5
Conclusions and Future Work

In this thesis, we present an extensible architecture that allows a multi-user stream-based interactive storytelling system to be available for different platforms. This architecture is currently working and two different clients are available for the Windows and Android systems. In this section we summarize our contributions and propose some future works.

5.1.
Main Contributions

We show how a multiplatform stream-based storytelling experience can happen. By using video streams, we move the CPU cost from the clients to the servers, allowing the architecture to support more platforms, since processors can be quite slow on low end Smartphones, tablets, and set-top boxes, for instance.

As shown in this thesis, the model can be extended for multiple platforms. This is a contribution towards reaching a bigger mass of users, in different environments. By doing so, the model can be applied in multiple mass storytelling applications, as the model can reach broader audience, using for instance, Digital Television.

Another important contribution of this work is the creation of multiple voting systems, inspired by literature, in order to handle a mass of opinions. The model is extensible in this aspect, but already provides multiple ways of solving the problem at hand: how to listen to the desires of a mass of users and then use these suggestions in a history.

These voting systems are also improved by the fact that the model supports harmonization of votes, that is, another contribution is that not only the framework supports handling multiple votes and ranking them, but also ways to combine them in the same voting sessions.

The user model in the approach in this thesis, even though simple, contributes for a more rewarding experience, both to the experienced users and to newer ones. This contribution helps create more interesting stories, since in the model, the user helps the authorial experience by interacting and changing the story. And if a user is a "good" voter (according to a give vote strategy that considers one as such), it means that these users help enrich the interactive experience.

The model presented here shows a new way of handling storytelling, combining multiple aspects together. There are no similar work in literature that presents a solution for this way of watching actual interactive stories. Even though there are excellent cases of single user storytelling systems, like Façade [11], and examples of massive multiplayer experiences like seen recently on [34] (which are not really interactive storytelling systems), none of these works are able to achieve what can be done by the presented model.

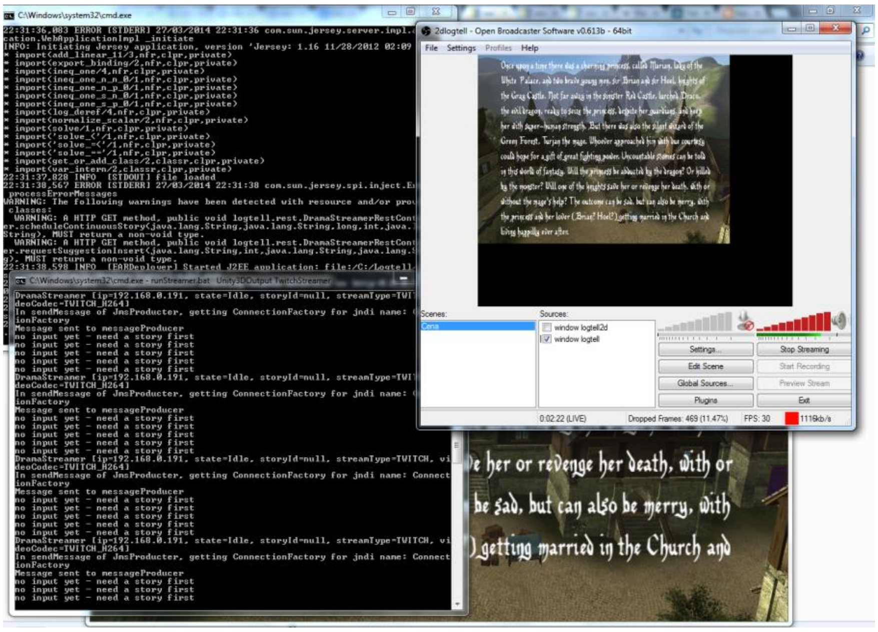

Figure 32 Streaming Server

The streaming server, shown in Figure 32, is one of the contributions of proposed model. It is capable of encoding and capturing the graphical dramatization of the story, be it 2D or 3D, and streaming it to clients. The presented model is extensible, as shown by already having 2 implementations: Twitch (using its network to stream to clients), and VLC (by direct streaming, from server to clients).

By combining the multiple issues at hand, this model is able to actually handle a mass of users watching the same interactive story. This can be said due to: the scalability of the model, which enables it to grow as needed, by adding extra servers; also by using already existing technologies that support thousands of simultaneous users, as in the Twitch implementation, by streaming to a central network of servers that already handles streaming the same video to all those clients The presented system can then provide mass experiences, based on logical coherence, while allowing them to vote and reach collective decisions that affect the story. This allows the creation of a new way of watching stories, in multiple devices, helping create the new concept of what can be the interactive television.

Another advantage of the proposed architecture is that it allows different types of Drama Streamers to be supported. That is, since the dramatization is taken from clients, it allows the same story to be rendered in different ways. In this particular aspect, a possible expansion of this architecture is to render the story in different graphical engines. This was a contribution of this work over Logtell’s previous version, where the dramatization rendering engine was replaced, now they can be attached and extended, shown as the simple 2D dramatization done after remodeling this aspect of the model.

Figure 33 Multiple Clients

Figure 33 shows how multiple clients can connect to the same story. Even when using a netbook, a Tablet and a Smartphone, the new model can display the same story simultaneously (with some delay). Although only a few simultaneous devices are shown in this photo, they are connected to Twitch's network that already supports thousands of simultaneous users. This shows how powerful this model can be for creating massive experiences. On the other hand, the model can also run on a cluster of scalable streaming servers, if so desired, by using the VLC implementation (or by further implementations of the model, for instance, using proprietary encoding hardware).

By making use of open patterns like REST and JSON through HTTP for the client server communication, the architecture presents itself as portable and extensible. The use of those patterns in APIs seems appropriate for this goal because there are multiple libraries in various platforms available to parse and process them. In some platforms, there are even native versions of those libraries, providing easier portability.. For instance, JSON is directly implemented in Javascript, where it origins from – thus making possible the creation of a new story client in pure HTML5, for instance.

5.2.
Future Work

For future works, different strategies of encoding will be investigated, for instance, generating multiple streams for the same story, in different formats and resolutions. With this alternative, the story clients could then have multiple choices that could be more appropriate for different platforms and screen sizes.

Regarding multiple platforms, more clients and different streaming servers can be created. For instance, Brazil’s GINGA [17][18] platform shows itself as a promising environment, as it is to be supported by the terrestrial TV broadcasting network. An implementation for digital television platforms is possible with the presented model, allowing it to reach a mass of simultaneous users. Future work can be done, by encoding the model's video stream into proper codecs and streaming the result through Digital Television for instance. In fact, the system could even be used for analog TV, by broadcasting the video of the story and creating another interface to capture votes, as it was done in the presented prototypes using Twitch. There are many possibilities.

As research expands, more tests will be made regarding usability tests. Performance is also an important question, so for a real world use case, different optimization alternatives will be pursued, and tests with more powerful hardware should also be carried out.

Furthermore, it should be noted that since the architecture supports multiple platforms, the forms to interact can be expanded upon the different interfaces that they support. For instance, the mobile version could support voice recognition and motion sensing. This will be considered in our future works.

As the model aims to support multiple platforms, it should also possible to coordinate multiple streams of multiple qualities for the same story. Research can expand further in this direction, by creating coordination and adaptation techniques in the architecture, to allow, for instance, to let users into different groups and show them the same story stream according to their profile, preferences and votes, or even split them mid story, thus allowing groups with more affinity to have a better experience. Also by further extending the model, clients should be able to automatically choose the best video stream for their specific devices.

An important contribution of the present thesis is the model for sharing massive interactive stories. As far as we are aware, no other work presents a model with strategies that are different from the simple “most voted suggestion”. The implementation and evaluation of all strategies we have proposed in the presented model are currently under development by our research group.

The question of voting and streaming delay can also be handled by future research: votes can be done after voting sessions, due to network and video delay issues that can occur. For such, the model can implement different strategies on how better synchronize information, and regardless, treat these interactions even when they would be lost. For instance, votes could always be analyzed for defining user preferences, also the timing of the votes can provide meaningful information - the moment where users decide to interact can be in itself input data for story generation process and user information inference.

One idea for future work is to improve the user model. For instance, by keeping track of users votes, the system could predict automatically what vote is more likely for that user profile (or group). In this case, even a non-vote could be significant. Also, future research can be done regarding voting weights. There are many possibilities on how they can be calculated in an automatic way. For instance, by analyzing user voting history in different ways, using different algorithms, in order to define how much a given user's votes should matter. One example would be using simulated annealing to modulate story tensions, and apply this as a criterion for votes and related emotions.

More work can be done also regarding the different ways groups are formed, and how they desire to vote. The model is configurable, and thus allows implementing multiple ways of voting. It is hard to know which is ideal, because that may depend on the target audience. Still, more voting methods can be done following the model's blueprints and more experimentation can be done with bigger masses of users.