Faculty of Informatics



Business Process Management with REST

Cesare Pautasso Faculty of Informatics University of Lugano, Switzerland

> c.pautasso@ieee.org http://www.pautasso.info @pautasso



Business Process Management

RESTful Web Services

3

WS-* Standards Stack



WS-* Standards Stack

Int	E	3PM	Ν	/lanage	ment	Messaging Specifications
eroperabi	Metada	Reliability	Security	Transactions	State	
lity	ਯ ਾ	*	Mess	agin	g	Transaction Specifications
Handards Enders DASH → Settematic in the set of the s	MAL Macrosoftications Mar. ************************************			Inner Contraction of the second secon		

Università

della Svizzera italiana

<image/> <section-header><section-header><section-header><text><text></text></text></section-header></section-header></section-header>	eroperabi	Int
XML Specif	Metada	6
Ca w	Reliability	BPM
n yo ith F	Security	
u do RES	Transactions	Manage
o it T?	State	ement
	Reliability Specifications Normality Specifications Normality Specifications<	Messaging Specifications

We believe there is huge potential to marrying REST with workflow and BPM.



Combined with the architecture of the Web, a workflow service can provide both a truly simple, portable, and flexible way to build workflow driven integrations and applications.

Jniversità

della Svizzera italiana

Business Process Management



WS-BPEL Primitives



The modeling language natively supports the RPC or message-based connectors

What is your SOA connector today?







REST as a new connector







Is REST really used?



PUT

REST in one slide

- Web Services expose their data and functionality trough resources identified by URI
- Uniform Interface Principle: Clients interact with resources through a fix set of verbs. Example HTTP: GET (read), PUT (update), DELETE, POST (catch all),
- Multiple representations for the same resource
- Hyperlinks model resource relationships and valid state transitions for dynamic protocol description and discovery

Business Process Management



BPM with **REST**



BPM with **REST**



Some Challenges for BPM engines

- Can you drive the execution of tasks with PUT/POST/DELETE requests?
- Can you monitor your processes with an RSS/ATOM feed?
- Can you bookmark a process instance?
- Can you send an email to your colleague with a link to a task from your worklist?
- Can you ask a process to give you links to its tasks left to be done?
- Can you publish your process as a resource?
- Can you publish resources from your process?
- Can you call RESTful APIs directly?

BPM REST

- Processes
- Tasks
- Control Flow
- Data Flow

- Resources/URIs
- Uniform
 Interface
- Representations
- Hypermedia

Everything is a resource





Follow links to discover the processes deployed as resources

Representations



Uniform Interface and Hypermedia



Starting or **Running** a new process?



Starting or Running a new process?



- The client starting a long running process is redirected to a location x representing the newly started process instance
- The process and the client run asynchronously
- The client may retrieve the current state of the process instance at any time

Università della

Svizzera italiana

Uniform Interface and Hypermedia



POST or PUT?



REST in Practice

Jim Webber, Savas Parastatidis, Ian Robinson http://restinpractice.com/27

Simple RESTBucks Example



Simple RESTBucks Example

Università della Svizzera italiana

Hypermedia-centric service design (and implementation) done with a business process model



Instantiating a process

Università della Svizzera italiana

GET / rest/restbucks/order/1.0/

🕘 Mozilla Firefox			
<u>File Edit View History Delicious Bookman</u>	ks <u>T</u> ools <u>H</u> elp		
🔇 💽 - C 🗙 🏠 🖬 🔝 🎰 🕻	http://localhost:8080/rest/restbucks/order/1.0/	☆ · W·	P 🗗 🐠 ·
Start Process res	stbucks.order [1.0]		
it			
ltem			
Start Run	•		
Get all instances of restbucks.order	[1.0]		
• GET this content in:			
 XML (application/xml) JSON (application/json) Plain Text (text/plain) 			
•	III		•
Done			🖬 🗏 🖂 🏶 🖾 🔚

Retrieve a form which describes how to instantiate a new process

Instantiating a process

Università della Svizzera italiana

POST / rest/restbucks/order/1.0/

Wozilla Firefox		_ 0 X
<u>File Edit View History Delicious Bookmarks Tools H</u> elp		
C X 🟠 🖬 📃 🗟 🗋 http://localhost:8080/res	st/restbucks/order/1.0/	₽ ₽ •
Start Process restbucks.or	der [1.0]	
item		
Start Run		
	Run = blocking	
	(client waits until	
Start = non blocking	the process replie	es)
(redirect to URI of		,
the new instance)		

Interacting with a task

GET



POST / rest/restbucks/order/1.0/0/payment



Interacting with a resource

Università della Svizzera italiana

GET / receipt/2fc7f6e2-8b43-4672-a7c4...



Interacting with a resource

Università della Svizzera italiana

DELETE / rest/restbucks/order/1.0/0

Poster Request	pr: Aurorg	X	
Select a file or you'd like or ju	enter content to POST or PUT to a URL and st use the GET, HEAD, or DELETE methods	then specify the mime type on a URL.	order
URL: h	ttp://localhost:8080/rest/restbucks/order/1.0/0		
User Auth:		Google Login	getPrice
Timeout:		- 30	A second s
Settings:	Save Import Store		payment
Actions GET PC	DST PUT DELETE V Submit		
Content to Ser	d Headers Parameters	1	checkPaymer
File:		Browse	
Content Type	e: text/xml		receipt
Content Opti	ions: Base64 Parameter Body		

Deleting a process resource

Università della Svizzera italiana

DELETE / rest/restbucks/order/1.0/0

Status: 200 OK		getPrice
		checkPayment
Headers:		
Date	Fri, 12 Mar 2010 09:01:16 GMT	
Server	Jetty/5.1.4 (Windows 7/6.1 x86 java/1.5.0_21	
Content-Length	0	
	Close	









	restbucks.om	I X			
	Adapter: R	EST_TASKAdapter (REST.TASK)			
	Insert Variable	Payload: Payload for the response			
Ê	- System Par	ameters			
	Code	200			
TASK	Headers			*	
		<		-	
	Payload	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes" <rb:payment xmlns="http://schemas.restbucks.com/dap" <link rel="latest" uri="/rest%instance%" /> <link rel="receipt" uri="/receipt/%id%"/> <rb:amount>\$amount\$</rb:amount></pre>	"?> xmlns:rb="http://schemas.restbucks.com">	*	
		<rb:cardholdername>\$name\$</rb:cardholdername> <rb:cardnumber>\$card\$</rb:cardnumber>	Specify the final state		
		<rb:expiry>\$expiry\$</rb:expiry> 	of the Payment task	-	
		•		•	
	Content_Type	application/vnd.restbucks+xml			
	Advanced S	ystem Parameters			
©2010 - Ce	Overview Prog	ram: Payment 1.0 Adapter: REST_TASKAdapter (REST.TASI	K)		





Open Challenges

Static vs. Dynamic Typing

- Myth: RESTful Web services cannot be composed (with BPEL) because they do not give a static contract description
- Reality: RESTful Web services can dynamically negotiate the most suitable representation format with their clients
- Challenge: How to support dynamic typing and content type negotiation in a BPM composition language?

- Myth: Processes cannot be mapped to resources because they can change their state (independently of their clients)
- Reality: REST Resources do not have to be passive "CRUD" services but can be active and have a life of their own.
- Challenge: How to best let clients control an active resource backed up by a process instance through the uniform interface?

- Myth: Processes run for a long time and need to interact asynchronously with their clients. This cannot be done with HTTP.
- Reality: HTTP supports non blocking interactions. Each process instance is mapped to a resource URI, which can be used by clients throughout its lifetime.
- Challenge: How to let processes send notifications back to their clients?

Jniversità

della Svizzera italiana

Conclusions

- REST resources are a good abstraction to publish processes on the Web
- RESTful HTTP is good enough to interact without any extension with process execution engines and drive the execution of process and task instances
- If done right, BPM can be a great modeling tool for Hypermedia-centric service design (and implementation!)

Cesare Pautasso Faculty of Informatics, USI Lugano http://www.pautasso.info/ @pautasso

JOpera RESTful Process Engine http://www.jopera.org/ @jopera_org

- R. Fielding, <u>Architectural Styles and the Design of Network-based Software Architectures</u>, PhD Thesis, University of California, Irvine, 2000
- C. Pautasso, O. Zimmermann, F. Leymann, <u>RESTful Web</u> <u>Services vs. Big Web Services: Making the Right Architectural</u> <u>Decision</u>, Proc. of the 17th International World Wide Web Conference (<u>WWW2008</u>), Bejing, China, April 2008
- C. Pautasso, <u>BPEL for REST</u>, Proc. of the 7th International Conference on Business Process Management (BPM 2008), Milano, Italy, September 2008
- C. Pautasso, <u>Composing RESTful Services with JOpera</u>, In: Proc. of the International Conference on Software Composition (<u>SC2009</u>), July 2009, Zurich, Switzerland.



Raj Balasubramanian, Benjamin Carlyle, Thomas Erl, Cesare Pautasso, **SOA with REST**, Prentice Hall, to appear in 2011