

adaptTo()

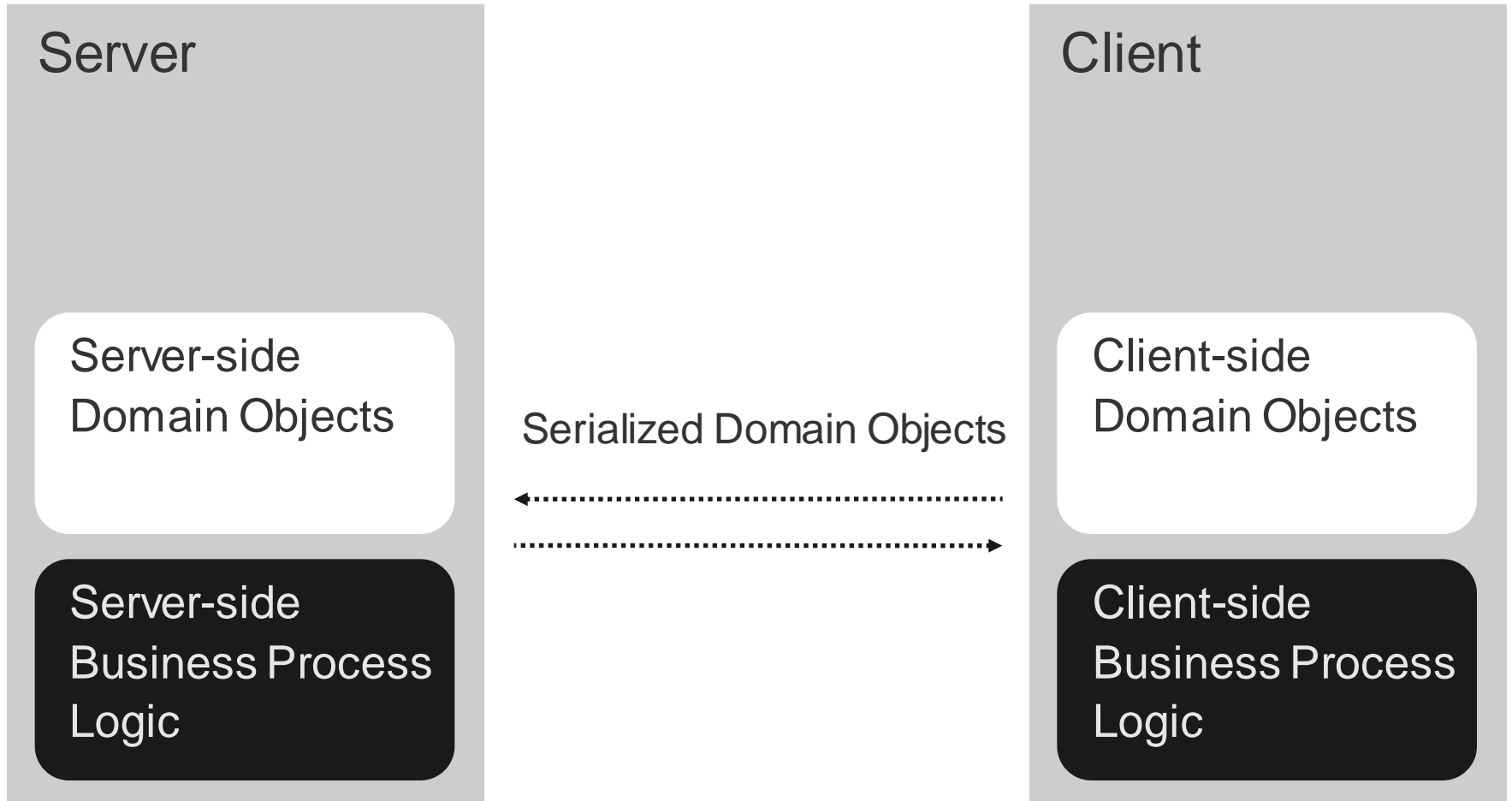
APACHE SLING & FRIENDS TECH MEETUP
BERLIN, 23-25 SEPTEMBER 2013

Hypermedia APIs on top of Apache Sling
with Granite UI

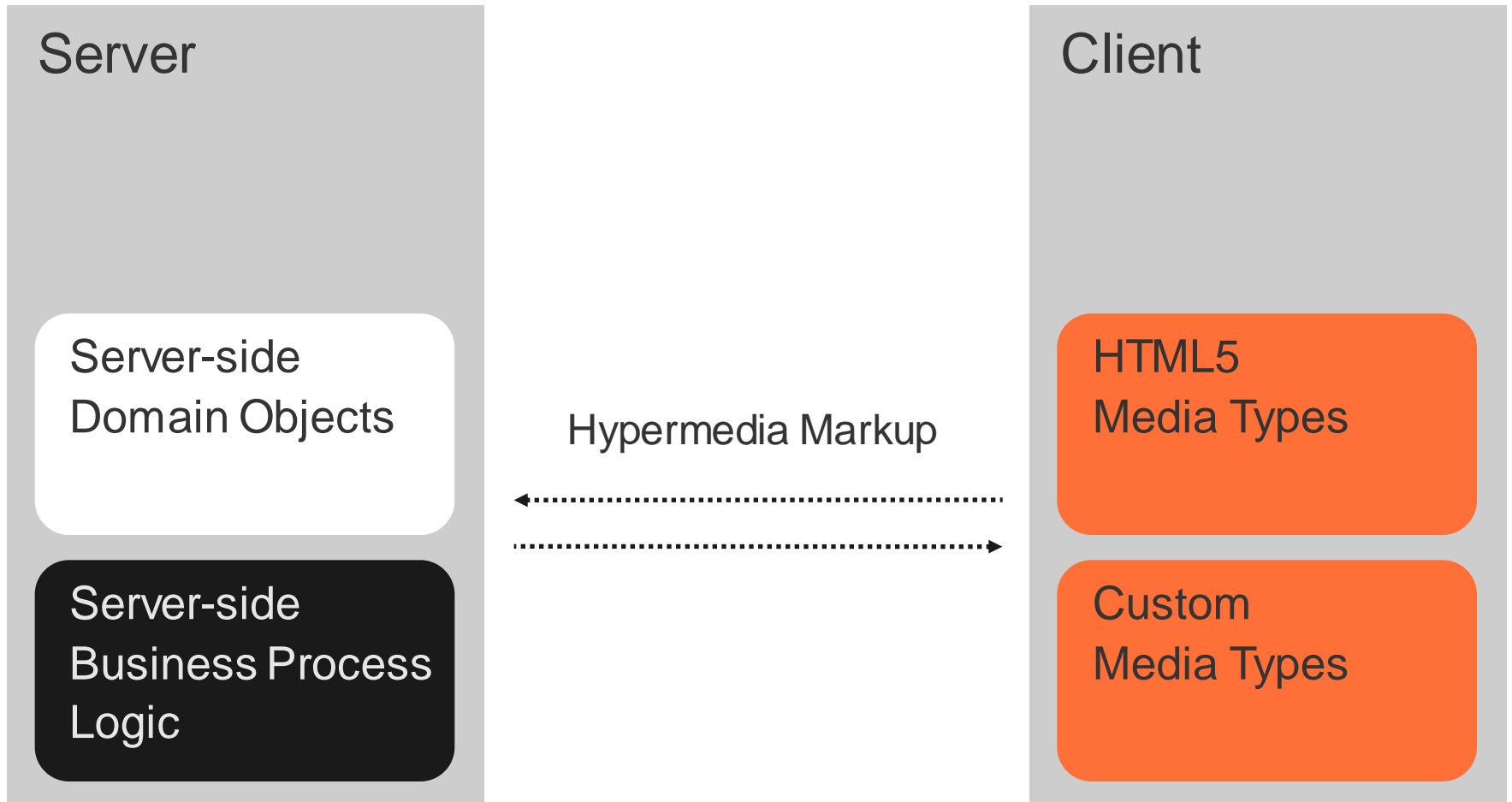
What is a Hypermedia API?

- Based on Representational State Transfer (**REST**) - **Uniform Interface** constraint.
- Hypermedia as the Engine of Application State (**HATEOAS**).
- We talk about **media types**, not domain objects.

How did it work in traditional APIs?



How does it work the Hypermedia way?



Goals and advantages of Hypermedia APIs

- Decoupling
- Encapsulation
- Reusability

Disadvantages

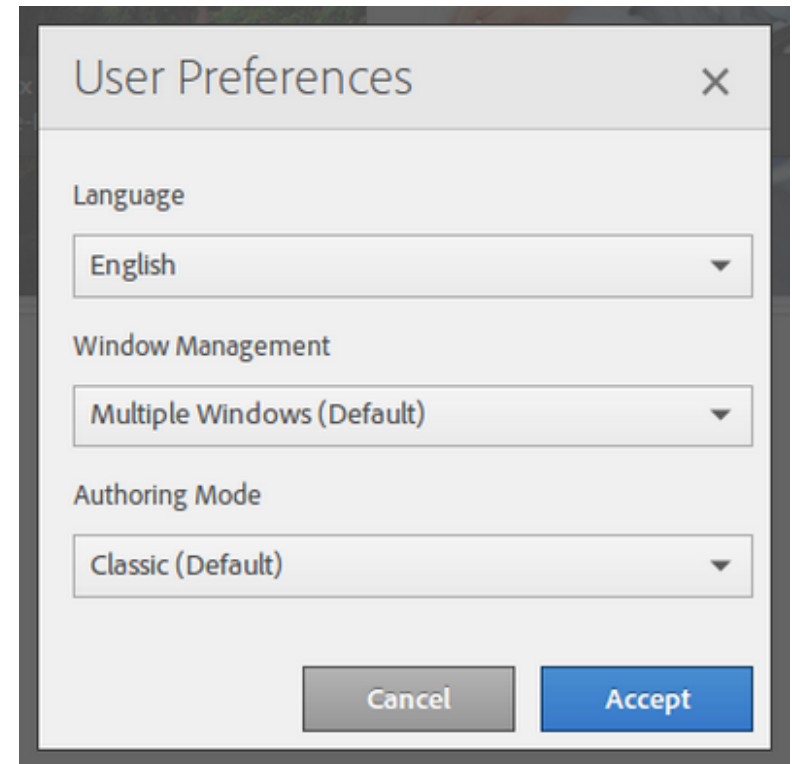
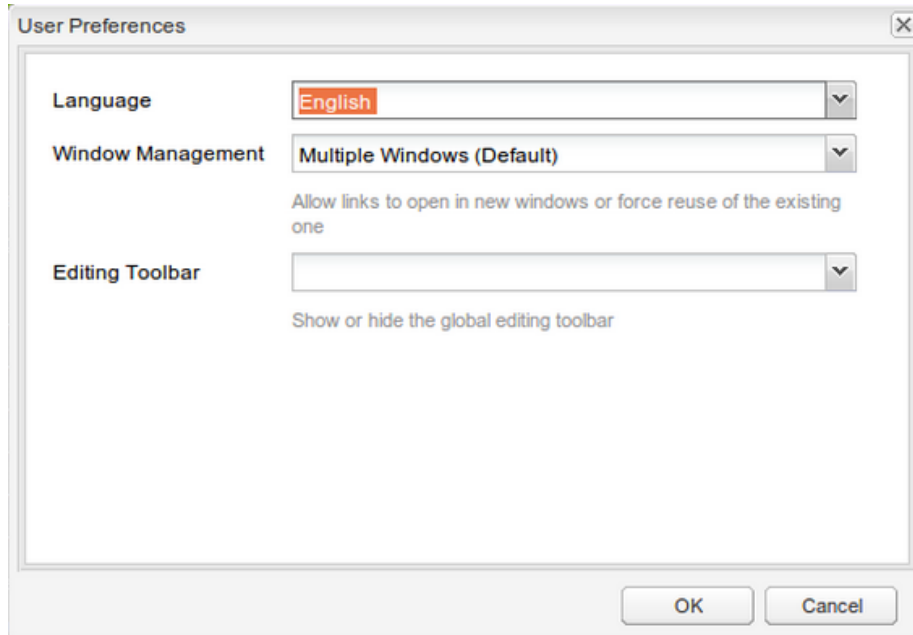
- More data to transfer: Textual representation versus compact data such as JSON.
- Latency may be greater than in more optimized APIs.

Example

We like . . .

- to use HTML5 as the hypermedia language - it already defines quite a lot of media types.
- to extend HTML5 on our own with further, specific media types as required.

Example: User Preferences Dialog



Example: Source code

```
<a href="#my-modal" class="button" data-toggle="modal">Show Modal</a>
```

```
<div id="my-modal" class="modal">
```

```
  <div class="modal-header">
```

```
    <h2>Modal from markup</h2>
```

```
    <button type="button" class="close" data-dismiss="modal">&times;</button>
```

```
  </div>
```

```
  <div class="modal-body">
```

```
    <p>This modal was created from markup.</p>
```

```
  </div>
```

```
  <div class="modal-footer">
```

```
    <button data-dismiss="modal">Close</button>
```

```
    <button class="primary" data-dismiss="modal">Save</button>
```

```
  </div>
```

```
</div>
```

Example: How did we define the new vocabulary?

- **class="modal"**

Defines the markup for modal

-> modal must have an **id**

- **data-toggle="modal"**

Shows or hides the modal

-> must have **href** to reference the modal

- **data-dismiss="modal"**

Closes the modal

-> always closes the modal it belongs to

modal.js

```
$(document).on("click", "[data-toggle=modal]", function(event) {  
    var id = $(event.target).attr("href")  
    $(id).show();  
    $(event).preventDefault();  
});
```

Contextual jQuery

Use the context (here: href/id) to get **all** your data.

Example: What makes it different?

- We define a “media type” / “web component” **modal dialog** that can be opened and dismissed.
- The client never knows about “User Preferences”, it only knows about modal dialogs and how to handle them.
- The form submit is a media implementation - free with HTML5.
- No need for special logic client-side.

What does Granite UI provide?

- CQ components that create semantic markup.
- Every CQ component creates markup for one specific web component (wizard, grids, form fields, etc.)
- Client-side javascript to enhance user experience.
- You configure pages instead of writing the markup on your own.

What next?

- Granite UI isn't currently meant to be used in customer projects.
 - > Create your own semantics / vocabulary in your projects.

Questions?