

Evaluating the Flash platform for web-based collaborative data visualization

Advisors: Andy Johnson (UIC),
Franca Garzotto (Politecnico di Milano)

Thesis Objective:

Test the effectiveness of **Adobe Flash** as a foundation platform for web-based visualization systems

- **Practical approach:** develop a framework and an application to test Flash capabilities in a real visualization context
- Evaluation will consider both end user and developer needs

Scientific Visualization



- Definition:

Transforming and viewing data as images (or other sensory forms) to gain insight into the data.

Visual representation of data is

- **Powerful:** allows to quickly understand the data and discover its features
- **Foundamental:** in the case of big datasets.

Visualization Systems Users

- Scientists / specialized people
 - Need powerful, custom tools for in-depth analysis of experimental results or observation.
 - Have specific training for tool usage and knowledge of data
 - Mainly interested in ***functionality***



Visualization Systems Users

- Scientists / specialized people

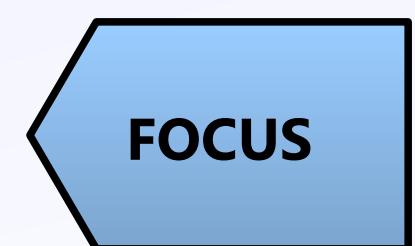


- Need powerful, custom tools for in-depth analysis of experimental results or observation
- Have specific training for tool usage and knowledge of data
- Mainly interested in ***functionality***

- General public



- They are interested in analyzing 'simpler' versions of the data, i.e. Results of work done by scientists
- No specific knowledge / training
- Need simplified access for tools and data
- Interested in both ***functionality*** and ***experience***



Scientific visualization and the Web

- Web is a powerful way to publish and distribute information.
- Web can be used to publish both data and applications that elaborate / visualize the data.



Visualization Systems for the People: Requirements

- Simple access
- Ease of use
- Collaboration
- 3D support
- Rich interactive experience.

Technological requirements

- A candidate platform for visualization system development needs:
 - Diffusion
 - 3D graphics support
 - Interface design and customization support.
 - Advanced communication layer
 - Easy development process

Web Technologies

- **Proprietary technologies**
 - Diffusion near 0%

- **Proprietary technologies**
 - Diffusion near 0%
- **Java + JOGL**
 - Quite powerful
 - Users have to download the JOGL host applet
 - JOGL is low level, compared to other alternatives

- **Proprietary technologies**
 - Diffusion near 0%
- **Java + JOGL**
 - Quite powerful
 - Users have to download the JOGL host applet
 - JOGL is low level, compared to other alternatives
- **VRML / X3D**
 - Pretty limited diffusion
 - No UI Support

- **Proprietary technologies**
 - Diffusion near 0%
- **Java + JOGL**
 - Quite powerful
 - Users have to download the JOGL host applet
 - JOGL is low level, compared to other alternatives
- **VRML / X3D**
 - Pretty limited diffusion
 - No UI Support
- **Shockwave**
 - IDE and language are somewhat limited

- **Proprietary technologies**
 - Diffusion near 0%
- **Java + JOGL**
 - Quite powerful
 - Users have to download the JOGL host applet
 - JOGL is low level, compared to other alternatives
- **VRML / X3D**
 - Pretty limited diffusion
 - No UI Support
- **Shockwave**
 - IDE and language are somewhat limited
- **FLASH**

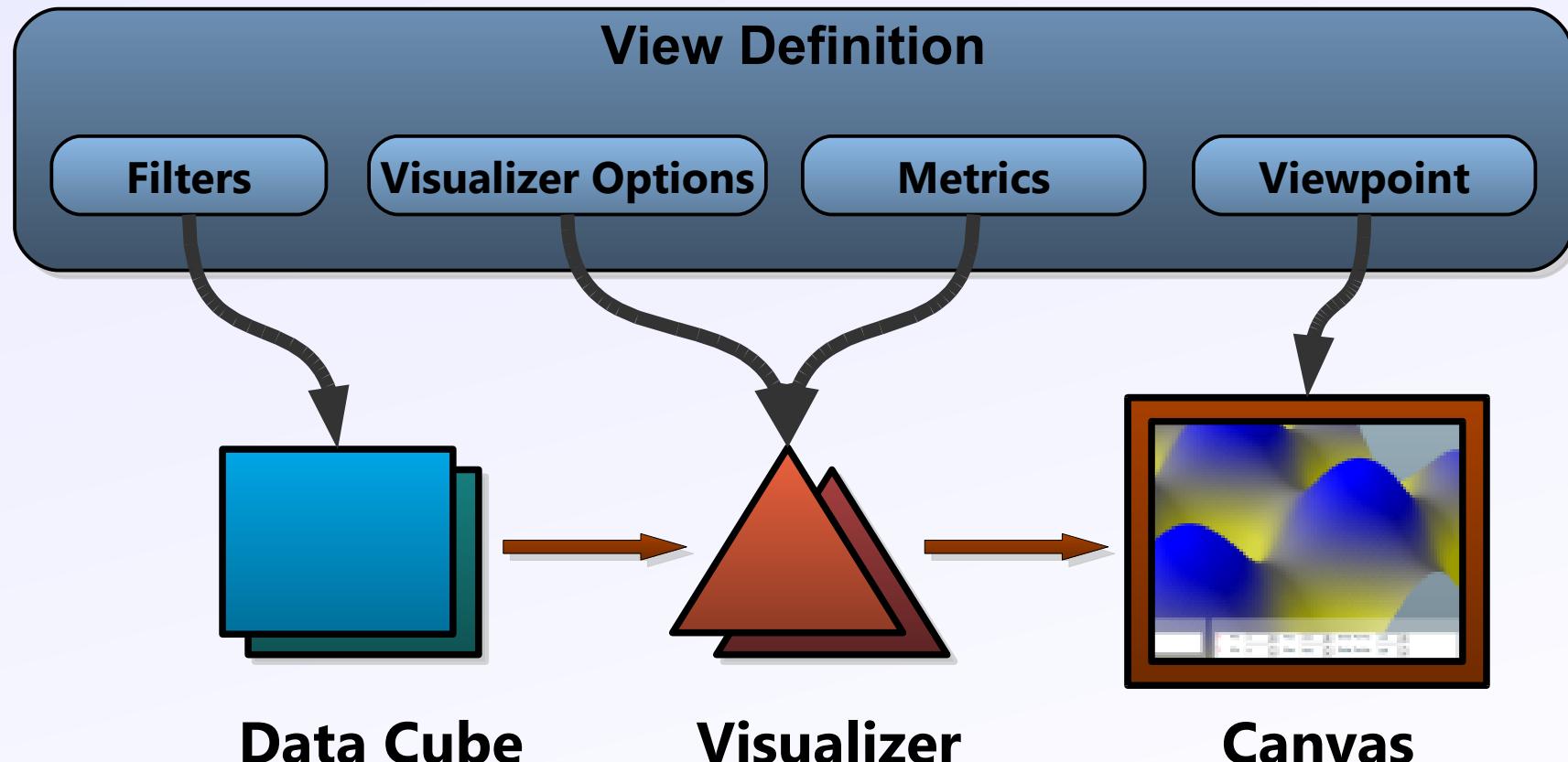
- Extremely diffuse (98%)
- Explicitly designed to deliver interactive graphics over the web
- Powerful yet easy programming language (ActionScript)
- Customizable UI implementation through *FLEX*
- Extensive data access and remoting capabilities
- Major disadvantage: content rendering is done by a **software rasterizer**

Executive Summary

- Define a conceptual framework for web-based collaborative visualization
- Implement a framework prototype using Flash
- Develop a proof-of-concept application on top of the framework
- Evaluate the platform choice wrt the identified requirements

Conceptual Framework

- Core of the framework: the **Visualization Pipeline**
 - Standard dataflow approach
 - Integrates interactivity and collaboration into the model

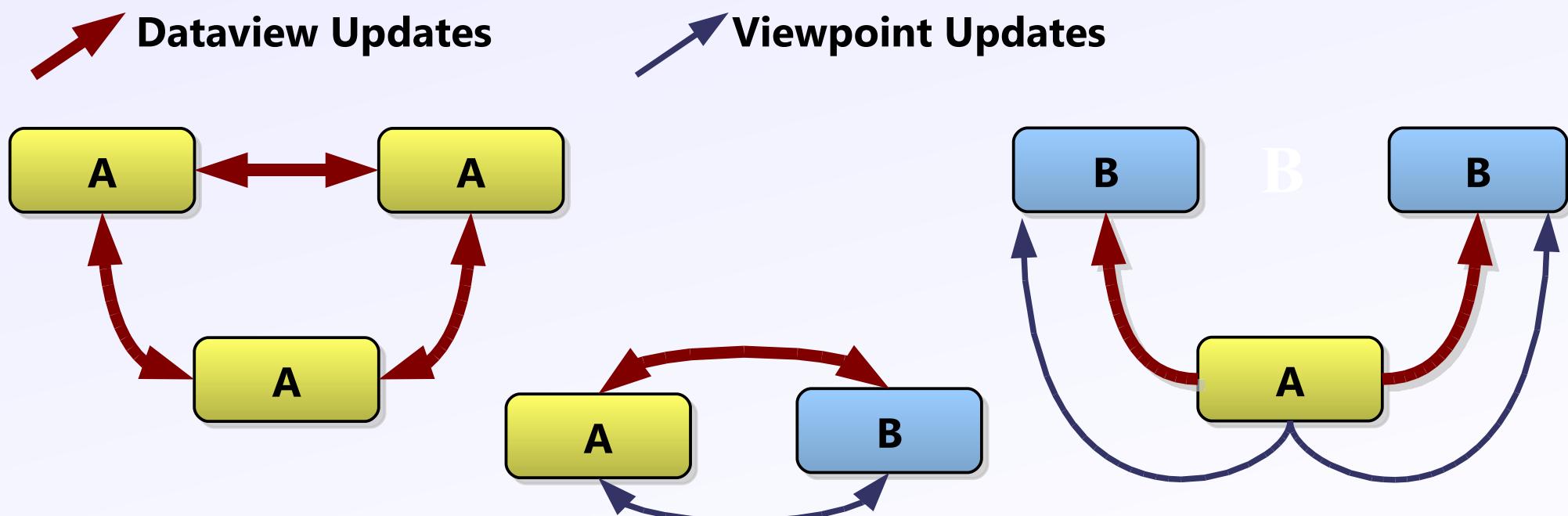


Collaboration

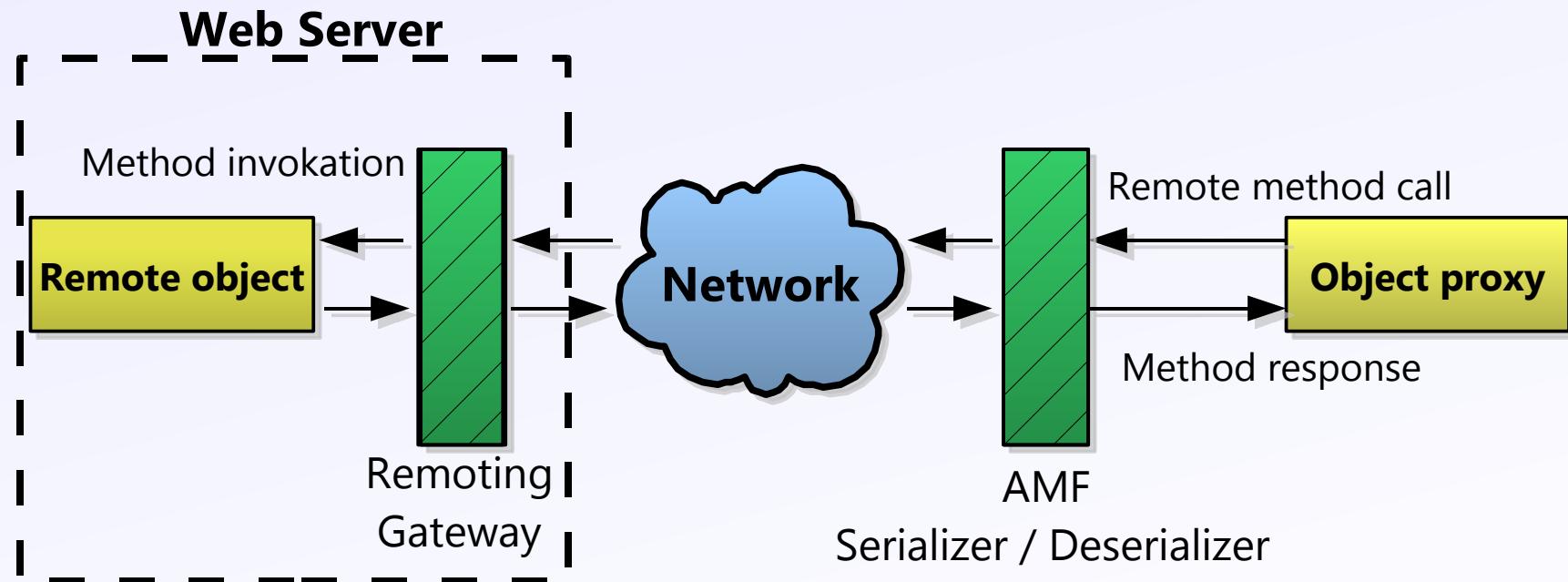
- View definitions are lightweight objects: collaboration is based on the **exchange of view definitions**.
 - through view definitions a client can render a precise instance of a view.
- View exchanges are mediated by a **Workspace Service**.

Collaboration (continue)

- **Filter policies** can be applied to view definition exchanges.
 - **Partial view definitions**
 - **Dataview updates:** filters, metrics, visualizer options
 - **Viewpoint updates:** viewpoint only
 - **Input / Output policies**



- Prototype developed as a set of ActionScript classes that directly implement conceptual framework entities.
- **Server side:** implemented in C#, hosted on a ASP.Net enabled Web server.
- **Communication:** Flash Remoting using a server side *Remoting Gateway*



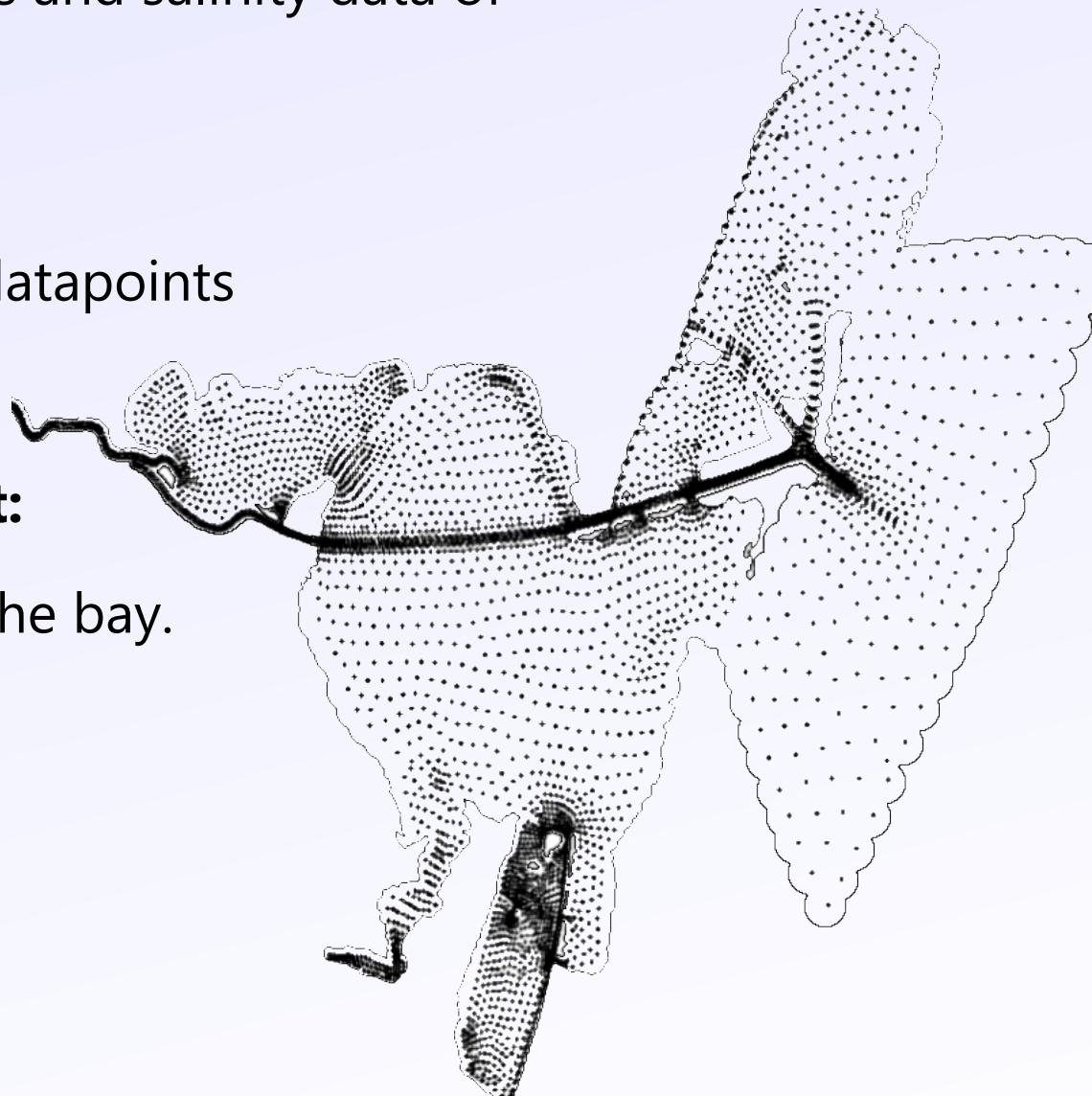
Hydroviz

- **Purpose:** visualization of currents and salinity data of Corpus Christy Bay, Texas.

- **Source dataset:** about 3 million datapoints

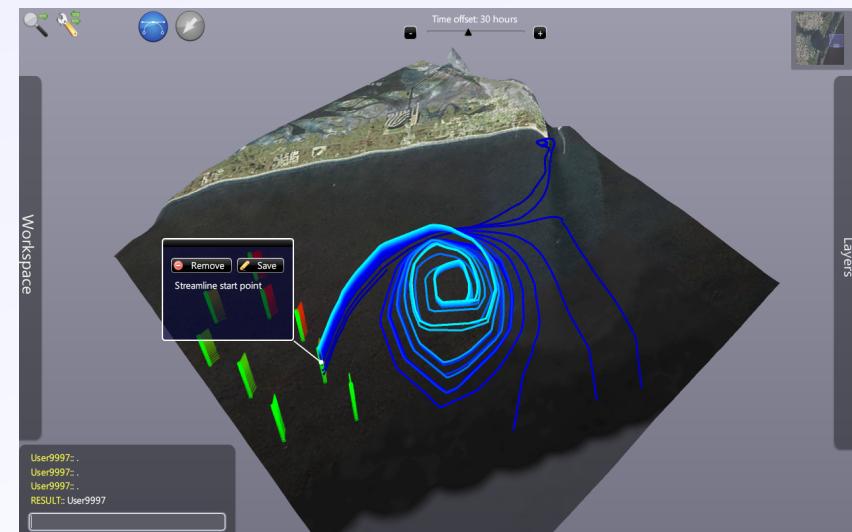
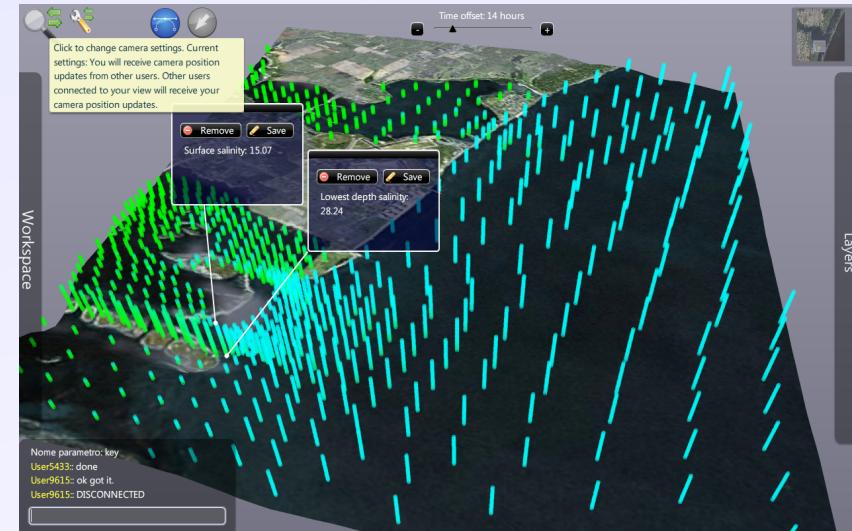
- **The application should support:**

- Rendering of a 3D model of the bay.
- Multiple data layers.
- Collaborative



Demo

- **Usage Scenarios:**
 - Basic Model Exploration
 - Datapoint Visualization
 - Collaborative Work



- **GOAL:** Evaluate the effectiveness of Flash for development of web-based visualization applications
 - **Input:** Implemented framework and application.
 - **Stakeholders:** developers and end users.

Discussion

- **Diffusion:** clearly satisfied by Flash.

Discussion

- **Diffusion:** clearly satisfied by Flash.
- **Ease of use:**
 - Simple interaction required to access most functionalities.
 - Not directly related, but **supported** by technology.

Discussion

- **Diffusion:** clearly satisfied by Flash.
- **Ease of use:**
 - Simple interaction required to access most functionalities.
 - Not directly related, but **supported** by technology.
- **Collaboration:** support was straightforward thanks to conceptual framework design **and** flash communication infrastructure..

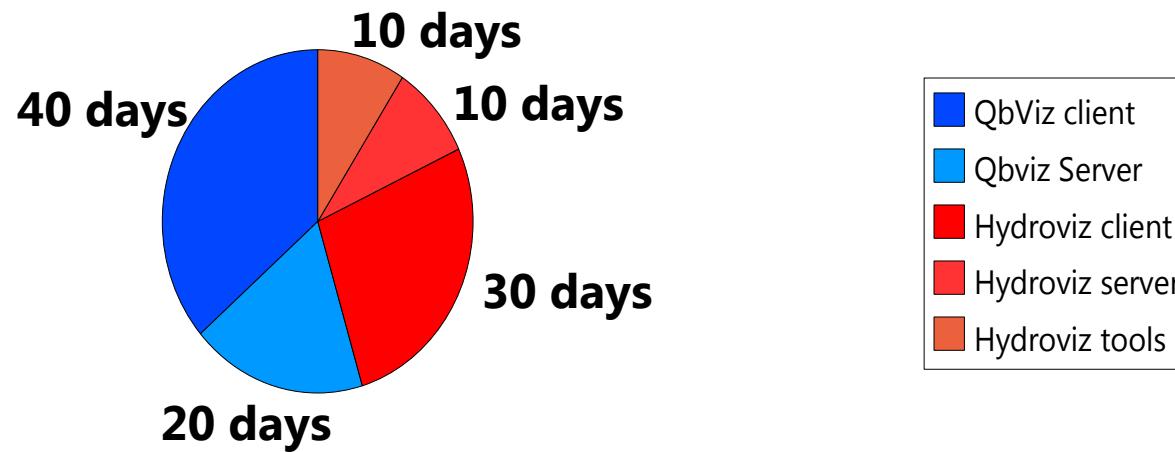
- **Diffusion:** clearly satisfied by Flash.
- **Ease of use:**
 - Simple interaction required to access most functionalities.
 - Not directly related, but **supported** by technology.
- **Collaboration:** support was straightforward thanks to conceptual framework design **and** flash communication infrastructure.
- **Enjoyement:**
 - Difficult to assess, needs thorough user testing.
 - Even if testing would suggest changes, they will be easy to implement **thanks to the platform.**

- **Diffusion:** clearly satisfied by Flash.
- **Ease of use:**
 - Simple interaction required to access most functionalities.
 - Not directly related, but **supported** by technology.
- **Collaboration:** support was straightforward thanks to conceptual framework design **and** flash communication infrastructure.
- **Enjoyement:**
 - Difficult to assess, needs thorough user testing.
 - Even if testing would suggest changes, they will be easy to implement **thanks to the platform.**
- **3D Visualization:** possible but limited.

Development Effort

- Implementation time: ~3½months; LOCs: ~10'000.
- 55% framework, 45% app.
 - Hopefully will get better.

Development Time

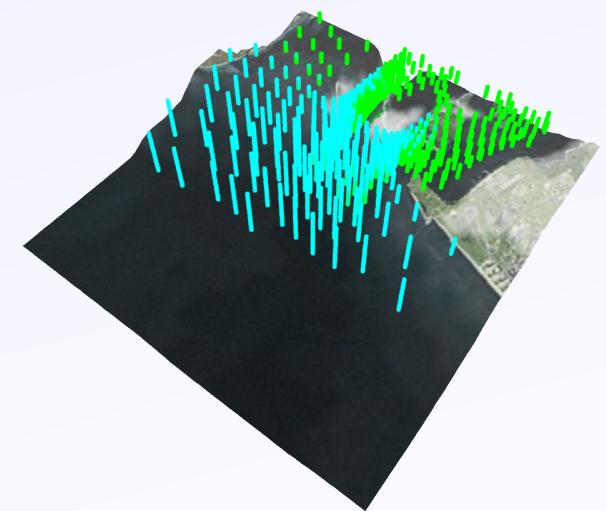
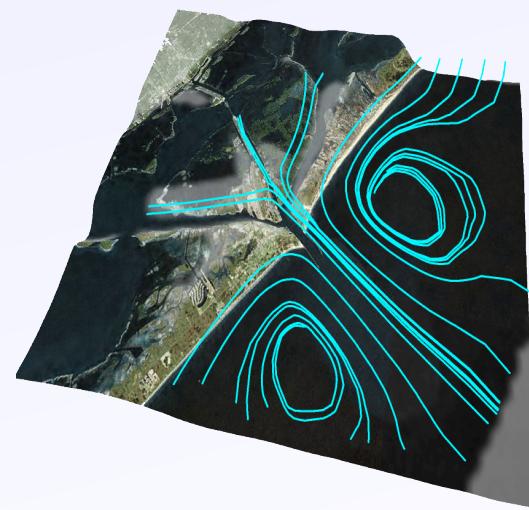
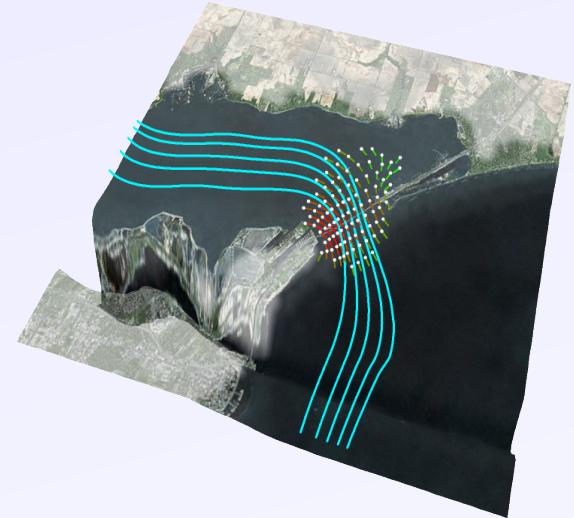


Flash can be considered a viable foundation platform for web-based visualization systems

- **NOT** a substitute for every other technology
- Final choice depends on the exact visualization scenario
 - General Public
 - Limited complexity

- **Hydroviz**

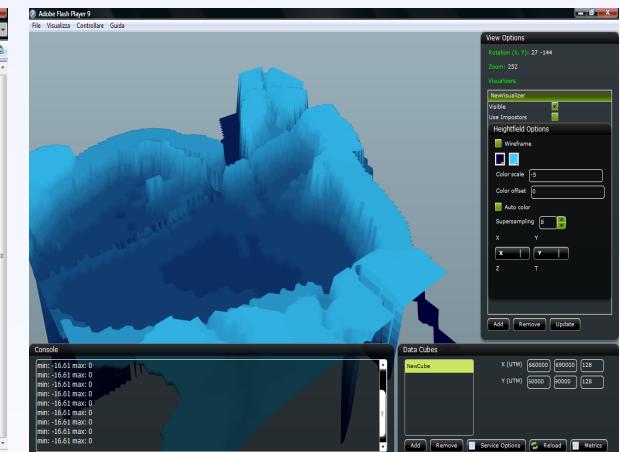
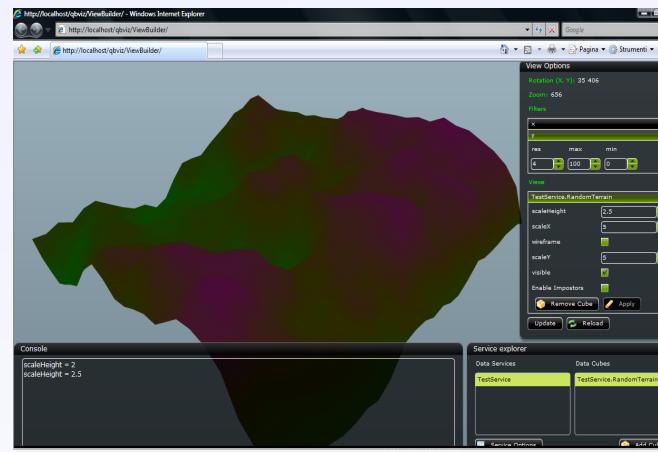
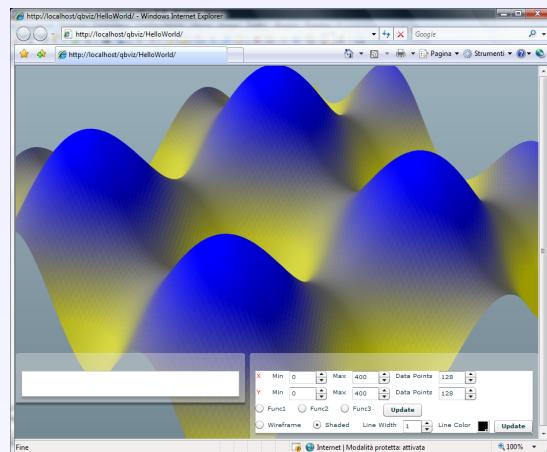
- User testing and validation
- Add data playback support
- Salinity isosurface visualizer



Future Work

- **Framework**

- Integrate a visualizer library
- Add *View Serialization* support to XML
- Create a *viewer* embeddable in HTML pages



THANK YOU!

QUESTIONS?