

#### REST in Practice

Jaakko Kangasharji

REST Refreshe

REST in Practice

**REST** Services

## REST in Practice Applications and Services in Internet 2009 (4 cr)

### Jaakko Kangasharju jaakko.kangasharju@futurice.com

Futurice Ltd.

Autumn 2009



### Outline

REST in Practice

Jaakko Kangasharju

REST Refreshei

REST in Practice

REST Services

### **1** REST Refresher

**2** REST in Practice

#### **3** REST Services

Jaakko Kangasharju (Futurice Ltd.)



### Outline

REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

REST Services

### 1 REST Refresher

**2** REST in Practice



### **REST** Constraints

#### REST in Practice

Jaakko Kangasharju

#### REST Refresher

REST in Practice

- Olient-Cache-Stateless-server
- 2 Uniform interface
- 8 Layered system
- 4 Code-on-demand



### **REST Data Elements**

#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

**REST** Services

Resource Nameable information Resource identifier Durable identifier for resource Representation Transferred version of resource Representation metadata Specific information on representation Resource metadata General information on resource

Control data Information on message



### Interface Constraints

#### REST in Practice

Jaakko Kangasharju

#### REST Refresher

REST in Practice

- Identification of resources
- Ø Manipulation of resources through representations
- Self-descriptive messages
- O Hypermedia as the engine of application state (HATEOAS)



### Outline

REST in Practice

Jaakko Kangasharju

REST Refreshei

REST in Practice

REST Services

### REST Refresher

**2** REST in Practice

#### **B** REST Services

Jaakko Kangasharju (Futurice Ltd.)



### **REST** Implementation

#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- REST is a general architectural style for distributed hypermedia systems
- World Wide Web based on REST through URI and HTTP
- Practical implementation often differs from theoretical ideal



### **REST** Implementation

#### REST in Practice

Jaakko Kangasharju

REST Refresher

#### REST in Practice

- REST is a general architectural style for distributed hypermedia systems
- World Wide Web based on REST through URI and HTTP
- Practical implementation often differs from theoretical ideal
  - Architecture does not support requirement (at least not easily)
  - Implementor does not understand architecture properly



## Uniform Resource Identifiers (URIs)

#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Resource identifier element on the Web
- In general, consist of a *scheme* and a scheme-specific part
- Schemes for locator URIs denote protocols, scheme-specific part includes address
- Example: http://www.cse.tkk.fi/en/
- Example: urn:ietf:rfc:2616



### Non-RESTful Practices

#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- URI does not (need to) map 1-to-1 into file system
- Embedding user or session information in URI
- Treating URIs as non-opaque



### **HTTP** Fundamentals

REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

**REST** Services

• НТТ	P: Appli	cation-layer	request-	response	protocol
-------	----------	--------------	----------	----------	----------

 Header-body structure, body contains representation, headers all metadata (resource, representation, control):

Status Line

Header

Body

Line HTTP/1.1 200 OK Content-Type: text/html Content-Length: ... Expires: ... Vary: Accept <name>: <value> : <opaque data>



### HTTP Methods

#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Four basic methods provide creation, read, update, and deletion (CRUD) functionality
  - GET, POST, PUT, DELETE
- Other common methods: OPTIONS, HEAD
- Special-purpose methods: TRACE, CONNECT
- Methods can be *safe* or *idempotent*



### Other HTTP Features

#### REST in Practice

Jaakko Kangasharju

REST Refresher

#### REST in Practice

- Types and encodings: Content-Type, Content-Encoding, Transfer-Encoding
- Negotiation of types and encodings
- Transport independence: Content-Length, chunked encoding
- Persistent connections
- Does not support asynchronous transports



### Non-RESTful Practices

#### REST in Practice

Jaakko Kangasharju

REST Refresher

#### REST in Practice

- Cookies: Site-based state instead of application state, opaque data
- Use of HTTP as a transport protocol



### Outline

REST in Practice

Jaakko Kangasharju

REST Refreshei

REST in Practice

**REST** Services

### REST Refresher

**REST** in Practice

#### 8 REST Services

Jaakko Kangasharju (Futurice Ltd.)



REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

**REST Services** 

• Web service designer: You cannot design services without session state!



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Web service designer: You cannot design services without session state!
- RESTafarian: Uniform interface! HATEOAS!



REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Web service designer: You cannot design services without session state!
- RESTafarian: Uniform interface! HATEOAS!
- Concept: Application state stored in hypermedia documents
- Concept: "Session" can be moved by sending URIs around



REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Web service designer: You cannot design services without session state!
- RESTafarian: Uniform interface! HATEOAS!
- Concept: Application state stored in hypermedia documents
- Concept: "Session" can be moved by sending URIs around
- Not just non-querying URLs, document-based interfaces, or XML-over-HTTP
- Representations not necessarily HTML, can be anything



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

**REST** Services

• Practical example: Online store with shopping carts



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Practical example: Online store with shopping carts
- What are the functional requirements?



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Practical example: Online store with shopping carts
- What are the functional requirements?
  - User must be able to browse the store, exploring items



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Practical example: Online store with shopping carts
- What are the functional requirements?
  - User must be able to browse the store, exploring items
  - User must be able to add items to the cart and remove them



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Practical example: Online store with shopping carts
- What are the functional requirements?
  - User must be able to browse the store, exploring items
  - User must be able to add items to the cart and remove them
  - Shopping cart needs to store items persistently (not just one-click ordering!)



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Practical example: Online store with shopping carts
- What are the functional requirements?
  - User must be able to browse the store, exploring items
  - User must be able to add items to the cart and remove them
  - Shopping cart needs to store items persistently (not just one-click ordering!)
  - User must be able to checkout after picking all desired items



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Practical example: Online store with shopping carts
- What are the functional requirements?
  - User must be able to browse the store, exploring items
  - User must be able to add items to the cart and remove them
  - Shopping cart needs to store items persistently (not just one-click ordering!)
  - User must be able to checkout after picking all desired items
  - Shopping cart must empty itself on checkout



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

**REST** Services

• What are the resources?



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- What are the resources?
- Each item in the store is a resource
- Each shopping cart is a resource



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- What are the resources?
- Each item in the store is a resource
- Each shopping cart is a resource
  - Alternative: shopping cart stored client-side?



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- What are the resources?
- Each item in the store is a resource
- Each shopping cart is a resource
  - Alternative: shopping cart stored client-side?
- Is the whole store a resource?
- Is the "checkout counter" a resource?
- What other resources must there be?



## Application State Diagrams





#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

**REST Services** 

# Create cart: POST to cart creator, receive new cart identifier



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Create cart: POST to cart creator, receive new cart identifier
  - Or: PUT an empty cart representation?



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Create cart: POST to cart creator, receive new cart identifier
  - Or: PUT an empty cart representation?
- Add item to cart: POST item representation to cart



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Create cart: POST to cart creator, receive new cart identifier
  - Or: PUT an empty cart representation?
- O Add item to cart: POST item representation to cart
- Ocheckout: POST payment and dispatch information to cart



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Create cart: POST to cart creator, receive new cart identifier
  - Or: PUT an empty cart representation?
- $\ensuremath{\textcircled{O}}$  Add item to cart: POST item representation to cart
- Ocheckout: POST payment and dispatch information to cart
  - Or: POST cart representation to checkout counter?



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Create cart: POST to cart creator, receive new cart identifier
  - Or: PUT an empty cart representation?
- $\ensuremath{\textcircled{O}}$  Add item to cart: POST item representation to cart
- Ocheckout: POST payment and dispatch information to cart
  - Or: POST cart representation to checkout counter?
  - Multi-step checkout process possible either way



### Representation Design

#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Storefront: Home page with links to items and cart creation
- Where is the link to a user's carts?



### Representation Design

REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Storefront: Home page with links to items and cart creation
- Where is the link to a user's carts?
- Item: Item description with link to add to cart
- Cart: List of items with remove links, form to fill in checkout data
- Make each item in the cart a separate resource?



### Representation Design

REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- Storefront: Home page with links to items and cart creation
- Where is the link to a user's carts?
- Item: Item description with link to add to cart
- Cart: List of items with remove links, form to fill in checkout data
- Make each item in the cart a separate resource?
- Mid-checkout cart: Form to fill in appropriate data, disallows adding or removing items



### Evaluation

REST in

Jaakko Kangasharju

REST Refresher

REST in Practice

- Benefit: Easy to move application state
- Benefit: More caching opportunities
- Drawback: Harder to do user-specific features



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

**REST Services** 

• What did I miss in the design?



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- What did I miss in the design?
- Security, obviously!



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- What did I miss in the design?
- Security, obviously!
- What are the security requirements of a shopping cart service?



#### REST in Practice

Jaakko Kangasharju

REST Refresher

REST in Practice

- What did I miss in the design?
- Security, obviously!
- What are the security requirements of a shopping cart service?
- But security in REST is a paper topic...



### **REST** Over

REST in Practice

Jaakko Kangasharji

## Questions?