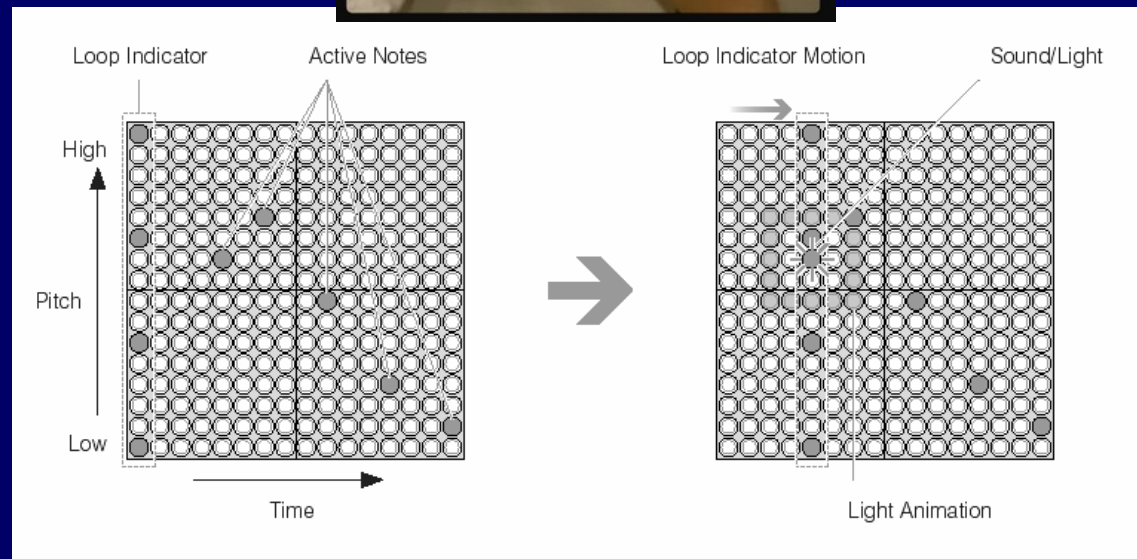
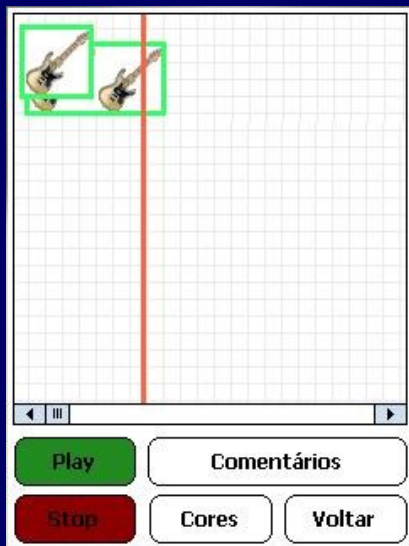
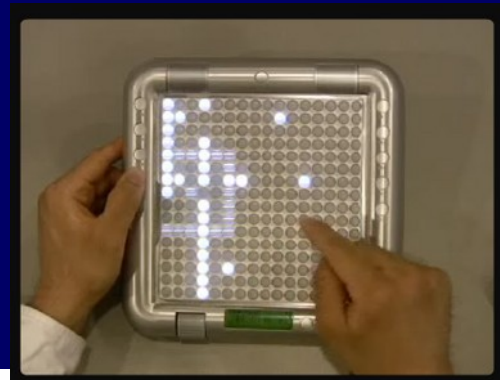
■ Bouncing Balls (Natural Behavior)



■ Event Sequencing

- Allows the arrangement of musical events in large sets (the timeline of the music).
- “Early scheduling” of events, asynchronous/early configuration. Distributing or organizing events in time is done in some moment before their actual occurrence (i.e., foreseen/planned). The focus is on the relative organization between events, seen together as a whole set in some “region” of time.
Allows/encourages epistemic actions
[Kirsh and Maglio 1994].

■ Event Sequencing



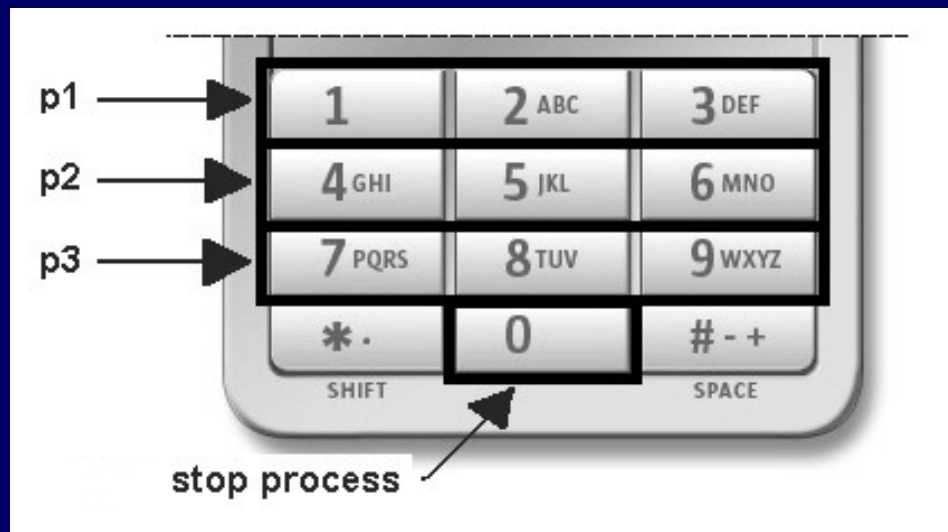
■ Event Sequencing

The screenshot shows the 'codes' web application running in Mozilla Firefox. The interface is designed for creating musical prototypes. At the top, there are navigation buttons and a search bar. Below the browser's address bar, the application header includes the 'codes' logo, a 'Create a new musical prototype' button, and a notification for '2 pending invitations'. The main workspace is a grid-based sequencer for a 'Hommer' prototype. It features a timeline at the top with a playhead and a 'Loop' checkbox. The sequencer is divided into two sections by a vertical line. The left section contains tracks for guitar (with fret numbers 1-4), keyboard (with notes 1-3), drums (with notes 1-4), and another keyboard track (with notes 1-5). The right section contains tracks for drums (with notes 1-2), keyboard (with notes 1-1), and another keyboard track (with notes 1-2). Below the sequencer is a control panel with buttons for undo, redo, and a 'hammer' icon. At the bottom, there is a 'Members Selection' panel with a list of members: 'Hommer Singing', 'Musium', and 'djnoise'. To the right of this panel are tabs for 'Rock', 'Funk', 'Jazz', 'Electronic', and 'Percussion', each with a grid of instrument icons. A 'Sound library' section is also visible on the far right.

■ Process Control

- Free the user from event-by-event music manipulation, by allowing him/her to control a process which, in turn, generates the actual musical events or musical material.
- A mapping from the (limited) interaction features of mobile devices, not to musical events, but to a small set of musical process parameters. Analog to the role of a conductor (in fact, corresponds to the “conductor mode”, as suggested by Dodge and Jerse [1997]).

■ Arpeggiator (Process Control, non-specific devices)



■ Mixing

- Music manipulation through real-time control of the parallel execution of long musical structures (musical material) – i.e. by mixing musical material.

A kind of “layered” composition of musical material, done in real-time.

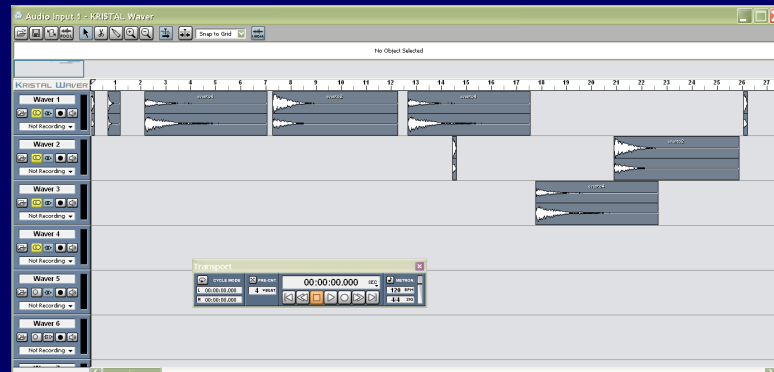
■ mixDroid (Mixing) [Radanovitsck et al. 2011]



[Tanaka
2004]

Final discussion

- Four musical interaction patterns that can be used for ubiquitous music systems design.
 - Accounts for user-device interaction.
 - Accounts for unavailability of resources.
- Preliminary tests on patterns comprehensibility (assimilation) and activity-independence.



Final discussion

- A necessary switch in CM design, from the current technology-oriented perspective to a more user-centered perspective.
- Future work:
 - More tests/experiments. Evaluate use in design.
 - Other hierarchic levels (musical interface patterns) – a pattern language?
 - Patterns for ubiquitous interaction: cooperation, sharing, emergence, location awareness, context awareness,...

References

- Alexander, C. et al. (1977) “**A Pattern Language: Towns, Buildings, Construction**”. New York, NY: Oxford University Press.
- Borchers, J. (2000) “**A Pattern Approach to Interaction Design**”, In: Proc. of the ACM Conference on Designing Interactive Systems, New York, USA. p. 369-378.
- Borchers, J. (2001) “**A Pattern Approach to Interaction Design**”. Chichester, UK: John Wiley & Sons.
- Dodge, C. and Jerse, T. A. (1997) “**Computer Music: Synthesis, Composition, and Performance**”. New York, NY: Schirmer Books.
- Flores, L. V. et al. (2010) “**Musical Interaction Patterns: Communicating Computer Music Knowledge in a Multidisciplinary Project**”, In: Proc. of the 28th ACM International Conference on Design of Communication, São Carlos, Brazil. p. 199-206.

References

- G-Ubimus (2012) “**Ubiquitous Music Group**”, <http://groups.google.com/group/ubiquitousmusic/>, April.
- Kirsh, D. and Maglio, P. (1994) “**On Distinguishing Epistemic from Pragmatic Action**”, Cognitive Science 18: 513-549.
- Radanovitsck, E. A. A. et al. (2011) “**mixDroid: Marcação Temporal para Atividades Criativas**”, In: Proc. of the 13th Brazilian Symposium on Computer Music, Vitória, Brazil.
- Riehle, D. and Züllighoven, H. (1996) “**Understanding and Using Patterns in Software Development**”, Theory and Practice of Object Systems 2(1): 3-13.
- Tanaka, A. (2004) “**Mobile Music Making**”, In: Proc. of the International Conf. on New Interfaces for Musical Expression, Hamamatsu, Japan. p. 154-156.

References

- Tidwell, J. (2005) “**Designing Interfaces: Patterns for Effective Interaction Design**”. Sebastopol, CA: O’Reilly Media.
- Wanderley, M. M. and Orio, N. (2002) “**Evaluation of Input Devices for Musical Expression: Borrowing Tools from HCI**”, Computer Music Journal 26(3): 62-76.
- Weiser, M. (1991) “**The Computer for the Twenty-First Century**”, Scientific American 265(3): 94-101.
- Wessel D. and Wright, M. (2002) “**Problems and Prospects for Intimate Musical Control of Computers**”, Computer Music Journal 26(3): 11-22.



Contact

- lucianovflores@google.com
- www.inf.ufrgs.br/lcm (Computer Music Lab)