

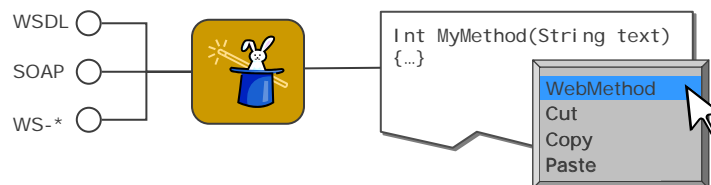
# SOA Patterns: New Insights or Recycled Knowledge?

Gregor Hohpe  
Software Engineer  
www.eaipatterns.com



© 2007 Google, Inc. All rights reserved.

## Could It Be So Easy?

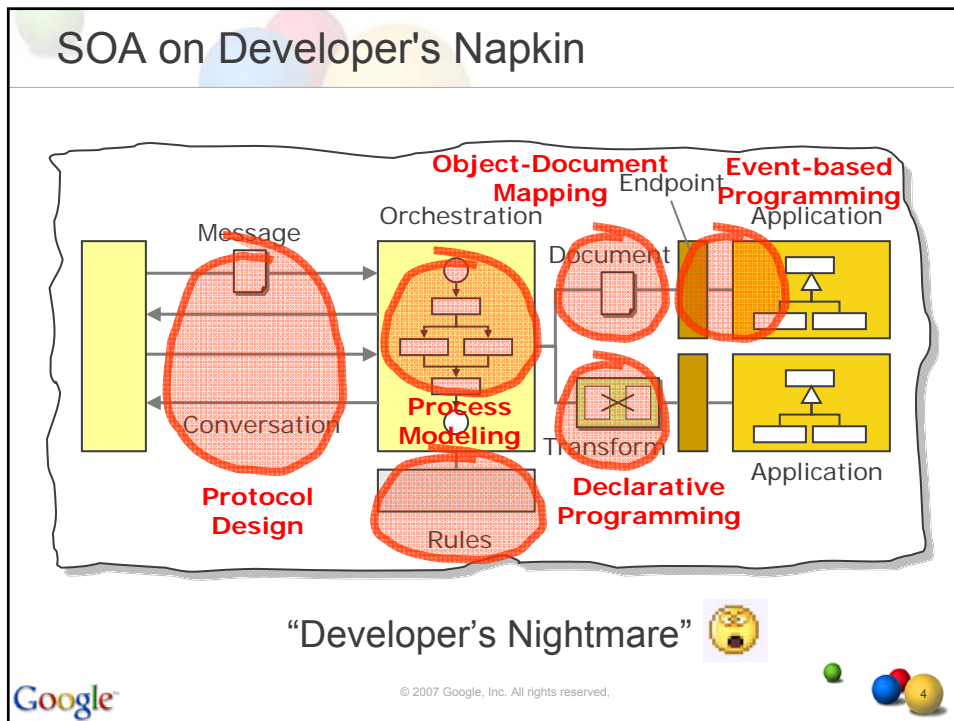
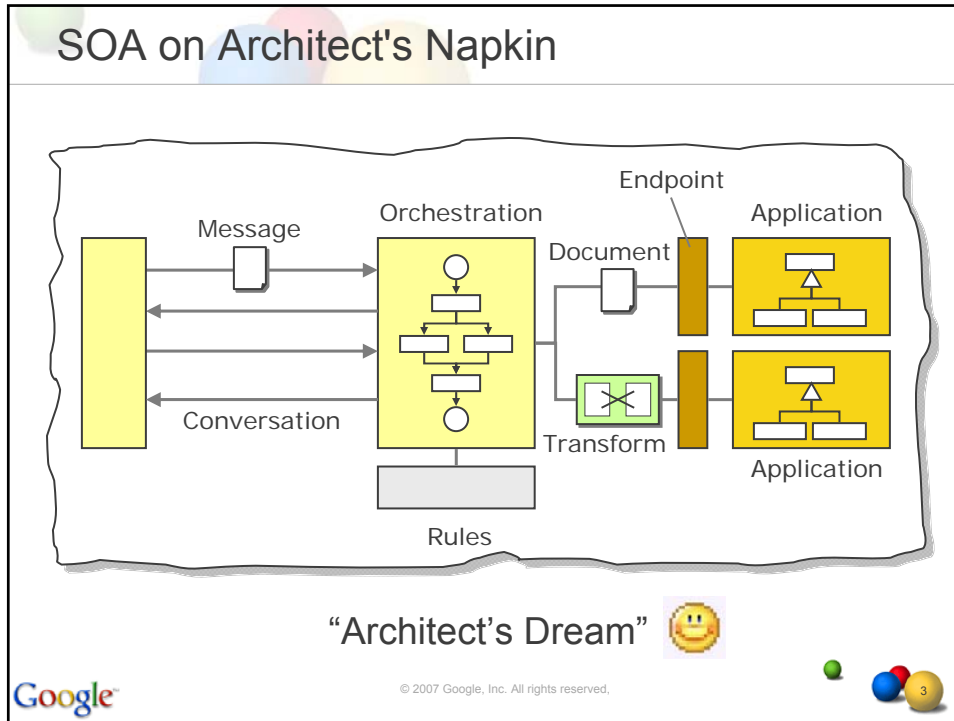



- Buzzword compliant, but not a service-oriented architecture
- Synchronous call stack mentality
- No interface-implementation separation



© 2007 Google, Inc. All rights reserved.









Service-oriented architectures  
involve a variety of new  
programming models.

We can expect to see a  
renaissance of design patterns that  
capture our experience in using  
them correctly.




© 2007 Google, Inc. All rights reserved.




## Understanding Technology

- Syntax
  - Basic language mechanism
  - Artefact of crude input devices
- Constructs
  - "Vocabulary": Objects, Classes, Interfaces, Inheritance
  - Easily explained but no guidance on good design
- Principles
  - Separation of Concerns, Open-Closed, etc.
  - Help evaluate a solution
- Patterns
  - Guidance



© 2007 Google, Inc. All rights reserved.



## Patterns Origin – C. Alexander

### BED ALCOVE

#### Design problem

Bedrooms make no sense.

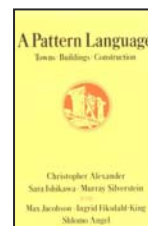
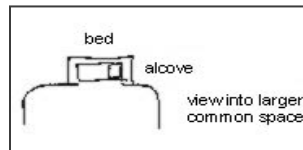
#### Forces

First, the bed in a bedroom creates awkward spaces around it: dressing, working, watching television, sitting, are all rather foreign to the side spaces left over around a bed.

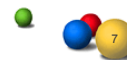
Second, the bed itself seems more comfortable in a space that is adjusted to it.

#### Solution

Don't put single beds in empty rooms called bedrooms, but instead put individual bed alcoves off rooms with other nonsleeping functions, so the bed itself becomes a tiny private haven.



© 2007 Google, Inc. All rights reserved.

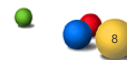


## Design Patterns

- “Mind sized” chunk of information (Ward Cunningham)
- Human-to-human communication
- Expresses intent (the “why” vs. the “how”)
- Observed from actual experience
- Not a firm rule
- Not copy-paste code



© 2007 Google, Inc. All rights reserved.

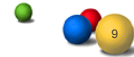


## Design Patterns in Software Development

- Substitute for industry standard vocabulary
- "Soft" around the edges
- Guide in the absence of strict rules
- Subject to hype cycle: "pattern" = "credible"



© 2007 Google, Inc. All rights reserved.



## 10 Years After GoF -- Why Revisit Patterns?

- New programming models bring new patterns.
- Patterns are expressed using the constructs of the underlying architectural style (e.g. OO, SOA).
- Good to document our knowledge when not all kinks are worked out yet.
- Patterns help discover higher levels of abstraction.
- Ultimately, some patterns can be implemented in the platform.



© 2007 Google, Inc. All rights reserved.



## New Programming Models in SOA

- Event-based, Asynchronous Programming
  - Explicit state management
  - Sequencing, timing uncertainty
- Declarative Programming
  - Execution path chosen at run-time
  - XSLT, Rules engines
- Object-Document Mapping
  - Analogous to O-R mapping: subtle, but important
- Process Modeling
  - Many concurrent, long-running instances
  - No two-phase-commit style transactions
- Protocol Design

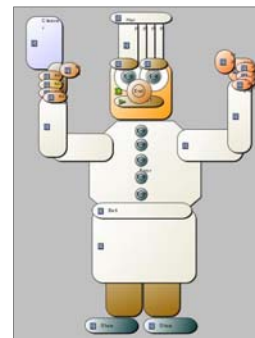


© 2007 Google, Inc. All rights reserved.



## “Doodeware” Only Limited Help

- Example
  - Graphical process editors
  - Graphical transformation editors
- We love pictures
- Programming in pictures tedious
  - Scalability issues
  - Diff, Merge mostly unsupported
- Often a thin veneer over a complex (or unfamiliar) programming paradigm



“EAI Art”



© 2007 Google, Inc. All rights reserved.

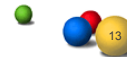


## New Patterns in SOA

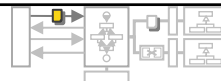
- Asynchronous Messaging Patterns
- Conversation Patterns
- Orchestration Patterns
- Process / Workflow Patterns
- Endpoint Patterns
- Security Patterns
- Architectural Patterns
- ...



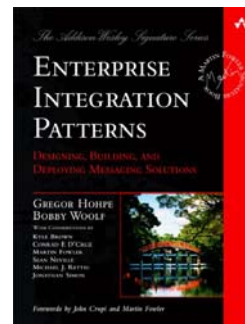
© 2007 Google, Inc. All rights reserved.



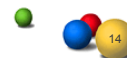
## Async. Messaging Patterns



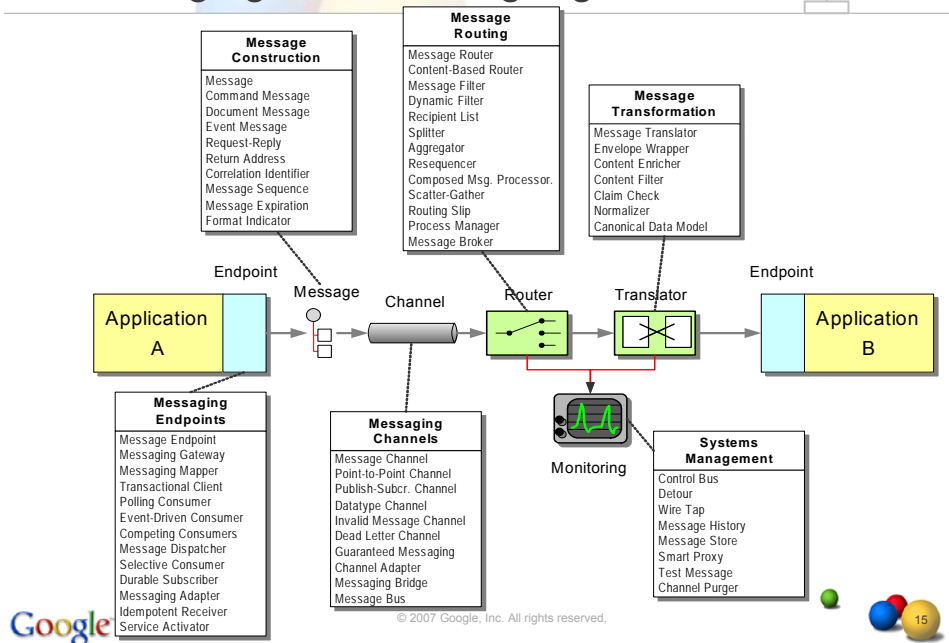
- Components communicate via message exchanges
- Simplified interaction model
- Compose solutions out of routing, transformation elements
- [www.eaipatterns.com](http://www.eaipatterns.com)
  - 65 patterns
  - Java, C#, Axis, JMS, MSMQ, TIBCO, BizTalk



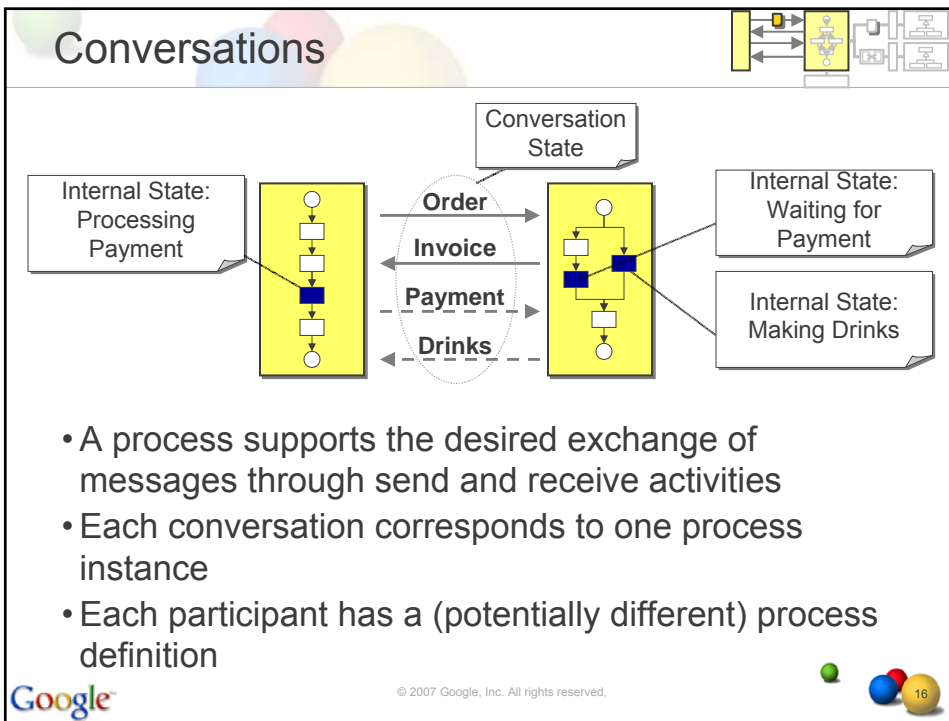
© 2007 Google, Inc. All rights reserved.



# Messaging Pattern Language

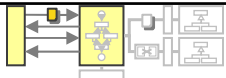


# Conversations





## Conversation Patterns

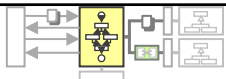


The diagram shows a central yellow box with a person icon, connected to several other boxes representing different conversation patterns. The patterns are arranged in two columns.

<b>Discovery</b> Dynamic Discovery Consult Directory Referral	<b>Multi-Party Conversations</b> Proxy Contingent Requests Reaching Agreement Leader Election
<b>Establishing a Conversation</b> Three-way Handshake Acquire Token First Rotate Tokens Address Verification	<b>Reaching Agreement</b> Receiver Cancels Sender Cancels Binding Request Binding Offer
<b>Basic Conversations</b> Fire-and-Forget Request-Response Request-Response with Retry Polling Subscribe-Notify Quick Acknowledgment	<b>Resource Management</b> Lease Renewal Reminder
	<b>Error Handling</b> Do Nothing Retry Compensating Action

Google © 2007 Google, Inc. All rights reserved. 17

## Process Patterns



The diagram shows a central yellow box with a person icon, connected to several other boxes representing different process patterns. The patterns are arranged in two columns.

- Long-running processes
- Multiple instances of a process template
- Orchestration  $\approx$  process across multiple services exposed as composite service
- WS-BPEL (Business Process Execution Language)

**Orchestration** describes the automated arrangement, coordination, and management of complex computer systems, middleware, and services. --Wikipedia

Google © 2007 Google, Inc. All rights reserved. 18

### Workflow Patterns (Subset)

- Exclusive Choice (XOR)
  - Choose one execution path from many alternatives

- Multiple Choice (OR)
  - Choose several execution paths from many alternatives

© 2007 Google, Inc. All rights reserved. 19

### Workflow Patterns (Subset)

- Synchronizing Merge
  - Merge many execution paths. Synchronize if multiple paths are taken.

- Multiple Merge
  - Merge many execution paths without synchronizing.

- Discriminator
  - Merge many execution paths without synchronizing. Execute the subsequent activity only once.

© 2007 Google, Inc. All rights reserved. 20

## Seems Trivial But...

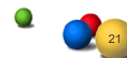
Support by standards can be spotty.

	XPDL	UML	BPEL	XLANG	WSFL	BPML	WSCSI
Parallel Split	+	+	+	+	+	+	+
Synchronization	+	+	+	+	+	+	+
Exclusive Choice	+	+	+	+	+	+	+
Simple Merge	+	+	+	+	+	+	+
Multi Choice	+	-	+	-	+	-	-
Synchronizing Merge	+	-	+	-	+	-	-
Multi Merge	-	-	-	-	-	+/-	+/-
Discriminator	-	-	-	-	-	-	-
Deferred Choice	-	+	+	+	-	+	+

Source: [www.workflowpatterns.com](http://www.workflowpatterns.com)



© 2007 Google, Inc. All rights reserved.

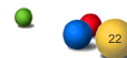


## Take-Away

- New architectural styles bring new patterns.
- Use pattern languages as design check list.
- Use pattern languages as catalog of options.
- Patterns often highlight subtleties that vendors like to glance over.
- Keep us from reinventing the wheel.
- Patterns harvested from actual usage.
- We are likely to experience a pattern renaissance.
- Share your knowledge.



© 2007 Google, Inc. All rights reserved.



## SOA Pattern Resources: Books



**Enterprise Integration Patterns**  
Gregor Hohpe, Bobby Woolf  
Addison-Wesley, 2004



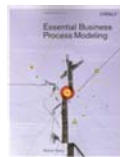
**Pattern-Oriented Software Architecture Vol.2**  
Schmidt et al  
Wiley, 2000



**Integration Patterns**  
Microsoft Patterns & Practices,  
2004



**Pattern-Oriented Software Architecture Vol.4**  
Buschmann et al.  
Wiley, 2007



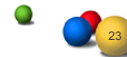
**Essential Business Process Modeling**  
Havey  
O'Reilly, 2005



**SOA Patterns**  
Arnon Rotem-gal-oz  
Manning, 2008 (?)



© 2007 Google, Inc. All rights reserved.



## SOA Pattern Resources: On-line

Integration Patterns	<a href="http://msdn.microsoft.com/practices/Topics/patterns">msdn.microsoft.com/practices/Topics/patterns</a>
Messaging Patterns	<a href="http://eaipatterns.com">eaipatterns.com</a>
Remoting Patterns	<a href="http://www.voelter.de/publications/books-rem.html">www.voelter.de/publications/books-rem.html</a>
Orchestration Patterns	<a href="http://orchestrationpatterns.com">orchestrationpatterns.com</a>
SOA Patterns	<a href="http://www.rgoarchitects.com/SOAPatterns/">www.rgoarchitects.com/SOAPatterns/</a>
Workflow Patterns	<a href="http://workflowpatterns.com">workflowpatterns.com</a>
Conversation Patterns	<a href="http://conversationpatterns.com">conversationpatterns.com</a> (WIP)
Service Interaction Patterns	<a href="http://serviceinteraction.com">serviceinteraction.com</a>
Data Integration Patterns	<a href="http://msdn.microsoft.com/practices/Topics/patterns">msdn.microsoft.com/practices/Topics/patterns</a>
Web Service Security Patterns	<a href="http://msdn.microsoft.com/practices/Topics/patterns">msdn.microsoft.com/practices/Topics/patterns</a>



© 2007 Google, Inc. All rights reserved.

