

Business Document Model

An ECM can only be as good as it's model

Something about me

- Heiko Robert
- ~15 years ECM, 8 years Alfresco



Agenda

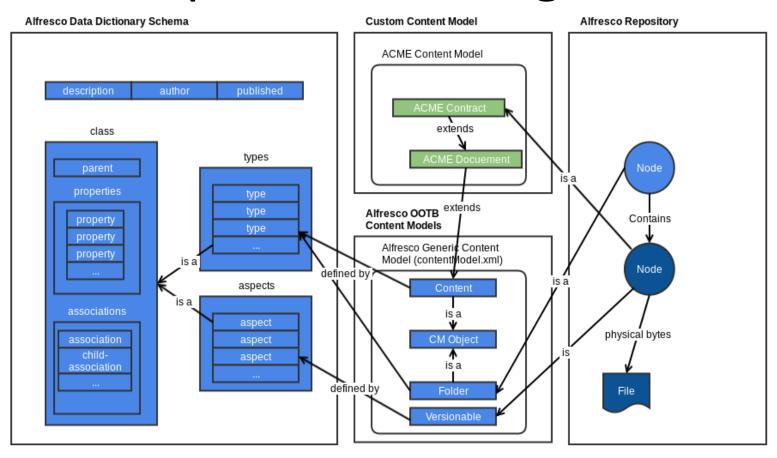
Motivation

Concepts

How to use



Oops! ... I Did It Again





Hands Up!

- Who already implemented a custom model?
- Who tried to optimize a custom model?
- Would you do it again if you could avoid it?



Business Documents

Why starting always from scratch?





Why?

should we share a basic, common model

- Easier combination of modules / apps / services
- Learn from each other
- Easier upgrade path, costs savings
- Share customization is expensive
- Very hard to fix errors in model





Requirements

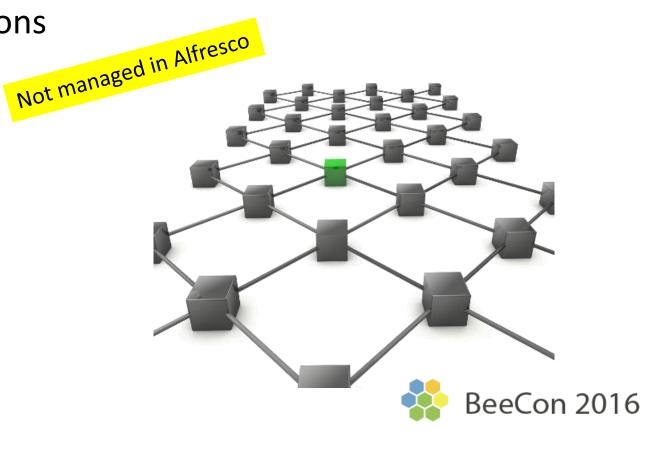
- Filing and finding documents
 - accounting documents
 - correspondence
 - cases and processes

— ...



Requirements

- Associate documents to business entities
 - Organisations
 - Contacts
 - Contracts
 - Products
 - Projects
 - Cases
 - **—** ...



Considerations

- Alfresco is not a relational database
 - No: relations, multi entity joins
 - Instead: associations, properties
 - Search capabilities may not meet expectations
- Alfresco Types are static
 - can be used as a vehicle for mandatory aspects
- Attach Aspects for Perspectivs
 - Business Partner, Contact, Tenant, Project, Product

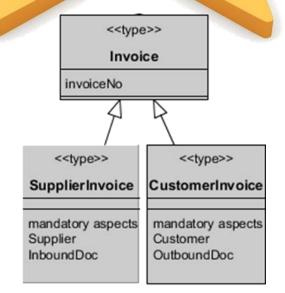




Direction Matters:

Inbound, Outbound

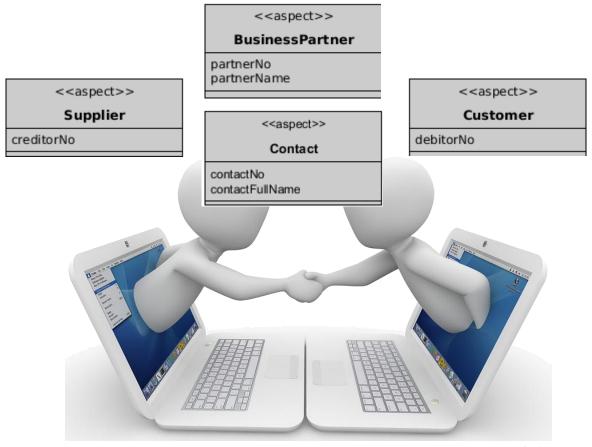
<aspect>>
InboundDoc
receivedDate
receivedNo



<aspect>>
OutboundDoc
sendDate



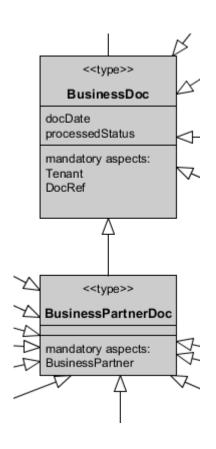
Roles

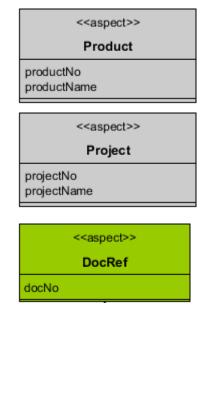




Business Partner Doc

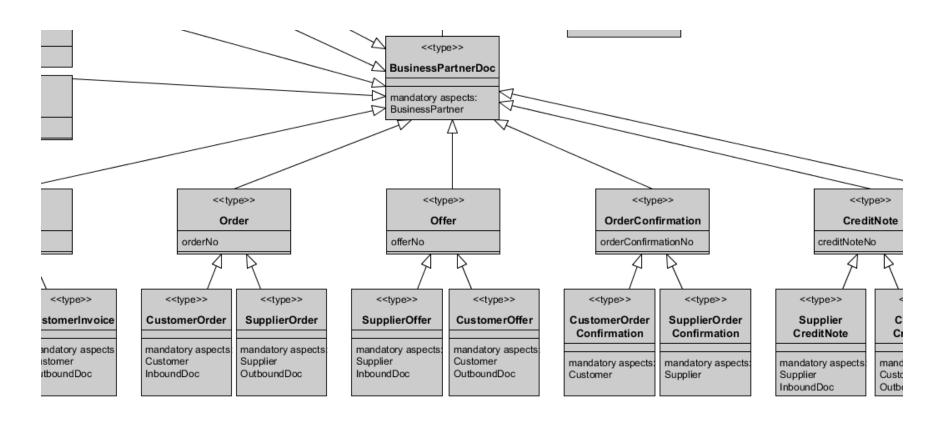
The Base Types







Business Partner Doc





Do's & Don'ts

- Avoid to copy master data
 - Copy as few data from your leading system as possible into Alfresco properties
 - →Use foreign key references like customer no
 - → Copied data may get inconsistent
- Avoid using docs without direction
 - Types without direction should only be used for migration (invoice, delivery note, order, offer, ...)

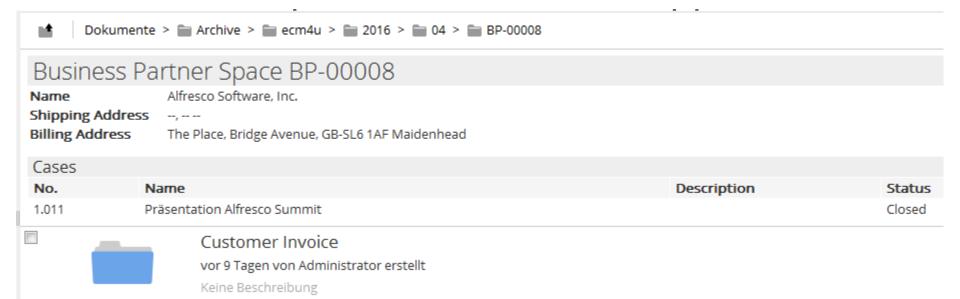


What's *not* in the box?

- Share Forms
 - No support for flexible inheritance / reuse

Tips

- Use folder rules
 - Adding required aspects (e.g. project folder)



Rechnungsadresse: The Place, Bridge Avenue,

GB-SL6 1AF Maidenhead Kunden-No.: DE-00001



Next Steps

- You should
 - check out the model and start to use it
 - give Feedback!
 - collaborate
- We
 - would be happy about any cooperation
 - may switch from "mandatory aspects" to a policy based logic to be more flexible / to remove specialized types



< <aspect>> Tenant</aspect>	< <aspect>> BusinessPartner</aspect>	< <aspect>> Contact</aspect>			< <folder>> BusinessPartnerSpace</folder>
tenantID	partnerNo partnerName	contactNo contactFullName			mandatory aspects: BusinessPartner
< <aspect>> InboundDoc receivedDate receivedNo</aspect>	< <aspect>> Supplier creditorNo</aspect>	< <aspect>> Customer debitorNo</aspect>	< <aspect>> Region regionCode</aspect>		< <folder>> ProjectSpace Project</folder>
< <aspect>> OutboundDoc sendDate</aspect>	<aspect>> Product productNo productName</aspect>	<aspect>> Project projectNo projectName</aspect>	<aspect>> Language languageCode</aspect>		<pre>Customer <<folder>> ProductSpace Product</folder></pre>
<aspect>> ContractDetails</aspect>					
contractNo < <aspect>> Revsion</aspect>		< <folder>:</folder>	·		< <aspect>> DocRef docNo</aspect>
revisionNo		caseNo caseObjectType caseObjectNo	< <type>></type>		<u> </u>
		caseType caseProcessStal casePersonInCh		quest	<aspect>> RecordData</aspect>
< <type>> AcceptanceCertificate</type>		caseTeamInCha < <type>>></type>	<type>> Documentation Revision</type>		recType recRetentionEndDate recRetentionStartDate recRetentionMinPeriod recRetentionMaxPeriod recRetentionDetionDate recRetentionDetionDate
< <type>></type>		docDate processedStatus	< <type>></type>		
NoticeToTerminate ContractDetails		mandatory aspect	cts:		< <aspect>> Hashable</aspect>
< <type>> Contract</type>			<type>> Calculation</type>		hashValue hashType timestamp
contractType ContractDetails		< <type>>> BusinessPartn</type>		_	
< <type>> Mail mailType</type>		mandatory asper BusinessPartner	cts:		
< <type>>> Invoice</type>	< <type>>> Order</type>	< <type>></type>	< <type> OrderConfir</type>		
invoiceNo	orderNo	offerNo	orderConfirmat		deliveryNoteNo
<type>> SupplierInvoice CustomerInv</type>	CustomerOrder Supplier	Order SupplierOffer C	<type>></type>	< <type>></type>	<type>></type>
mandatory aspects Supplier InboundDoc mandatory as Customer OutboundDoc	Customer Supplier	mandatory aspects: Supplier CL InboundDoc	andatory aspects ustomer arboundDoc mandatory aspects: Customer	Supplier Supplier C	mandatory aspects: Customer Supplier OutboundDoc InboundDoc mandatory aspects: Customer OutboundDoc

Q & A

Thank you!

