Component web-services architecture

Steve Battle, Tony Wiley
steve_battle@hp.com, anthony_wiley@hp.com
HP Labs

The challenge of Information Integration

'Enterprise Information Integration' is becoming more important than 'Enterprise Application Integration'

Frank Gilbane fourth forum XML, Paris 2001

In an information economy, improving the utility of information is synonymous with creating economic value.

Mohanbir Sawhney & Deval Parikh , *Where value lives in a networked world*, Harvard Business Review, Jan 01

Permeable organizations form external relationships easily.. They lower barriers to circulation of value.

Stan Davis, Christopher Meyer, Blur, 1998

Overview

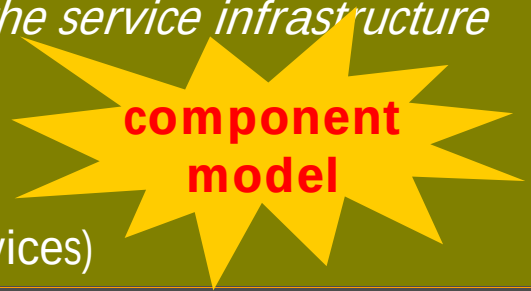
- Building on today's web-service technologies
- A document-centric view of information
- What if business objects were web-resources?

Inverting the business

The web services stack

Vertical services : *domain specific applications & processes*
- Financial services
- ERP
- CRM

Horizontal services : *providing the service infrastructure*
- UDDI registries
- Billing & payment services
- Service hosting
- Delivery (e.g. physical mail services)

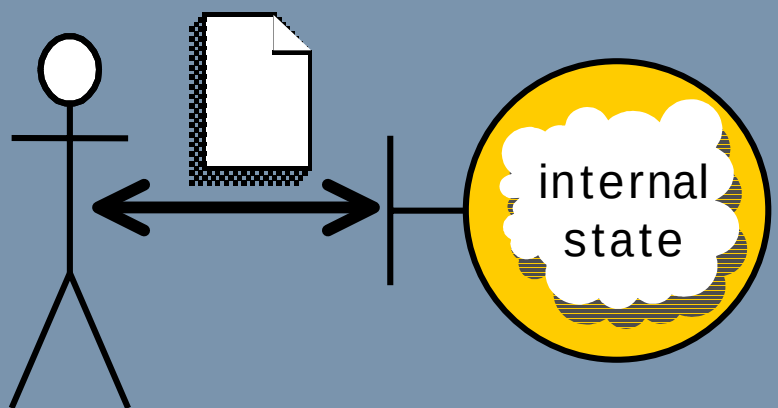


Standards
- UDDI - Universal Description, Discovery and Integration
- WSDL - Web Services Description Language
- SOAP - Simple Object Access Protocol
- XML - eXtensible Markup Language

It’s a document centric world *- or is it?*

- *Document exchange*
- *Document state*
- *Collaborative documents*

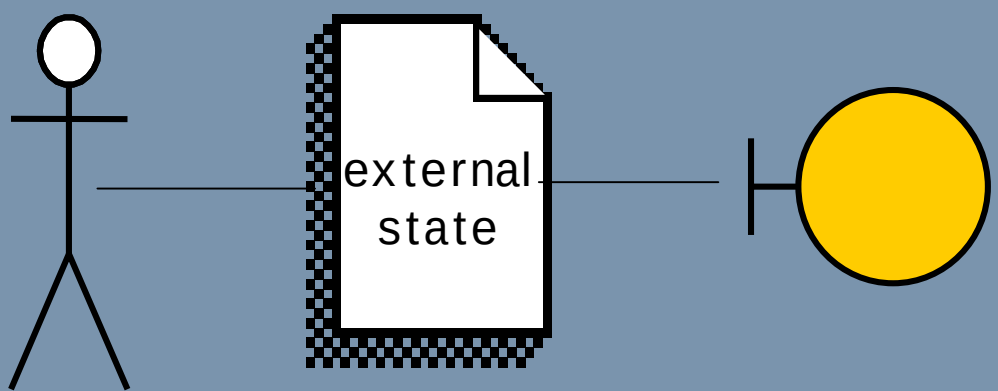
Web services are defined by document exchange.
But information assets remain locked away behind the public interface.



Objectifying service state

- *Document exchange*
- *Document state*
- *Collaborative documents*

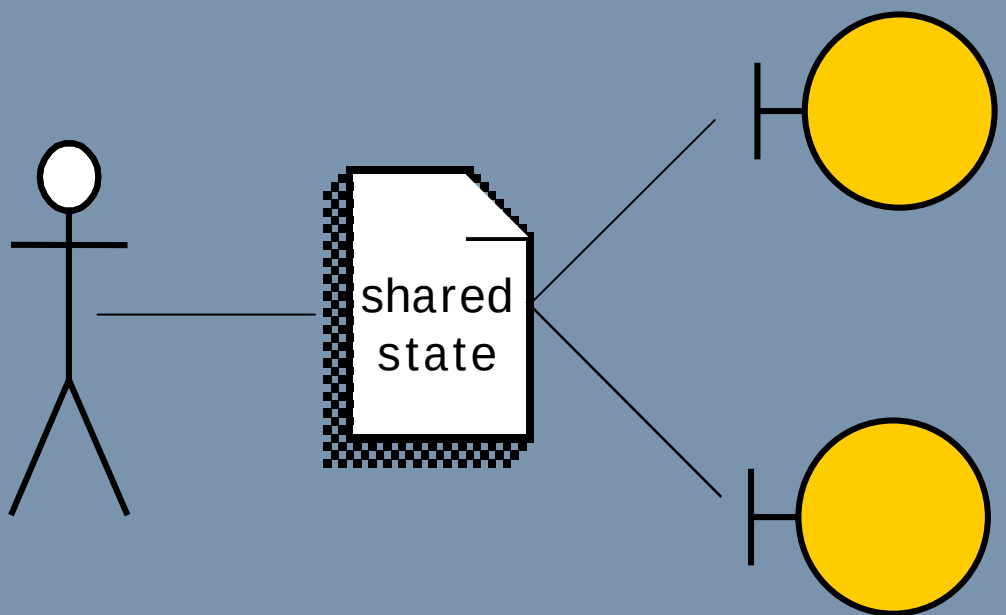
Service state is externalised in the form of a document, representing a business object. It becomes a web resource.



Information sharing

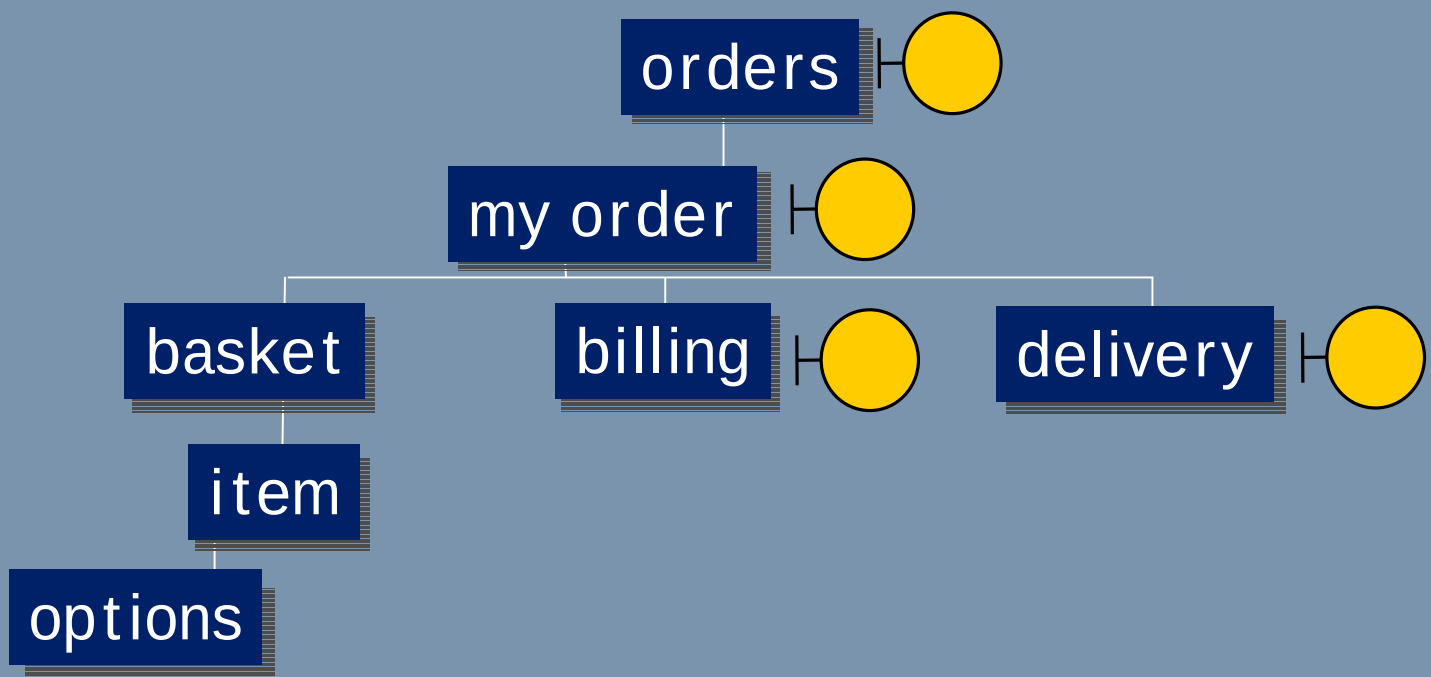
- *Document exchange*
- *Document state*
- *Collaborative documents*

The state can now easily be shared, enabling cross-enterprise workflows and collaborative business models.



Document parts

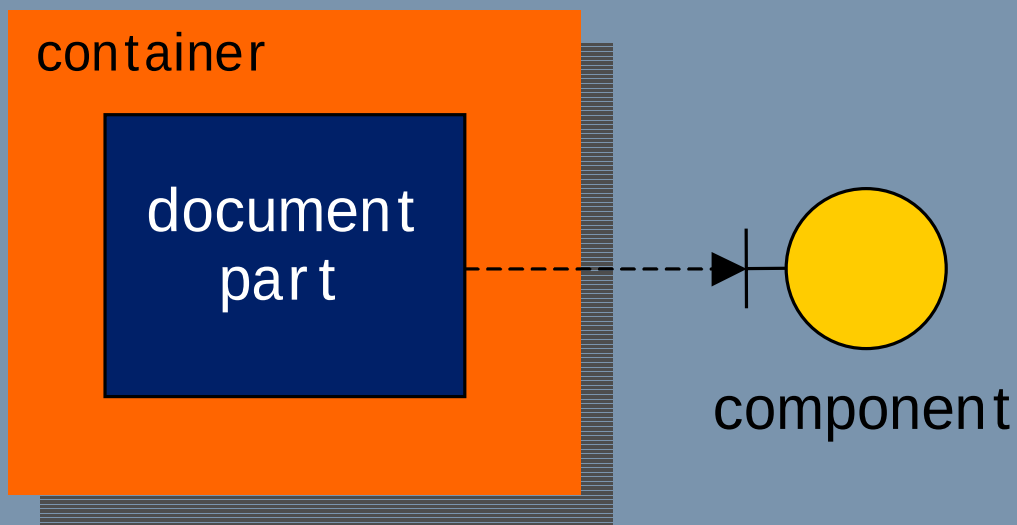
A compound document may have many parts, aggregating content used by different services.



Adding behaviour to documents

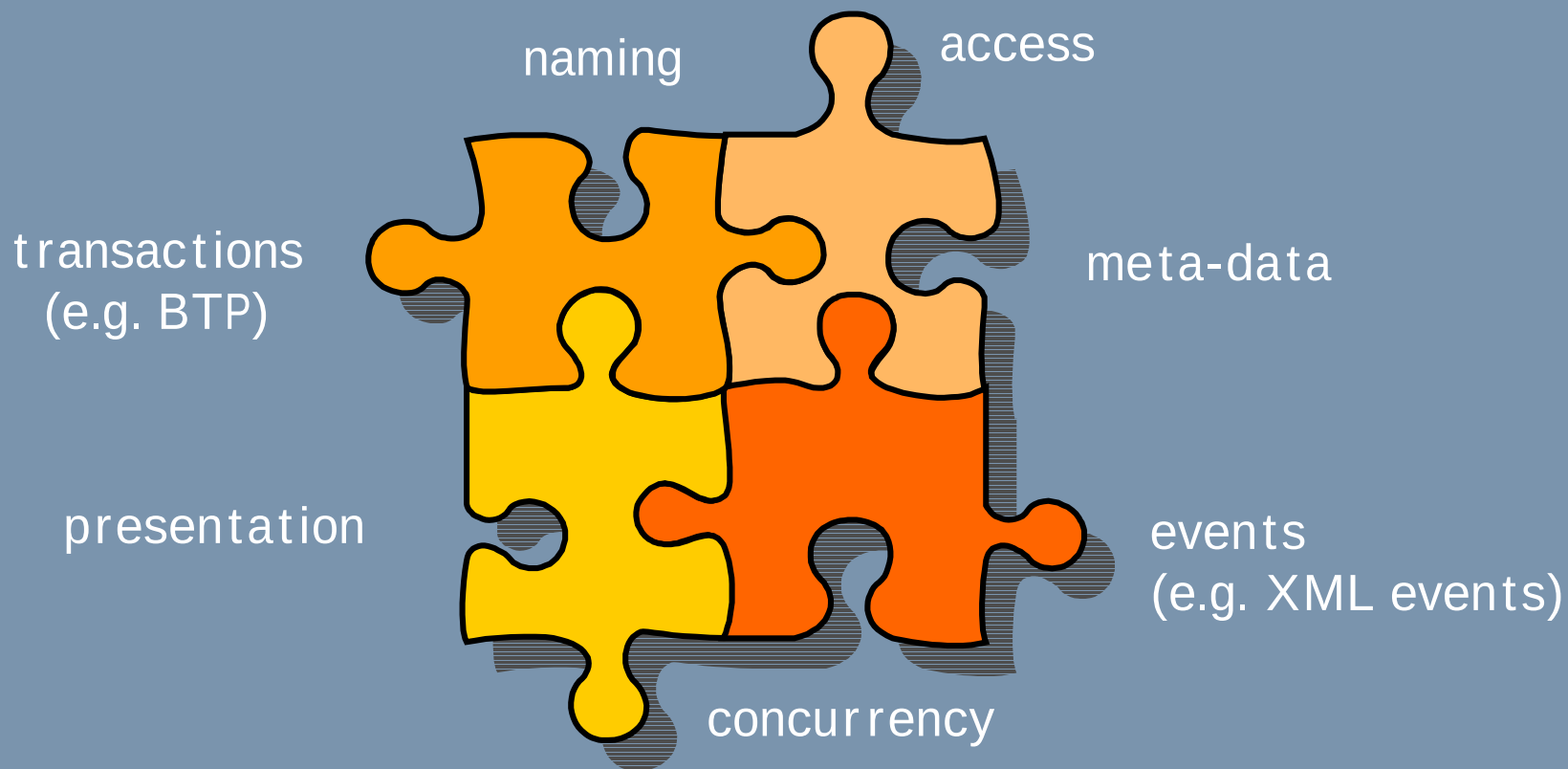
Component services

- Web services that support a prescribed set of component interfaces
- A part can use different services for different operations
- Services are composed within compound documents

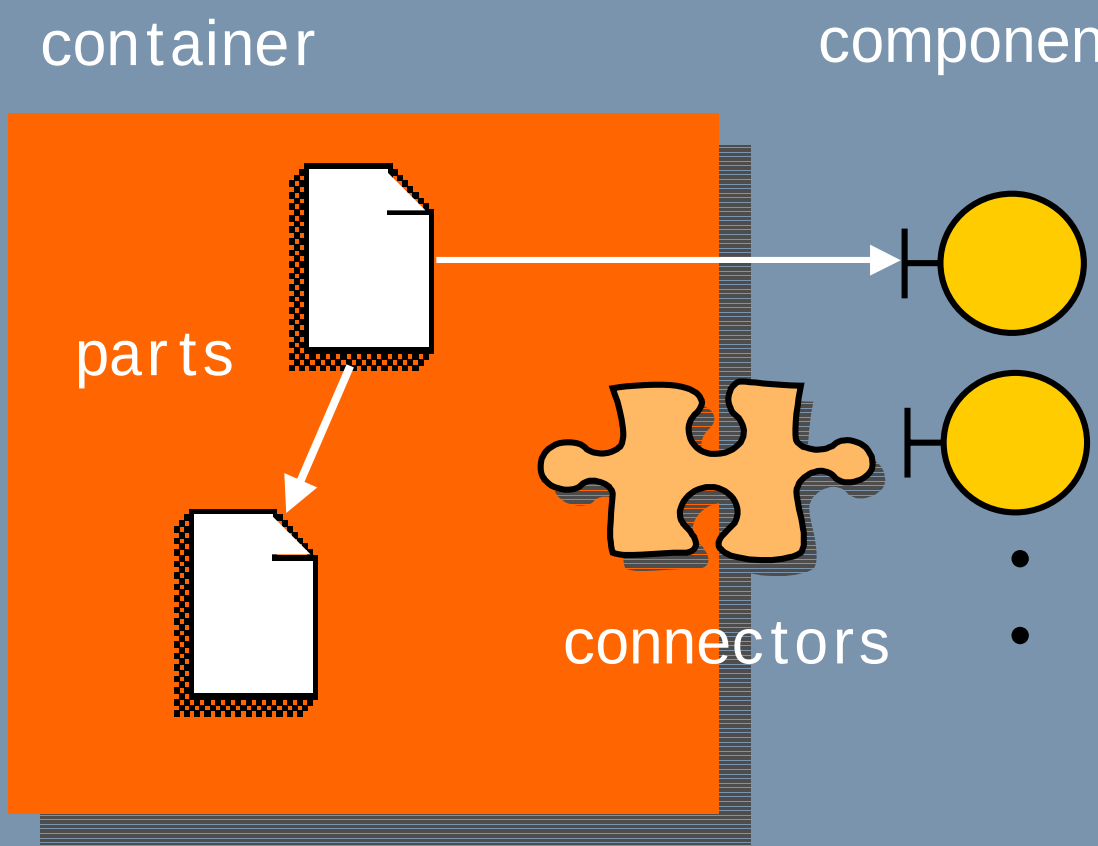


Component inter-operability

Components are defined by the connectors they support.
These will be industry standard, open interfaces.



Component architecture



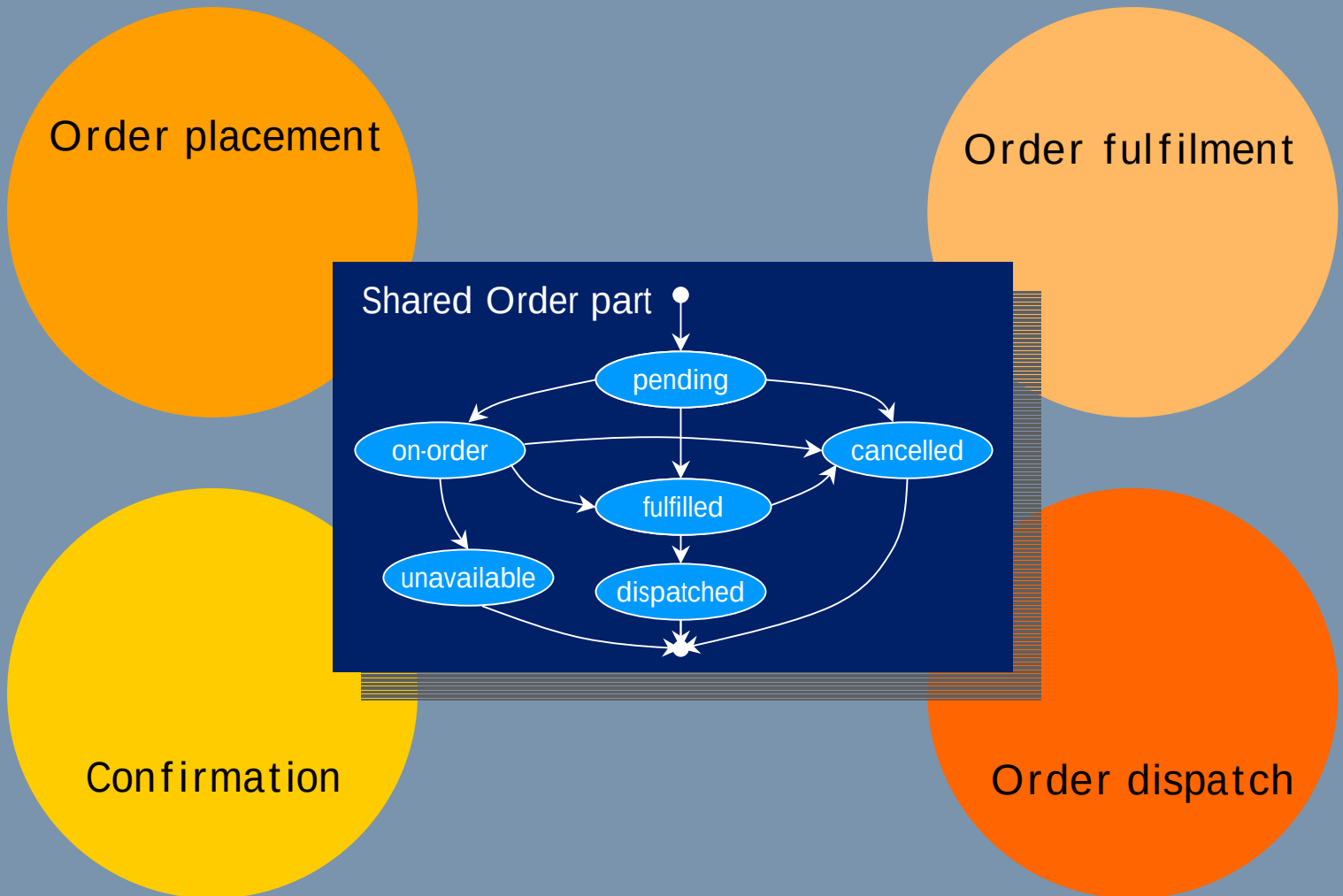
What is it good for?

Shared objects are useful if you..

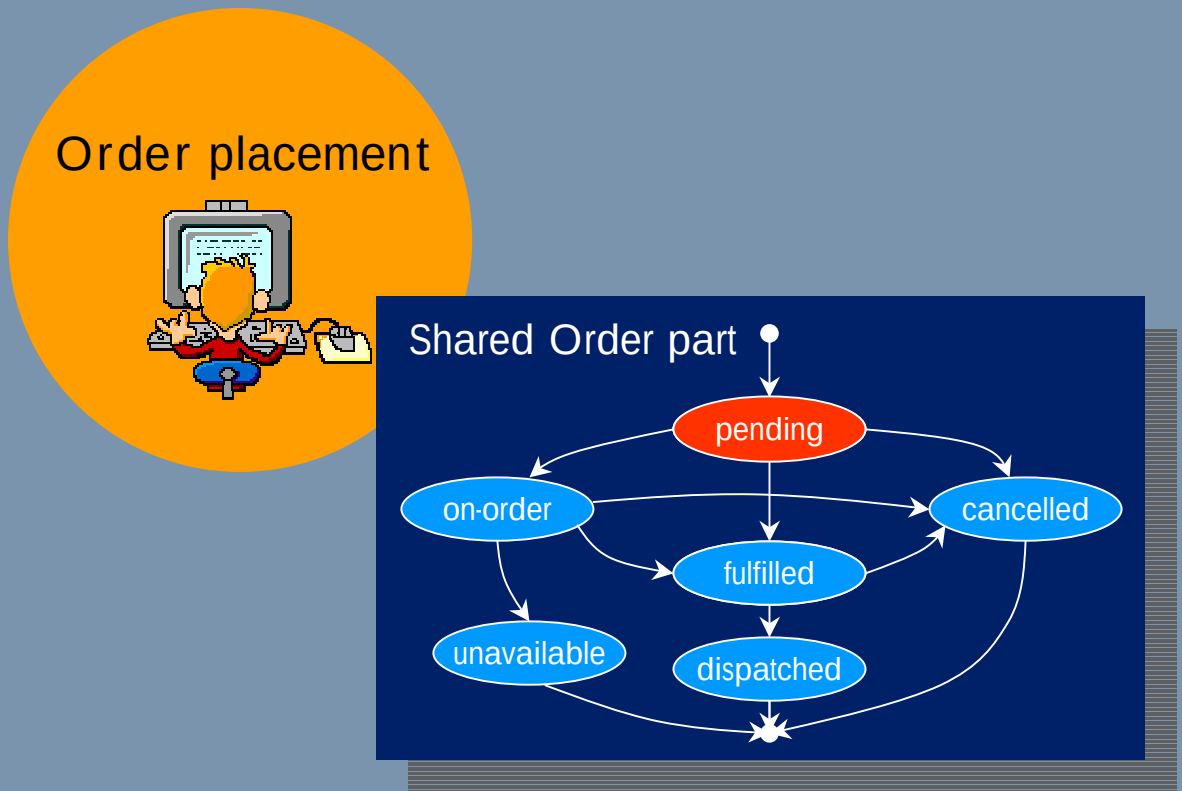
- need to outsource specific business functions.
- have a high degree of shared information.
- have a collaborative business process.
- want to create a new *virtual* enterprise.

The following example is used to highlight these points…

Book-buying at Nile.com



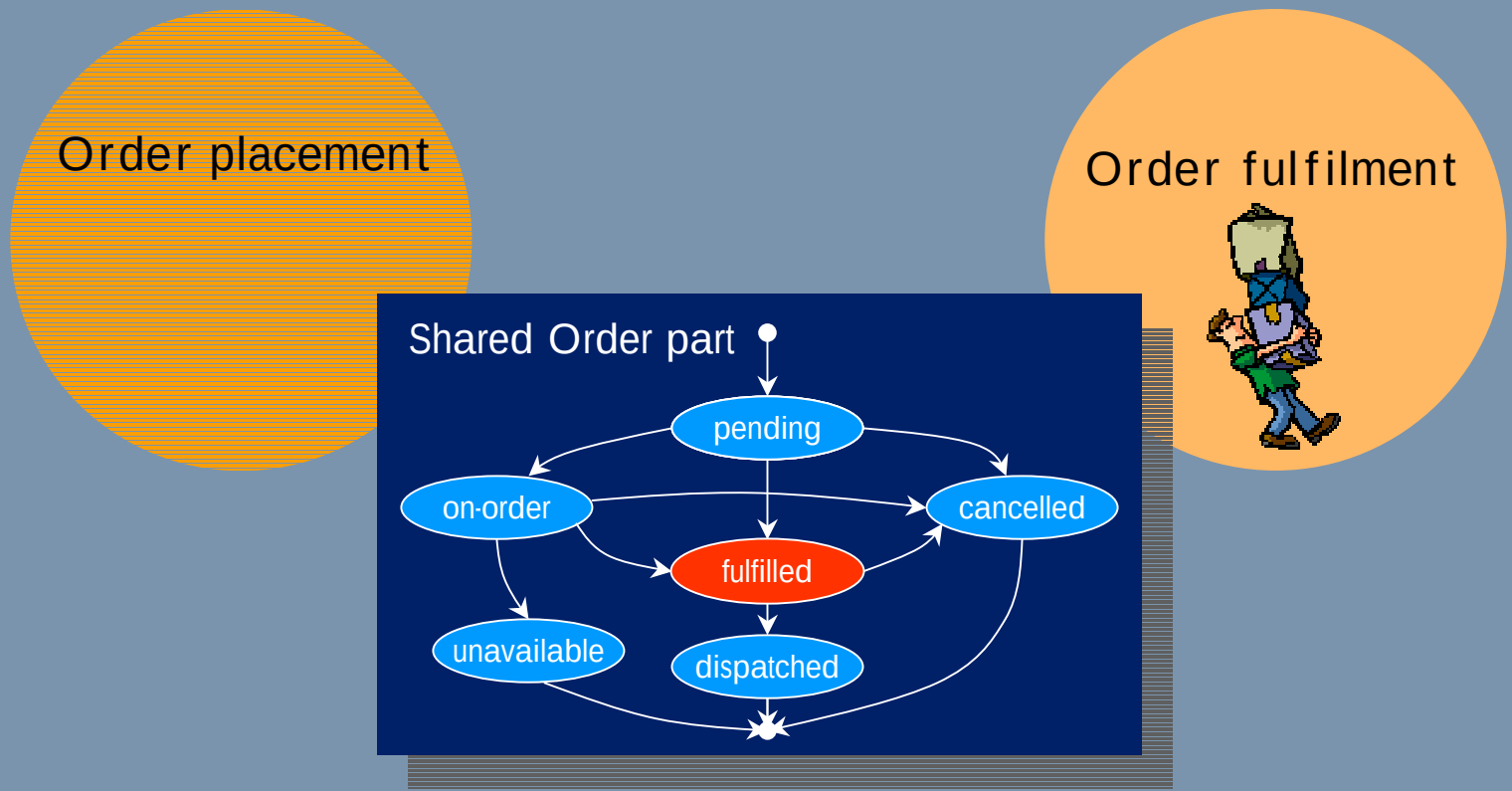
Order placement



Customer places order with Nile books:

- Kicks off ordering process..
- creating a new shared order part in state *pending*

Order fulfilment



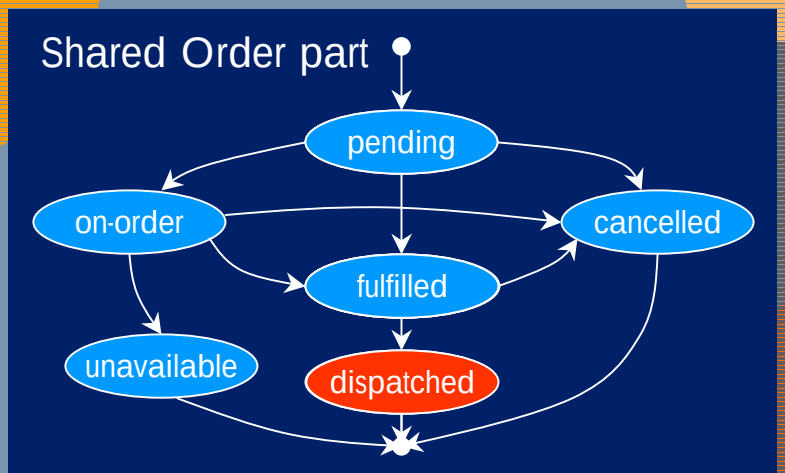
Order fulfilment: on pending orders..

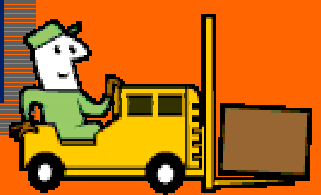
- Locate and package books
- Updates order part (fulfilled)

Order dispatch

Order placement

Order fulfilment





Order dispatch

Order dispatch:

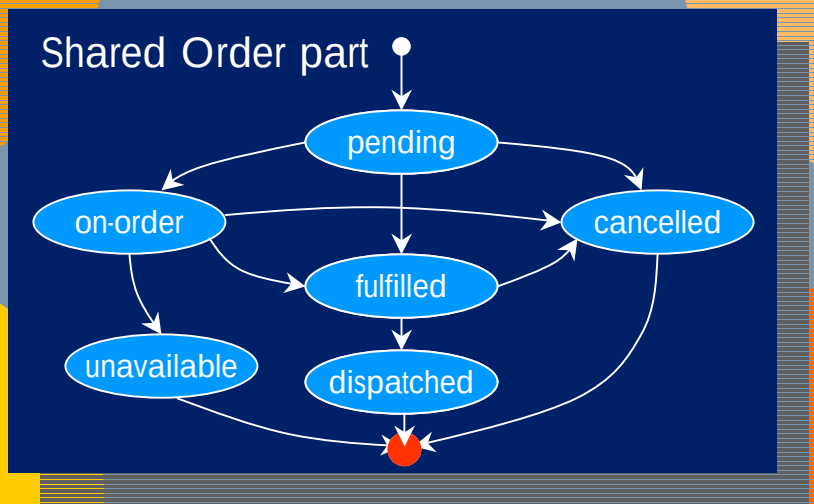
fulfilled orders are dispatched

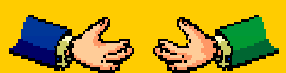
once loaded onto carriers van, order is *dispatched*

Order confirmation

Order placement

Order fulfilment





Confirmation

Confirm order:
emails customer

Order dispatch

Conclusions …

- *Document centric approach to information integration.*
- *What next? Working towards open standards and tools.*

For further information, please contact the authors

or alternatively visit our web-site:

http://www-uk.hpl.hp.com/people/stebat/components