

# DeepTalk: A complete conference in a picture.

## Abstract

Particle physics conferences lasting a week (like CHEP) can have 100's of talks and posters presented. Current conference web interfaces (like Indico) are well suited to finding a talk by author or by time-slot. However, browsing the complete material in a modern large conference is not user friendly. Browsing involves continually making the expensive transition between HTML viewing and talk-slides (which are either PDF files or some other format). Further the web interfaces aren't designed for undirected browsing. The advent of multi-core computing and advanced video cards means that we have more processor power available for visualization than any time in the past. This poster describes a technique of rendering a complete conference's slides and posters as a single very large picture. Standard plug-in software for a browser allows a user to zoom in on a portion of the conference that looks interesting. As the user zooms further more and more details become visible, allowing the user to make a quick and cheap decision on whether to spend more time on a particular talk. The project, DeepConference, has been implemented as a public web site and can render any conference whose agenda is powered by Indico. The rendering technology is powered by the free download, Silverlight. The poster discusses the implementation and use as well as cross platform performance and possible future directions. A demo will be shown.

<http://www.codeplex.com/DeepTalks>

## What Motivated This Conference Viewer?

I can't travel to as many conferences as I'd like and I was looking for a way to quickly browse conferences. Something that would allow me to quickly look at a conference over a lunch break, examine a talk or two, quickly load a talk, glance at a few slides, etc.

The current state-of-the-art is pretty good—and indico agenda, PDF's or PPT's. But that interface isn't built for quick glances—it is built for deep and careful examination. Or presentations.

I had no idea how to go about this until I saw the Hard Rock Memorabilia site. They had a collection of very high resolution images—small thumbnails of which are all displayed at once in your web browser window. You then use the mouse wheel to zoom in progressively closer and closer. As you zoom in higher resolution versions of the images are loaded. This is all done seamlessly—the effect is quite smooth. Once I discovered that the tools required for this were all free (both rendering and display) it was only one or two conversations with my wife (another physicist) before the basic idea of what you see here was born.

I was also interested enough to devote weekends and evenings to this project because it tickles another one of my favorite hobby horses: HEP is taking very little advantage of the latest in image processing and the web. Mostly we are doing what we did before, but with more cores—this is an attempt to re-purpose some of the cool technology out there for something for science.

## What Is Deep Zoom?

The algorithm is fairly common, actually. Most web mapping programs use it. The iPhone uses it. As you zoom in on a map, the map web site loads more detailed images of just the area you are interested in. Zoomed out you may only see major roads and land features. The closer you get the more you see of city streets or even houses.

Microsoft implemented this algorithm in their Silverlight 2.0 technology-called it Deep Zoom. The high resolution image is rendered as a single low resolution image, 4 higher resolution images, and then 16 very high resolution images. The web browser is instructed to load only the relevant images. Zoomed out it will load only the Level 0 image, and as the user zooms in it will select the appropriate images from Level 1 and Level 2.

Instead of the picture of the flower, think of all the slides of a conference laid out on a big floor. That is the high resolution image. A decent length talk rendered this way can require almost 20 levels when rendered at 200 DPI. Each tile is configured to be 256x256 pixels. When the jpeg is written with a 95% quality level the sizes vary between 4 and 30 KB each, depending on complexity—small files that can be quickly downloaded over the internet.

Once the images arrive at the web browser, Silverlight stitches them back together and scales them correctly to appear in the user's web browser.

## The Technology

There are only two key components that are being used - everything else could be replaced with little effort.

- Silverlight 2.0.** This is Microsoft's competitor to Adobe's Flash/AIR. The key thing is the *MultiscaleImage* control. This control does the heavy lifting on the user's computer: it blends images as the user zooms in, it coordinates the http requests for the base images, and it stitches the tiles together. Silverlight works on Windows and Mac OS X and in most modern browsers like IE, Firefox, and Chrome (not Opera). Linux is coming as Novel gets their open source *Moonlight* project off the ground.
- Composer Tools.** This is an application and a library that can be used to generate the Deep Zoom images. As input it takes a data structure that describes the layout of a collection of images, and the image files themselves.
- Silverlight is the only thing used on the user's computer. On the render computer a range of software is required to turn an agenda into a Deep Zoom image. There is no reason this toolset couldn't be expanded in the future:
  - Custom code downloads and parses an indico agenda in XML format. The information in the Agenda is fairly free form—for example it is hard to get all talks but avoid papers. Worse, MIME-Types are only set for some sub-set of document types. The system currently doesn't render PowerPoint 2007 pptx files because the MIME-Type returned by most indico servers is "unknown".
  - PowerPoint* is used to turn a ppt file into a PDF file. I do see a few crashes with PPT that contain unknown picture formats (mostly PPT generated on Apple computers).
  - The *GhostScript(GS)* program has the job of turning a single talk (pdf or ps format) into a sequence of jpeg images. There are lots of problems with GS:
    - Crashes on lots of PDF files that Adobe Acrobat renders correctly.
    - Doesn't always get talk rotation correct; I've got code to fix up most cases and re-rotate the image. However, it isn't perfect.
    - Has trouble with funny DPI settings (e.g. output from the TeX slide package). Sometimes adds oceans of whitespace around the outside of the talk content.
    - I no longer think of PDF as a standard. :-)
    - There is no reason why other file formats couldn't be rendered—the current architecture isn't extensible, but that is a reasonably simple change.
  - Layout is done by some custom code—a simple set of in-memory transforms. No images are loaded to accomplish this, so this is quite memory efficient.
  - Rendering is done by the *Composer Tools* referenced above. Each talk is rendered separately
    - If a single talk's rendering causes problems it can be dropped
    - Allows for possibility of rendering talks as they are uploaded without wasting CPU time to re-render every talk.
    - Rendering can be a memory hog. A talk with 50 slides can use more than 4 gigs of memory. So talks are split into sub-parts if they are too big in order to avoid this.
    - This tool is constantly being updated by Microsoft; its performance varies from release to release...

Finally the server is built out of several off-the-shelf server products. There is no reason this couldn't have been served from a Linux box using PHP, for example (or any other framework or platform). A very nice thing about Silverlight is that it can be distributed as static content from any server platform. Indeed, once the talk is rendered this is very much just that—static content.

- Windows Server 2008 and IIS 7.0 running on an old Athalon 3000+ with 2 gigs of ram.
- Web site is written with Microsoft's MVC (model-view-controller) framework. This is a framework that makes it easy to write *testable* web pages.
- Web services are written in Windows Communication Foundation (WCF). Both http and TCP/IP end points are exposed. Wire format is currently SOAP, but a trivial change in a config file will make them REST.
- Database is a simplistic Compact SQL Server 2005. The server is backed by the *Entity Framework* that allows for easy mapping between an OO framework and a relational database. It includes the ability to read only what is necessary from the DB as opposed to the full structure. This projects database requirements are very simple at the moment.

## Resources and Performance & Conclusions

This system is a resource hog—it requires high bandwidth network, lots of memory, and lots of disk space. Most of the tools worked well for the project, though some of them are more complex than they need to be for this project (i.e. the webservice infrastructure). It was very nice to be able to one program end-to-end in a real programming language (C#) - from the server (CLR) to the web browser (Silverlight).

There are a lot of small issues that need to be fixed—the long tail. Since this is a web project I'll slowly update the system and make it more robust over time. It would also be nice to take advantage of multiple cores to process talks more quickly. And render conferences not using the indico system. There are lots of user possibilities for user interface improvements. Probably the biggest one is being able to move from slide to slide within one talk. Or bookmarking a spot so you can send it to someone.

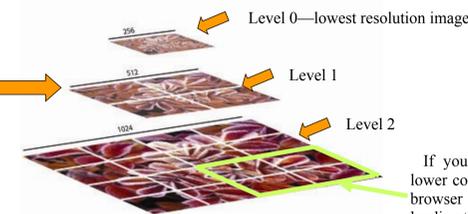
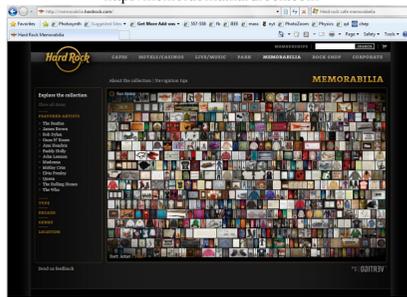
I now use it regularly; so in that sense it is a success. Robustness must be improved so I can ignore it until I'm ready to work on it, however. The code is open-source—so anyone with an idea can help!

## Gordon Watts

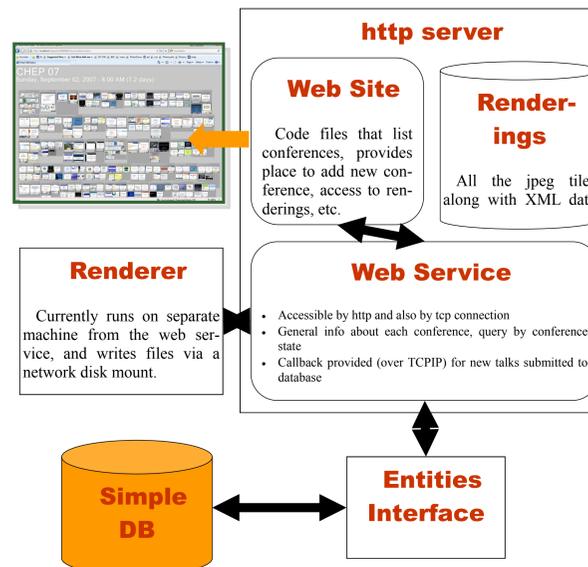
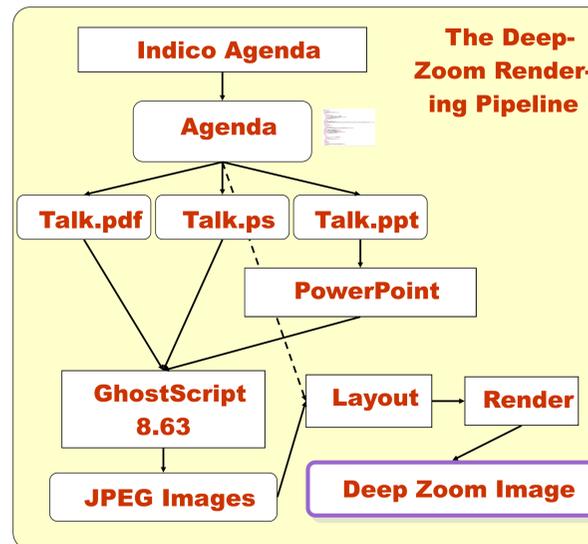
University of Washington, Seattle



<http://memorabilia.hardrock.com>



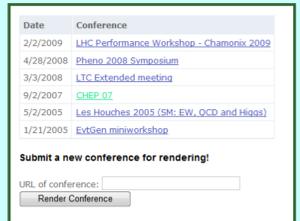
If your view is only this lower corner, why should the browser waste bandwidth loading the rest of the image?



<http://deeptalk.phys.washington.edu>

## Jump Off Point

The address <http://deeptalk.phys.washington.edu> lists the most recent 20 conferences and you can enter the URL of any conference you wish to view. Clicking on a link or entering an already known URL will take you to the initial view of the whole conference. If you enter an unknown URL, DeepTalk will queue the rendering request.

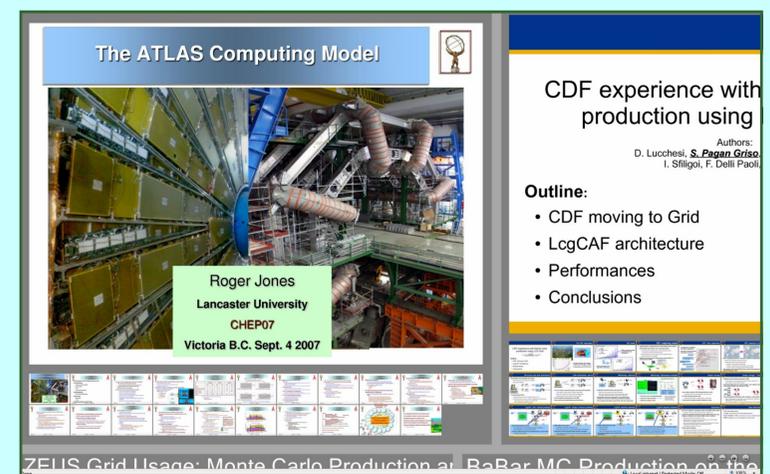
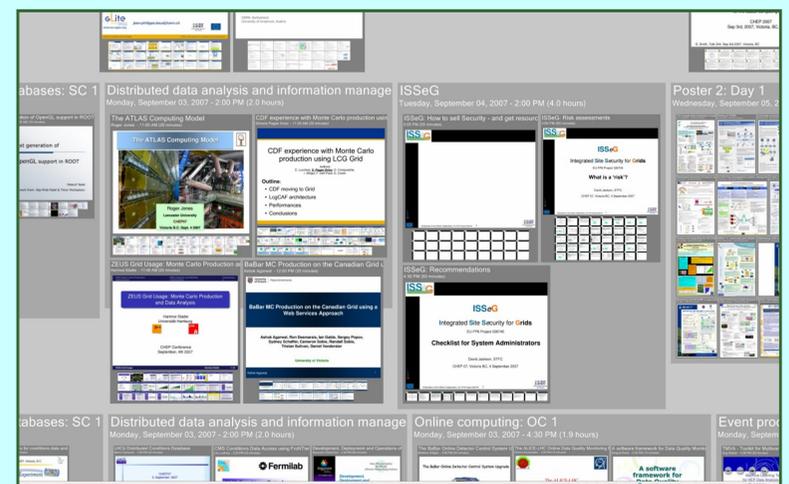
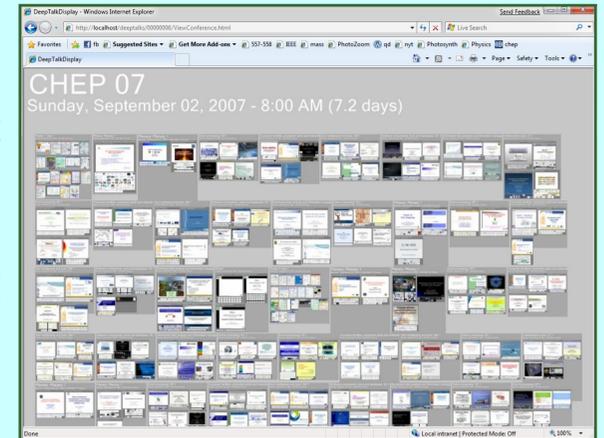


## Conference View

The complete conference is rendered as a single picture. You can zoom in (or clicking on the image or using the "+" and "-" buttons). Panning is accomplished by click-and-drag with the mouse or the arrow keys.

It is possible to quickly zoom in, out, and pan around and explore all the talks in the conference. The layout should make it easy to explore what you want to look at quickly, allowing your eye to determine if you want to delve deeper into a talk.

Below is a quick example of zooming into a conference.



## Viewing Talks

I have found the following makes this most useful to me:

- Make browser full screen
- Turn off status bar at bottom of browser so you have nothing-but-talk on the screen.
- Use a mouse with a scroll wheel—zooming in and out is much faster!
- Use a high speed internet connection. Lots of small jpeg's have to be moved from the server to your browser!
- Sit back and explore!

