Developing on the Clove Core

- Vanilla Framework
- Working with Clove
- What's planned for the future

Vanilla Framework

Built to solve issues with our SDK

- Overwriting issues when loading SWF's

Youtube Plugin

untitled 4

```
package com.spice.clove.plugin
{
    interface IClovePlugin
    {
        function initialize(controller:IClovePluginController);
    }
}
```

Line: 7 Column: 2 ActionScript Tab Size: 4 initialize

Twitter Plugin

untitled 2

```
package com.spice.clove.plugin
{
    interface IClovePlugin
    {
        function initialize();
    }
}
```

Line: 7 Column: 2 ActionScript Tab Size: 4 initialize

Built to solve issues with our SDK

- Overwriting issues when loading SWF's
- Inconsistent User Interface, and User Experience
- code wasn't reusable

Vanilla fixes these issues

- Plugins define their own concrete implementation, and expose available “calls”

Youtube Plugin

untitled 2

```
package com.spice.clove.plugin.core
{
    import com.spice.vanilla.core.proxy.IProxyOwner;

    interface IClovePlugin extends IProxyOwner
    {
        function getProxy():IProxy;
    }
}
```

Line: 7 Column: 36 ActionScript Tab Size: 4 getProxy

Exposed Calls

untitled 4

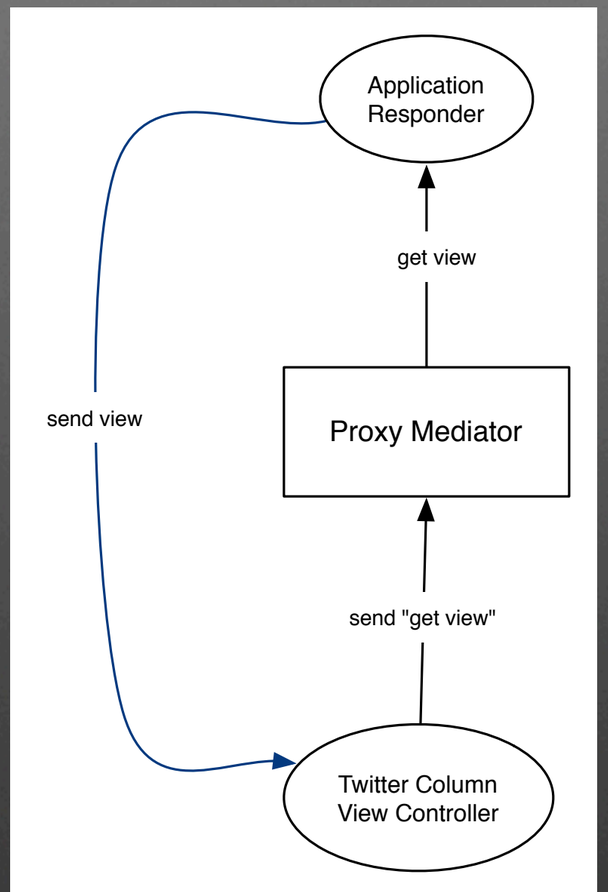
```
package com.spice.clove.plugin.impl.calls
{
    class CallClovePlugin
    {
        public static const INITIALIZE:String = "initialize";

    }
}
```

Line: 6 Column: 9 ActionScript Tab Size: 4 CallClovePlugin

Vanilla fixes these issues

- Plugins depend on the application to provide registered views



Vanilla fixes these issues

- Plugins are easily portable to other languages
  - Language specific features are replaced with language safe features
    - [Bindable] - Binding Framework
    - IEventListener - Notifications Framework
    - serializing objects with ByteArray - Settings Framework

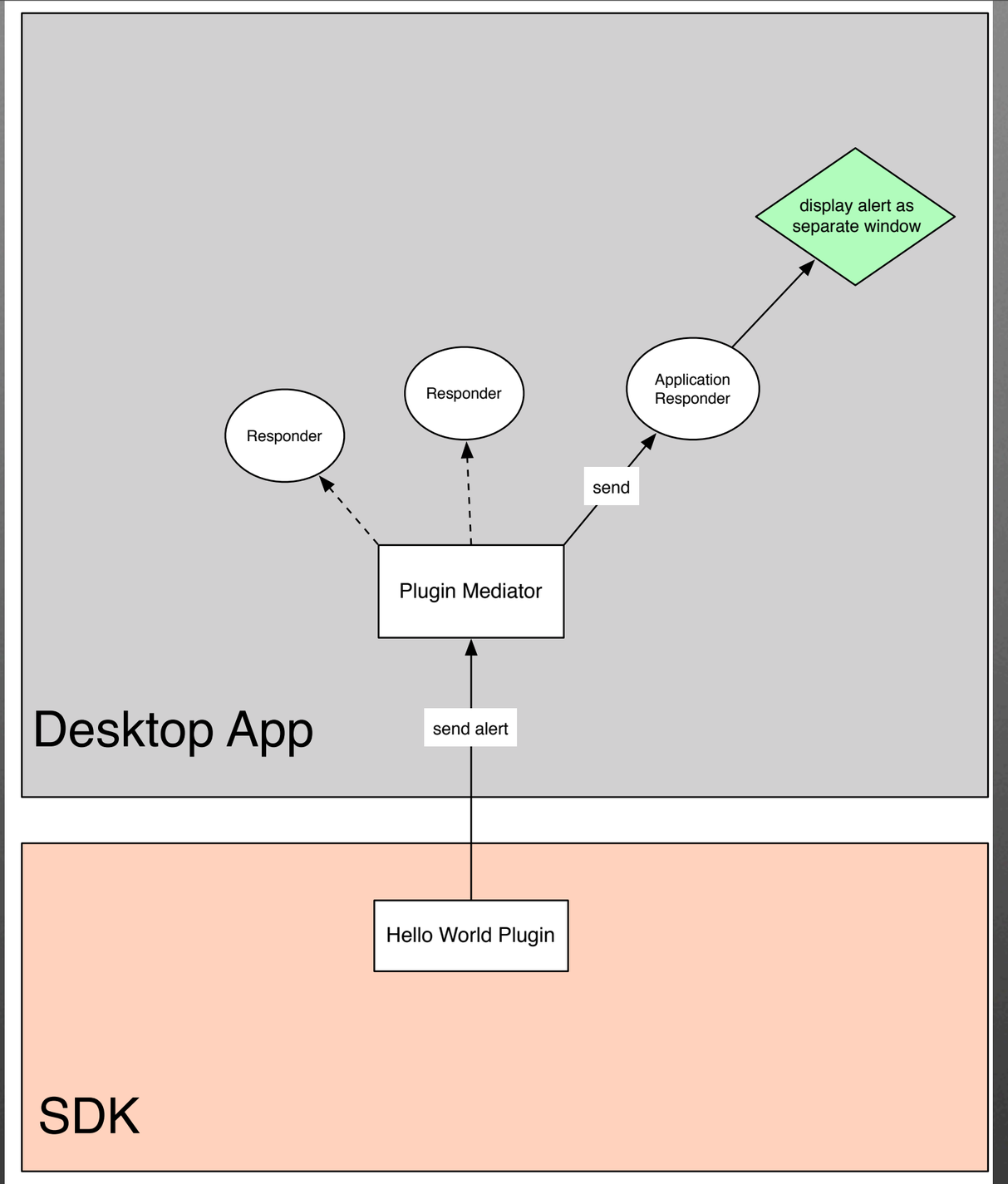
Clove SDK

- Rules for developing portable plugins
  - Follow the folder structure: com/site/clove/plugin/platform
  - Examples:
    - com/spice/clove/twitter/core - shared classes for other plugins
    - com/spice/clove/twitter/impl - global implementation
    - com/spice/clove/twitter/flash - flash specific
    - com/spice/clove/twitter/mx - flex specifc
    - com/spice/clove/twitter/java - Hopefuly sometime soon...
- Always use registered view factories
  - Removes verbose code
  - Ensures plugins are portable
  - If custom views are needed, register the view factories to the plugin's proxy controller

Demo

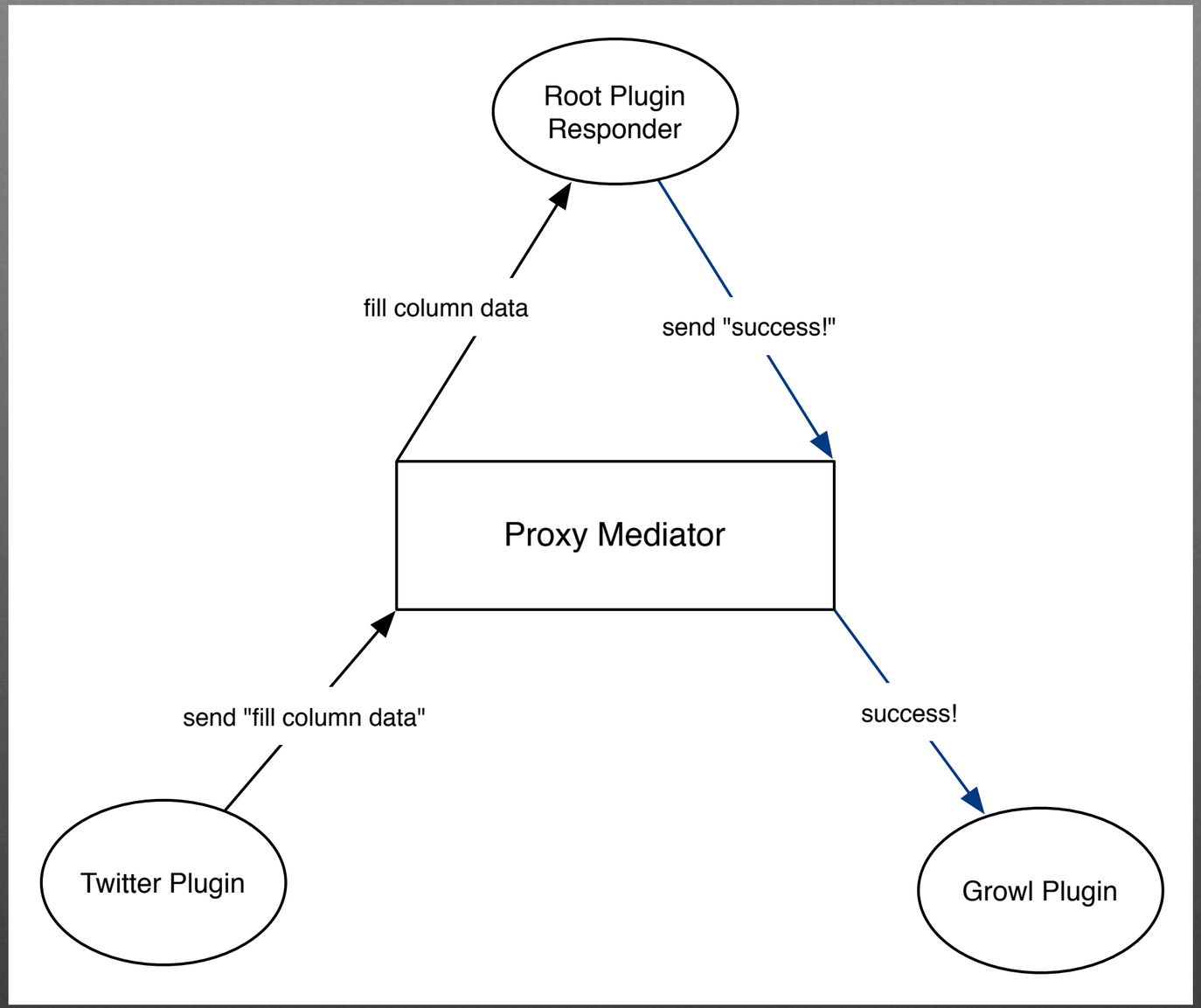
Hello World Plugin

- Written once, usable across multiple platforms
  - Desktop
  - Web
  - Easily portable to other languages



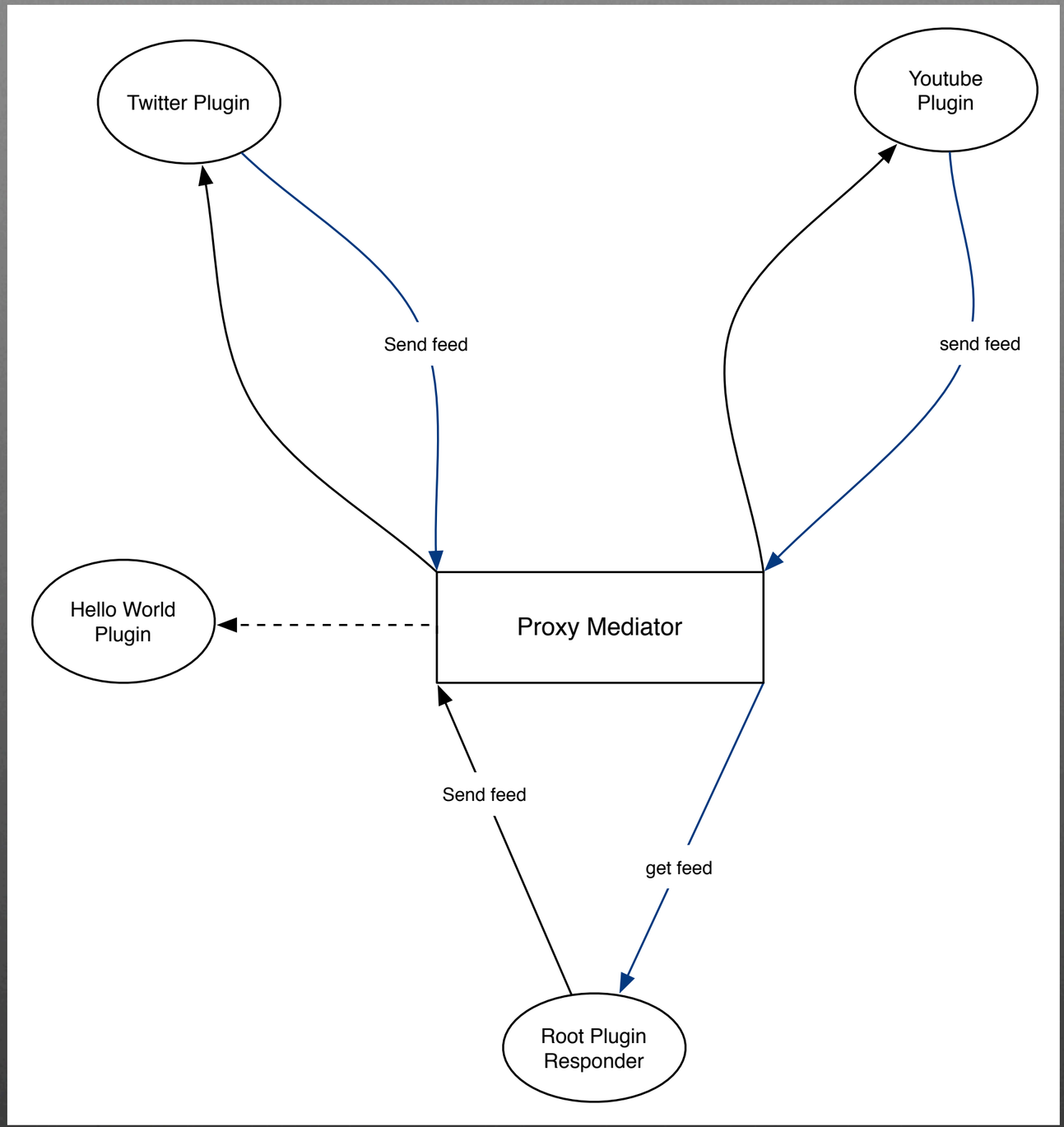
Growl Plugin

- Desktop only
- Listens to all plugins that dispatch a “fill column with data” command



Web Demo

- Clove core running on the web
- All plugins are reused, none modified



What's planned for the future?

- Chocolate Framework
  - A framework to help connect other applications running Vanilla
  - Extending the plugin framework to other languages, not just the platform it's currently running on
    - Clove daemon for Mac, and Windows

Sign up for public beta at:

cloveapp.com