Who Are the Web Composers?

Evandro Manara Miletto, Marcelo Soares Pimenta, Aurelio Faustino Hoppe,
and Luciano Vargas Flores

Instituto de Informática – UFRGS. Av. Bento Gonçalves, 9500. Porto Alegre, RS. Brazil
{miletto,mpimenta,afhoppe,lvflores}@inf.ufrgs.br

Abstract. Web 2.0 with RIA (Rich Internet Applications) becomes a wide field for social networks and new distributed collective practices. In this paper we explain why and how CODES, a novice-oriented Web-based environment for cooperative music prototyping, provides support to a new practice in which novices in music may produce (not only consume) music cooperatively. CODES stimulates the emergence of new user roles – these users not only create and edit cooperatively their own music but also may participate in discussions and exchange ideas about their contributions. The implications of this Web-based group music making and shared authorship – some of them identified through actual experiments – are also presented.

Keywords: cooperative music prototyping in the Web, Web composition, networked music, novice-oriented interaction.

1 Introduction

Day by day the Internet is growing as an environment for communication, data exchange, and information in all fields, shortening distances and providing interaction features that support the growth of virtual communities. Web 2.0 sites such as YouTube, MySpace, Facebook, and Flickr have improved and enhanced the interaction between users and systems over the Web, making it a potential field to explore distributed collective practices (DCP) [12].

Thereby, nowadays we have the growth of other types of user profile, interested in an effective participation, with the freedom of the "playful" element which allows them to create and express themselves, and not only act as "passive" and consumer users anymore. At least, most of these Web 2.0 sites have two profiles of end users: the "browsers" and the "producers" of content [11].

These practices are typically related to Rich Internet Applications (RIAs), a class of more sophisticated Web applications that behave likewise the desktop-based software programs. Compared to normal Web pages, they are rich in interaction, in content, and so in functionality. The primary advantage of RIAs is an enhanced user experience. Knowing and keeping the behaviors and practices that RIAs allow can be crucial to bring the ordinary Web-users to experiment with music and technology where most of them never tried before.

A.A. Ozok and P. Zaphiris (Eds.): Online Communities, LNCS 5621, pp. 381–390, 2009.

The field of music has traditionally served as a natural motivation for community formation, even considering that the most widely supported practice is sharing ready-made music files. We are convinced that Web-based RIAs are able to address the main concerns for breaking barriers that keep non-musicians away from expressing themselves musically.

CODES is a Web-based environment for cooperative music prototyping (CMP) by novices in music, where the main task is experimenting with music by combining, listening, rearranging sound patterns, and cooperating with partners in order to produce their music, which we call a Web (music) composition. Indeed, with Web-based RIAs like CODES, everyone having access to the Internet, using a personal computer as a general virtual musical instrument (instead of having a real one) and using a high level music representation to experience with music (instead of traditional music notation) may produce music [9]. CODES was designed to support cooperative activities between users while prototyping musical pieces. This implies requirements that range from HCI, CSCW, and Computer Music disciplines integrated in the same environment. The ultimate goal of CODES is to support activities which allow the Web composers (typically novices in music) to experiment with music, discuss, and exchange ideas with others users about each step of their experiments. Even more than producers of content, these users create and edit cooperatively their own culture.

This paper presents and discusses this new user profile and the challenges it poses to the design of an environment providing features which support this goal, like group composition, authorship, and other characteristics identified through some experiments.

The text is organized as follows. Next section discusses the characteristics of the music prototyping process in the Web. Section 3 presents some requirements related to the design for Web composers, which we detected during the development of CODES. Some social aspects of CMP are discussed in section 4. Section 5 presents some preliminary test results, and section 6 presents the concluding remarks.

2 Musical Prototyping in the Web

The design of CODES has been carefully conceived to support the cooperative music prototyping process, taking into account concepts from other disciplines and the overall characteristics of the user profile. Differently from other Web-based RIAs, where ordinary users use video cams and mobiles to capture and send their contents, CODES should also provide mechanisms and tools to allow users (or groups of users) to edit, refine and publish their contributions and musical experiments – which we call Web compositions – in a process that we call musical prototyping.

"Prototyping" is not a common expression in music literature. There is a convention that musical composition is done by composers. There are people who are not formal composers in a strict sense but like to do music experiments, discussing and reinventing their own music. So, in our point of view, the product of this kind of cooperative musical experimentation over the Web can be called a Web composition and their authors, the Web composers, use and reuse sound samples reinterpreting and mixing musical styles. Furthermore, the emphasis of our work is mainly on the process (prototyping), and not on the quality or characteristics of the product itself. The cyclic nature of cooperation in CODES, where online partners refine a musical sketch until a form considered “final” is reached, clearly resembles incremental software prototyping cycles, and thus we call this process “music prototyping”.

Music prototyping is a simple process: first prototype creation, then prototype sharing – including prototype edition –, and finally prototype publishing. Prototype creation is typically an individual activity but each user does not need to worry about music theory. CODES offers a high level music representation and user interface features to allow easy direct manipulation (drag-and-drop) of icons representing sound patterns for music creation. The sound patterns are available as a sound library, obviously extensible to add new sound patterns (created by users or imported from other sound libraries). For prototype sharing, the prototype creator (called the “prototype owner”) can “invite” CODES members (searching other existing CODES users) or send invitations by e-mail to non-members, asking them for cooperation. If someone accepts the invitation of the prototype owner, she becomes a prototype partner, and she can edit the prototype like the owner does. While “editing” a musical prototype, any user may choose sound patterns and insert, delete and position them in the editing area. At any time, users can listen to the prototype and arguments may be linked to all decisions/activities made on it, in a structure similar to a design rationale structure. Thus all prototype partners may discuss and change ideas about each step of the prototype refinement, a good way to understand someone else’s decisions. In fact, arguments and modifications of a prototype are equally considered as typical contributions in a cooperative musical prototype. Also, users can send a “modification request” in order to make changes to other users’ contributions, which have to be approved. When everyone thinks the result sounds good, a “publication request” can be sent and the cooperative Web composition will be available on the CODES home page. This activity is named prototype publishing. As an alternative to publishing their music, users may export (download) their musical prototype as an MP3 file and share it as they want.

2.1 General Prototyping Interaction Issues

CODES exploits the interactive and cooperative aspects of music creation activities in a high level. The exploratory nature of the way people engage themselves in a process of musical experimentation suggests that such characteristics in user interfaces for music could yield great user benefits.

In contrast with YouTube “producers”, who upload ready videos, CODES “producers” can use CODES support to create, edit, browse, and manipulate their production. This implies a focus not only on community management (i.e., discovering, building, maintaining a community), but also on experimenting and constructing specific practices by using a suitable interactive practice vocabulary.

As any other RIAs, CODES extends the types of possible interactions on the Web. Examples include things such as real-time filtering with sliders, configuring objects with drag-and-drop, and panning across large surfaces or zooming in on images. One example can be seen in CODES Editing Window, where users drag sound patterns (instrument icons) from the sound library and drop them to the editing area (Fig. 1.). Such actions also require new kinds of mechanisms and controls compared to normal Web navigation.

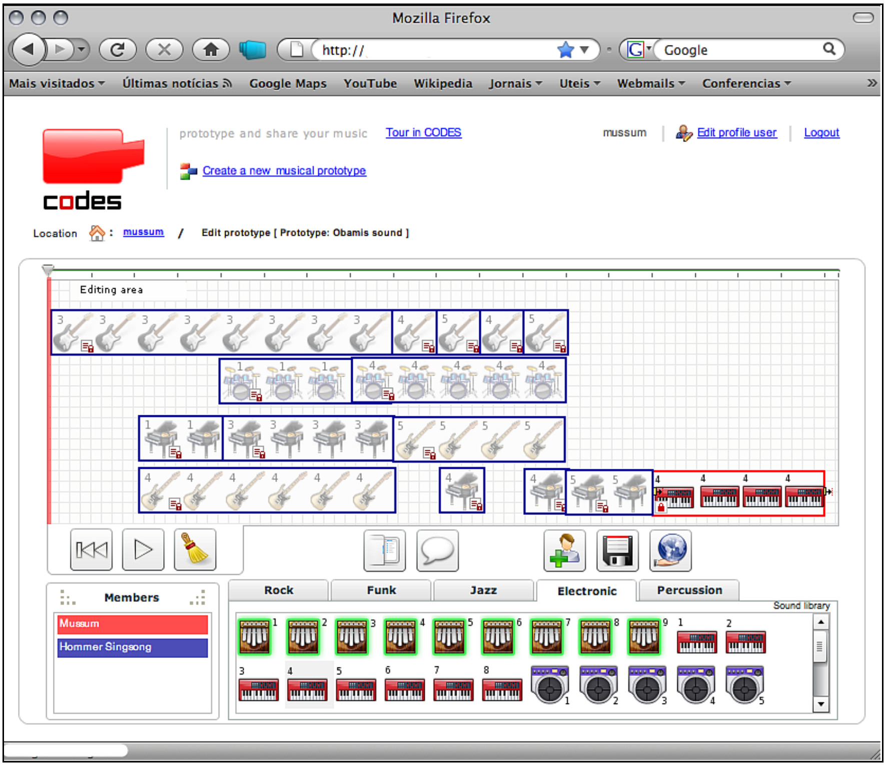


Fig. 1. CODES editing window

3 Designing for Web Composers

Since we consider the Web composers as ordinary users – novices in music –, some aspects of the interface and interaction should be specified and adapted to allow these users to perform the task of music experimentation. In this section, the requirements of user interfaces for musical activities (including networked ones) are investigated and discussed, particularly focusing on the necessary distinction between interfaces for musical activities and interfaces for musicians.

Usually, computer music systems are designed for experienced musicians, and with rare exceptions (e.g. the networked music systems PitchWeb [6], Daisyphone [2], and PSO [1]) they require previous mastering of specific skills and knowledge of specific concepts for a better use. Besides musicians, novices are also interested in creating music and participating in musical experiments, but they lack these abilities and also lack environments oriented to their profile.

If the intention is to design interaction so that a musical system can be useful and usable even to non-musicians, we believe this problem must be approached from a Human-Computer Interaction (HCI) perspective, combined with concepts from the fields of Computer Music, New Interfaces for Musical Expression (NIME), and even Computer Supported Cooperative Work (CSCW) if cooperation is also a requirement.

To investigate what should be a musical interface for novices, it is convenient to start by considering the context of use of traditional music software, including here its user profile (which is normally a musician or amateur musician). By doing so, we can understand why some of the features of interfaces for musicians are only suitable for that kind of user, and we may think about how to modify those features in order to suit also the non-musician profile.

First of all, musicians know music theory. They know how to read scores, the traditional music notation with its staff and musical symbols. Moreover, they know these symbols refer to concepts like notes, rests and tonalities – a novice may not even know what these musical concepts are all about! Even alternative notations (like tablature) contain alternative symbols for the same concepts, and the problem remains: these concepts are not part of a novices' world. Notation is a hard and non-intuitive concept for any novice to learn. At least, we must regard this as a true possibility when designing the user interface. In addition, musicians also have theoretical and practical knowledge about musical instruments, have access to them, and know the technical issues related to how to play them.

As a consequence of the above, usual music software often relies on traditional music representations and on metaphors from a musician's experience. The MIDI protocol itself, which is designed to interconnect digital musical instruments and computers, is based upon "musical performance events", like keys being pressed, changes in timbre and in tonality, tempo changes, etc. Even some more recent interaction styles (like for example the style adopted by IRCAM's Max/MSP [88]) are metaphors of something musicians are used to do, requiring experienced musician's knowledge and vocabulary, and they are consequently inadequate for novices.

3.1 Novice-Oriented Requirements

Usually there are some obstacles that make it complicated for novices to participate in music creation. In short, these obstacles are:

1. How to play music? Novices need to own a musical instrument and to know how to play it;
2. How to represent music? Novices need to represent the result of a creative process in order to repeat it later and to communicate it for anyone else;
3. Where and when? Novices need to have access, at any time, to places where musical activities and group meetings happen.

Based on our group's experience in applying HCI concepts to improve musical systems interfaces, we suggest some requirements to be taken into account when designing interfaces for musical activities in general, so to allow their use by novices as well as by musicians:

- Do not rely solely on traditional music notation, nor demand from users the knowledge of music theories and concepts for them to work with music. For CODES, we developed mechanisms to represent sound patterns as icons, and the option to smartly suggest them to the user, by offering him an easier access to those patterns which could fit well in his music prototype.
- Use musical metaphors from the real life, known by anyone, and not metaphors from a musician's reality. Such a metaphor needs obviously to include everyday

concepts and vocabulary, avoiding technical or specific terms from a musician's world.

- Use conventional interaction mechanisms. Prefer not to demand sophisticated interaction devices (like complex controllers, gesture interfaces, VR, etc.), but everyday technologies (mouse, keyboard, and usual audio features available on most commercial PCs).
- Avoid conflict with musical tasks (which involve sound), by avoiding sound feedback (apart from the sound being created, of course).
- Offer alternatives of music representation/encoding formats, making it easy for users to export/import their music between different systems.
- Don't forget other common usability requirements, which become even more important when focusing on non-expert users: easiness of learning, interaction flexibility, interaction robustness, and constant feedback [1010].
- Make the system multi-platform if possible, minimizing requirements of use and thus increasing user access (this is an architecture/implementation requirement, but it has an effect on system usability).
- For a cooperative system like CODES, a very important interface characteristic should be the users' possibility to perceive and analyze group members' actions on the object they are working on, and to know the reasons behind each one of these actions. These are aspects related respectively to awareness and rationale mechanisms, which then must be provided on the interface. See section 4 for more details.

This, of course, is a non-exhaustive list of requirements. Some are very obvious, but others are not so straightforward. Still, we see these requirements as very important ones, and in the next section we will discuss some social aspects of CMP in the Web.

4 Social Aspects of CMP: Roles and Activities

Music has been described as a social activity in which we share a musical experience [7]. Clearly, technology has created new social modalities for music listening, but we are convinced that Web technology also offers great contributions to social ways of music making, mainly with respect to cooperation and discussion among novices. Indeed, exchanging ideas is a sound way to know and perhaps understand the distinct points of views of all different users involved in a shared prototype.

CODES is a Web-based environment designed to support Cooperative Music Prototyping (CMP), with special focus on novices in music. But, differently from YouTube, Flickr, and even MySpace, we are also interested in providing ways for users to experiment with music, to contribute and discuss about it. For this reason, we consider CODES as a system for music design (group musical authoring) that allows novices to "compose-by-dragging-and-dropping". It is designed to introduce ordinary users to musical composition. Through CODES, non-experts may have the opportunity to be – like experienced musicians are – the actors of their own musical experiences. This means they can draft simple musical pieces – the Musical Prototypes (MPs) – that can be tested, modified, and repeatedly listened to, both by the first authors and by their partners, who will be cooperating in the refinement of the MP.

As this process emerges from the cyclic interactions of the group, based on contributions from/to each other, the "control" of the process is done by negotiation between members, without the need for the role of an explicit controller. Thus, the "decisions" are supposed to be consensual by negotiation, and not imposed by the authority of a leader. We believe that it is not necessary to make a distinct and explicit representation of the leader, because usually in a hierarchical group, the leader's opinions and actions may inhibit the other users' participation. Indeed, interactions can evolve as time passes, and the "more skilled" users can be recognized and respected naturally by the group while suggesting and justifying their contributions. This allows total flexibility without needing prior role definition, task allocation or responsibility assignment for members.

Awareness and conflict resolution are already considered critical issues in general CSCW systems. However, mechanisms existing in most of the related work do need some adaptation to take into account the idiosyncrasies of the CMP context. The ultimate goal is to provide actual cooperation, social knowledge construction, argumentation and negotiation among the different actors – most of them being novices in music – of MP design activities. This cooperation is supported here by a set of mechanisms, borrowed from the Software Engineering and HCI areas [33], [5], and specially adapted for CODES to handle awareness, music prototyping rationale, authorship, version control, and conflict resolution. In CODES, the notion of "authorship" is related basically to author's contributions inside a MP. Preliminary experiments have shown that users need to keep their contributions and compare with others' contributions, and with the new and past versions of their contributions, and it is also important to make explicit the evolution of the CMP process.

Our main adapted cooperative mechanisms are modification marks, argumentation and negotiation, version control with layers, contribution locking or selective edition, publication request, and modification requests. They enable group members to use and reuse contributions from several versions of a MP, without losing track of each individual author's ideas. Some preliminary results of initial experiments (see section 5) show us that CODES effectively provides support for cooperation between novices toward a collective creation of musical content.

4.1 Novices as Creators of Content: Social and Artistic Considerations

Since the involvement with music is naturally human, CODES aims to encourage people to use and create music, at least as a means of expression. We believe that making music is of more value for the individual than consuming it. Moreover, making music in a group context is a rich experience: collaboration, discussion and negotiation among users may also encourage reflection about social issues raised in CMP.

Currently, people freely share their creative works in the Web – music in this case – and they use anything they find online, with or without permission to download or use it. The Creative Commons [4], an organization that has defined an alternative to copyrights by filling in the gap between full copyright and public domain, comes in as an interesting option for non-technical products that emerge from cooperative creation as in CODES.

A discussion about collective ownership of an music prototype may occur: there are still some open questions to be answered, such as "who owns the result of my contribution?" and "how to give credits appropriately?" that highlight our challenge while designing the system. Please notice that, until now, we assume authorship as a concept related only to artistic contributions and not considering the legal aspects. Obviously, traditional copyright or intellectual property laws cease to be applicable, because they are too limiting. Interesting alternatives may be those of the Creative Commons initiative [4], and as far as it concerns the users, it will be made clear why they should "open their work to the world", at least to agree with the philosophy of the project. However, we are convinced that this discussion is just beginning.

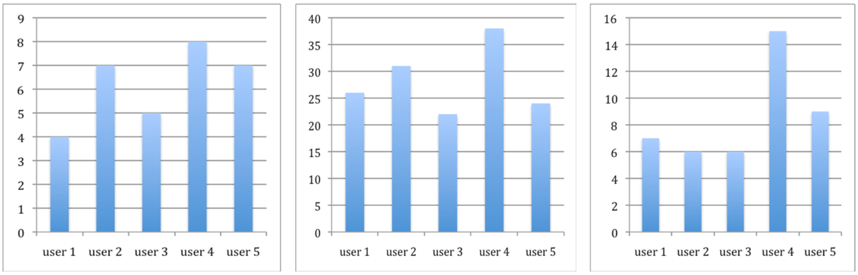
5 Some Preliminary Results

We have planned an experiment with five subjects in which the main task should be to create and publish a music prototype in CODES – for this experiment considered as an entertainment environment. This task included activities as to create a new music prototype, invite other users, decide about the style, edit, discuss, contribute and decide when the musical experiment was ready to be published.

All the subjects were people from the computer science field, with ages from 27 to 42 years and with no previous knowledge about music editing. They could participate during their free time in every place they wanted. This was usually done at university labs and at home in some cases. So, the Internet connection was always broadband and the sessions in average took about 30 minutes. During a period of 3 weeks the group interacted and discussed about the cooperative musical prototype they produced. It was about four to 8 sessions per subject in a total of thirty-one sessions. See Figure 2 for the general results of the test, where: 2.a shows the number of the sessions of each user; 2.b shows the duration of each session in minutes; and 2.c shows the number of contributions of each user.

The interactions were recorded in the system log file, containing the date, time, actions, and the duration of the session. After the MP was published, the subjects were asked to fill out a form with qualitative and quantitative questions related to the interactive aspects of CODES. The subjects should comment about their actions, options and decisions, as always as possible, while working in the CMP.

We analyzed the interrelationship between arguments with the log file and the questionnaire filled out by the users. For instance, user 4 does access the Web frequently to download music and read articles about musical groups and styles.

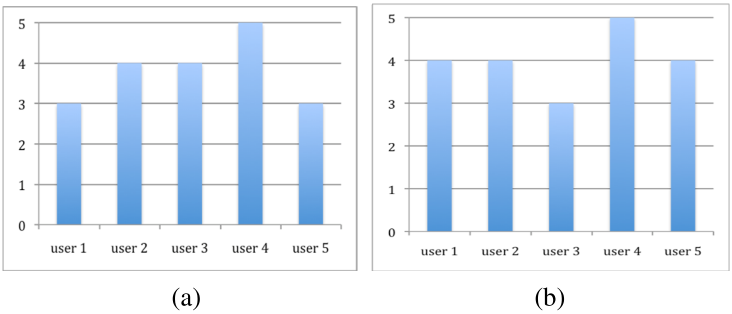


(a) (b) (c)

Fig. 2. General results about the number of sessions, duration and contributions

We can relate this fact to the highest level of interactivity of user 4 in comparison with the rest of his group (user 4 had the biggest number of sessions – 8 as shows Figure 2.a –, had the biggest online participation – 38 as shows Figure 2.b –, and has 15 messages/arguments – Figure 2.c). This kind of behavior reflects our expectative about the natural coordination inside the group, lead by interactions and argumentation, as mentioned in section 4.

All the users performed all the possible activities in the main task (editing, argumenting, sending modification requests and saving new contributions).



(a) (b)

Fig. 3. Excerpt of the results of the questionnaire

Figure 3 shows part of the results of the questionnaire, where each user should choose from 1 to 5 (the best rating) for each question. The questions about user satisfaction shown a positive user feedback about CODES as an easy to use system for entertainment (as shown in Figure 3.a), and so as a system with an effective mechanism of collaboration to experiment with music (see Figure 3.b).

On the other side, some users have mentioned the limitation of the sound library, the need for creating their own sound patterns, and even the desire of editing the existing ones. These points will be considered in the next system versions.

6 Conclusions

So, who are the Web composers? We are all potentially Web composers if we have available the appropriate support and tools to experiment with music, as the ones presented here. We are convinced that CODES stimulates the new Web user profile in which users are authors of their own culture.

While most of the related RIAs consider only producers and browsers of content, CODES allows collective music experimentation and stimulates group discussion about music creation, providing to the users the feeling of being the actors of their own artistic production as creators of content.

Our work is still in progress. The system is currently being deployed for evaluation in a restricted context. Initial evaluations have shown the system to be engaging for novices in music, but more systematic evaluations are forthcoming.

Future work will include the possibility of sound pattern editing by users, as well as creating their own, through sound recording on the client-side (with a computer microphone) and saving in the CODES server. We find this a very important feature to be implemented for the success of the cooperative music prototyping process.

References

1. Barbosa, A.: Public Sound Objects: a shared environment for networked music practice on the Web. Organised Sound 10(3), 233–242 (2005)
2. Bryan-Kinns, N.: Daisyphone: the design and impact of a novel environment for remote group music improvisation. In: DIS 2004, Cambridge, MA, August 2004, pp. 135–144. ACM Press, New York (2004)
3. Buckingham, S.S.: Design argumentation as design rationale. The Encyclopedia of Computer Science and Technology 35(suppl. 20), 95–128 (1996)
4. Creative Commons, `http://creativecommons.org/` (accessed, March 2009)
5. Dourish, P., Bellotti, V.: Awareness and coordination in shared workspaces. In: ACM Conference on Computer Supported Cooperative Work (CSCW 1992), Toronto, Ontario. ACM Press, New York (1992)
6. Duckworth, W.: Making music on the Web. Leonardo Music Journal 9, 13–18 (2000)
7. Gurevich, M.: JamSpace: designing a collaborative networked music space for novices. In: The 2006 International Conference on New Interfaces for Musical Expression (NIME 2006), Paris, France (2006)
8. Max/MSP, `http://www.cycling74.com/products/maxmsp/` (accessed, March 2009)
9. Miletto, E., Pimenta, M., Vicari, R., Flores, L.: CODES: a Web-based environment for cooperative music prototyping. Organised Sound 10(3), 243–253 (2005)
10. Preece, J., Rogers, Y., Sharp, H.: Interaction Design: Beyond Human-Computer Interaction. John Wiley & Sons, New York (2002)
11. Silva, P., Dix, A.: Usability – Not as we know it. In: HCI 2007 The 21st British HCI Group Annual Conference, University of Lancaster, UK (2007)
12. Turner, W., et al.: Information Infrastructures for Distributed Collective Practices. Special Issue of Computer Supported Cooperative Work: the Journal of Collaborative Computing 15(2-3) (June 2006)