

GWT: Cool and Useful Stuff

Google Web Toolkit (Beta)

[Google Code Home](#) > [Google Web Toolkit](#)

Google Web Toolkit - Build AJAX apps in the Java language

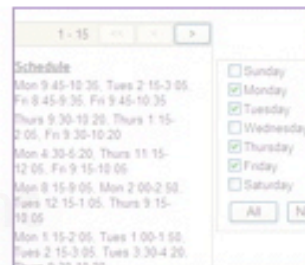
[Download Google Web Toolkit \(GWT\)](#)

Google Web Toolkit (GWT) is a Java software development framework that makes writing AJAX applications easy for developers who don't speak browser quirks as a second language. Writing dynamic applications today is a tedious and error-prone process; you spend 90% of your time working around subtle differences between web browsers and platforms, and JavaScript's lack of modularity makes sharing, testing, and reusing components difficult and fragile.

GWT lets you avoid many of these headaches while offering your users the same dynamic, standards-compliant user interface. You write your front end in the [Java](#) programming language, and the GWT compiler converts your Java code to browser-compliant JavaScript and HTML.

[Learn more »](#)

Featured Example Projects



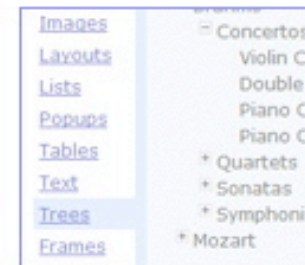
Dynamic Table

A table of data that supports paging using RPCs.

GWT features used:

- [UI widgets](#)
- [Polymorphic RPC](#)

[All example projects »](#)



Kitchen Sink

An application that uses every UI widget GWT provides.

GWT features used:

- [UI widgets](#)
- [History manager](#)

David Geary,
Clarity Training, Inc.

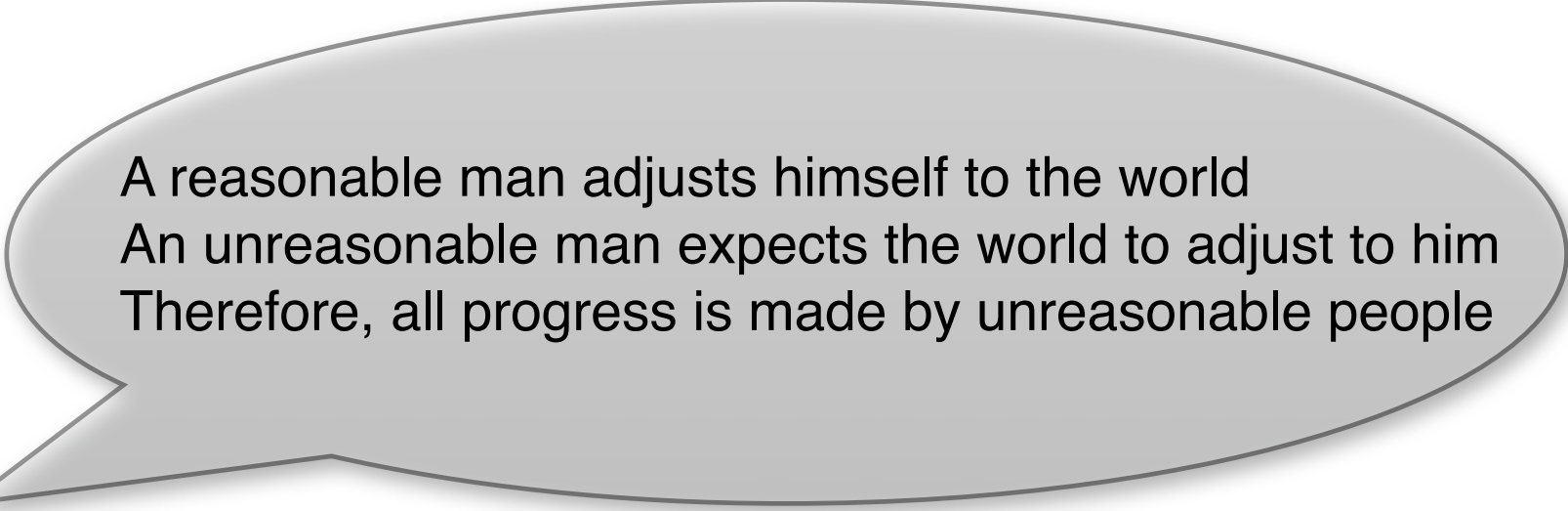
David Geary

- [GWT Solutions](#) (with Rob Gordon, 2007)
- Best-selling books on both component frameworks
 - [Graphic Java Swing](#) (1997)
 - [Core JavaServer Faces](#) (with Cay Horstmann, 2004)
- Member of [JSF](#) 1.0 and [JSTL](#) Expert Groups
- 2nd [Struts](#) committer, [Shale](#) contributor
 - Inventor of the Struts Template Library ([Tiles](#))
- Wrote questions for Sun's Web Dev. [Certification](#) Exam
- [JavaOne Rock Star](#) (2004/2006), [NFJS](#) (2003-present)

Code for this talk is from...



All progress...



A reasonable man adjusts himself to the world
An unreasonable man expects the world to adjust to him
Therefore, all progress is made by unreasonable people

Confucious

Agenda

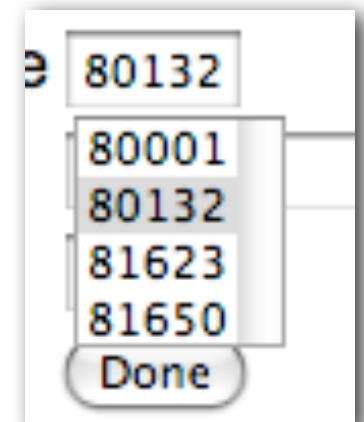
- Custom widgets
- Drag and drop
- Viewports

Three kinds of custom widgets

- **Composite** widgets
 - Extend `Composite` (which extends `Widget`)
 - Compose widgets from existing widgets
- **Intermediate** widgets
 - Extend or delegate to subclasses of `Widget`
- **Low-level** widgets
 - Extend `Widget`
 - Create and manipulate DOM elements

A composite

```
abstract public class AutoCompleteTextBox extends Composite {
    private final TextBox textBox = new TextBox();
    private final PopupPanel popupPanel = new PopupPanel();
    private final ListBox listBox = new ListBox();
    ...
    public AutoCompleteTextBox() {
        initViewWidget(textBox);
        ...
    }
    ...
    public String getText() { // A pass-through method
        return textBox.getText();
    }
    ...
}
```



An intermediate widget

```
public class ResizableCellPanel extends HorizontalPanel {  
    ...  
    public ResizableCellPanel(Widget cellWidget) {  
        ...  
        add(cellWidget);  
        add(new HTML("&nbsp;"));  
        ...  
    }  
    ...  
}
```



A low-level widget

```
public class Message extends Widget {
    ...
    public Message() {
        setElement(DOM.createDiv());
        setName("gwtsolutions-Message");
    }
    public void setText(String text) {
        DOM.setText(getElement(), text);
    }
    public void setVisible(boolean visible) {
        super.setVisible(visible);
        if (visible) {
            applyEffect(getElement(), effect, duration);
        }
    }
    ...
}
```

*Invalid name/pwd combination.
Please try again.*

The DOM Class

- The DOM class has more than 80 static methods
 - Three types of methods
 - HTML element creation methods, eg:
 - `createAnchor()`, `createDiv()`, `createImg()`
 - DOM manipulation methods, eg:
 - `appendChild(Element parent, Element child)`
 - Event methods eg:
 - `eventGetTarget(Event)`, `eventGetType(Event)`

Demos

The image shows three overlapping browser windows demonstrating GWT applications:

- Top Window (GWT Solutions: AutoComplete):** A form titled "Please Tell Us Where You Live" with input fields for Zip code (80132), City (80001), and State (80132). A dropdown menu for the State field shows options: 80132, 81623, and 81650.
- Middle Window (GWT Solutions: EasyFlexTableResizableAndPageableExample):** A table with columns: First Name, Last Name, City, and State. The data rows are:

First Name	Last Name	City	State
Richard	Tattersall	New York	NY
Lynn	Seckinger		
Gabriella	Sarantini		
Homer	Kenney		
Anna	Richards		
- Bottom Window (GWT Solutions: Login):** A login form titled "Please log in" with input fields for Name and Password, and a "Login" button. A red-bordered box highlights an error message: "Invalid name/pwd combination. Please try again."

Next...

- Custom widgets
- Drag and drop
- Viewports

Implementing drag and drop

- Drag and drop is easy with GWT
 - AbsolutePanel
 - MouseListener
 - FocusPanel

A drag panel

```
public class DragPanel extends AbsolutePanel {
    private static final MouseDragger defaultDragger =
        new FollowsMouseDragger();

    public addWidget(Widget widget) {
        addWidget(wrapWidget(widget));
    }

    protected Panel wrapWidget(Widget widget) {
        FocusPanel fp = new FocusPanel();
        fp.add(widget);
        fp.addMouseListener(getMouseDragger());
        return fp;
    }

    protected MouseListener getMouseDragger() {
        return defaultDragger;
    }
}
```

On mouse down

```
public void onMouseDown(Widget draggable, int x, int y) {  
    xoffset = x;  
    yoffset = y;  
    DOM.setCapture(draggable.getElement());  
    dragging = true;  
}
```

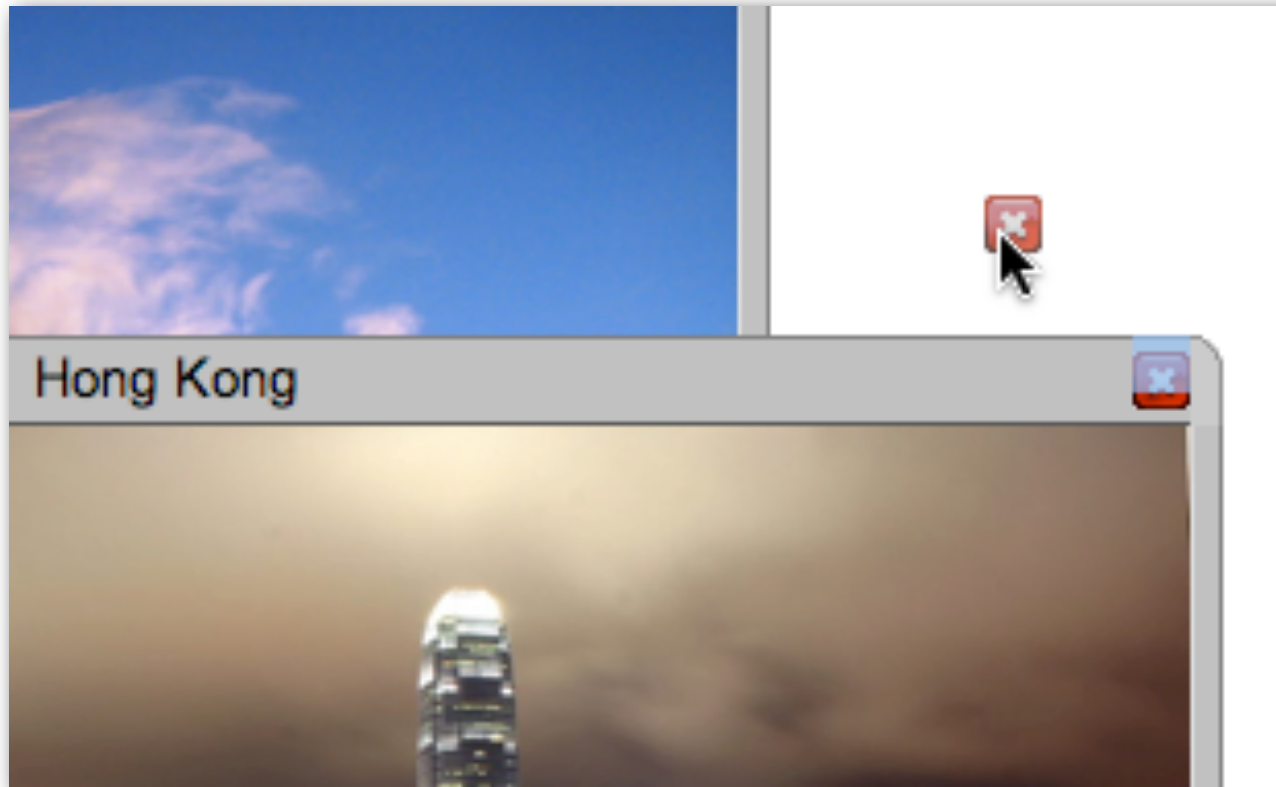
On mouse move

```
public void onMouseMove(Widget draggable, int x, int y) {  
    if (dragging) {  
        AbsolutePanel p = (AbsolutePanel)draggable.getParent();  
  
        p.setWidgetPosition(draggable,  
            x + p.getWidgetLeft(draggable) - xoffset,  
            y + p.getWidgetTop(draggable) - yoffset);  
    }  
}
```

On mouse up

```
public void onMouseUp(Widget draggable, int x, int y) {  
    dragging = false;  
    DOM.releaseCapture(draggable.getElement());  
}
```

Unwanted browser event handling



Tell the browser to butt out

- Implement an event preview


```
private final EventPreview preview = new EventPreview() {  
    public boolean onEventPreview(Event event) {  
        switch (DOM.eventGetType(event)) {  
            case Event.ONMOUSEDOWN:  
            case Event.ONMOUSEMOVE:  
                DOM.eventPreventDefault(event);  
        }  
        return true;  
    }  
};
```

Prevent the
browser's default reaction to
mouse events

Tell the browser to butt out (cont)

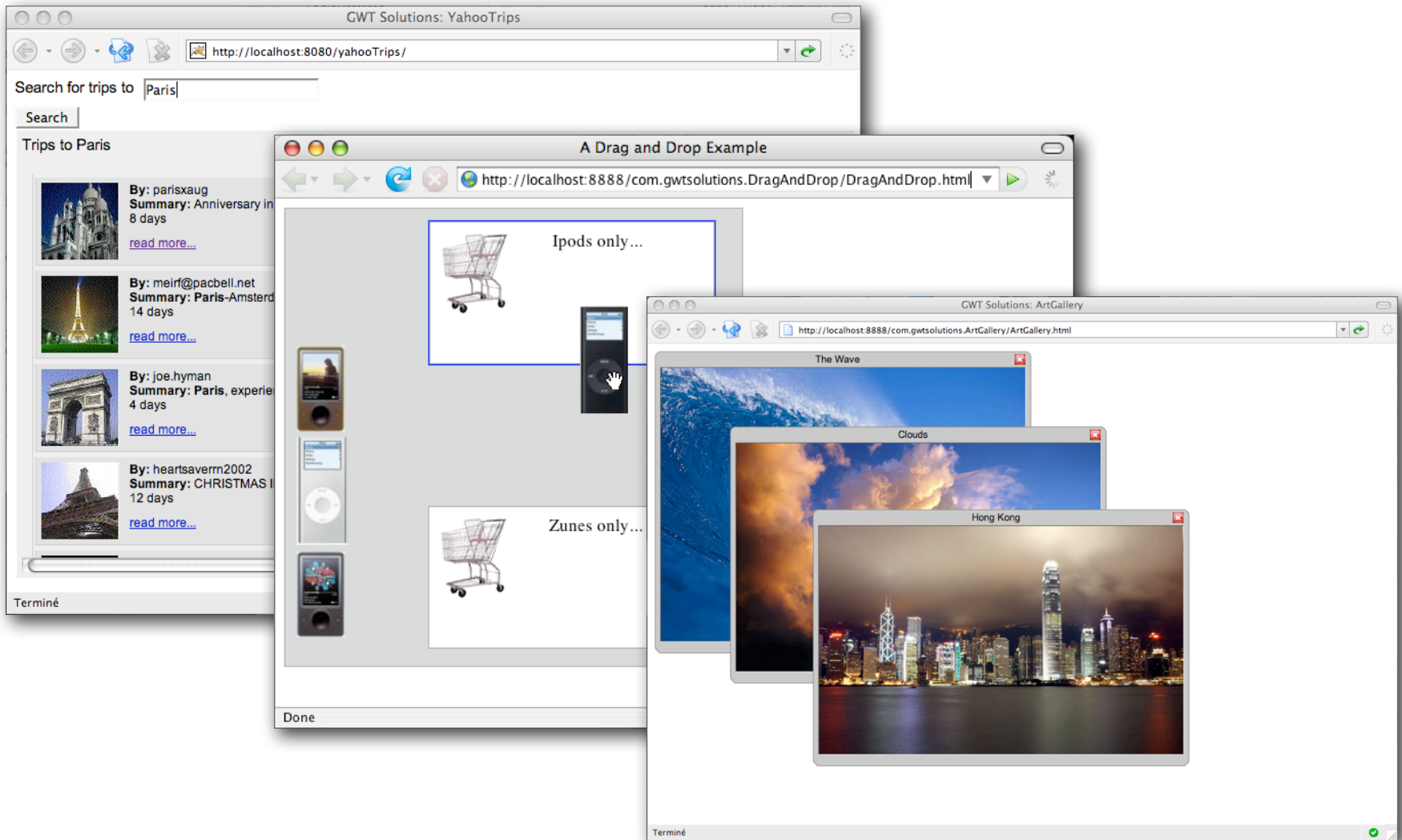
- Add the event preview to the event stack

```
public void onMouseEnter(Widget sender) {  
    DOM.addEventPreview(preview);  
}  
public void onMouseLeave(Widget sender) {  
    DOM.removeEventPreview(preview);  
}
```



Don't forget to
remove
the event preview

Demos



Next...

- Custom widgets
- Drag and drop
- Viewports

Viewports

- Many uses for viewports:
 - Maps
 - Scrolling backgrounds for games
 - Apple's iPhone
- The GWT does not have a Viewport class, but...
- ...you can implement one in ~100 LOPJC!

The mouse panel

```
public class MousePanel extends SimplePanel
    implements SourcesMouseEvents {
    private MouseListenerCollection mouseListeners = null;

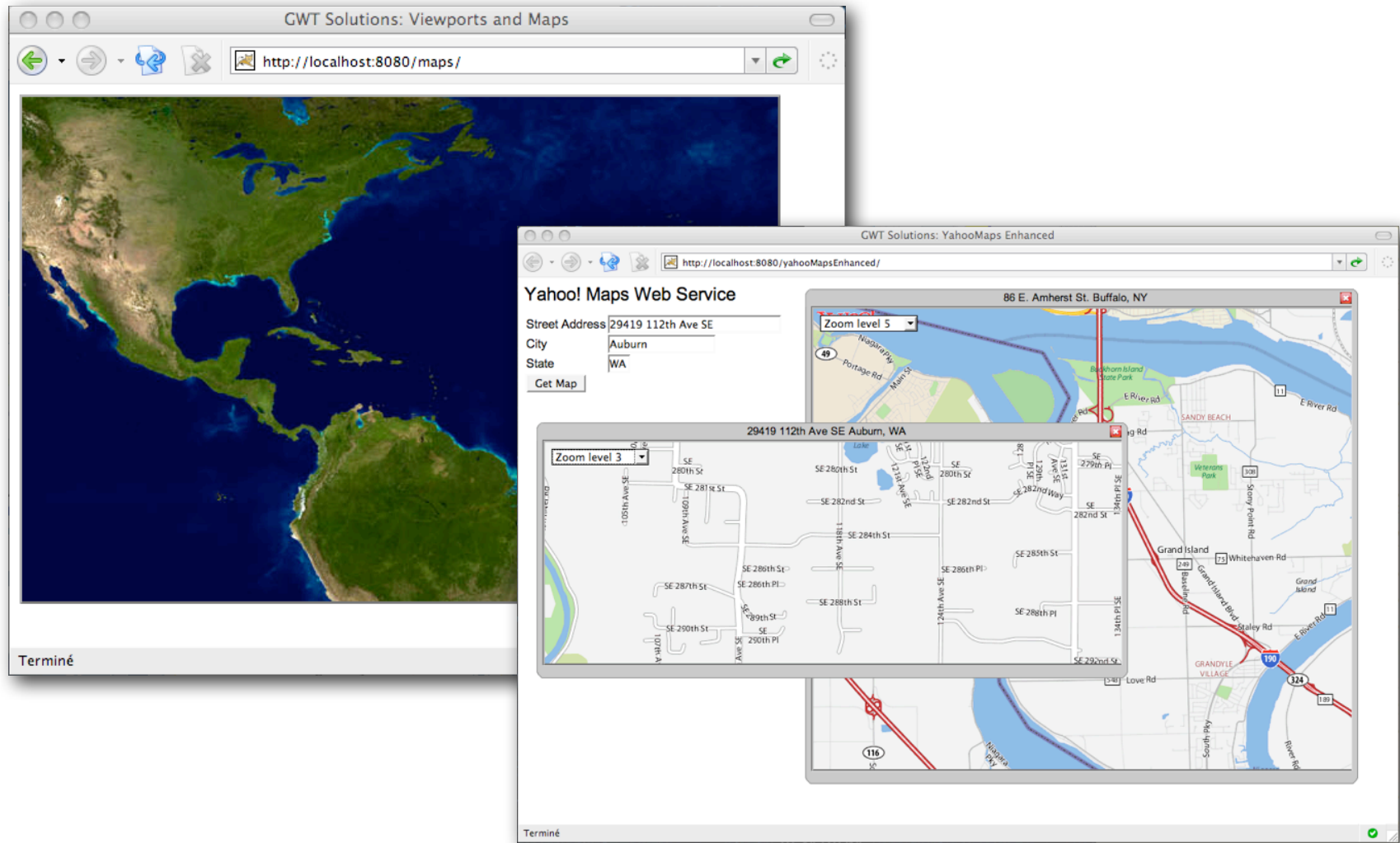
    public MousePanel() {
        super();
        sinkEvents(Event.MOUSEEVENTS);
    }
    public void onBrowserEvent(Event event) {
        if (mouseListeners != null)
            mouseListeners.fireMouseEvent(this, event);
    }
    public void addMouseListener(MouseListener listener) {...}
    public void removeMouseListener(MouseListener listener) {...}
```

The viewport

```
public class Viewport extends AbsolutePanel {
    private MousePanel mousePanel = new MousePanel();

    public Viewport() {
        add\(mousePanel\); // Add the mouse panel to the absolute panel
        ...
        mousePanel.addMouseListener(new MouseListenerAdapter() {
            // A good deal of code to let the user drag the view, in addition
            // to user gesture implementation
        });
    }
    ...
    public void setView(WWidget view) {
        mousePanel.setWidget\(view\);
    }
}
```

Demos





The End

Thanks for coming!

Excellent

Excellent

Excellent

Excellent