



Basic Forms Modeling

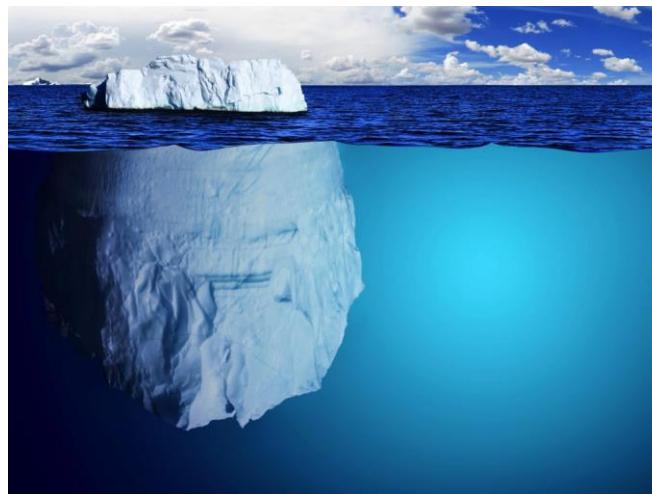
Oct 2015

Backbase Forms

- Martijn Scheers
 - Forms Modeller & trainer
 - Some programming experience
-
- Feel free to interrupt
 - Please introduce yourself!
(what are your expectations)

Goal of this training

- Deliver a basic understanding of Backbase Forms



Expectations

- Theory
- Modeling
 - Lab exercises: Building your first Forms application
 - Questions
- Practicing some of the different ways to model

Basic Forms Modeling

Day 1: Basic Forms Modeling

Day 2: Complex Domain model

Day 3: Documents

Agenda – Basic Forms Modeling

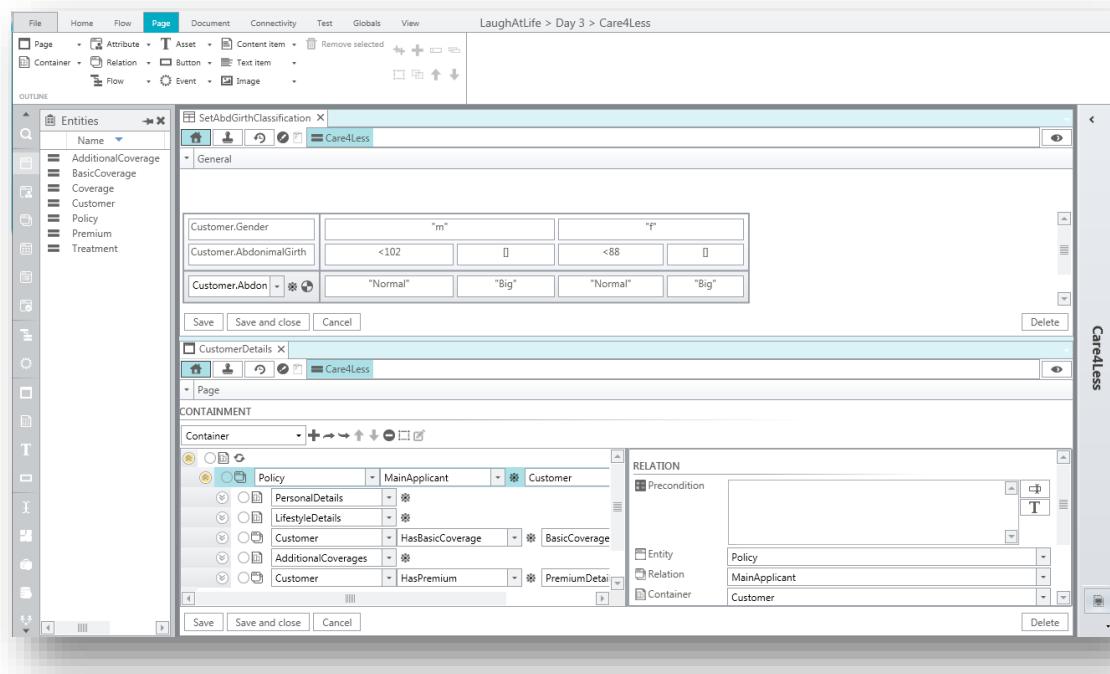
- **Introduction to Backbase Forms**
- Domain Model
- Interaction (pages and flows)
- Business logic

Introduction to Backbase Forms

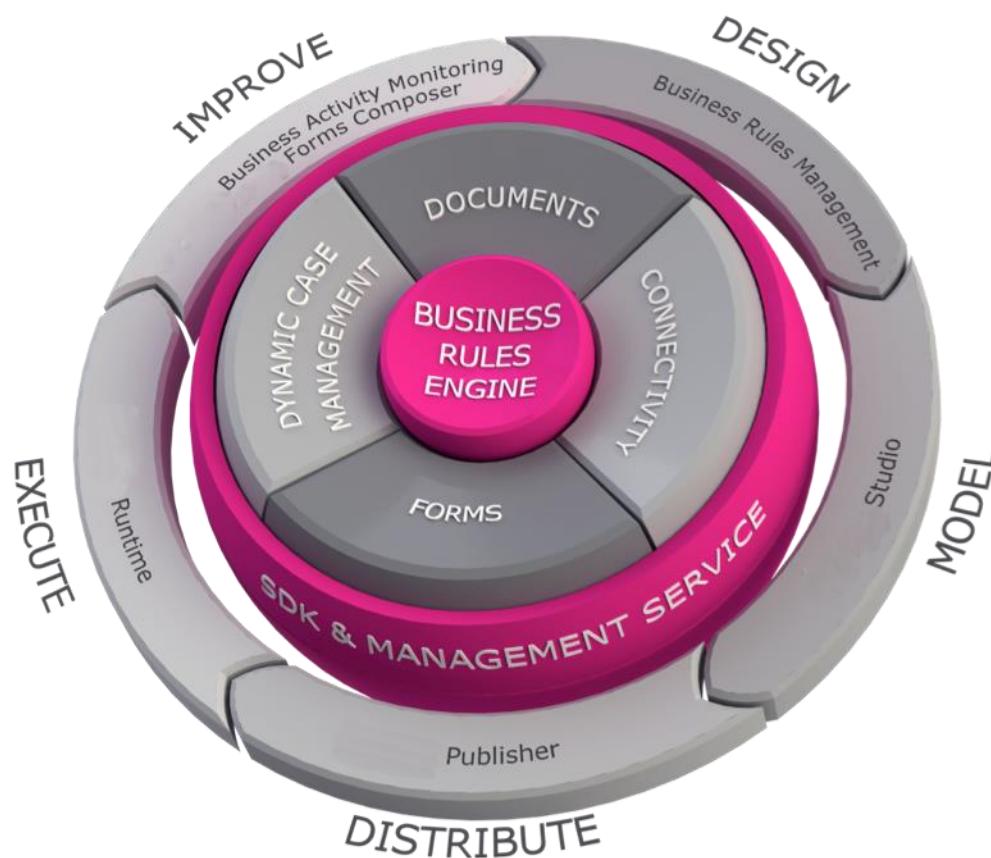
A software suite aimed at analysts and ‘business users’ to model, execute and adapt business applications.

-Studio

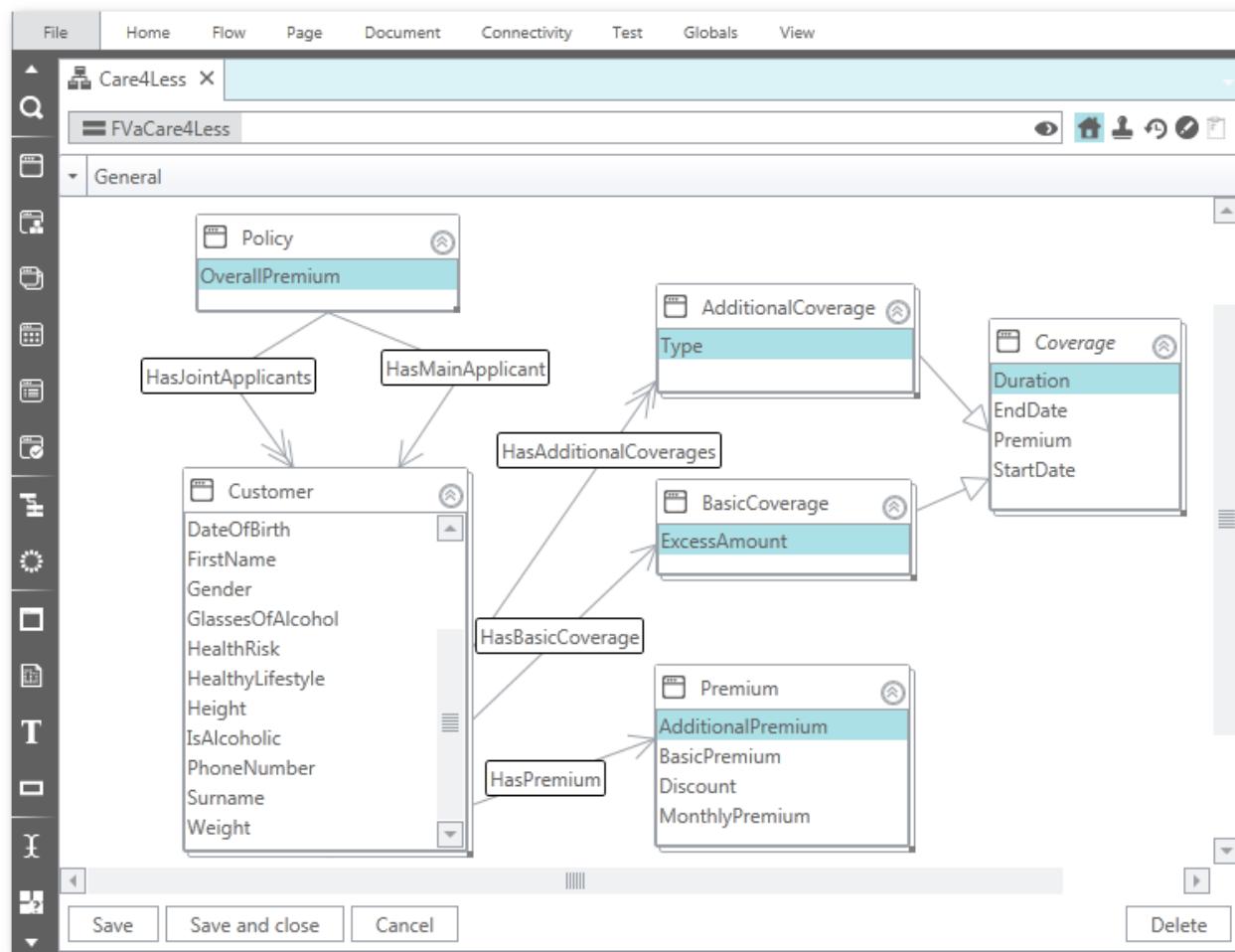
-Runtime



Development lifecycle



Domain Model



Knowledge as rules (1)

- Business Rules

IF

`Customer.Age > 21 OR Customer.IsMarried`

THEN

Customer ▾ AgeCategory ▾

IS

`"Adult"`

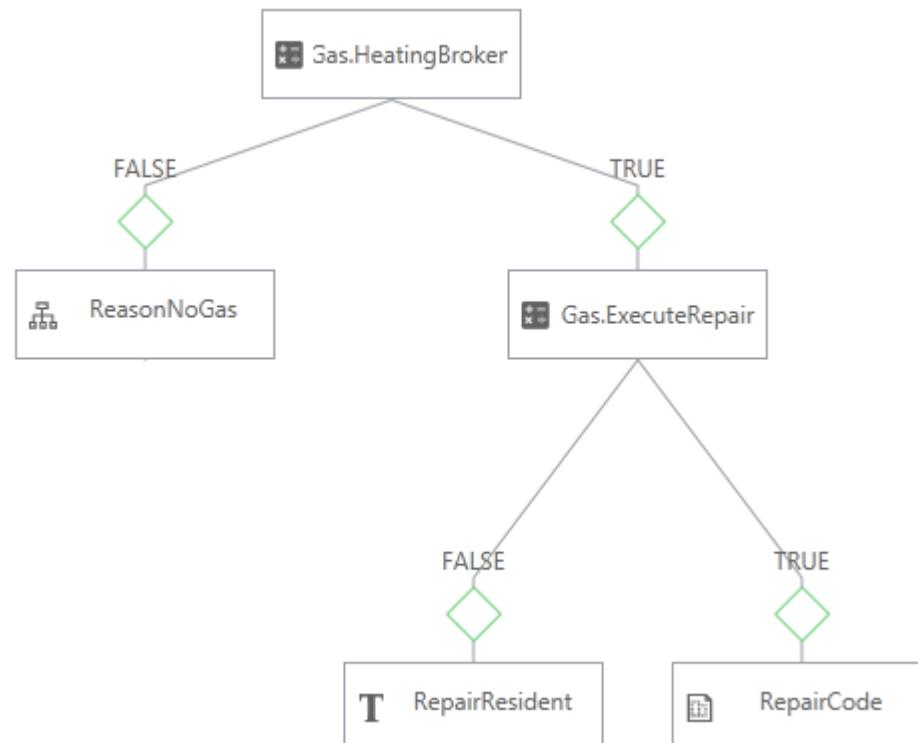
- Decision Tables

Car.NrOfDoors	3	5
Car.NrOfSeats	>2	*
Car.EngineCC	<=2000	>2000
Car.Boot	"Small"	"Large"
Car.Type	*	"Small" "Large"
	"Sport" "HatchBack" "Sedan"	"Sport" "Sedan" "Saloon"



Knowledge as rules (2)

- Decision Trees



Forms

Lening aanvragen of bestaande lening wijzigen

1. Uw situatie 2. Inkomsten en uitgaven 3. Resultaat 4. Gegevens 5. Afronden

Informatie over uw financiële situatie is van belang om te bepalen of en hoeveel u bij ons kunt lenen.

Thuishuisituatie

Burgerlijke staat: Maak een keuze

Voor hoeveel kinderen bent u financieel verantwoordelijk? Maak een keuze

Uw inkomsten

Werksituatie: Maak een keuze

Uw uitgaven

Woonlasten

Woonsituatie?

- Ik ben huiseigenaar
- Ik huur een woning
- Ik ben inwonend / thuiswonend

Alimentatie

Betaalt u alimentatie? Ja Nee

Andere leningen

Heeft u nog andere leningen? (inclusief ING-leningen)

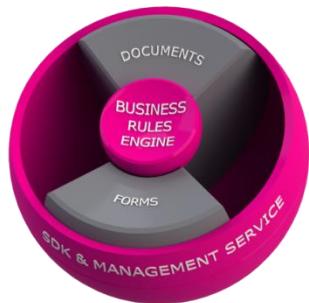
Ja Nee

Volgende **Vorige**

Your email: e.o. name@domain.com



Documents



The image displays two windows illustrating document management and generation.

Left Window: Confirmation

This window shows a hierarchical tree structure for a document template named "Confirmation". The tree includes sections like "Letter", "Header", "Logo", "Your_address", "Date", "Body", and "TermsAndConditions". The "Your_address" node is currently selected. To the right of the tree is a preview pane showing the generated document content:

```

Policy.MainApplicant_
Veemarktkade 8
5222 AE Den Bosch

TODAY

Dear Mr. [REDACTED] Policy.MainApplicant

Thank you for your application.

Enclosed you will find the q
persons. The total monthly pr
months, starting Policy.Mai

We take this opportunity to

If you have any questions, p
info@care4less.com. Our insu

Yours faithfully,

John Doe
Assistant General Manager,
Sales Planning & Operations
  
```

At the bottom of the preview pane are buttons: "Save", "Save and close", and "Cancel".

Right Window: Document.pdf - Adobe Reader

This window shows a PDF document titled "Document.pdf" viewed in Adobe Reader. The PDF contains several sections of text and some form fields:

- Indicatiestelling**
- Persoonsgegevens**
 - Voorletters: Bram
 - Tussenvoegsel: van
 - Gebodenam: Aalst
 - Geborendatum: 23-09-1982
 - Burger Service Nummer (BSN): 121
- AWBZ - Aanleiding**

Aanleiding	Diagnose gesteld?	Wanneer is de diagnose gesteld?
test	Ja	Tussen 3 en 6 maanden geleden
- AWBZ - Problemen bij bewegen en verplaatsen**

Problemen waardoor u aan de orde zijn bij bewegen en/of verplaatsen:
Trap op en af met voorwerp
Coördinaten benen en voeten (staan lopen/richting houden)
Lopen binnenshuis

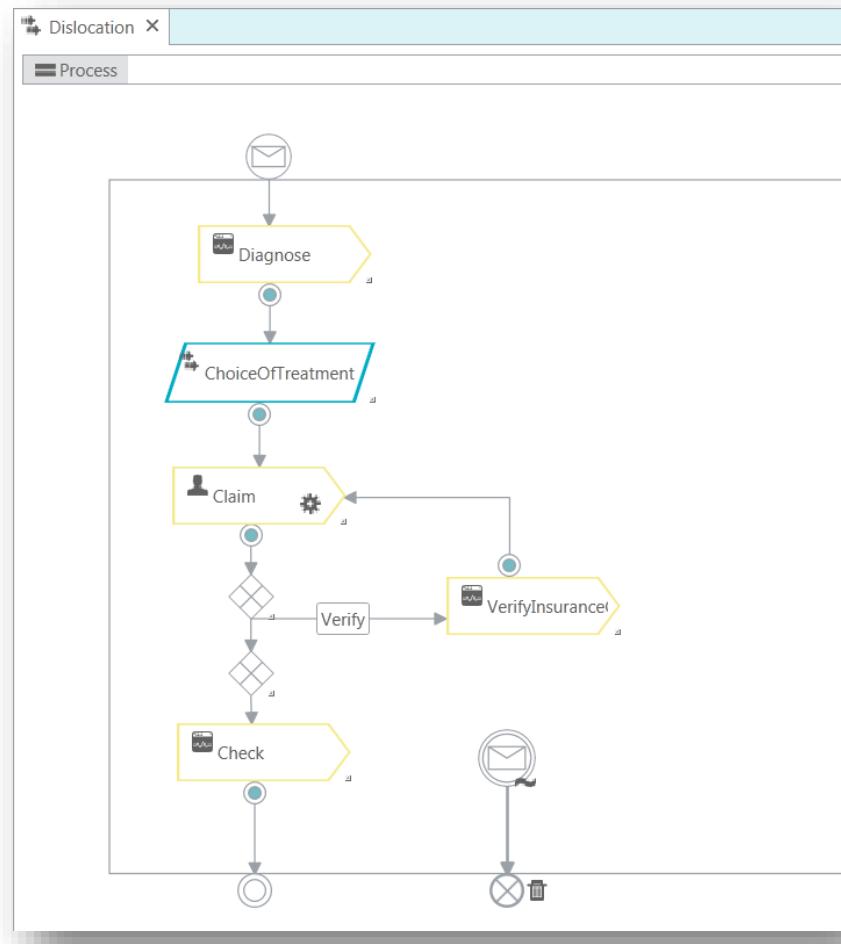
Opmaking die u hierbij gegeven heeft: Van alles!
- Aanvullende informatie**

POLISNUMMER	UZOVIN nummer zorgverzekerar
Hoe wilt u uw zorg regelen?	Ik wil dat het zorgkantoor mijn zorg regelt (zorg in natura)
Hoe lang denkt u de zorg nodig te hebben?	3 of 4 jaar
- Afroeling**

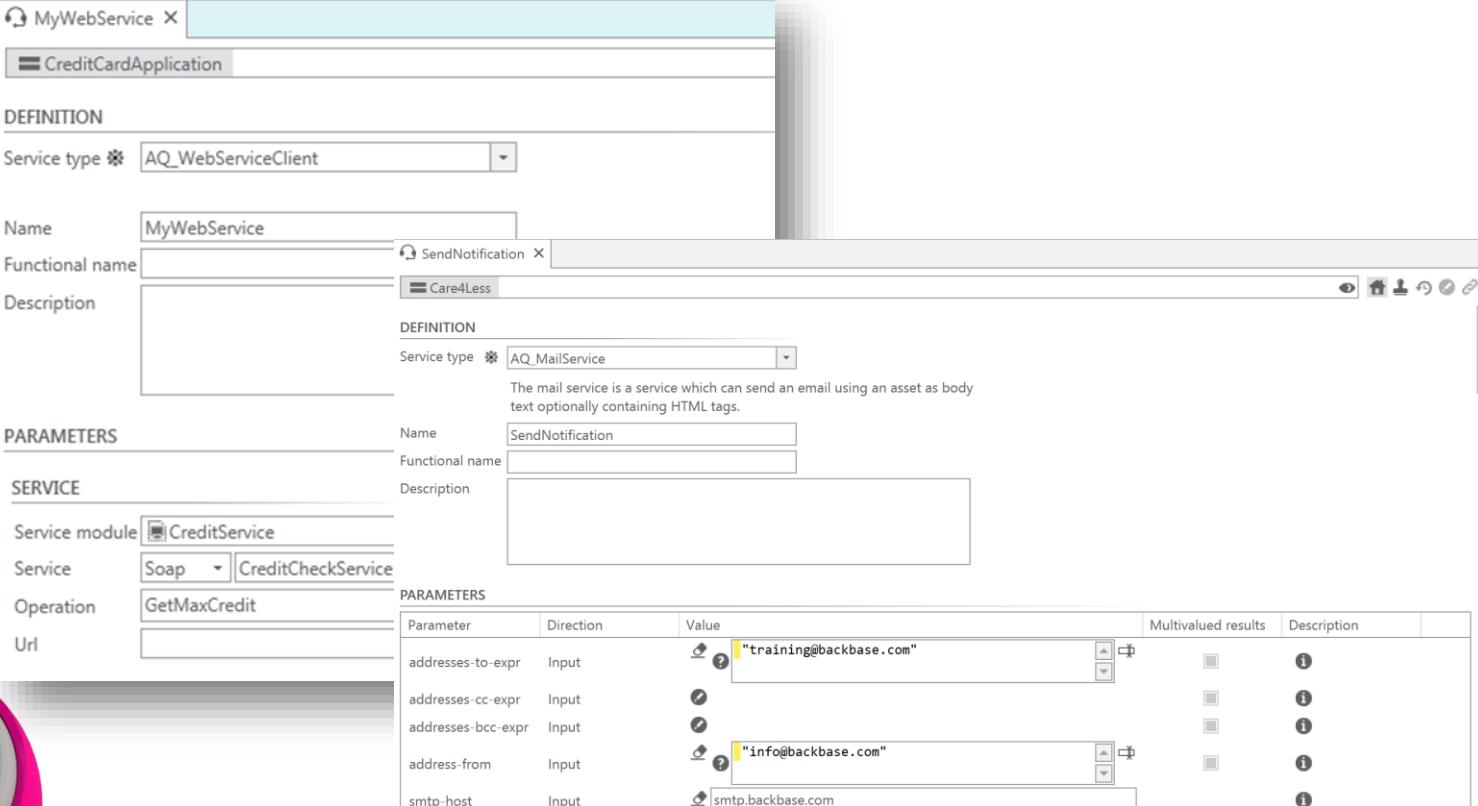
U geeft een huisarts **geen** toestemming om (medische) gegevens aan het CIZ te geven, als dat nodig is voor het indicatieonderzoek.
U geeft uw andere behandelarts **geen** toestemming om (medische) gegevens aan het CIZ te geven, als dat nodig is voor het indicatieonderzoek.
U geeft personen of organisaties die u op dit moment zorg verlenen **geen** toestemming om (medische) gegevens aan het CIZ te geven, als dat nodig is voor het indicatieonderzoek.
U geeft het CIZ **geen** toestemming om uw (medische) gegevens die al bekend zijn bij het CIZ te gebruiken bij het indicatieonderzoek.
- Indicatielijst**

Ziekenhuisid	Omschrijving	Omschrijving in WR	Registratiedatum	Einddatum
Ventilator	Andere woning	150	01-01-2012	01-06-2018
Huishoudelijke verzorging	Poetsen	5	01-01-2012	01-01-2015
Verpleging	Pleisteren plakken	10	01-01-2012	01-01-2016

Dynamic case management



Connectivity



The screenshot shows the Backbase Connectivity configuration interface. At the top, there's a navigation bar with tabs like 'MyWebService' and 'CreditCardApplication'. Below it, the 'DEFINITION' section is expanded, showing:

- Service type: AQ_WebServiceClient
- Name: MyWebService
- Functional name: SendNotification
- Description: Care4Less

Under 'DEFINITION' for the 'SendNotification' service, it says: "The mail service is a service which can send an email using an asset as body text optionally containing HTML tags."

The 'PARAMETERS' section contains:

- Service module: CreditService
- Service: Soap / CreditCheckService
- Operation: GetMaxCredit
- Url: (empty)

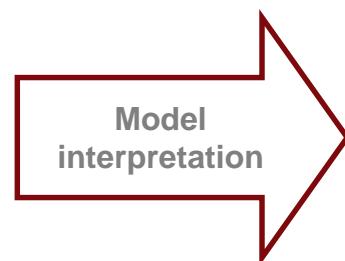
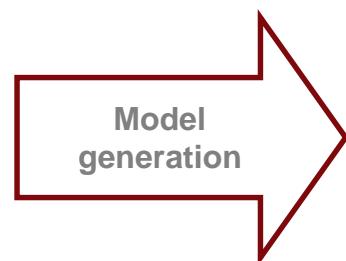
The 'SERVICE' section is collapsed.

On the right, under 'PARAMETERS', there's a table:

Parameter	Direction	Value	Multivalued results	Description
addresses-to-expr	Input	"training@backbase.com"		i
addresses-cc-expr	Input			i
addresses-bcc-expr	Input			i
address-from	Input	"info@backbase.com"		i
smtp-host	Input	smtp.backbase.com		i

A circular navigation menu on the left includes: DOCUMENTS, DYNAMIC CASE MANAGEMENT, BUSINESS RULES ENGINE, CONNECTIVITY, FORMS, and SDK & MANAGEMENT SERVICE.

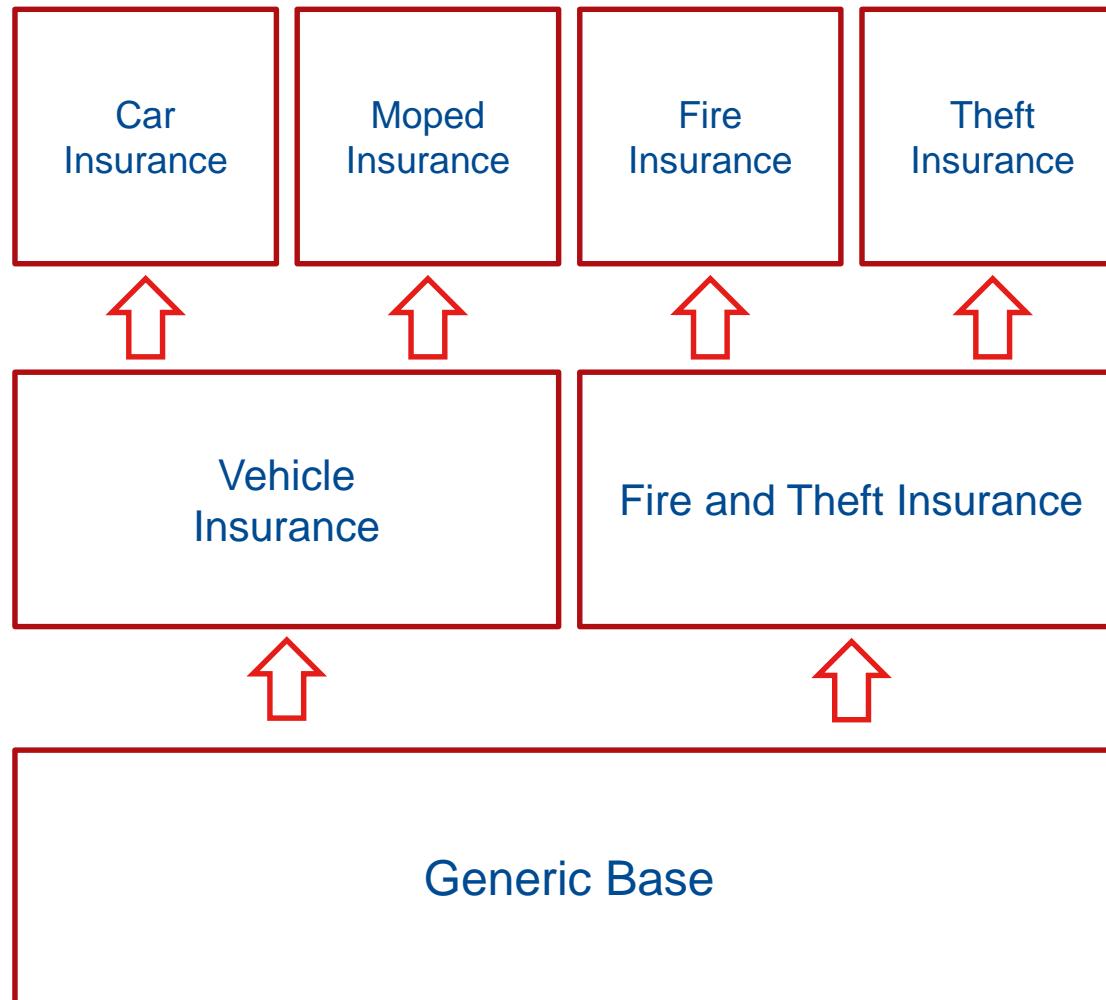
How does it work



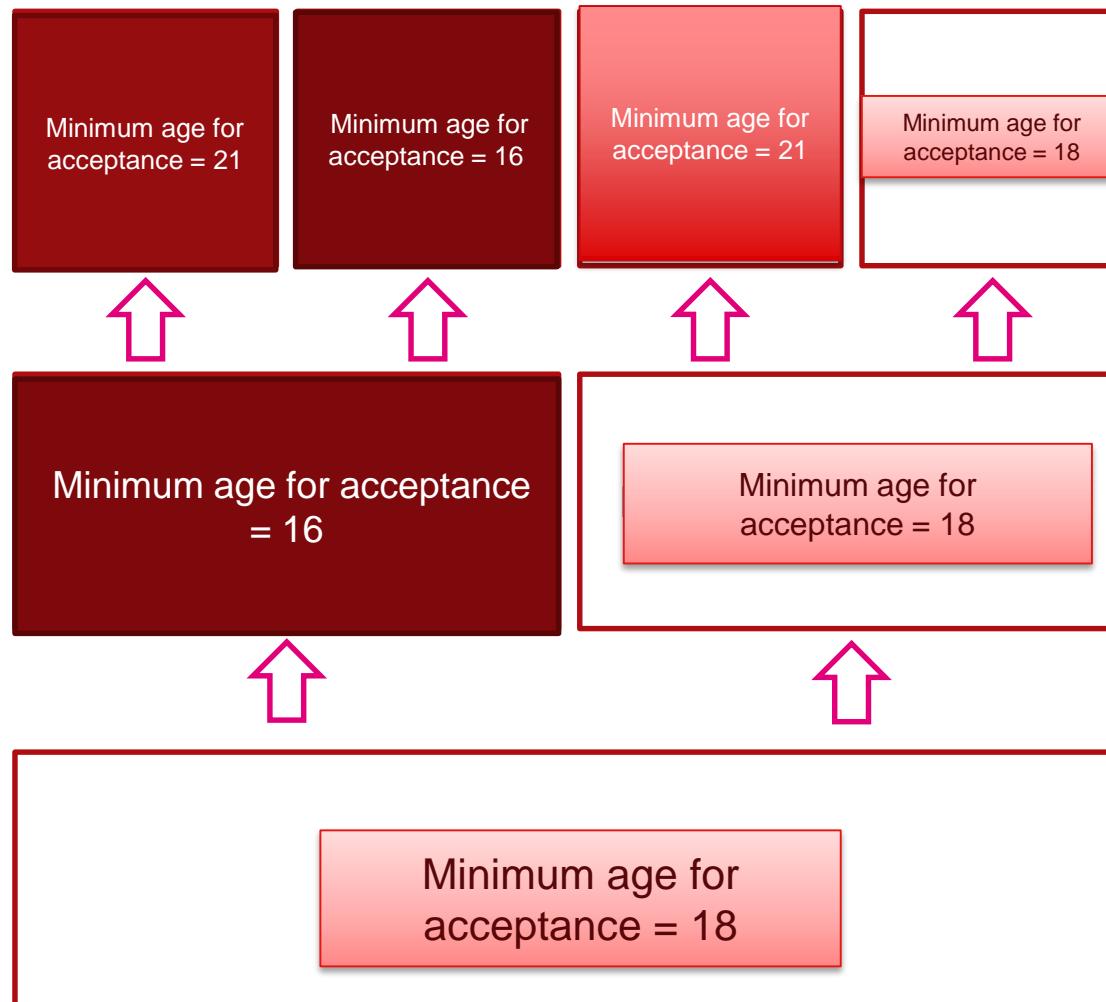
Backbase Forms
Studio

Backbase Forms
Runtime
(JAVA)

Project and Interaction Modules



Specialization



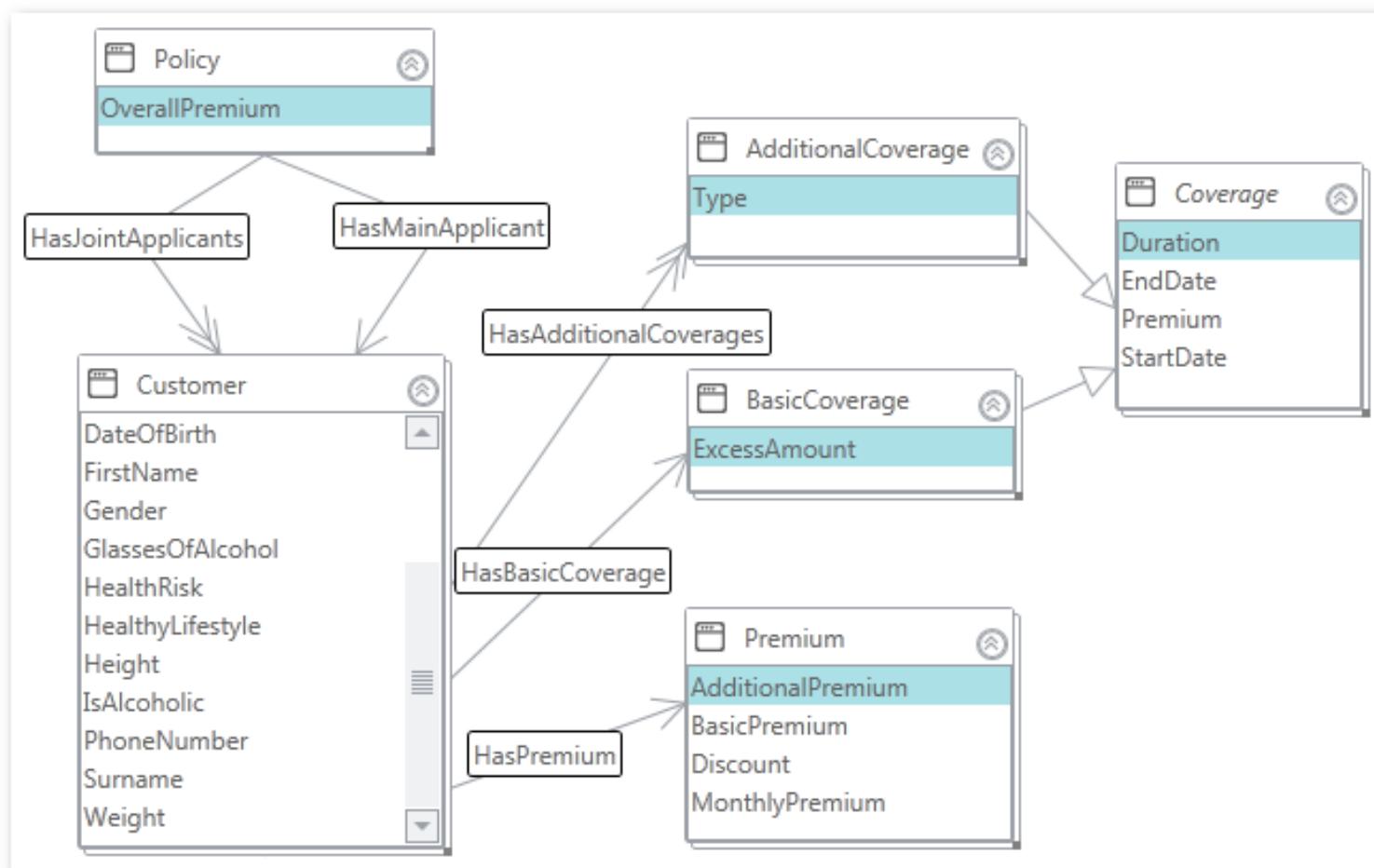
Agenda – Basic Forms Modeling

- Introduction to Backbase Forms
- **Domain Model**
- Interaction (pages and flows)
- Business logic

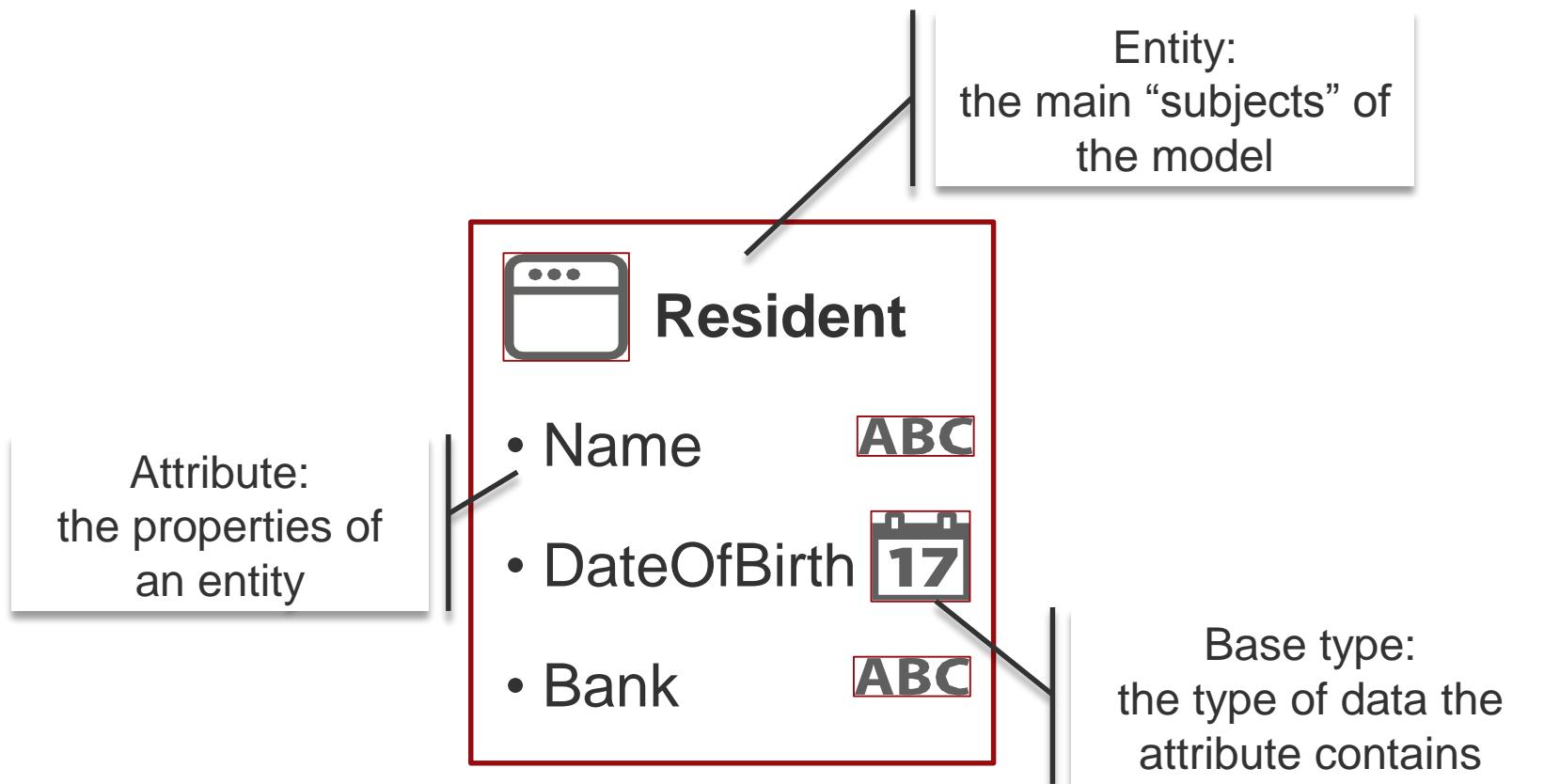
Domain Model



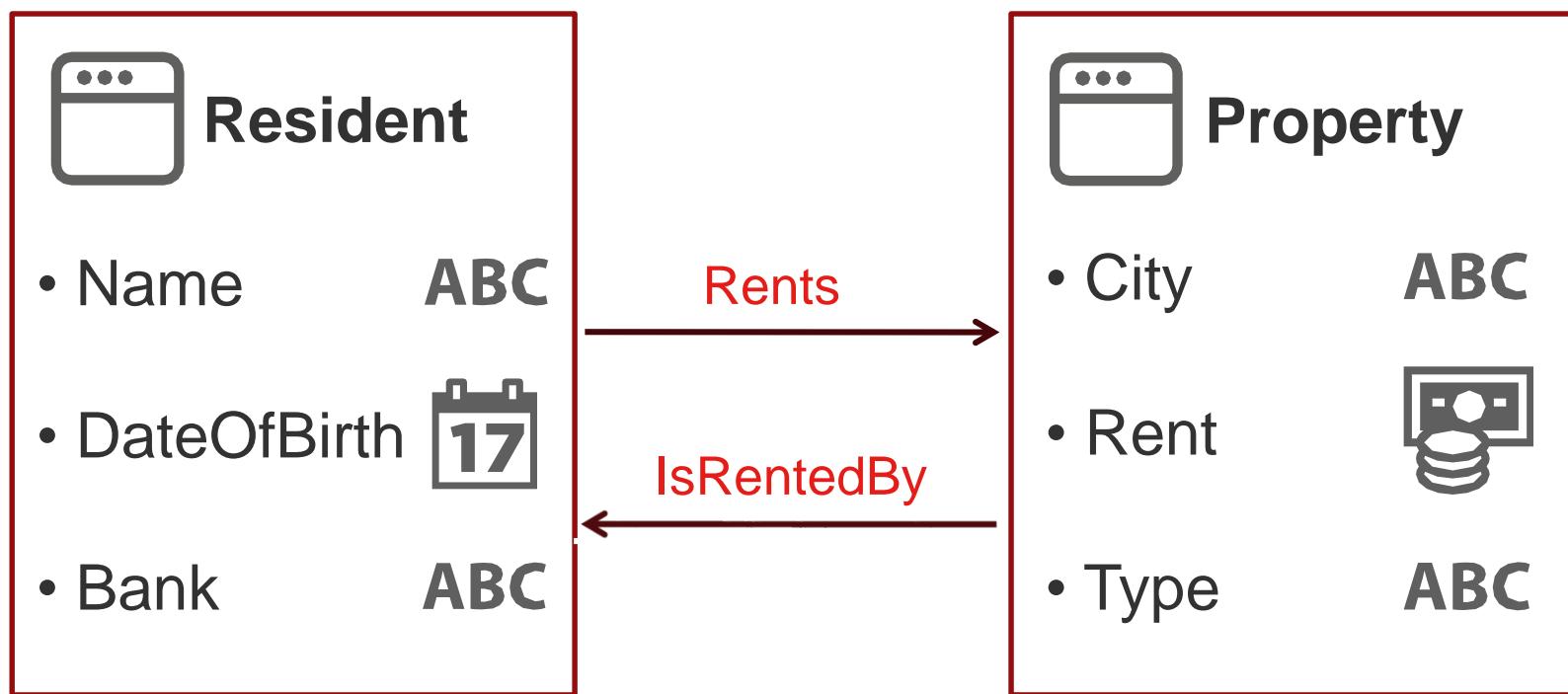
Domain Model (1)



Domain Vocabulary



Domain Vocabulary - Relations



Domain Vocabulary – Value List



PropertyTypes ABC

- Cabin
- Country home
- Townhouse
- Residential lot
- Villa
- Cottage
- Apartment

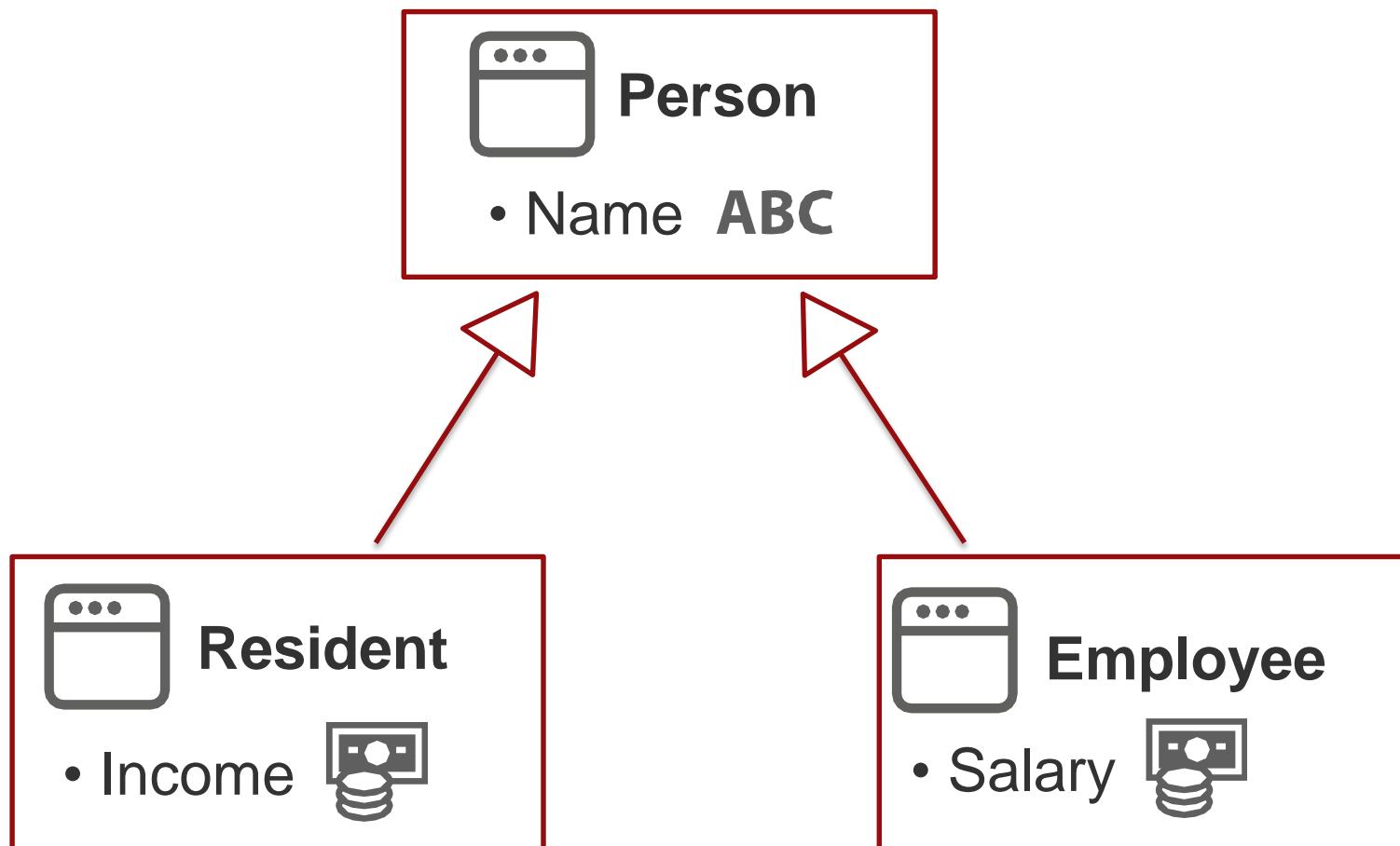
Domain Vocabulary – Attribute Properties

In a conditional value list you can decide which values are shown under which conditions.

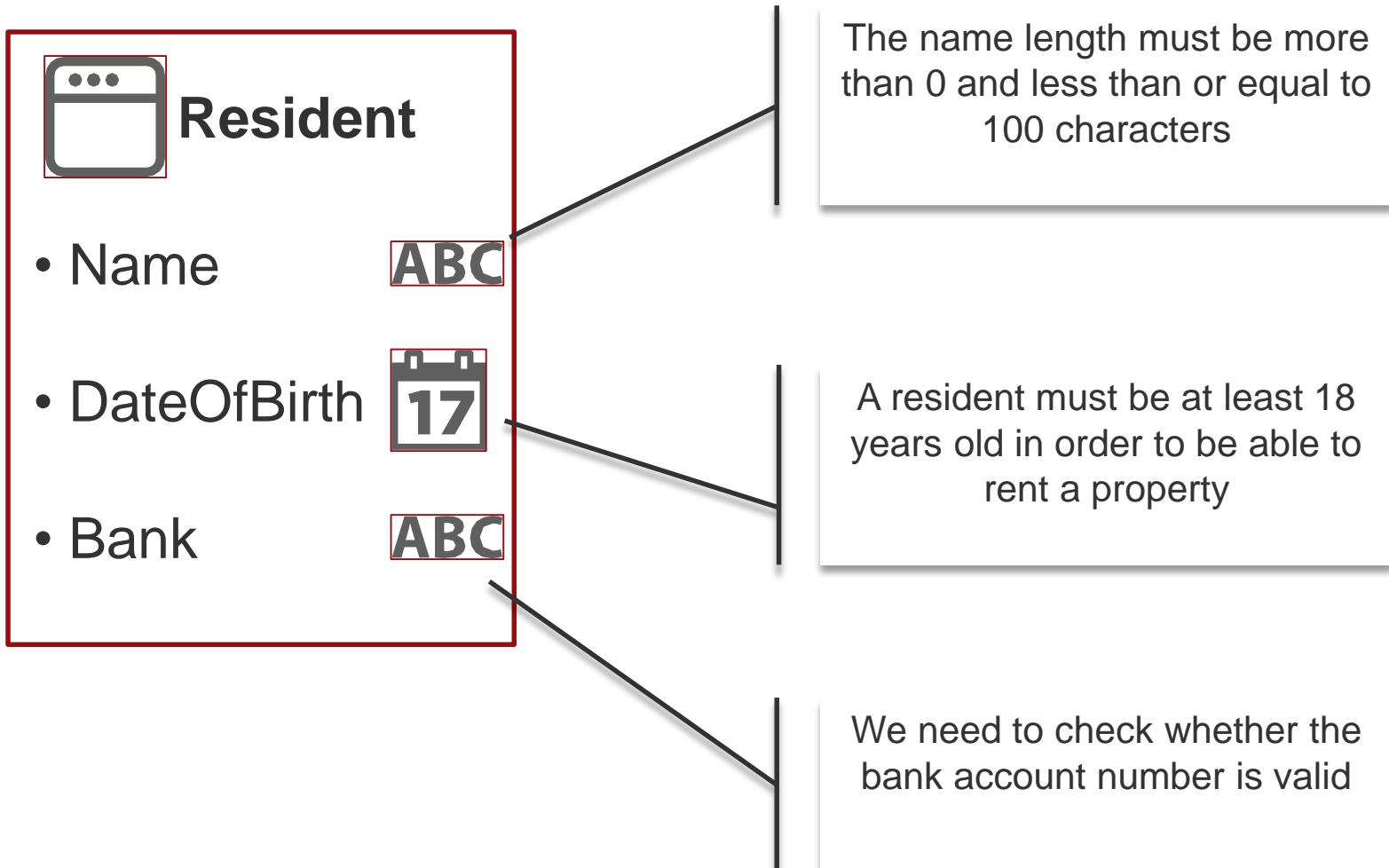
If an attribute is multivalued it can represent more than one value at the time, e.g.

Person.Hobbies= [watching tv, skating, sailing]

Domain Vocabulary - Inheritance



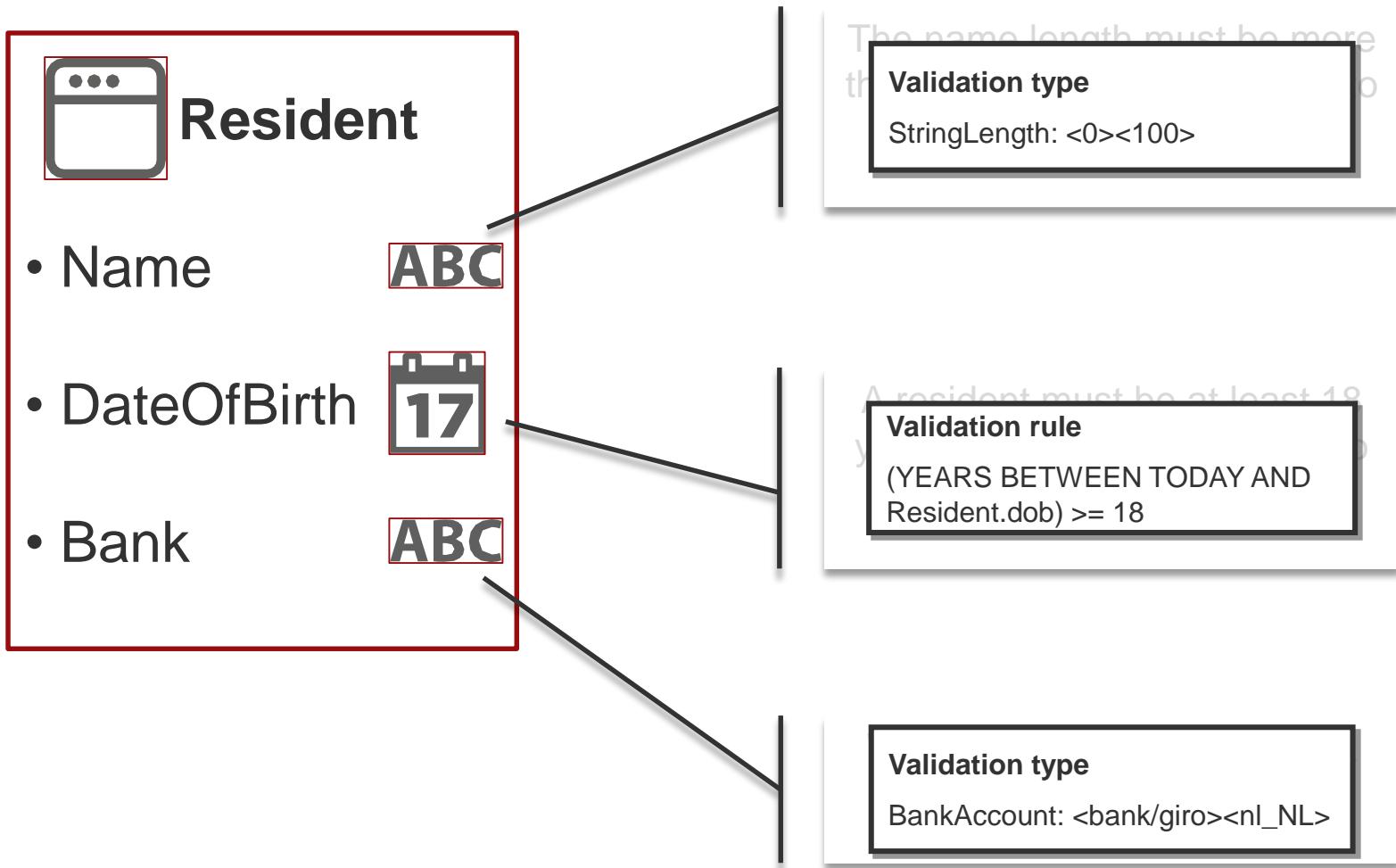
Validations (1)



Validations (2)

- A validation type is used to check if a single attribute value is valid at runtime.
This type of validation is specific for the attribute.
- A validation rule specifies the condition(s) the value of the attribute has to meet in order to make it a valid entry.
This element can be defined in Studio and reused.

Validations (3)

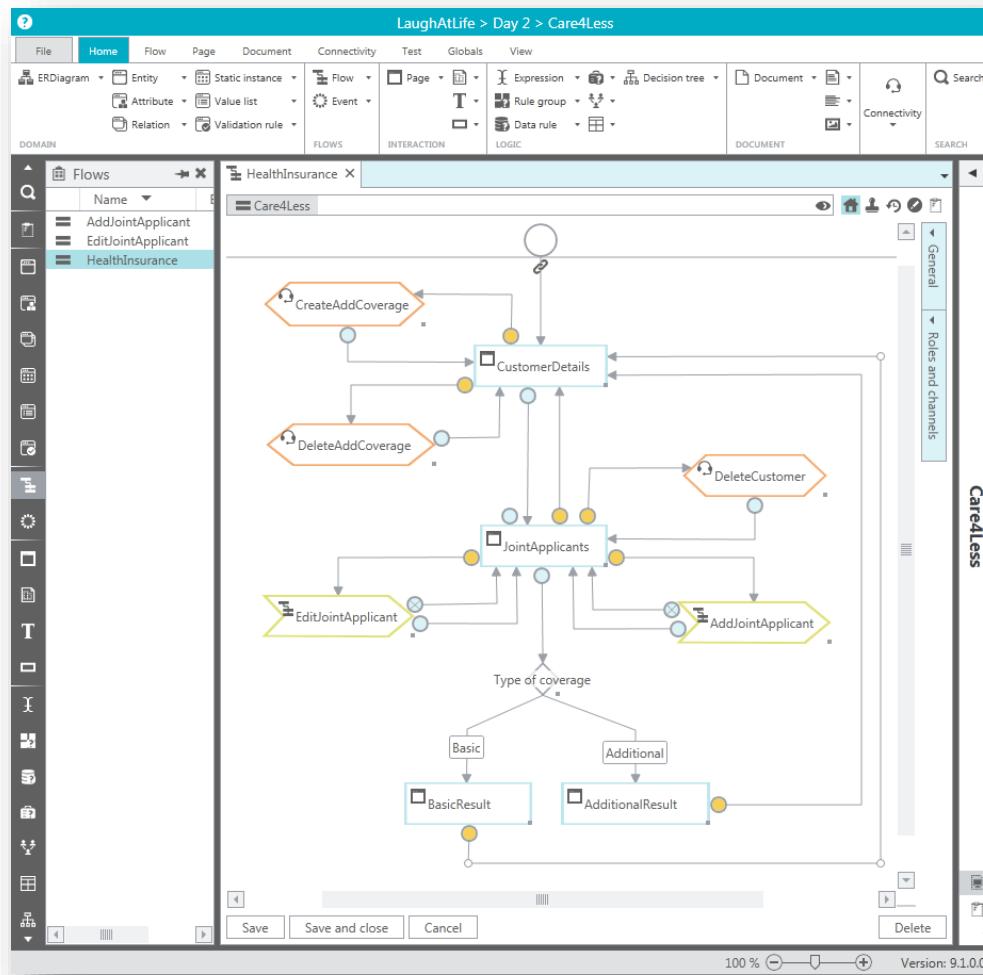


Validations (4)

When do validation rules apply?

- connected to an attribute AND
- the attribute is put on a page AND
- the attribute is not read-only AND
- the page is submitted

“Care for Less”



BACKBASE

YOUR DETAILS

First Name:

Surname:

Gender: Male Female

Date of Birth: yyyy-mm-dd

Phone number:

HEALTH QUESTIONS

Alcohol Usage: Never Daily Weekly

Height:

Weight:

BASIC COVERAGE

Basic Premium: € 99.00

Excess Amount: € 170.00

ADDITIONAL COVERAGES

Type of Coverage: None Health Dental

PREMIUM

Additional premium: € 0.00

Basic Premium: € 99.00

Discount: 0.00 %

Monthly Premium: € 99.00

Next



Demo

Assignment 1

- “Care for Less”
- Create a domain model
- Read carefully, the instructions are in your handouts
- Stick to the names used in the assignments
- Use English names

Workshop environment

Connect IE browser to:

<http://forms5train.backbase.com>

Select: Studio Client

User name :

- s[xx] {01, 02, 03, 04, 05, 06, 07, 08, 09, 10}

Password

- Same as username
- Select the branch for your user

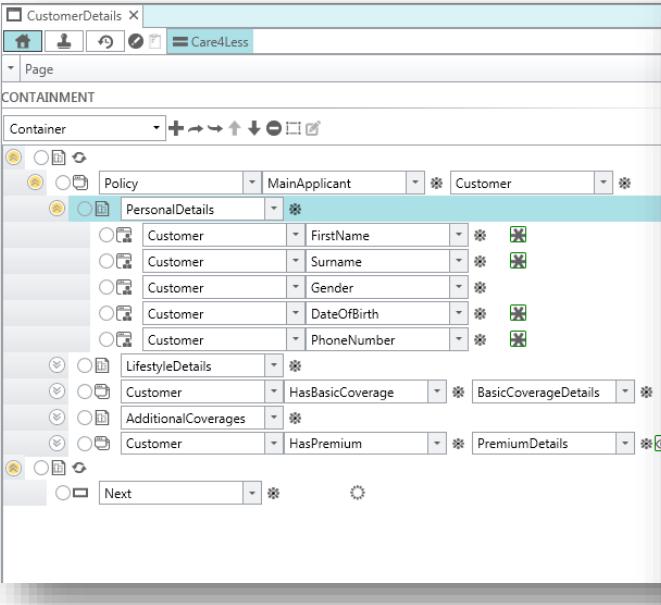
Agenda – Basic Forms Modeling

- Introduction to Backbase Forms
- Domain Model
- **Interaction (pages and flows)**
- Business logic

Pages/Forms



Page examples



UNIVÉ VERZEKERINGEN | Particulier | Zakelijk

Over Univé | Werken bij | Mijn Univé | Contact

Home | Verzekeringen | Hypotheek en financiën | Klantenservice

Home > Particulier > Verzekeringen > Zorgverzekering > Premie berekenen en afsluiten

Zorgverzekering

Basisgegevens  **Premie**  Uw gegevens Meeverzekerden Slotvragen

1  2  3  4  5 

Kies hieronder uw basisverzekering en eventuele aanvullende verzekering(en). Wilt u meer informatie over de inhoud van de aanvullende verzekeringen, kijk dan eens naar onze [keuzehelp](#).

Op basis van het ledencollectief komt u in aanmerking voor **2,5% korting** op de premie van de basisverzekering. Deze korting is al in de premies verwerkt.

Verzekeringnemer (14-09-1976)

Basisverzekering	 Geregeld polis (€ 105,25)	€ 105,25
Lees meer		
Eigen risico	 € 350,- eigen risico	€ 0,00
Lees meer		
Aanvullende verzekering	 Maak uw keuze	
Lees meer		
Tandartsverzekering	Kies eerst uw aanvullende verzekering	
Premie per maand	€	105,25

Uw partner (19-07-1983)

Basisverzekering	 Geregeld polis (€ 105,25)	€ 105,25
Lees meer		
Eigen risico	 € 350,- eigen risico	€ 0,00
Lees meer		
Aanvullende verzekering	 Maak uw keuze	
Lees meer		

Andere mensen kiezen voor



89% kiest voor de Univé Geregeld polis

96% kiest voor een eigen risico van € 350,-

Basisgegevens

Postcode: 5691 DG
 UW geboortedatum: 14-09-1976
 Geboortedatum
 partner: 19-07-1983
 Geboortedatum kind: 26-08-2012
 Ingangsdatum: 05-03-2013
 Ledencollectief: 2,5% korting

[Wijzig](#)

Premie

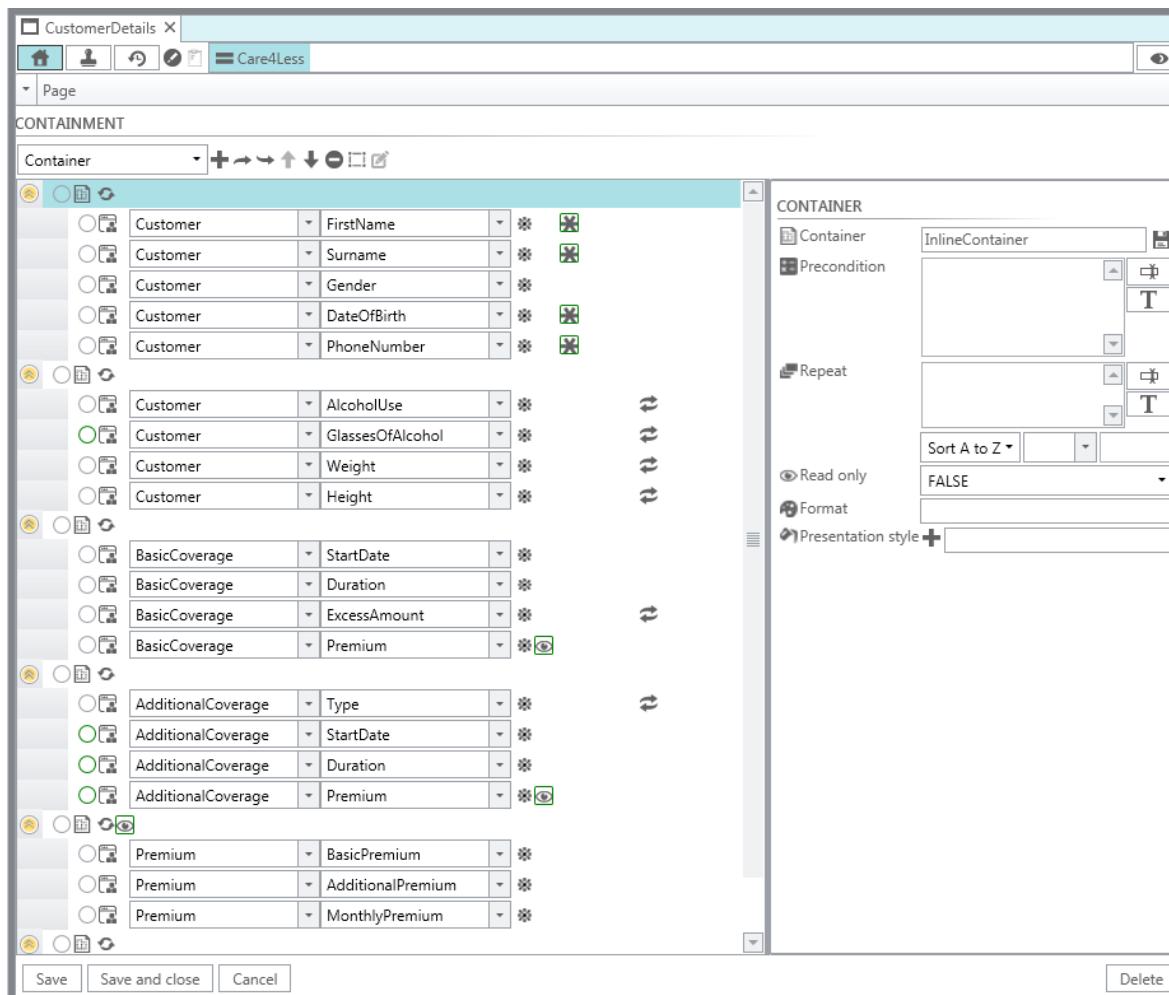
Verzekeringnemer (14-09-1976)
 Geregeld polis € 105,25

37

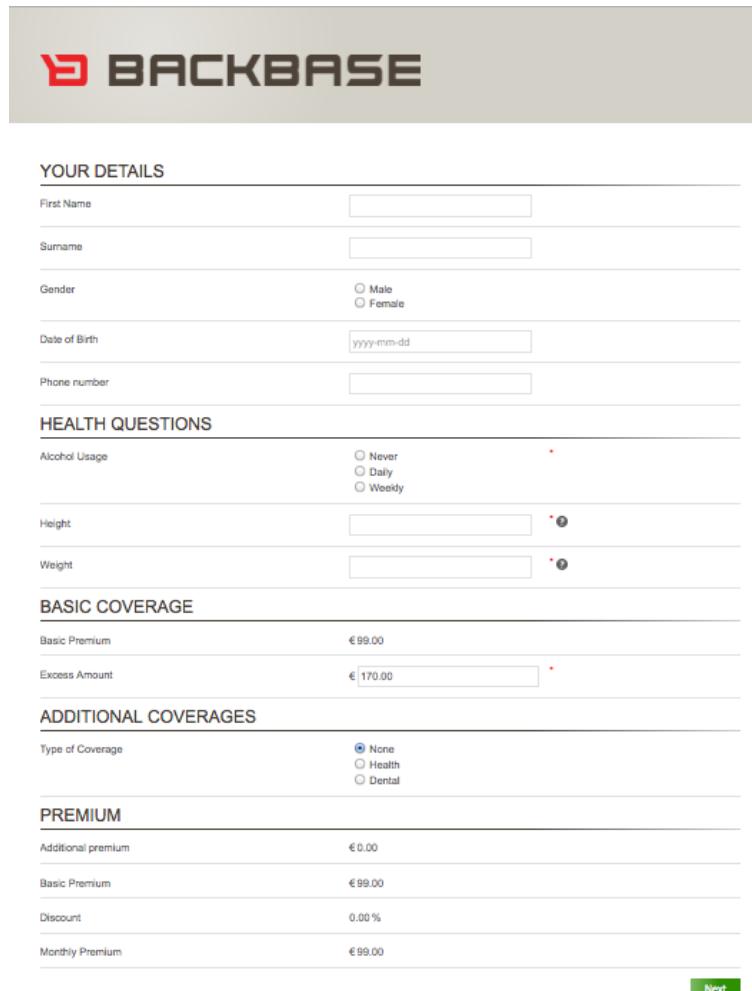
Pages

- A page represents a dialogue or form. A page is composed of containers
- A container is used to group information. It can be composed of other containers, attributes, relations, text, images and / or buttons
- An inline container can not be reused on other pages

Pages with containers in Forms Studio

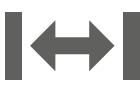


Pages with containers in Forms Runtime



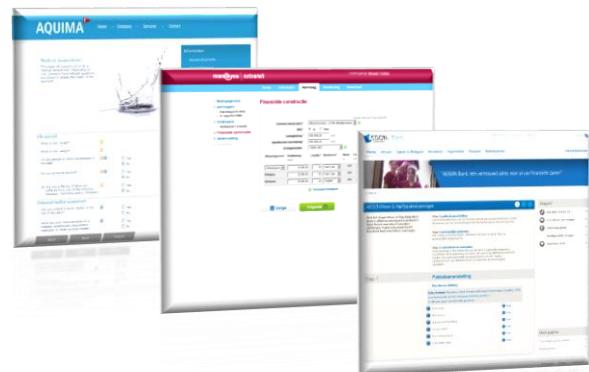
The screenshot displays a web form interface from Backbase. At the top, there is a header with the Backbase logo and the text "YOUR DETAILS". Below this, there are several input fields: "First Name" (text input), "Surname" (text input), "Gender" (radio buttons for Male and Female), "Date of Birth" (text input with placeholder "yyyy-mm-dd"), and "Phone number" (text input). A section titled "HEALTH QUESTIONS" follows, containing "Alcohol Usage" (radio buttons for Never, Daily, and Weekly) and two empty text inputs for "Height" and "Weight". Next is a section titled "BASIC COVERAGE" with "Basic Premium" (text input showing € 99.00) and "Excess Amount" (text input showing € 170.00). The "ADDITIONAL COVERAGES" section includes a "Type of Coverage" field with radio buttons for "None", "Health", and "Dental", where "None" is selected. The final section is "PREMIUM", which lists "Additional premium" (€ 0.00), "Basic Premium" (€ 99.00), "Discount" (0.00 %), and "Monthly Premium" (€ 99.00). At the bottom right, there is a green "Next" button.

Page element options

Icon	Explanation
	Precondition
	Read only
	Required
	Length
	Format
	Presentation style
	Event
	Refresh

Page and style

**BACKBASE
RUNTIME**

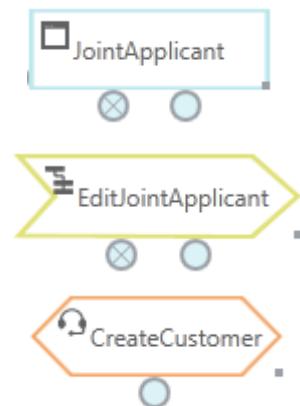


Agenda – Basic Forms Modeling

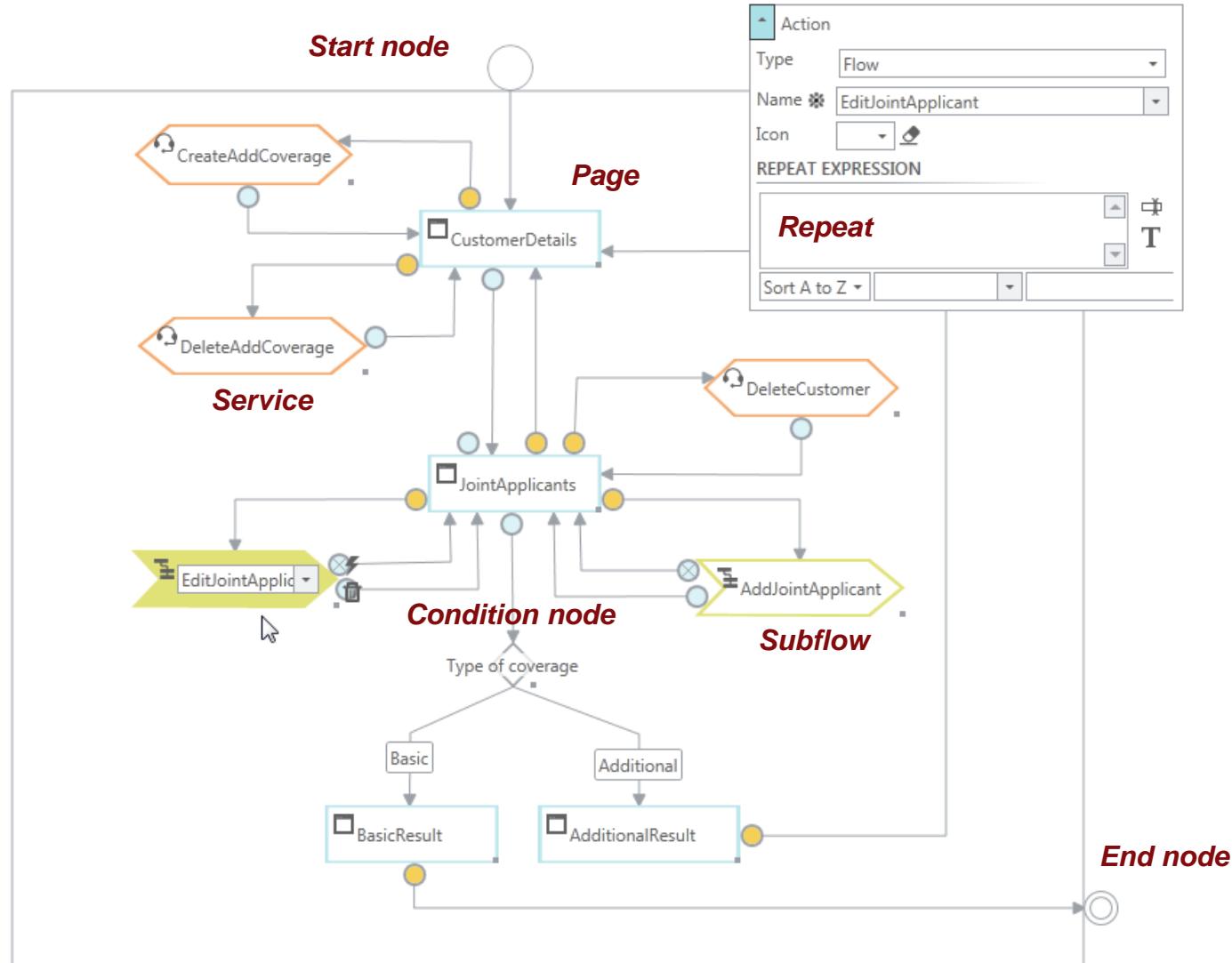
- Introduction to Backbase Forms
- Domain Model
- **Interaction (pages and flows)**
- Business logic

Flows

- A flow is a graph-like structure containing
 - Pages
 - sub-flows and
 - services

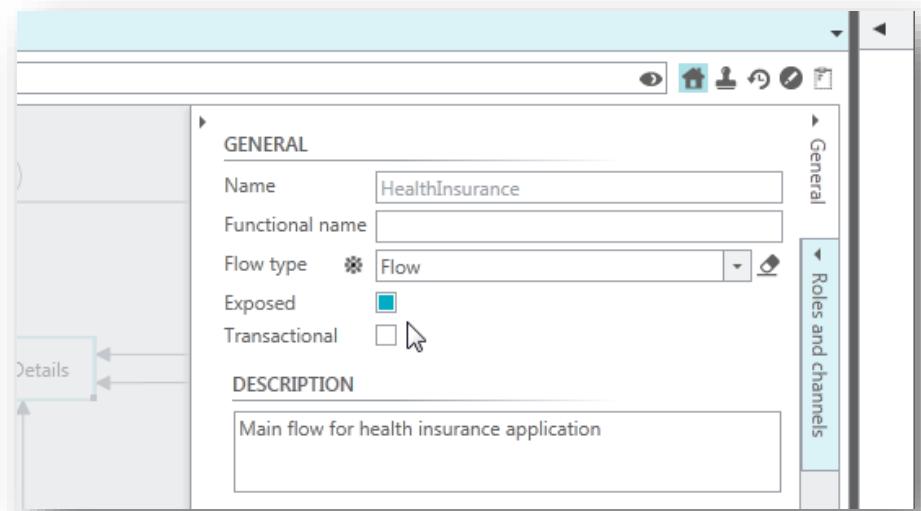


Flows (2)



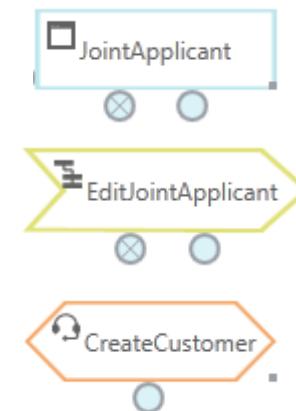
Flows (3)

- An exposed flow can be approached and used as start flow, either by a user, the process engine, or by a service.
- It also means that the flow shows up in the list of flows you can select in the Backbase Forms Runtime.



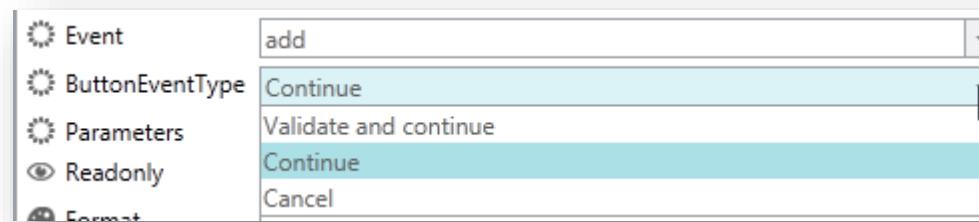
Events (1)

- Events
 - displayed as small circles,
 - can be used to direct the flow.
- Events are set by
 - pressing a button
 - flowing through a flow endpoint
 - executing a service



Events (2)

- There are three event types:
 - Continue: stores all user input, even if it is not valid
 - Validate and Continue: validates and stores all user input
 - Cancel: rollback current page or flow, nothing from this page or transaction is stored



Assignment 2

- Apply for health insurance
- Create your first flow and pages

Introduction to Backbase Forms

- Domain model
- Pages (or forms)
- Flows
- **Business logic**
 - Business rules
 - Decision tables
 - Default values
 - Writing expressions

Logic



Inference Engine

- The heart of Backbase Forms is its inference engine
- It tries to derive answers from a knowledge base

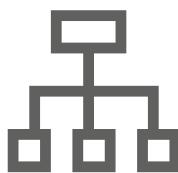
Business Logic



Business rule



Decision table



Decision tree

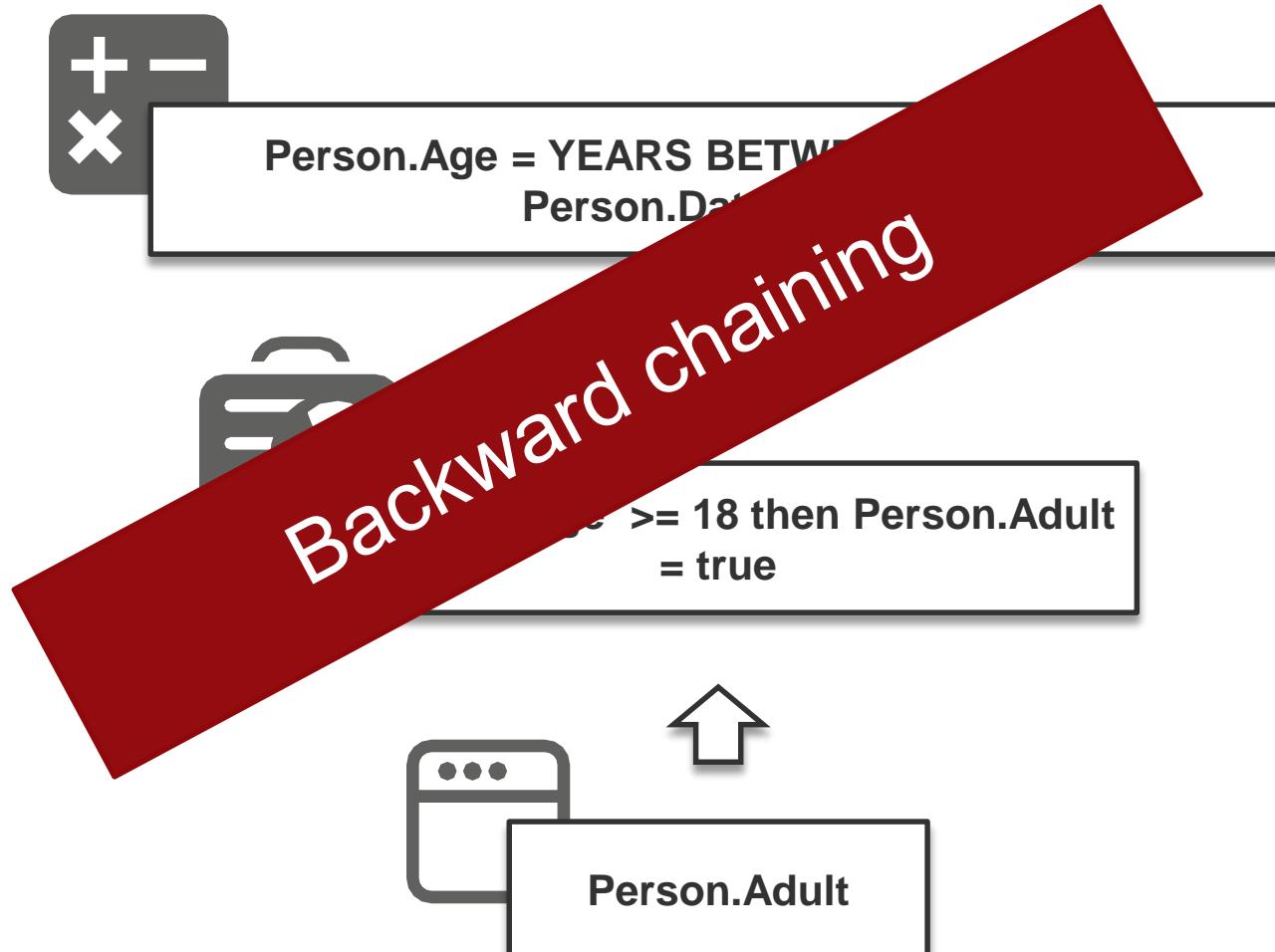


Default value



Can have reusable
parts / expressions

Inference Engine



Inference Engine

- User set values prevail
- Order of attribute sourcing
 1. Business rule/decision table/external rule
 2. Default value
- Truth maintenance

Introduction to Backbase Forms

- Domain model
- Pages (or forms)
- Flows
- Business logic
 - Business rules
 - Decision tables
 - Default values
 - Writing expressions

Business Rule

- A business rule derives one decision.
- IF-THEN construction:
 - IF a condition is met
 - THEN an attribute
 - IS set to a value/expression

The screenshot shows a Business Rule editor interface with the following structure:

IF	<code>Customer.IsObese OR Customer.IsAlcoholic</code>	
THEN	Customer	HealthyLifestyle
IS	FALSE	

The interface includes a toolbar with icons for copy/paste, delete, and search, as well as text input fields and dropdown menus for each rule component.

Introduction to Backbase Forms

- Domain model
- Pages (or forms)
- Flows
- Business logic
 - Business rules
 - **Decision tables**
 - Default values
 - Writing expressions

Decision Table

- A decision table derives multiple decisions (or actions).
- To derive these actions we need one or more conditions. The answers to these conditions are called condition alternatives.
- Typical knowledge tasks for which you use decision tables are assessing and classification.

Customer.AlcoholUse	"daily"	"weekly"	
Customer.GlassesOfAlcohol	<= 4	>4	<=18 >18 *
Customer.IsAlcoholic	FALSE TRUE	FALSE TRUE	FALSE

Decision Table - Example

- Goal:
 - Person.TypeOfSportPlayer
{Never, Recreational, Active}
- Necessary information:
 - Does the applicant play sports?
 - How often does he play sports? {yearly, monthly, weekly, daily}
 - Is it an intensive sport?

Person.PlaysSports	Yes			No	?
Person.SportFrequency	“Yearly”	“Monthly”	[]	*	*
Person.IntensiveSport	*	Yes	No	*	*
Person.TypeOfSportPlayer	Recreational	Active	Recreational	Active	Never

Decision Table – Unknown ?

- An attribute value will remain unknown when it cannot be determined or asked.
- Use the ‘Unknown’-symbol in the decision table to deduct the goal parameter.

Person.PlaysSports	Yes			No	?
Person.SportFrequency	“Yearly”	“Monthly”	[]	*	*
Person.IntensiveSport	*	Yes	No	*	*
Person.TypeOfSportPlayer	Recreational	Active	Recreational	Active	Never

Decision Table – Don't care *

- Some information may be irrelevant for certain decisions. Use the ‘Don’t care’-symbol to indicate in which cases the information is not relevant.
- The information will not be deducted or asked in those cases.

Person.PlaysSports	Yes			No	?
Person.SportFrequency	“Yearly”	“Monthly”		[]	*
Person.IntensiveSport	*	Yes	No	*	*
Person.TypeOfSportPlayer	Recreational	Active	Recreational	Active	Never
					2

Decision Table – Else []

- Usually attributes/expressions can have a range of values. These values are specified in the condition alternatives. Use the ‘Else’-symbol to indicate all the other possible values.
- Please notice: ‘Unknown’ is not a part of those ‘other values’ and therefore has to be included separately.

Person.PlaysSports	Yes			No	?
Person.SportFrequency	“Yearly”	“Monthly”	[]	*	*
Person.IntensiveSport	*	Yes	No	*	*
Person.TypeOfSportPlayer	Recreational	Active	Recreational	Active	Never

Decision Table System (1)

- A decision table system is one large table, split into separate related tables.

Person.Smoker	Yes	No													
Person.ChronicCondition	*	Yes						No							
Person.PlaysSports	*	Yes						No	Yes						No
Person.SportFrequency	*	"Yearly"		"Monthly"				[]	*	"Yearly"		"Monthly"		[]	*
Person.IntensiveSport	*	*		Yes			No	*	*	*		Yes	No		*
Person.Age	*	<60	>=60	<40	>=40 AND <60	>=60	*	*	*	<50	>=50	*	<50	>=50	*
Person.Accepted	No	More	No	Yes	More	No	No	Yes	No	More	No	Yes	Yes	No	Yes

Decision Table System (2)

Person.Smoker	Yes	No												
Person.ChronicCondition	*	Yes						No						
Person.PlaysSports	*	Yes						No	Yes					
Person.SportFrequency	*	"Yearly"	"Monthly"				[]	*	"Yearly"	"Monthly"				[]
Person.IntensiveSport	*	*	Yes			No	*	*	*	Yes	No			*
Person.Age	*	<60	>=60	<40	>=40 AND <60	>=60	*	*	*	<50	>=50	*	<50	>=50
Person.Accepted	No	More	No	Yes	More	No	No	Yes	No	More	No	Yes	No	Yes

Person.PlaysSports	Yes					No	?
Person.SportFrequency	"Yearly"			"Monthly"		[]	*
Person.IntensiveSport	*			Yes	No	*	*
Person.TypeOfSportPlayer	Recreational		Active	Recreational	Active	Never	Never

Sub-table

Person.Smoker	Yes	No									
Person.ChronicCondition	*	No									
Person.TypeOfSportPlayer	*	Active				[]	Active	Recreational		Never	
Person.Age	*	<40	>=40 AND <60		>=60	*	*	<50	>=50	<40	>=40
Person.Accepted	No	Yes	More		No	No	Yes	Yes	No	Yes	More

Main table

Decision Table System (3)

- Consequences:
 - Splitting the table enhances the clarity and maintainability of the table.
 - Often the table will be smaller (less columns) and therefore more efficient.



Decision Table – Exclusive and Completeness

A decision table has to meet certain requirements:

- Exclusivity:
 - knowledge may not be ambiguous. This means that the same combination of condition alternatives may not result into various actions.
- Completeness:
 - knowledge may not be incomplete. This means that there are no situations that are not modeled in the decision table.

Decision Table – What is wrong?

Distance	<=100	>100 AND <= 500	>300 AND <= 3000	>5000			
Rain	*	y	n	y	n	*	
Thunderheads	*	y	n	*	y	n	*
Transportation	walk	car	walk	car	car	bike	car

Not exclusive

Not complete

Not complete

Not exclusive

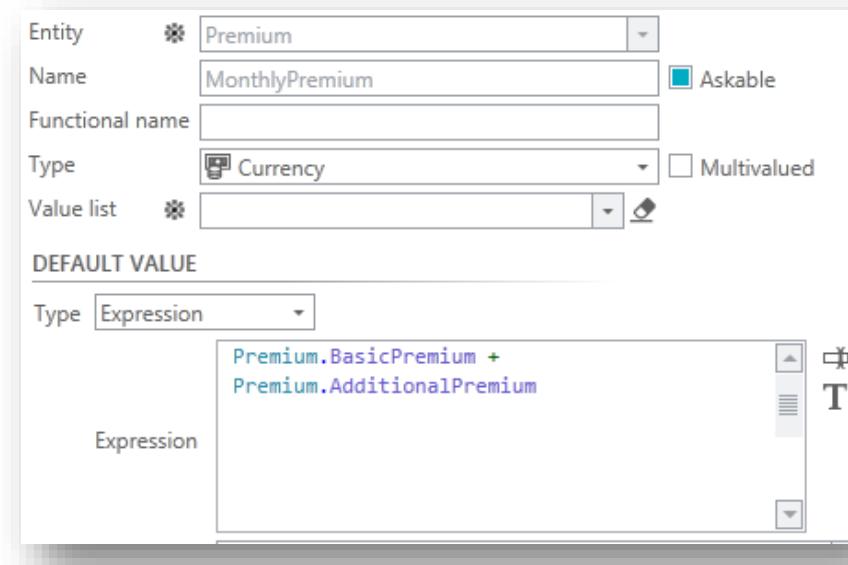
Introduction to Backbase Forms

- Domain model
- Pages (or forms)
- Flows
- Business logic
 - Business rules
 - Decision tables
 - **Default values**
 - Writing expressions

Default Values

- A default value sets the initial value of an attribute.
 - None: no initial value will be set
 - Constant: fill in a value of the type defined by the base type
 - Expression: create an expression which derives the value in the Runtime

The derived value must be of the same type as the base type of the attribute.



Introduction to Backbase Forms

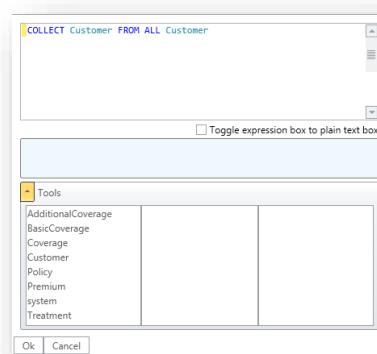
- Domain model
- Pages (or forms)
- Flows
- Business logic
 - Business rules
 - Decision tables
 - Default values
 - Writing expressions

Writing Expressions (1)

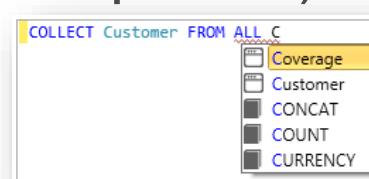
- A (reusable) expression is a combination of attributes and formulas that are evaluated.
- Expressions can be used in:
 - Attributes,
 - Validation rules,
 - Containers,
 - Business rules,
 - Decision tables,
 - Documents,
 - Etcetera.

Writing Expressions (2)

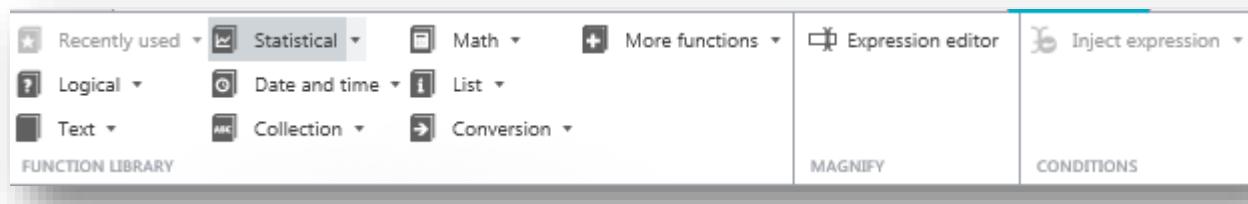
- Expression editor



- IntelliSense (auto completion)



- Formula's ribbon



Writing Expressions (3)

- For basic math you can use math operators +-*/
- Use attribute references instead of values
 $Premium.BasicPremium + Premium.AdditionalPremium$
- Use “advanced” mathematical, statistical, or date formula’s from the formula’s ribbon
 $SUM([22, 3, 17])$

Writing Expressions - parentheses

- Specify the order of execution
 $(\text{Premium}.\text{Base} + \text{Premium}.\text{Addition}) * \text{Premium}.\text{Discount}$
- Separate the various logical components in conditions.
 $(A \text{ AND } B) \text{ OR } C$
- If the input of a formula can be more than one value
 $\text{SIZE}([12, 29, 66, 5])$
 $\text{SUM}(\text{Premium}.\text{MonthlyPremium})$

Writing Expressions - Examples

- Determine if the excess amount is over € 250,-

BasicCoverage.ExcessAmount > 250

- Calculate the average of a list

AVG([1,2,3])

- Determine the age of a customer

YEARS BETWEEN TODAY AND Customer.DateOfBirth



Demo

Assignment 3

- Premium calculation and acceptance
- Model business rules and decision tables
- Add logic to your flow

Day 2

- Complex domain model
 - Instances, singleton and multiple instances, inheritance
 - Instance management, relations and pages & flows
- Complex logic
 - Relations and expressions
 - Overwrite inherited logic

Complex domain model



Complex domain model



Day 2

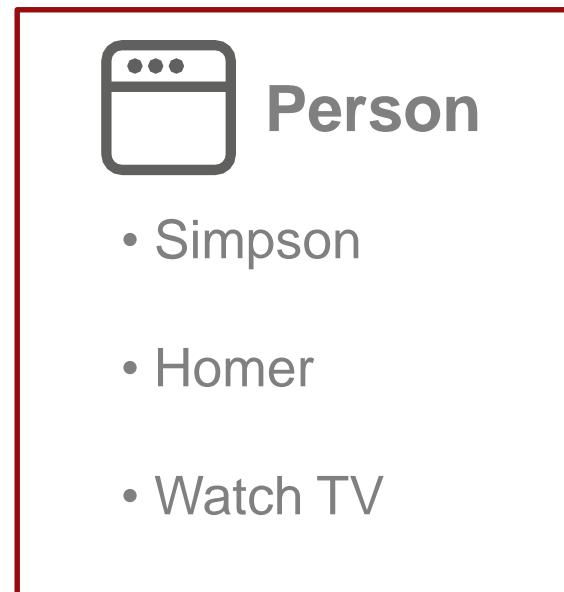
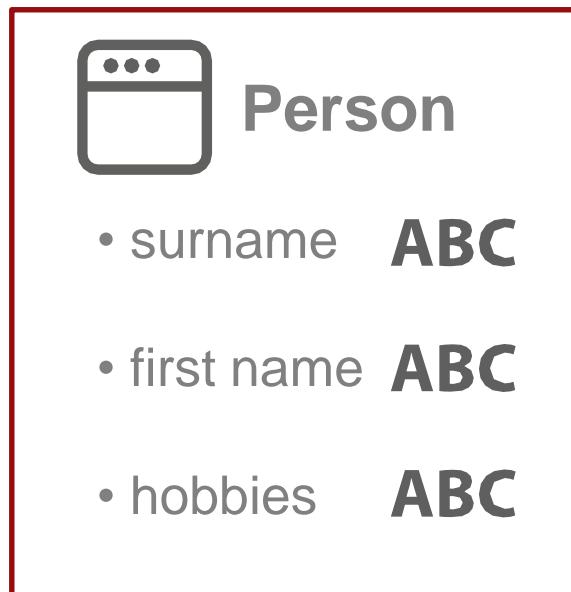
- Complex domain model
 - Instances, singleton and multiple instances, inheritance
 - Instance management, relations and pages & flows
- Complex logic
 - Relations and expressions
 - Overwrite inherited logic

Instances (1)

- An entity is used as a blueprint to create instances. It describes the attributes and logic that all created instances share.
- An instance contains specific values for all attributes defined by the entity.

Instances (2)

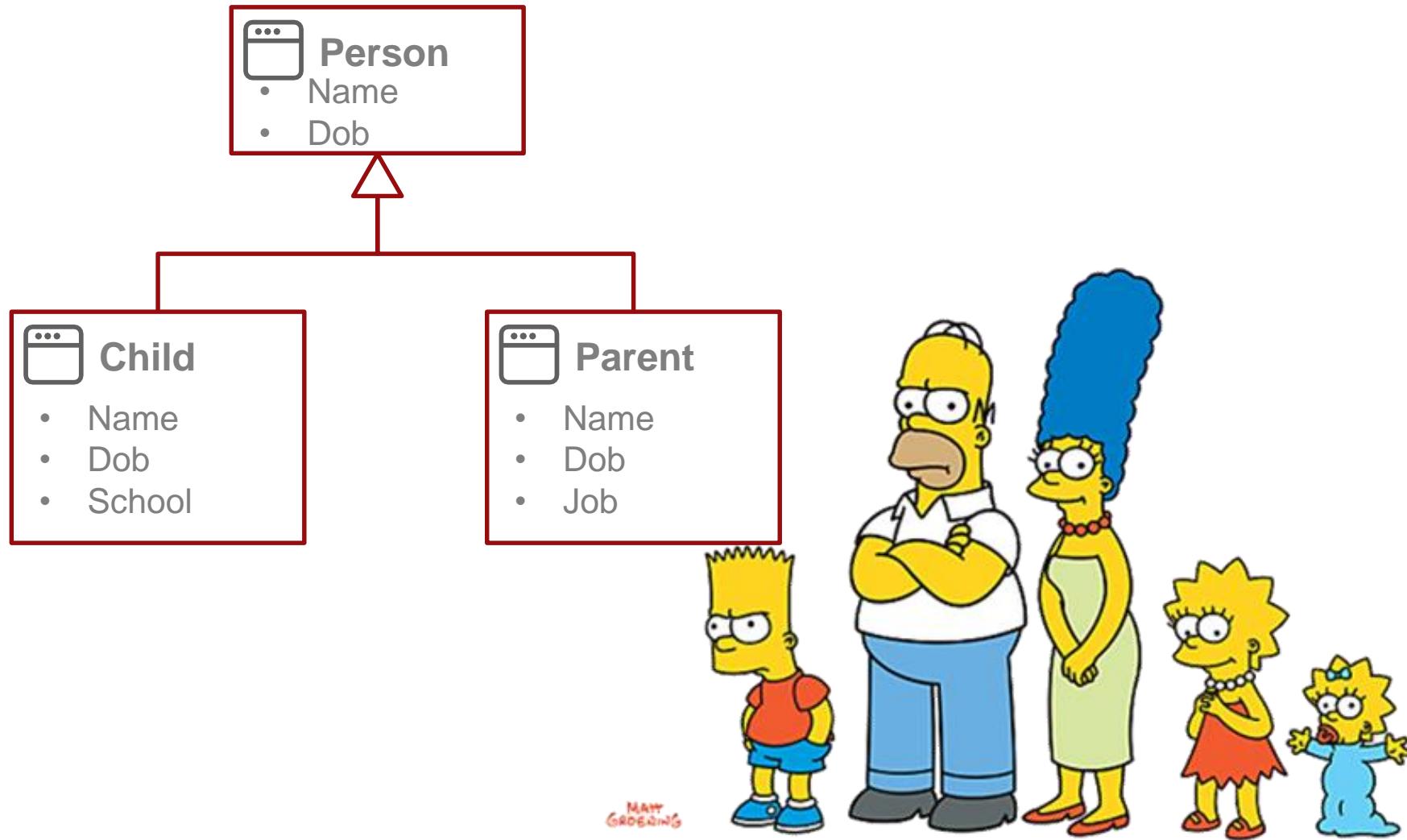
- Homer Simpson is an instance of Person



Instances (3)

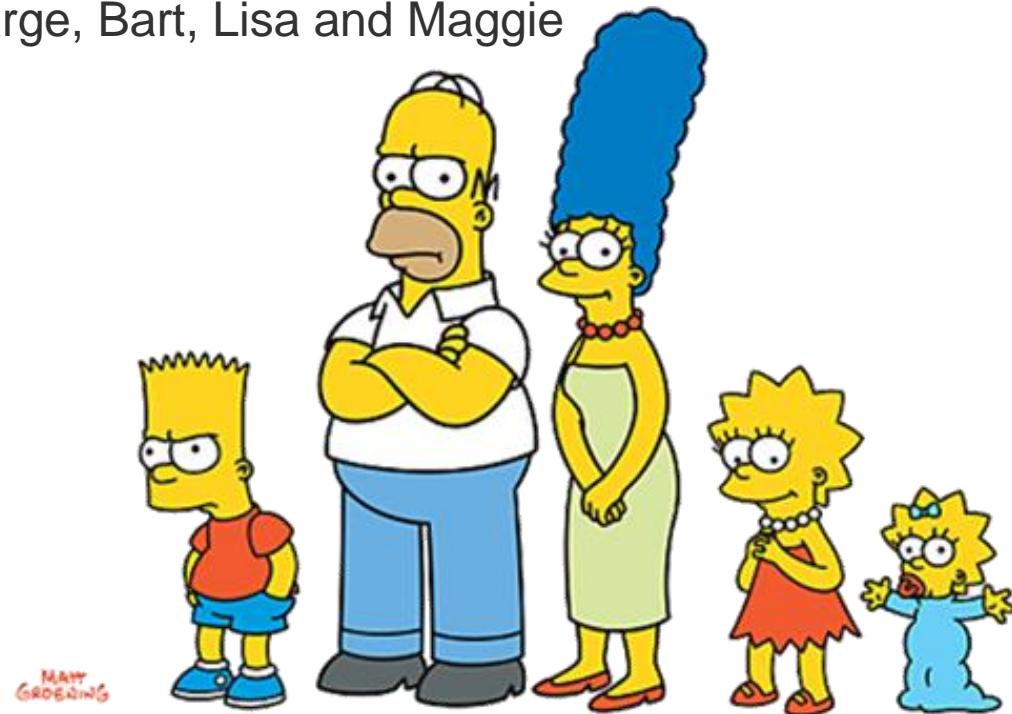
- An entity can have multiple instances.
- A singleton entity can only have one instance.
- An abstract entity can't explicitly be created, but is only accessed through inheritance.

Instances and inheritance (1)



Instances and inheritance (2)

<i>Entity</i>	<i>Instances</i>
Parent	Homer and Marge Simpson
Child	Bart, Lisa and Maggie Simpson
Person	Homer, Marge, Bart, Lisa and Maggie Simpson



Day 2

- Complex domain model
 - Instances, singleton and multiple instances, inheritance
 - Instance management, relations and pages & flows
- Complex logic
 - Relations and expressions
 - Overwrite inherited logic

Instance Management

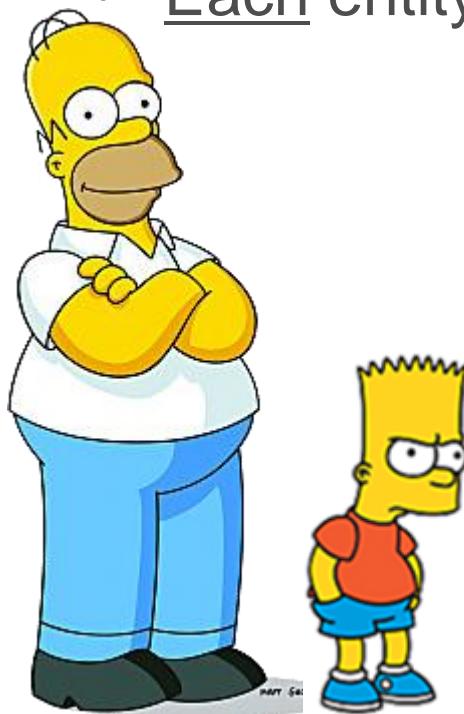
Operations on instances:

- Displaying a (list of) instances
- Creating an instance
- Editing an active instance
- Deleting an active instance

Active: Telling which instance to use, behave as singleton

Instance Management - Activating

- Telling the application which instance to use
- Used as if it is singleton
- Each entity has maximum 1 active instance



Instance Management - Creating

- There is always an instance available for a singleton entity.
- Dynamic instances for an entity must be created explicitly.
- Instances are created with a service:

AQ_CreateInstance

- Or using a container by setting a relation onto a page

Instance Management - Editing

- You can only edit active instances.
- Each entity has only one active instance.
 - A singleton entity always has one active instance.
- Instances can be activated by:
 - AQ_InstanceSelectorPlus container
 - Repeat on a container
 - AQ_ActivateInstance service

Instance Management - Deleting

- An instance for a singleton entity can never be deleted.
- You can delete an instance with the service:
AQ_DeleteInstance

Instance Management - Displaying

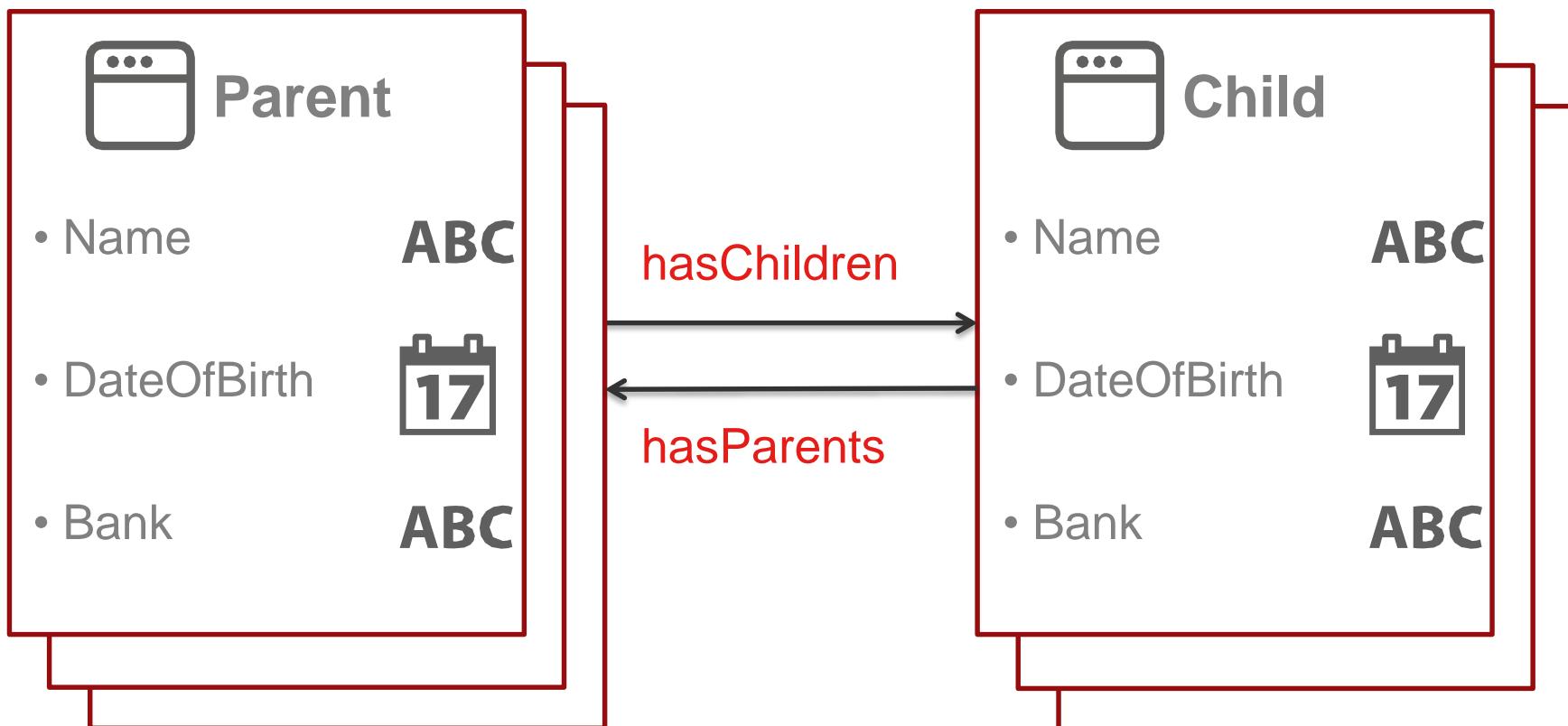
There are several ways to display instances:

- **AQ_InstanceSelectorPlus** container
 - Display, create, edit and delete instances.
 - Instances are activated before editing or deleting.
- Repeat on containers
 - Specify a repeat expression that results in a set of instances.
 - With each repetition an instance from the set is activated.

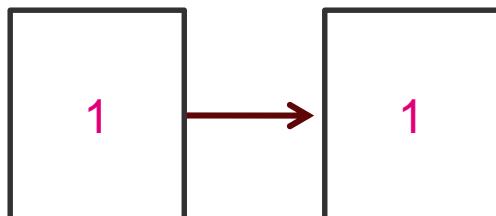
Instance Management - Transactions

- If a transactional flow ends with a success  end point all changes to instances are saved.
- If a transactional flow ends with a cancel  end point all changes to instances made in this flow are made undone.

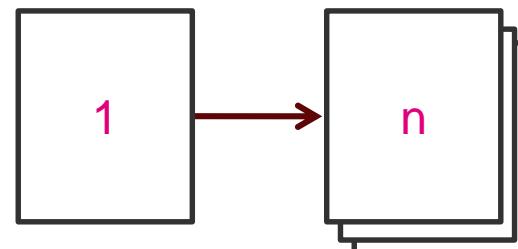
Relations



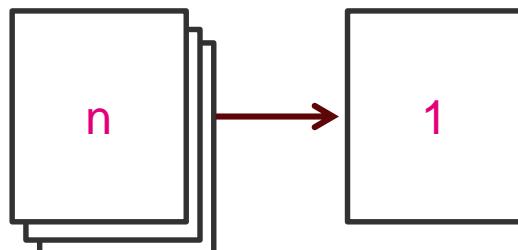
Relations and cardinality



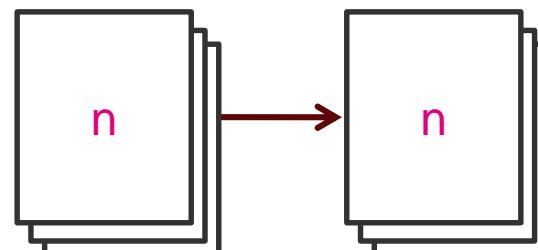
one-to-one



one-to-many



many-to-one

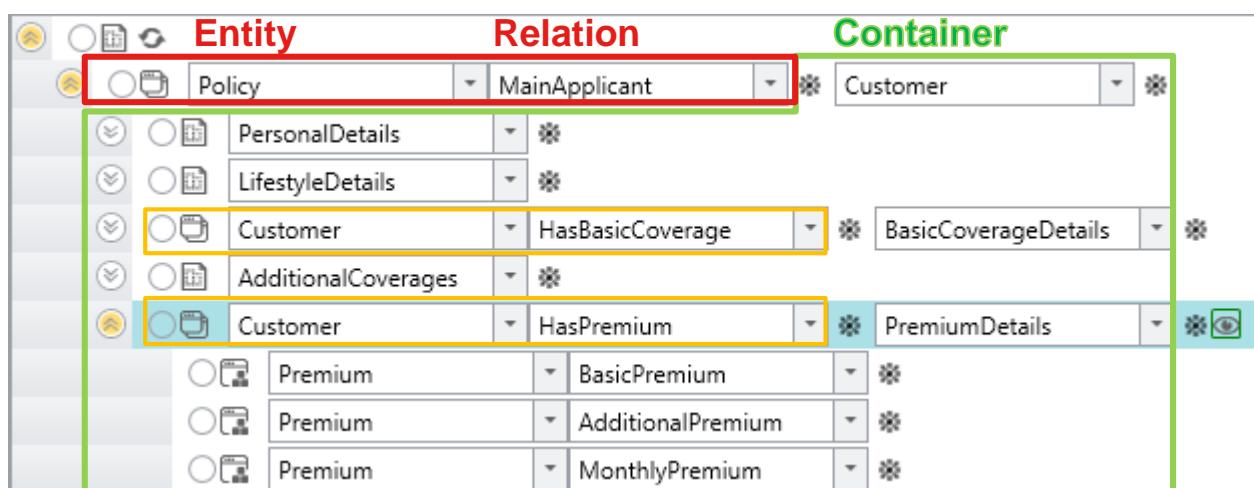


many-to-many

Relations don't define how many instances can exist!

Relations on a page

- Use a 1-1 relation on a page to display the instance within this relation:



Relation Creation

- Relations are created in the Domain section of the ribbon



- Setting the relation cardinality is done via selecting the multivalued tick box underneath the relation.
- It is set towards the relation not the entity.

The screenshot shows the 'RELATION' configuration dialog. It includes the following fields:

- RELATION**:
 - Functional name:
 - From entity:
 - Relation:
 - Multivalued:
- DESCRIPTION**: A large text area for entering descriptive text.
- REVERSE RELATION**:
 - To entity:
 - Reverse relation:
 - Multivalued:
- QUESTION TEXT**: A text input field with a language dropdown set to English and a rich text editor icon.
- EXPLAIN TEXT**: A text input field with a language dropdown set to English and a rich text editor icon.
- VALIDATION RULES**: A section containing a 'Validation rule' dropdown and several small icons for managing validation rules.

Assignment 4

- Advanced domain modeling
- Additional coverages
- Joint applicants
- Flows and multiple instances

Day 2

- Complex domain model
 - Instances, singleton and multiple instances, inheritance
 - Instance management, relations and pages & flows
- Complex logic
 - Relations and expressions
 - Overwrite inherited logic

Relations & Expressions



Relations & Expressions (1)

- Dotted notations can be used for relations.
- Entity.Relation
- Examples

– Customer.HasPremium

Returns the Premium of the active Customer

– Policy.JointApplicants

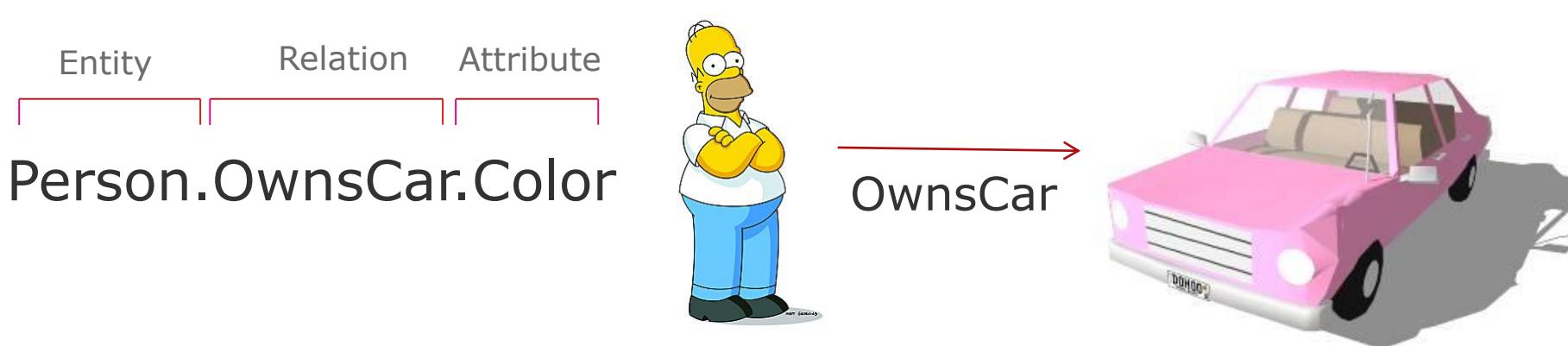
Returns a list of Customers

– Policy.MainApplicant.HasCoverage

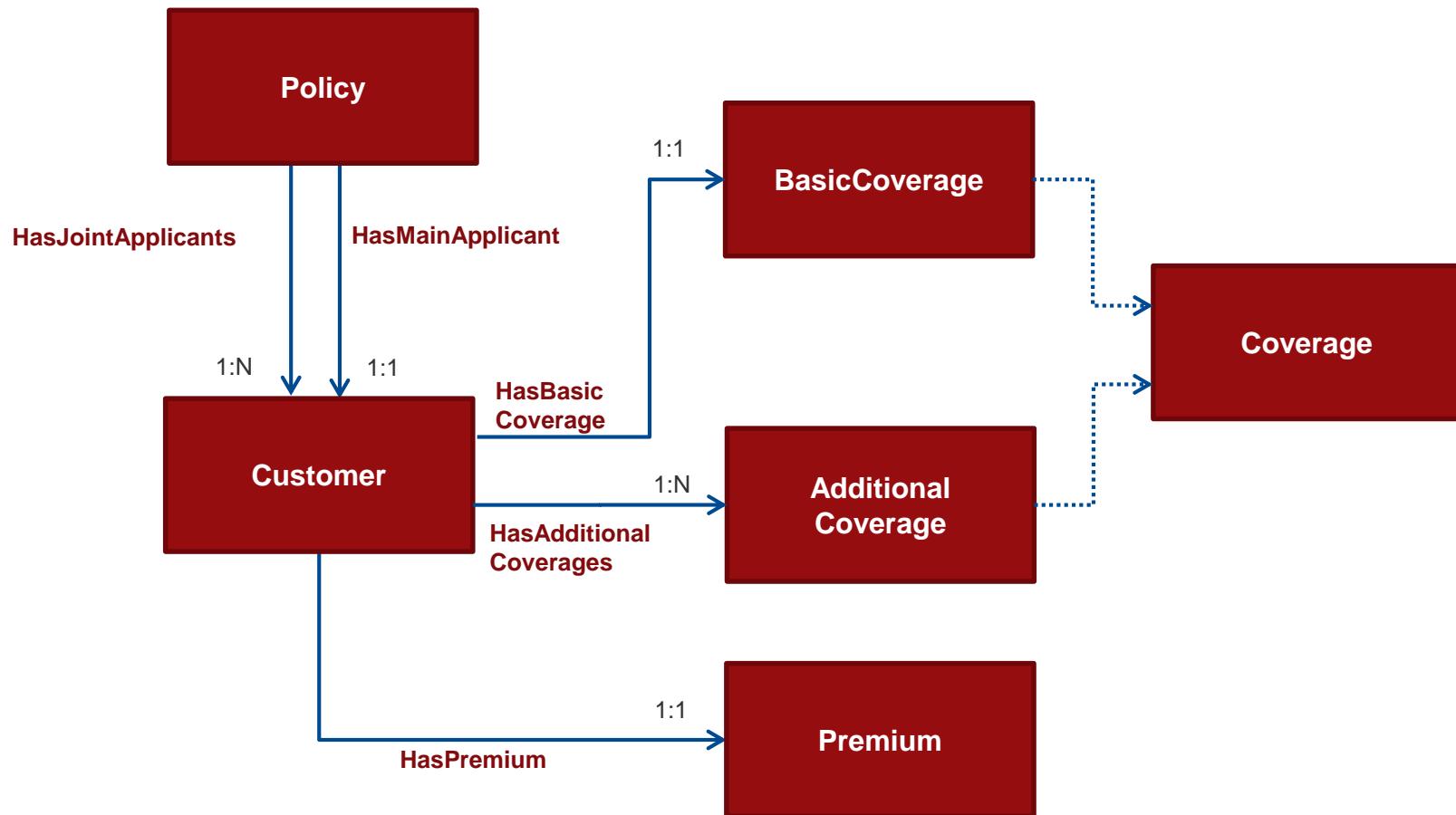
Returns the Coverage of the MainApplicant of the Policy

Relations & Expressions (2)

- Dotted notations can be used for single valued relations and attributes.
- Entity.Relation.Attribute



Relations & Expressions (3)

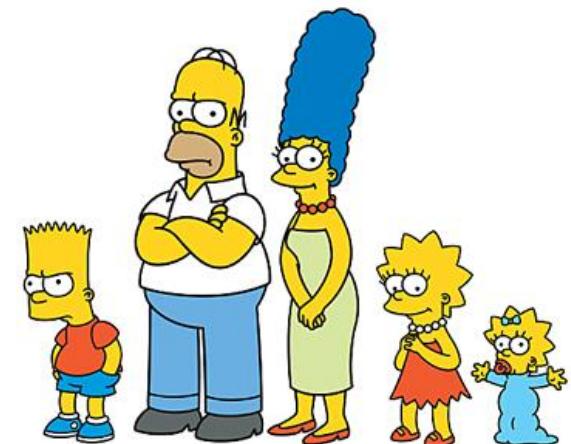
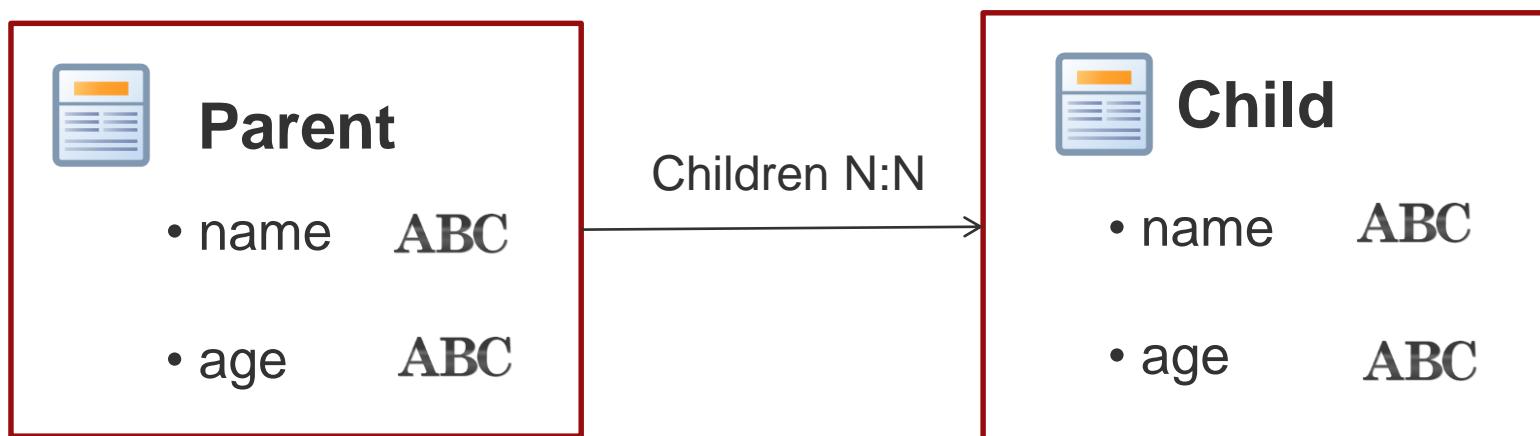


Relations & Expressions (4)

- If you are dealing with multiple instances, you will have to specify which instances you are referring to.
- Use the ALL statement if you wish to refer to all instances that are present in your application.
- Use the COLLECT statement to select the instances via a specified relation.

Relations & Expressions (5)

- Multiple instances

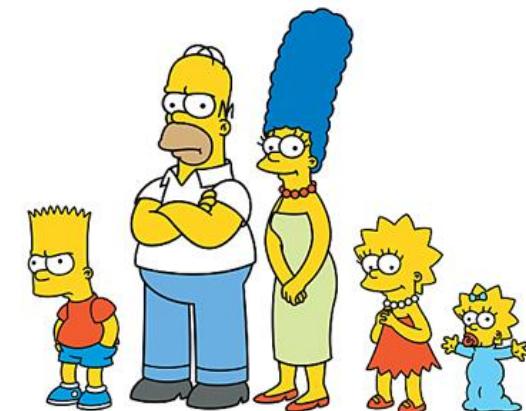


Relations & Expressions (6)

- ALL entity
 - ALL Parent
 - returns a list of all instances of entity Parent
- EXISTS entity [WHERE condition]
 - EXISTS Person WHERE (Person.age < 18)
 - returns true if an instance of Person exists with age < 18, otherwise false
- COLLECT entity | attribute FROM collection [WHERE condition]
 - COLLECT Child.name FROM ALL Child WHERE (Child.age < 18)
 - returns a list of names of all children with age < 18

Relations & Expressions (7)

- Select all children
ALL Child
- Select all children of a parent (in scope)
COLLECT Child FROM Parent.Children
- Select all of Homers' children
COLLECT Child FROM ALL Child
WHERE (Child.HasParent.Name = "Homer")
- Count all of Homers' children
SIZE(COLLECT Child FROM ALL Child
WHERE (Child.HasParent.Name = "Homer"))



Day 2

- Complex domain model
 - Instances, singleton and multiple instances, inheritance
 - Instance management, relations and pages & flows
- Complex logic
 - Relations and expressions
 - Overwrite inherited logic

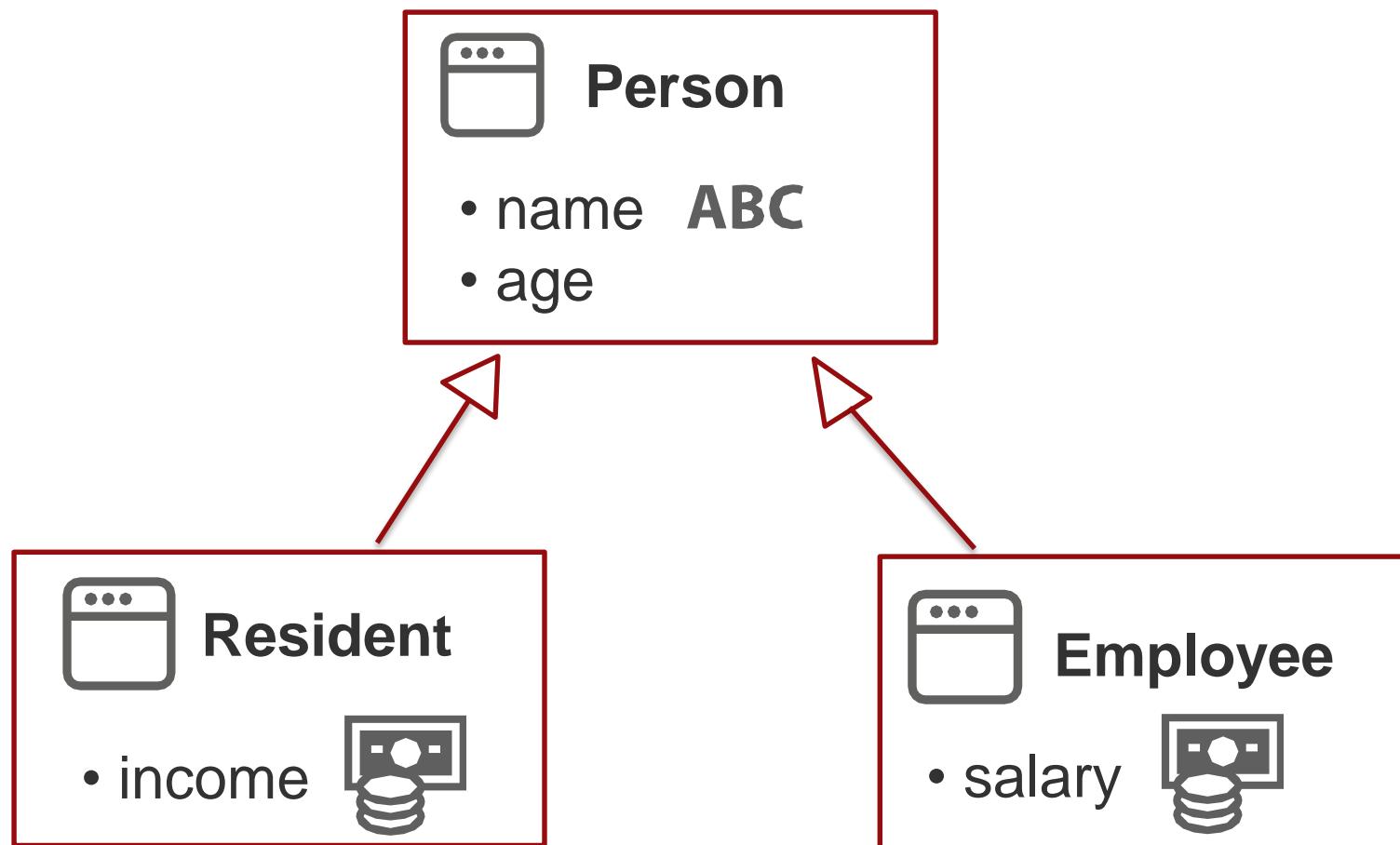
Overwriting inherited logic

Order of attribute sourcing

- Derived entity
 - Business rule/decision table/external rule
- Base entity
 - Business rule/decision table/external rule
 - Default value

First, the inference engine looks for a specific deduction. If no specific deduction exists, then a generic deduction will be used.

Overwriting inherited logic – Example (1)



Overwriting inherited logic – Example (2)



```
IF Person.age >= 18 then  
    Person.Adult = TRUE
```



```
IF Resident.age > 21 then  
    Resident.Adult = TRUE
```

Assignment 5

- Joint applicants
- Complex logic

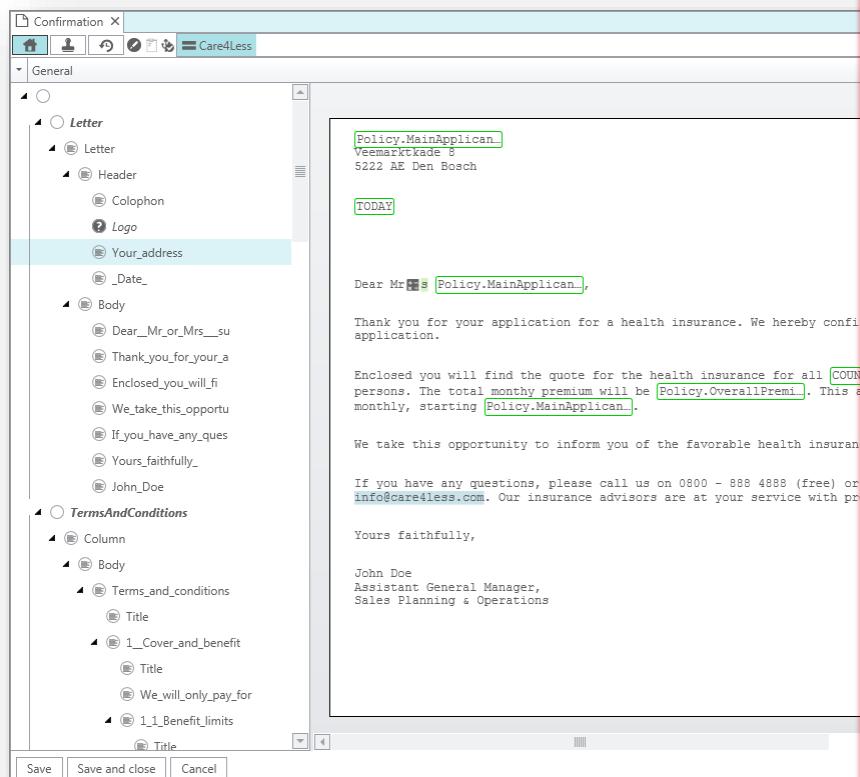
Day 3

- Documents
- Optional: Decision trees

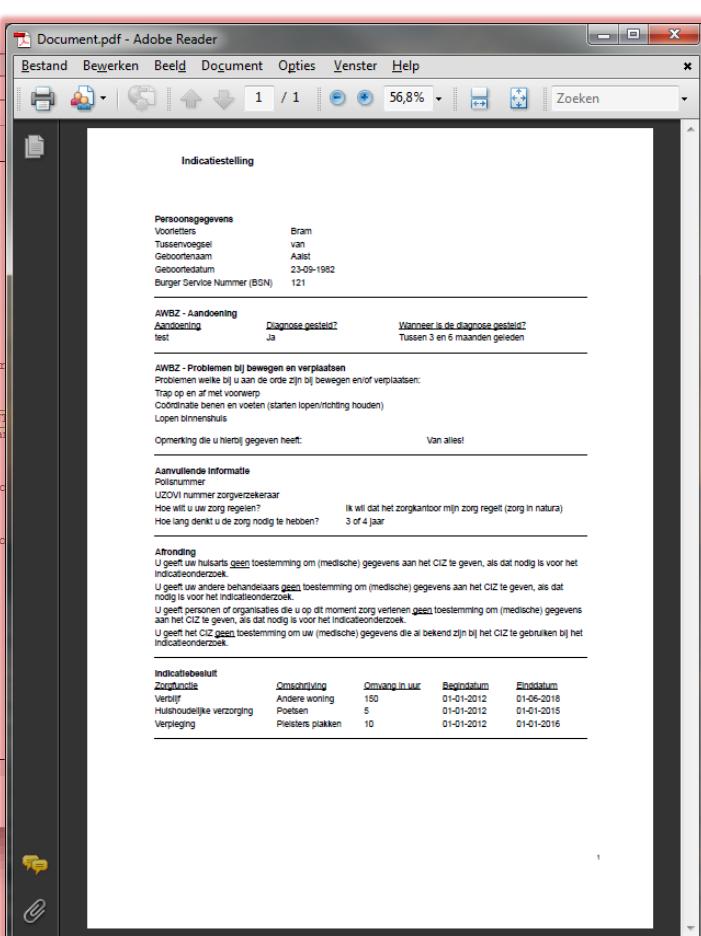
Documents



Documents



The screenshot shows the Care4Less software interface. On the left, a tree view displays document components: Letter (Letter, Header, Colophon, Logo, Your_address, Date, Body, Dear_Mr_or_Mrs_su, Thank_you_for_your_a, Enclosed_you_will_hi, We_take_thi_opportu, If_you_have_any_ques, Yours_faithfully_, John_Doe), TermsAndConditions (Column, Body, Terms_and_conditions (Title, 1_Cover_and_benefit (Title, We_will_only_pay_for), 1_1_Benefit_limits (Title))), and a general section. The 'Your_address' node is currently selected. On the right, a preview window shows a letter to 'Policy.MainApplicant' at 'Veemarktkade 8, 5222 AE Den Bosch'. The letter starts with 'TODAY' and ends with 'Yours faithfully,' followed by 'John Doe, Assistant General Manager, Sales Planning & Operations'.



The screenshot shows a PDF document titled 'Document.pdf' in Adobe Reader. The document includes sections for 'Indicatiestelling' (indications) and 'Aanvullende informatie' (additional information). In the 'Indicatiestelling' section, there is a table with columns for 'Voorletters' (First names), 'Tussenvoegsel' (Middle name), 'Geboorenaam' (Name at birth), 'Geboortedatum' (Birth date), 'Burger Service Nummer (BSN)' (CIT number), 'Aanleiding' (Reason), 'Diagnose gesteld?' (Diagnosis established?), and 'Wanneer is de diagnose gesteld?' (When was the diagnosis established?). The table shows Bram van Aalist, born 23-09-1982, with a BSN of 121. The 'Diagnose gesteld?' column shows 'Ja' (Yes) and the 'Wanneer...' column shows 'Tussen 3 en 6 maanden geleden' (Between 3 and 6 months ago). The 'Aanvullende informatie' section contains a table with columns for 'Polisnummer' (Policy number), 'UZOVNI nummer zorgverzekeraar' (UZOVNI number of healthcare provider), 'Hoe wilt u uw zorg regelen?' (How do you want your care arranged?), 'Ik wil dat het zorgkantoor mijn zorg regelt (zorg in natura)' (I want the healthcare provider to manage my care (care in nature)), 'Hoe lang denkt u de zorg nodig te hebben?' (How long do you think you will need care?), and '3 of 4 jaar' (3 or 4 years). The 'Altronding' (Conclusion) section contains a note about giving permission for medical data to be used for research. The 'Indicatielijst' (Index of indications) table lists various medical conditions and treatments with their respective codes, descriptions, and dates.

Documents

- A document represents a piece of text in any form: a letter, quote, email etc.
- Documents can be:
 - Downloaded from a page
 - Sent by email
 - Stored in the internal DMS
 - Stored in an external source (database, DMS)
 -

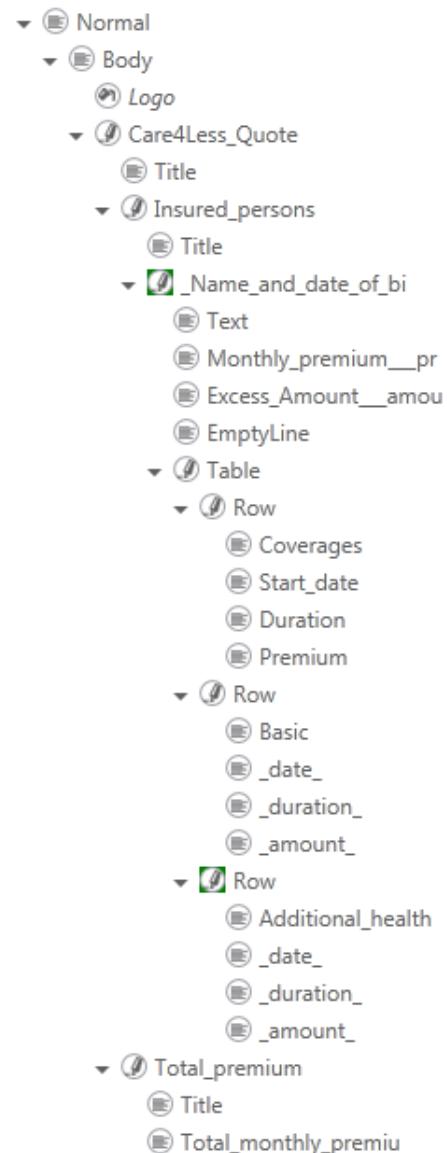


Documents

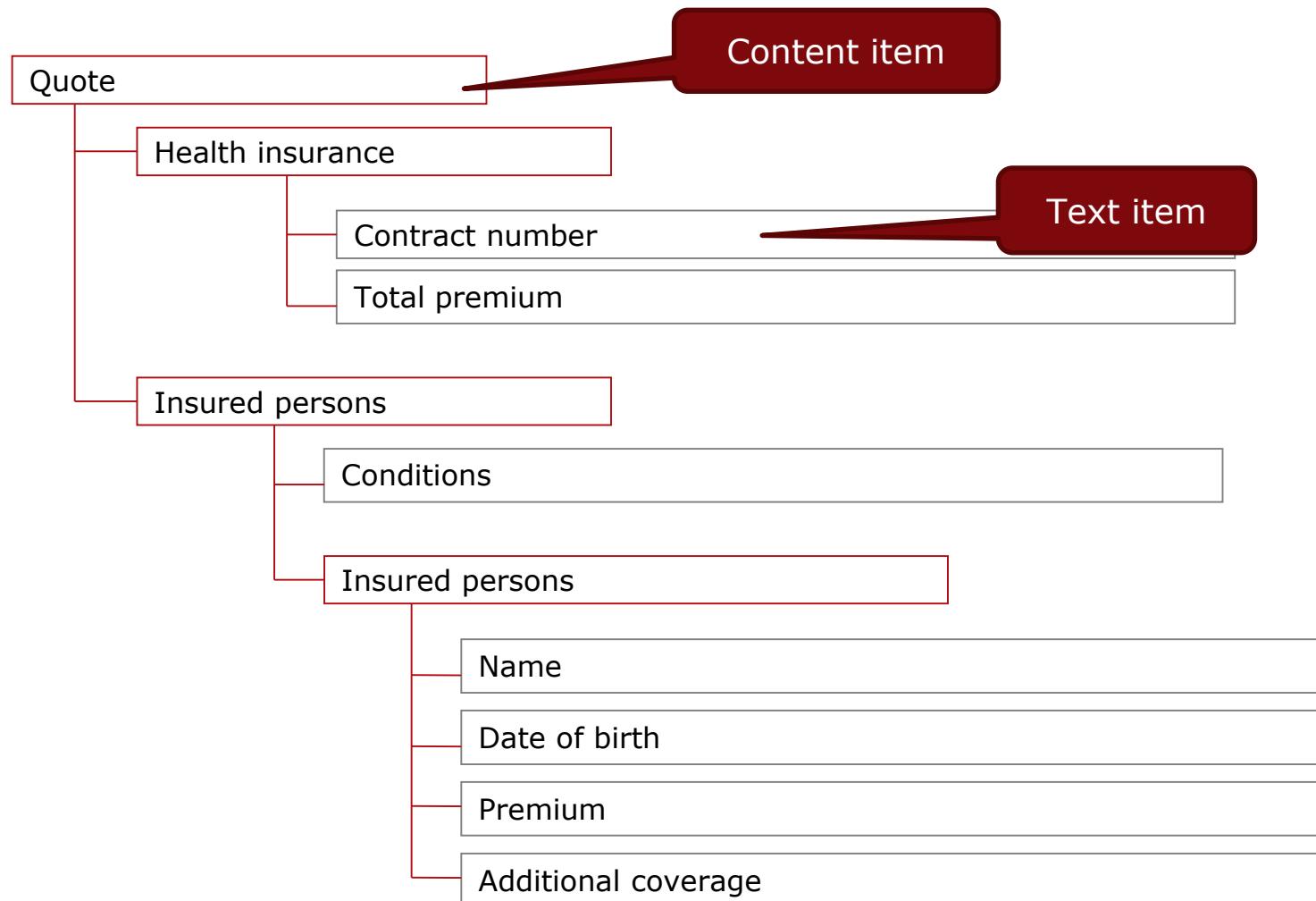
- Backbase Forms documents separate:
 - Content
 - Structure
 - Layout

Document Structure (1)

- The structure of a document is defined in the document tree
- The document tree contains content items
- Content items are used to group information. They can contain other content items, text items and images



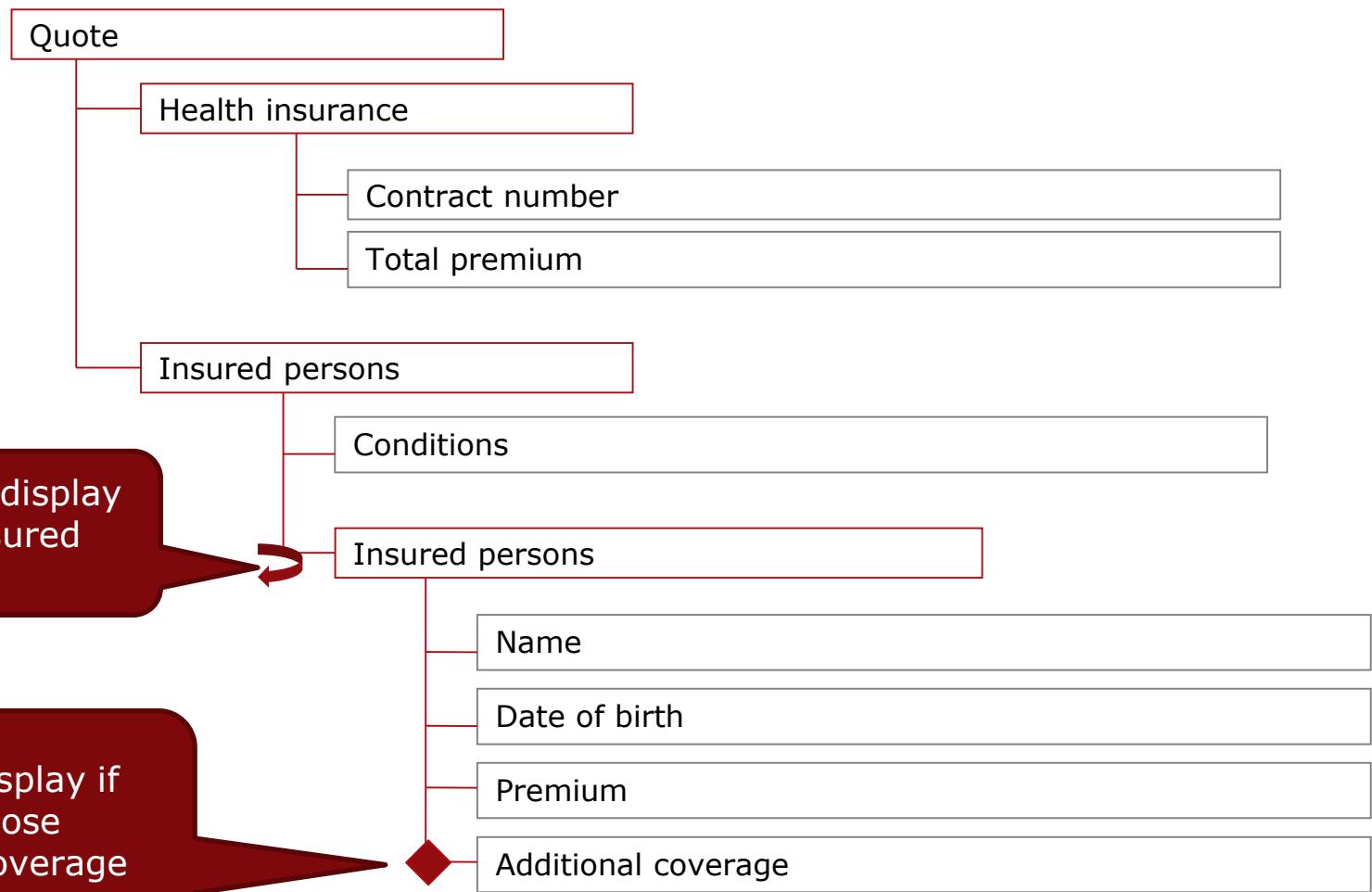
Document Structure (2)



Documents Structure (3)

- A condition will determine if the content item including its children is to be displayed
- Repetition allows the content item to be displayed for each instance of an entity

Document Structure (4)



Document Content

- A text item is a block of formatted multi-lingual text and fields.



- Fields can be added to the text item to show attribute values



- Pieces of text and fields within a text item can contain a condition.

Name	Thank_you_for_your_a
Content style	Paragraph
Presentation style	
Condition	Customer.PhoneNumber != ?

- An image can be imported into Backbase Forms Studio and can then be added to the document.



Document Layout

- Content styles and presentation styles are applied to content items for layout and formatting.
- Content styles and presentation styles are no more than names.
- The document renderer uses this information to determine the layout and the formatting.

Document Layout – Content style (1)

- Content styles are applied to content items for the layout.
- The type of content is determined by the content style, for instance heading, paragraph, list or table.
- Custom content styles can be added as a global object. FE developer needs to do his part!

Document Layout – Content Style (2)

Normal

HeaderFirst

Header

HeaderLast

Body

 Heading1

 Heading1Title

 Heading2

 Heading2Title

 Paragraph

 Paragraph

 Paragraph

 Paragraph

FooterFirst

Footer

FooterLast

Document Layout – Presentation Style (1)

- Presentation styles are applied to content items and text items for the formatting, for instance emphasis or alignment
- Model only functional elements
 - Emphasis versus Bold
 - Negative versus Red
- Custom presentation styles can be added, again the FE developer has to do his part!

Document Layout – Presentation Style (2)

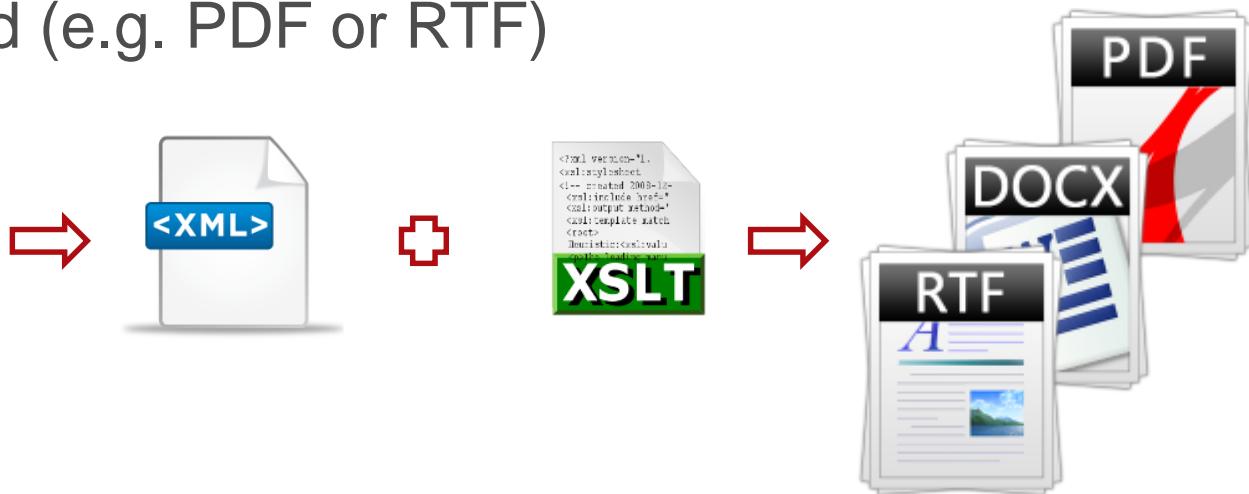
- Custom presentation styles can be added, for instance:
 - Negative
 - Positive
 - Large
 - Small

Document Layout - Masks

- A mask is used to determine the presentation of a field.
- There are masks for all attribute types, for example:
 - Boolean
 - presenting in words
 - Number/Currency/Percentage/Integer
 - formatting with varying precision 3.14159265359 : 3.14
 - using thousand separators
 - Date
 - selecting which part of the date or time to show
 - spelling it out in words

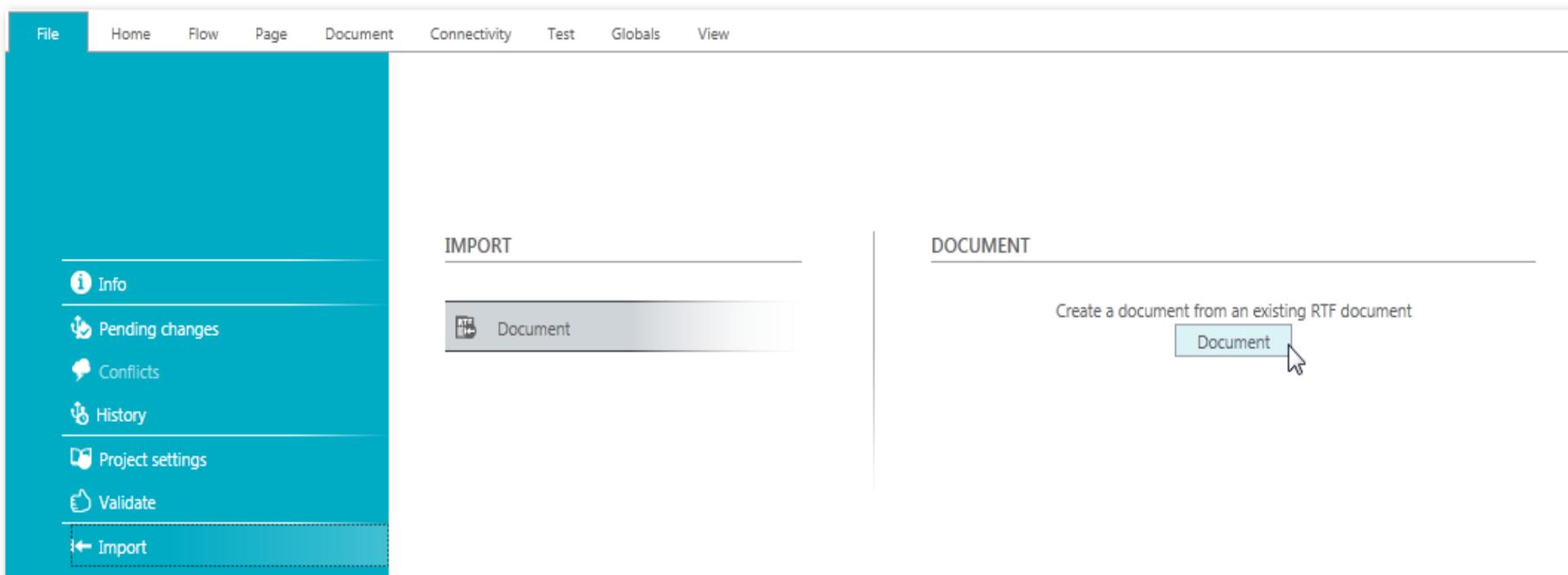
Document Output

- A document is rendered for the desired output channel with a stylesheet
- The stylesheet determines the type of document that is rendered (e.g. PDF or RTF)



Documents – RTF import wizard (1)

- Making a new document in Backbase Forms starts with
 - creating a Document element or ...



Documents – RTF Import wizard (2)

- Make document in Word, incl. styles, images, etc.
- Save as RTF
- Import RTF into Backbase Forms
 - Map styles from the source document to predefined Backbase Forms styles
 - Map hierarchical structure from source document to parent-child relations in Backbase Forms

The screenshot shows the 'RTF Import' wizard interface. At the top, there's a header bar with the title 'RTF Import X'. Below it is a teal-colored section titled 'Page Caption' with the sub-instruction: 'We found several styles that need your attention. Please match the corresponding styles.' A green box highlights the section 'Care4Less Quote'. Underneath, the text 'Insured persons' is shown. Below that is a placeholder '[Name and date of birth main applicant]'. Further down, there's a table for 'Cov coverages' with three rows: 'Basic', 'Additional health', and 'Additional dental'. Each row has columns for 'Start date', 'Duration', and 'Premium'. Below the table is another placeholder '[Name and date of birth joint applicant]'. At the bottom, there's a large mapping table with columns for 'Style', 'Content style', 'Presentation style', and 'Style acts as parent node'. Several rows are listed, including 'Text with style' (selected), 'Empty line', 'Text with s', 'Text with sty', 'Text with style', 'Table', 'Table row', and 'Table cell'. The 'Text with style' row is highlighted with a teal background.

Style	Content style	Presentation style	Style acts as parent node
Text with style	Paragraph		<input checked="" type="checkbox"/>
Empty line	EmptyLine		
Text with s	Heading1		<input checked="" type="checkbox"/>
Text with sty	Heading2		<input checked="" type="checkbox"/>
Text with style	Heading3		<input checked="" type="checkbox"/>
Table	Table		
Table row	TableRow		
Table cell	TableCell		

Assignment 6

- Documents
 - Quote

Links

- Home Page: www.backbase.com
- Extranet: my.backbase.com
- Feedback: www.backbase.com/feedback

Day 3

- Documents
- Optional: Decision trees

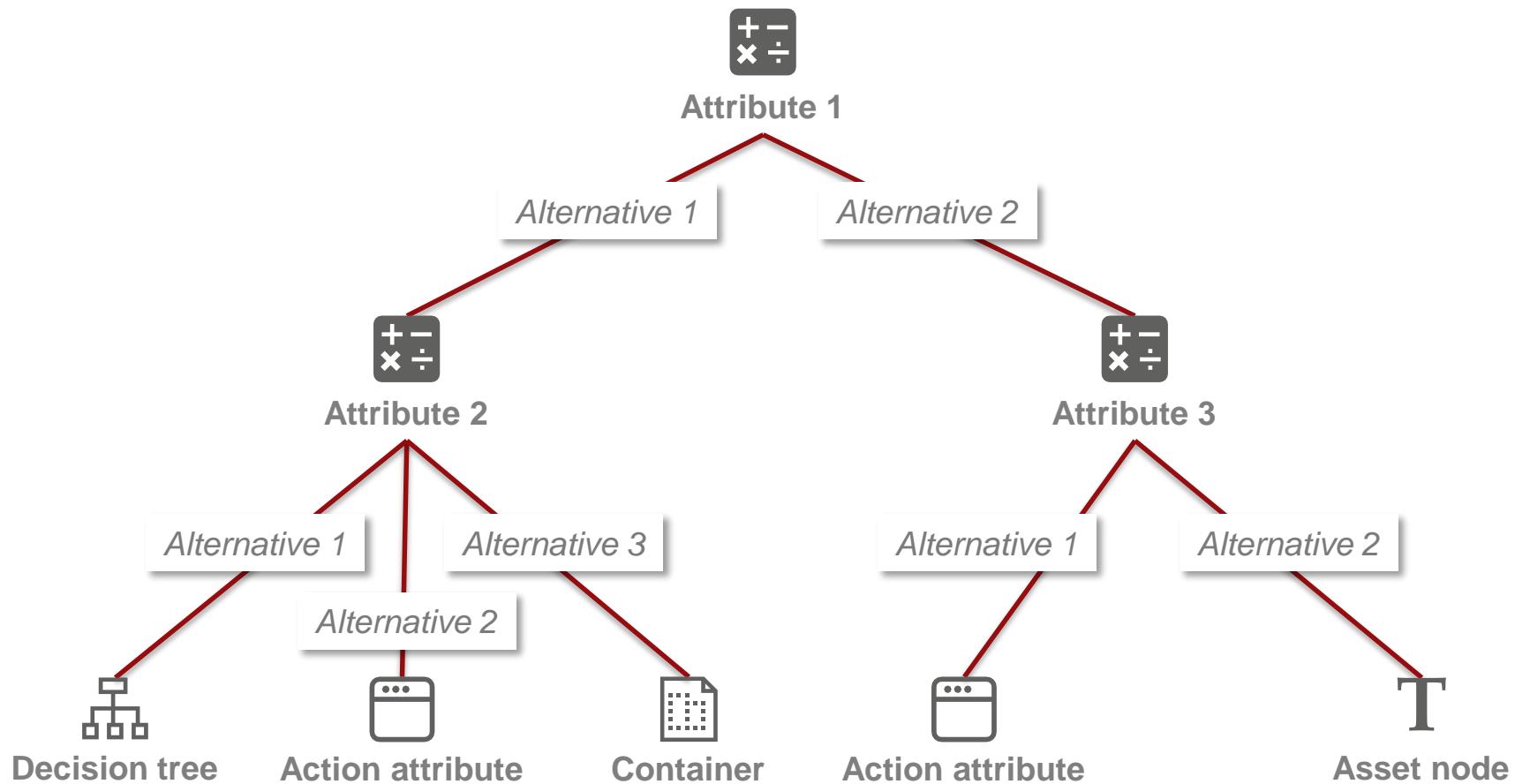
Decision Trees



Decision Trees (1)

- A decision tree is a tree-like graphical representation of all alternatives in a decision making process.
- Decision trees are commonly used for advice or diagnosis.
- A decision tree is the implementation of a question script.

Decision Trees (2)



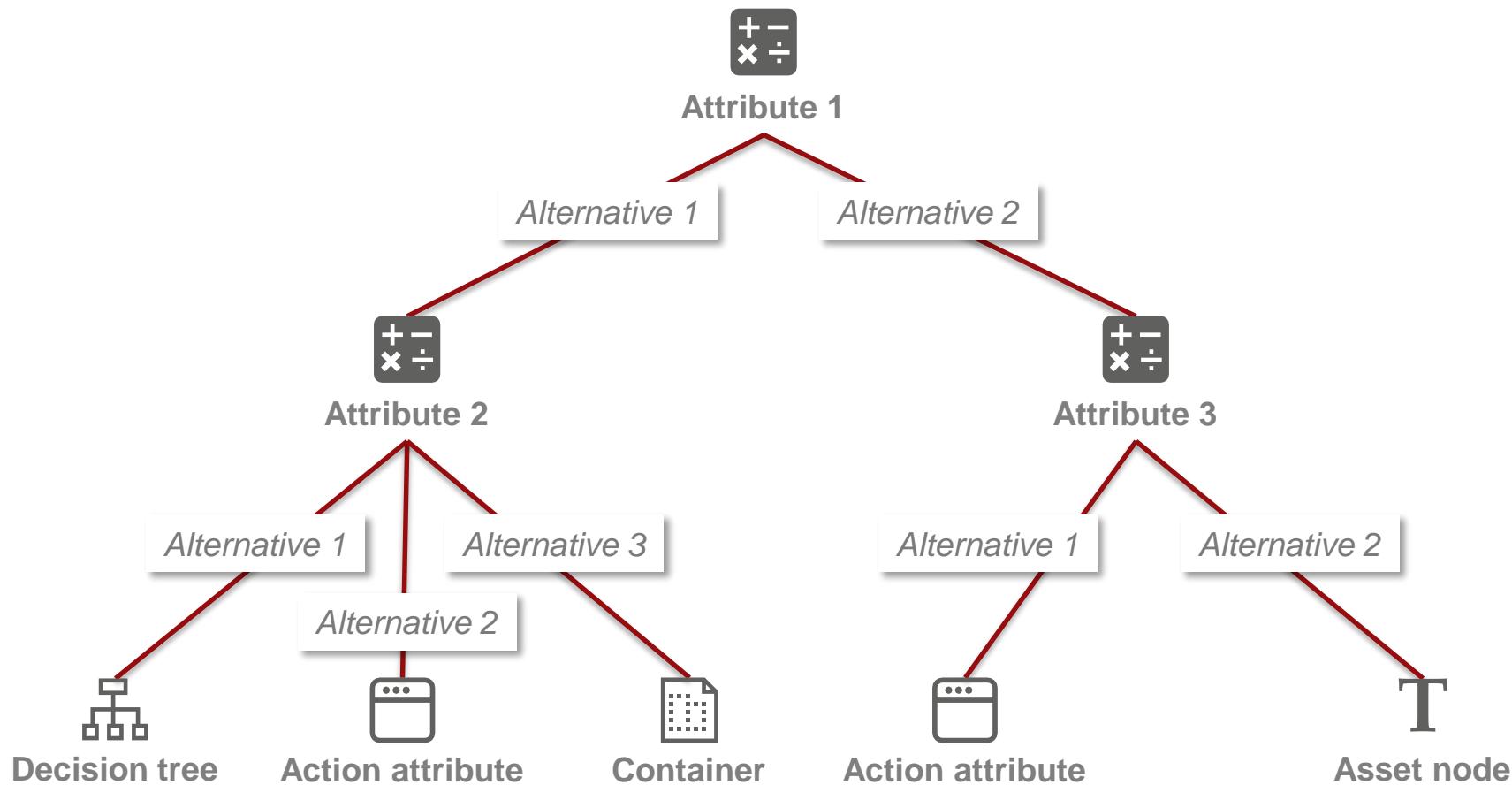
Decision Trees (3)

- A question node corresponds to a question (attribute) asked to the user and the possible answers (attribute values).
- If an attribute is askable the value of the attribute can be asked to the user.
- If an attribute is not askable, the value of the attribute has to be derived.

Decision Trees (4)

- A leaf represents the conclusion based upon the path of given answers
- A leaf can be:
 - Action node: set an attribute
 - Container node: show the user a container
 - Asset node: show the user a message
 - Decision tree: start another tree

Decision Tree



Decision Tree Selection

- A decision tree has to be explicitly called on a page to show it to a user.
- The AQ_DecisionTreeSelector is a type of container that can be used to search for a decision tree that matches certain symptoms.
- A symptom is a keyword which describes (part of) the content of a decision tree.

Decision Tree Evaluation

- An AQ DecisionTreeEvaluator is a type of container that executes a decision tree.
- The dynamic container asks the user questions and depending on the given answer, it determines the next node to execute.
- A decision tree can be searched based on its symptoms and then started, or it can be started directly.

Symptoms – Trigram matching

INFERENCE

CONFERENCE

_IN
INF

NFE
FER
ERE
REN
ENC
NCE
CE_

NFE
FER
ERE
REN
ENC
NCE
CE_

_CO
CON
ONF

NFE
FER
ERE
REN
ENC
NCE
CE_

Decision Table vs. Decision Tree

Decision table

- The conclusion is the deduction of one or more attribute values.
- All question paths have the same length.
- The decision table is used by inference engine to source the action attribute.

Decision tree

- There are 4 different types of conclusions: action nodes, container nodes, asset nodes, decision trees.
- The question paths have various lengths.
- A decision tree is triggered and NOT used by the inference engine.

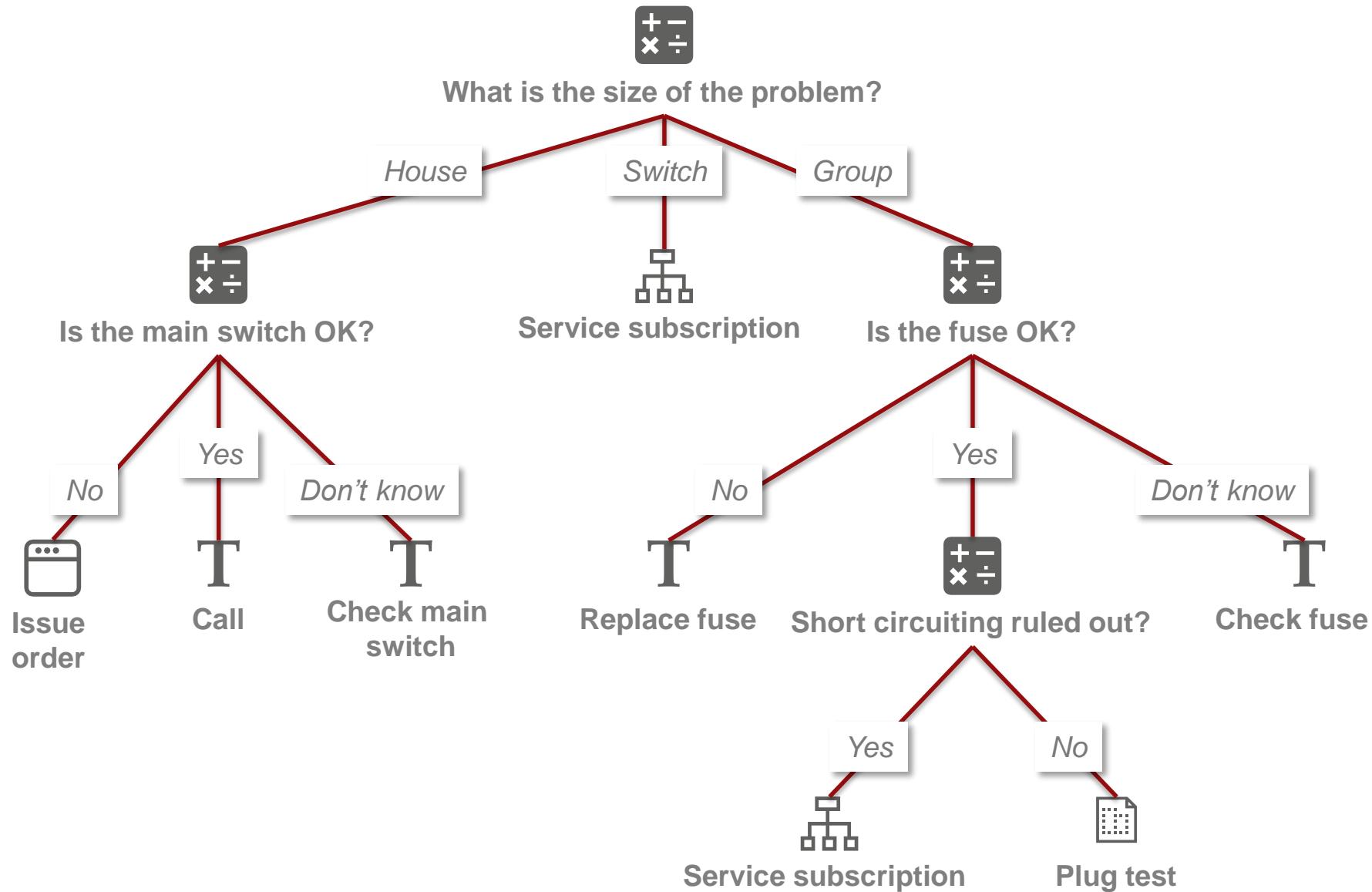
Building a decision tree - questions

1. Identify the questions that must be asked
 - The less questions the better
 - Does the question contribute to get to the solution?
2. Decide in which form each question should be asked
 - Try to use value lists instead of yes/no questions
 - Avoid negations in questions
3. Determine the question order
 - Avoid duplication when there are multiple paths that lead to the same conclusion

Building a decision tree - conclusions

1. Identify the possible conclusions
2. Determine which situation should lead to which conclusion
3. Outline a diagram with questions, answers and conclusions

Decision Tree - example



Links

- Home Page: www.backbase.com
- Extranet: www.mybackbase.com
- Feedback: www.backbase.com/feedback