

Ananda Jacobs
ISE 599
Paper review
February 26, 2004

Software Architecture for Immersipresence - Alexandre R.J. Francois

Francois begins the article by introducing a new term: immersipresence. Immersipresence refers to immersive, interactive applications that necessitate the mixing of multimedia data such as live input or prerecorded audio or video. The example he cites is that of teleconferencing. In this case, the Immersipresence version allows for a much closer interaction between participants. Especially intriguing was the possibility of virtual eye-contact; this is an important psychological factor in communication, and therefore has the potential to be a very useful tool.

The purpose of the paper was to present a new model, the Software Architecture for Immersipresence (SAI). Software Architecture is a relatively new field, according to the author. Among the challenges faced in creating them is that a particular style may be more appropriate to one application, but less suited to another. So, my interpretation of this is that there is a challenge of finding an appropriate style for instances where many different types of dynamic media are going at once. A fundamental concept in such systems is *distributed parallel processing of generic data-streams*. So, SAI strives to provide a method for distributed algorithms, so that they can be integrated under one system. I imagine it as being like a umbrella, one that is able to encompass each component of Immersipresence. (But then, without a background in software engineering, my interpretation may be skewed.)

Francois brings up the important point of there being a lack of cross-disciplinary system integration. Most algorithms, he says, are designed with a narrow field of application in mind. This is where SAI fills in the gap; it addresses a multitude of components, as stated above.

Some of the graphics Francois uses are hard to follow, while others are insightful. The first one he presents summarizes the goal of Immersipresence: to combine high bandwidth (lots of data) with low latency (good real-time interaction)—the best of both worlds.

The implications for SAI are far-reaching. Its versatility is impressive, as it can be used for projects ranging from a single video processing to an integrative session involving live sound, images, and online video streaming. Francois presents a number of applications, in order of complexity. MuSA.RT is included as an example, and is cited as using four independent data streams: MIDI input and event processing, real-time tonal analysis, rendering of Spiral Array structures, and gamepad or camera manipulation. The author cites the SAI architectural style as being potentially important to research, education, and industrial projects.

Overall the article was organized very well, and the only problem I encountered was the very in-depth technical explanation of how SAI works. However, the context surrounding the explanation provided me with enough of a general background to garner an appreciation of the usefulness of this tool.