

Cloud Environment Selection and Configuration: A Software Product Lines-Based Approach

Clément Quinton

Ph.D. Defense

Wednesday, October 22th 2014

Rapporteurs: Roberto Di Cosmo, *Université Paris Diderot*

Christian Perez, *LIP - ENS Lyon*

Examinateurs: Patrick Heymans, *Université de Namur*

Daniel Le Berre, *Université d'Artois*

Daniel Romero, *Université Lille 1*

Directrice: Laurence Duchien, *Université Lille 1*

Cloud Computing

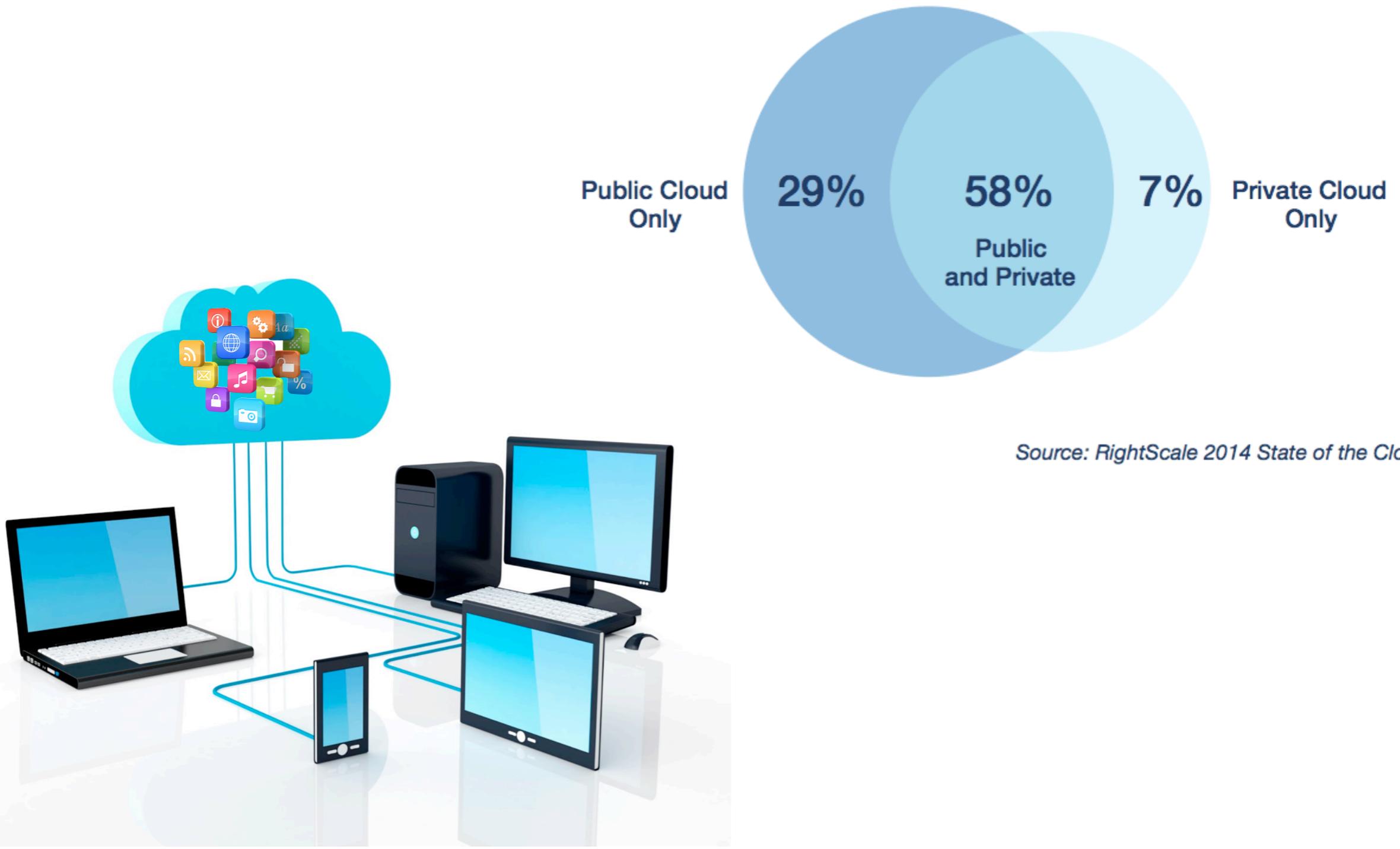


Cloud Computing



Cloud Computing

94% of Respondents Are Using Cloud



Cloud Computing

“Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction”

[NIST 2009]

Cloud Computing

*“Cloud computing is a **model** for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction”*

[NIST 2009]

SaaS

Application

PaaS

Libraries, Database

IaaS

Infrastructure

Cloud Computing

*“Cloud computing is a **model** for enabling convenient, on-demand network access to a shared pool of **configurable** computing **resources** that can be rapidly provisioned and released with minimal management effort or service provider interaction”*

[NIST 2009]

SaaS

Application

PaaS

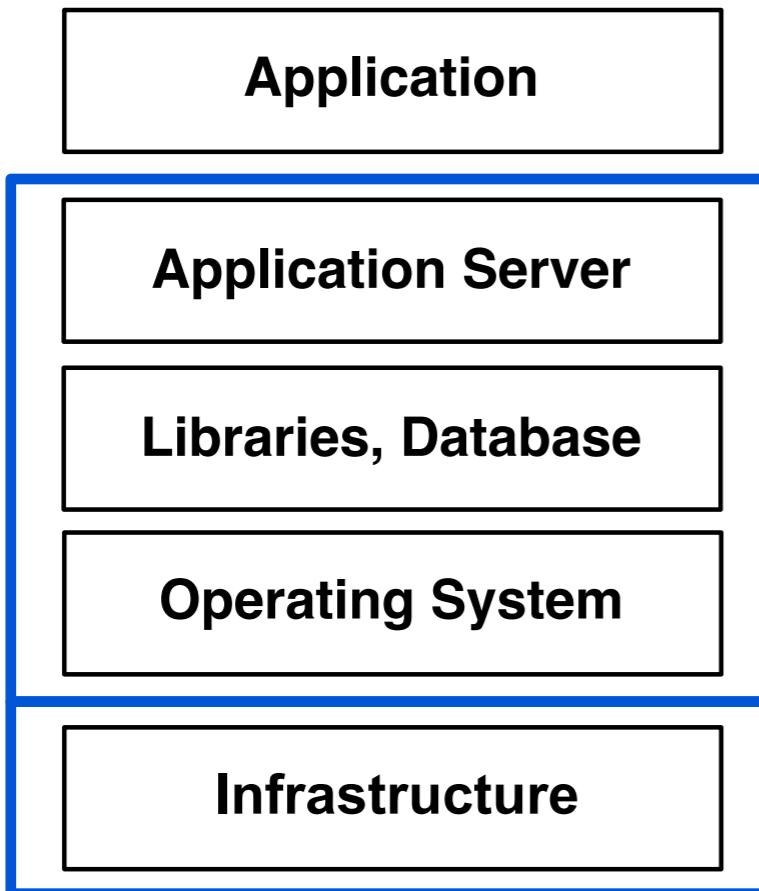
Application Server

Libraries, Database

Operating System

IaaS

Infrastructure



Cloud Computing

*“Cloud computing is a **model** for enabling convenient, on-demand network access to a shared pool of **configurable** computing **resources** that can be rapidly provisioned and released with minimal management effort or service provider interaction”*

[NIST 2009]

SaaS

Application

PaaS

Application Server

Libraries, Database

Operating System

IaaS

Infrastructure

Virtual Machines, resources (RAM, CPU, disk), etc.

Cloud Computing

*“Cloud computing is a **model** for enabling convenient, on-demand network access to a shared pool of **configurable** computing **resources** that can be rapidly provisioned and released with minimal management effort or service provider interaction”*

[NIST 2009]

SaaS

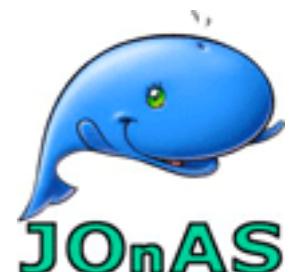
Application

PaaS

Application Server



jetty://



IaaS

Libraries, Database



MySQL

maven



Operating System

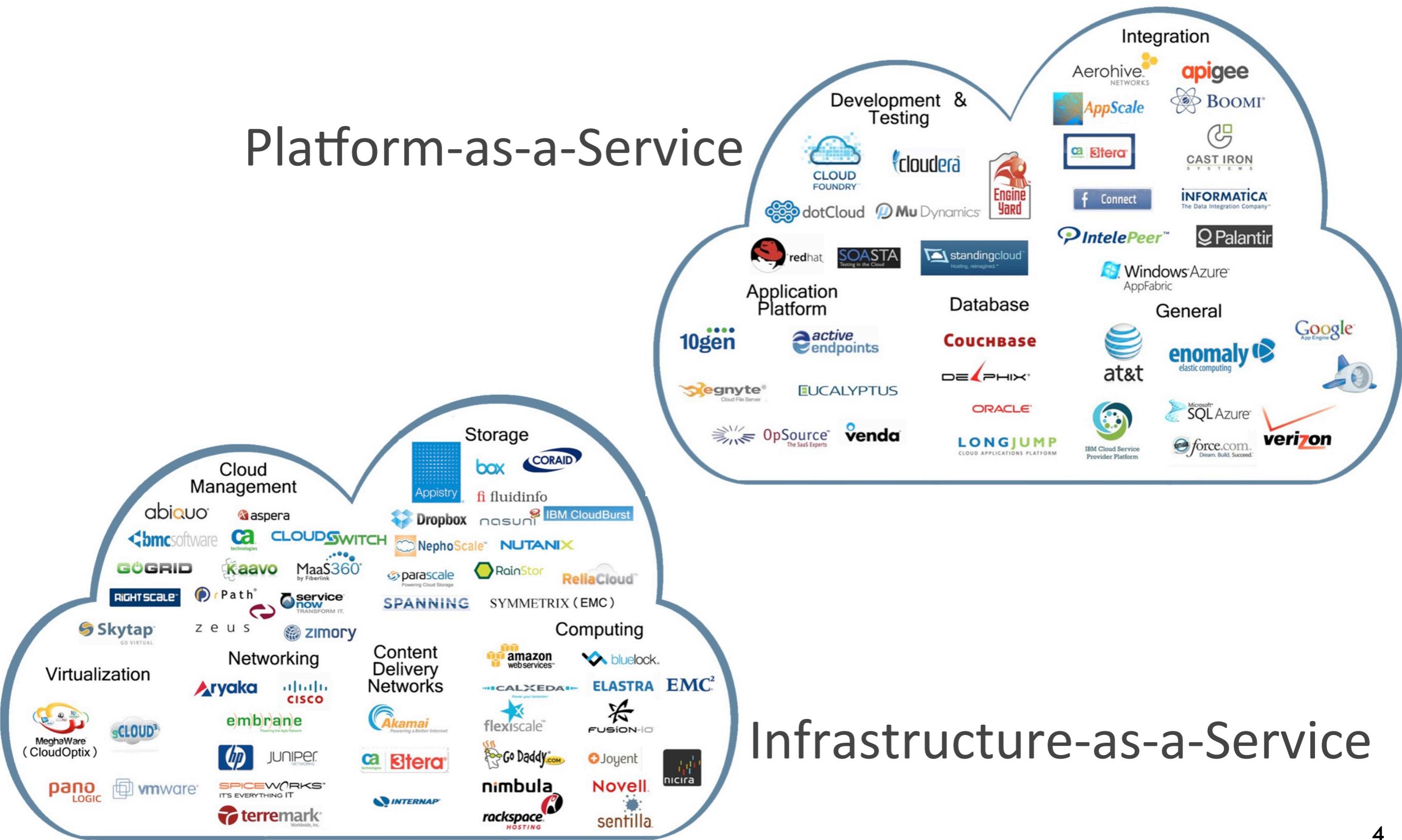


Infrastructure

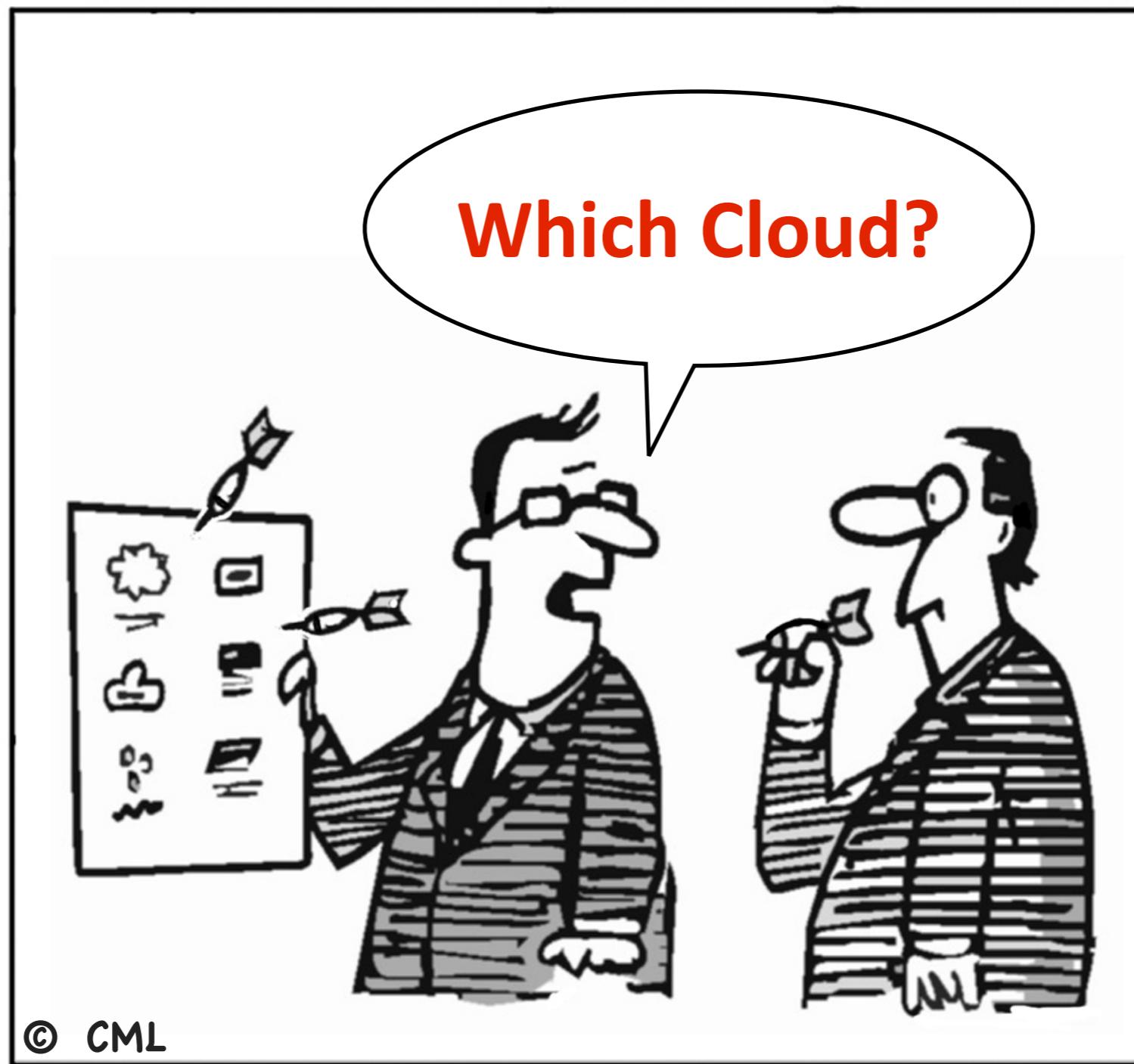
Virtual Machines, resources (RAM, CPU, disk), etc.

Cloud Computing

Platform-as-a-Service



Problem #1



Problem #1



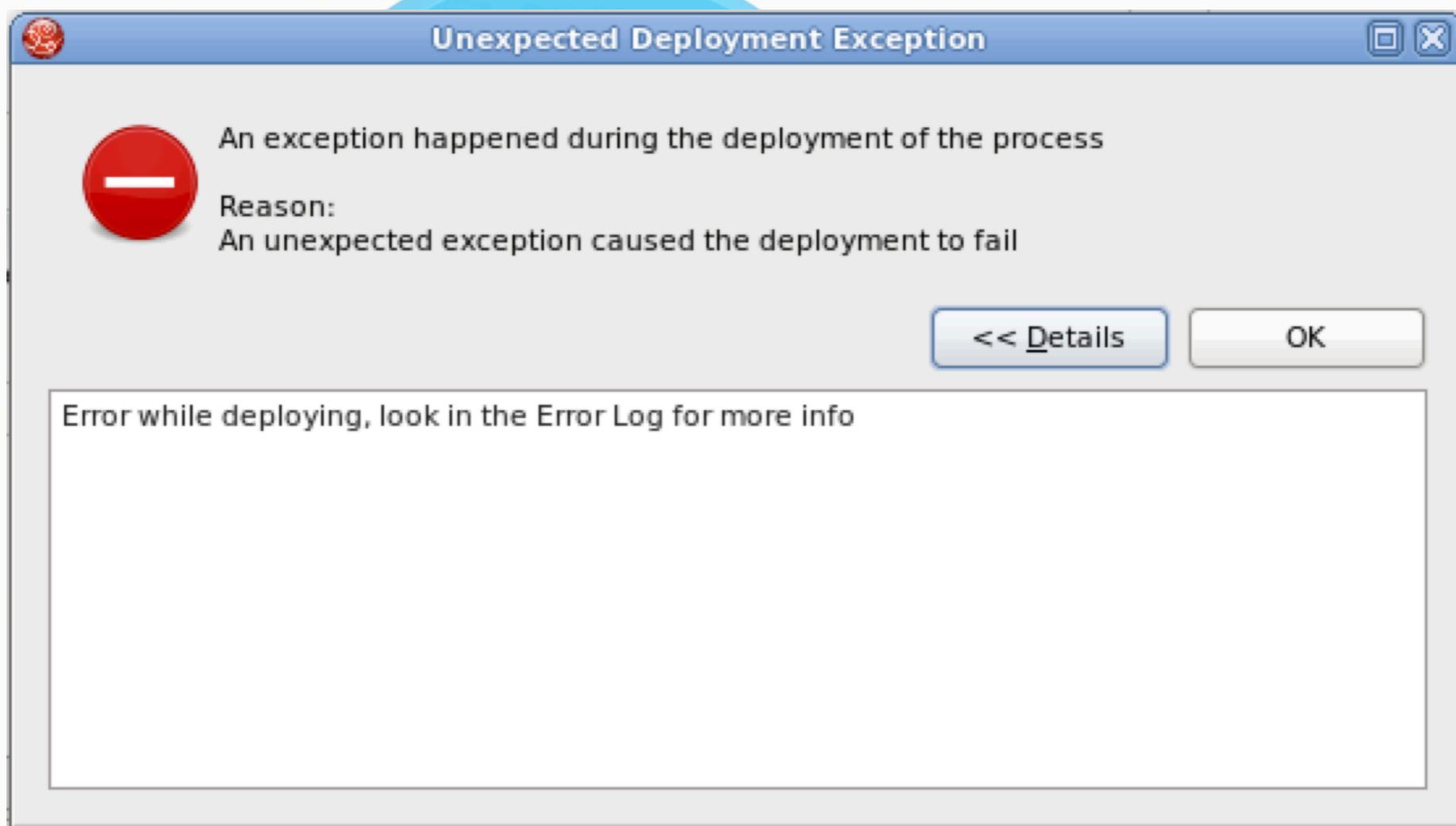
Problem #2



Problem #2



Problem #2



Problem Statement

Problem Statement

- Lack of selection support

Problem Statement

- Lack of selection support
- Lack of unified representation

Problem Statement

- Lack of selection support
- Lack of unified representation
- Heterogeneous configuration processes

Problem Statement

- Lack of selection support
- Lack of unified representation
- Heterogeneous configuration processes
- Limited automated support

Main Criteria

Main Criteria



Main Criteria

- Flexibility



Main Criteria

- Flexibility
 - Requirement specification, reasoning mechanism



Main Criteria

- Flexibility
 - Requirement specification, reasoning mechanism
- Maintainability



Main Criteria

- Flexibility
 - Requirement specification, reasoning mechanism
- Maintainability
 - Add, remove, update Clouds



Main Criteria

- Flexibility
 - Requirement specification, reasoning mechanism
- Maintainability
 - Add, remove, update Clouds
- Abstraction



Main Criteria

- Flexibility
 - Requirement specification, reasoning mechanism
- Maintainability
 - Add, remove, update Clouds
- Abstraction
 - Correct granularity level(s)



Main Criteria

- Flexibility
 - Requirement specification, reasoning mechanism
- Maintainability
 - Add, remove, update Clouds
- Abstraction
 - Correct granularity level(s)
- Heterogeneity



Main Criteria

- Flexibility
 - Requirement specification, reasoning mechanism
- Maintainability
 - Add, remove, update Clouds
- Abstraction
 - Correct granularity level(s)
- Heterogeneity
 - Different Clouds



Existing Approaches

Approach	Flexibility	Maintainability	Abstraction	Heterogeneity
Aeolus	✓	(✓)	✓	
ConPaaS	(✓)			(✓)
mOSAIC	(✓)	(✓)		
MODAClouds	(✓)	(✓)	✓	
REMICS		✓	✓	
CloudGenius	(✓)			(✓)
CloudMIG		(✓)	✓	✓
Zeel/i	(✓)			(✓)

✓: proposes or deals (partially) with the criteria

Research Goals

Research Goals

- Manage Cloud variability

Research Goals

- Manage Cloud variability
- Guarantee environment independance

Research Goals

- Manage Cloud variability
- Guarantee environment independance
- Provide a flexible and practical solution

Research Goals

- Manage Cloud variability
- Guarantee environment independance
- Provide a flexible and practical solution
- Deliver an automated support

Research Goals

- Manage Cloud variability
- Guarantee environment independance
- Provide a flexible and practical solution
- Deliver an automated support
- Maintain consistency

Agenda

I. Introduction

Agenda

I. Introduction

II. Contributions

Agenda

I. Introduction

II. Contributions

- Cloud environments variability modeling

Agenda

I. Introduction

II. Contributions

- Cloud environments variability modeling
- SALOON

Agenda

I. Introduction

II. Contributions

- Cloud environments variability modeling
- SALOON
- Consistency checking for evolving Cloud models

Agenda

I. Introduction

II. Contributions

- Cloud environments variability modeling
- SALOON
- Consistency checking for evolving Cloud models

III. Conclusion and Perspectives

Agenda

I. Introduction

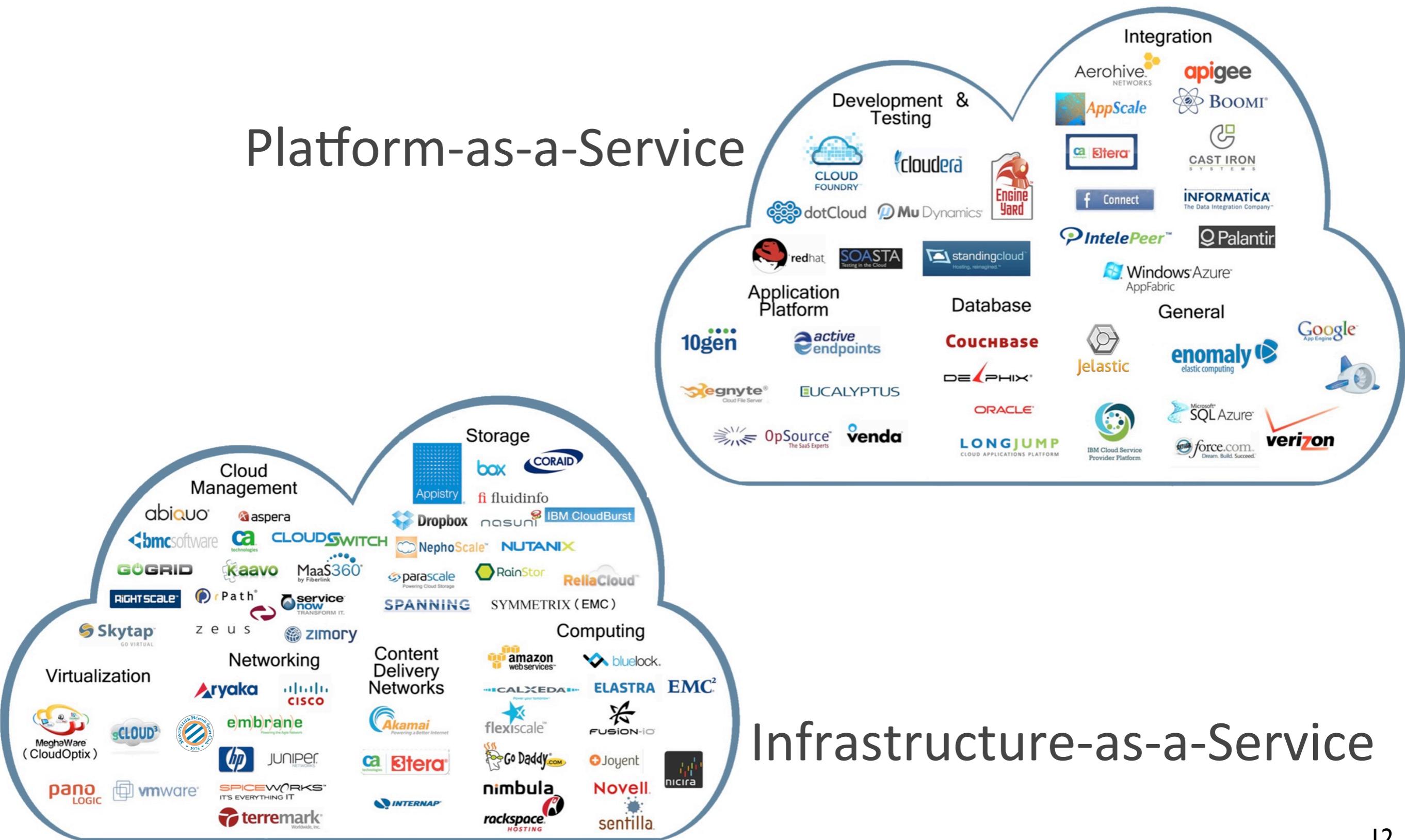
II. Contributions

- Cloud environments variability modeling
- SALOON
- Consistency checking for evolving Cloud models

III. Conclusion and Perspectives

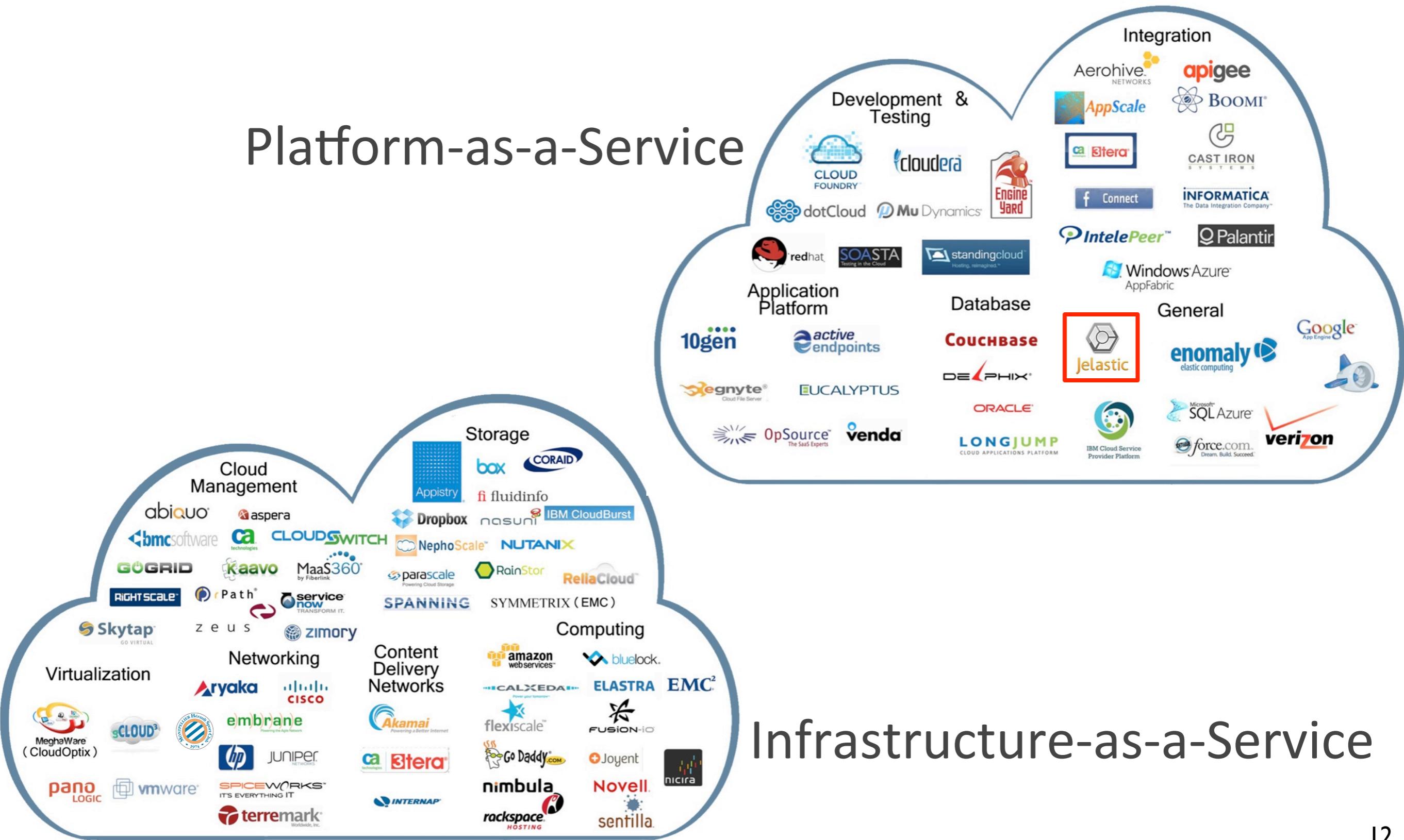
Cloud Environment Variability Modeling

Platform-as-a-Service



Cloud Environment Variability Modeling

Platform-as-a-Service



Infrastructure-as-a-Service

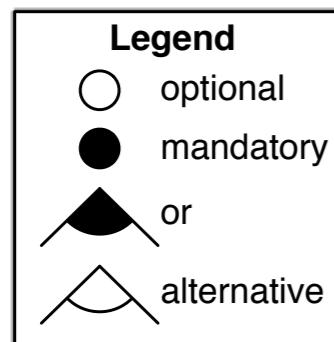
Use Case : Jelastic

The screenshot shows the Jelastic interface for creating a new application environment. The left side features a flowchart diagram for a Java application, starting with 'Balancing' at the top, followed by a central application node (represented by a cat icon), and then 'Cache', 'SQL', and 'NoSQL' databases at the bottom. Below the database layer are 'VDS', 'SSL', and 'Build' options. The right side of the interface contains configuration settings:

- Application Servers**: A switch is set to **ON**.
 - Vertical scaling per node**: Shows 1 reserved cloudlet(s) with 128 MB, 200 MHz, and 0.0085 EUR per unit. A discount of 15% is applied.
 - Horizontal scaling**: Shows 1 node(s) selected from a range of 0 to 64.
 - Tomcat 7.0** and **Java 7** are chosen as the application server and Java version respectively.
 - High-availability** and **Public IPv4** options are turned off.
- Resources (cloudlets)**:
 - Reserved cloudlets**: Now: 1, Next: 8. Discount: 15% (green) and 25% (green).
 - Total Reserved Cloudlets**: €6 per month. A note says "You're saving €1/month by using Reserved Cloudlets".
 - Total monthly cost**: €6*. A note says "*Free of charge for the trial period."
- Environment name**: The input field contains **env-2897066** and **.jelastic.dogado.eu**.

At the bottom right are **Cancel** and **Create** buttons.

Use Case : Jelastic



The screenshot shows the Jelastic application builder interface. At the top, there are tabs for JAVA and PHP, with JAVA selected. Below the tabs is a legend box containing the following items:

- optional
- mandatory
- ▲ or
- △ alternative

The main area displays a Java application stack. At the top is a blue box labeled "Balancing" with a cat icon. Below it is a blue box labeled "Application Servers" with a switch set to "ON". A tooltip for "Application Servers" indicates "Vertical scaling per node" with "1 cloudlet(s)" reserved, "128 MB, 200 MHz" resources, and a "Discount: 15%". A slider for horizontal scaling ranges from 0 to 64, currently set at 0. Below the servers are three blue cylinders labeled "Cache", "SQL", and "NoSQL". At the bottom of the stack are three blue boxes labeled "VDS", "SSL", and "Build". To the right of the stack, there are sections for "Resources (cloudlets)", "Total Reserved Cloud", "Total monthly cost", "Cost details", "Quotas & pricing", and "Environment name".

Use Case : Jelastic

Legend

- optional
- mandatory
- or
- alternative

Application Server

Resources (cloudlet)

Reserved cloudlets
Now: **1**
Discount: **15%**

Total Reserved Cloud

You're saving **€1/month** b

Vertical scaling per node

Application Servers **ON**

Reserved **1 cloudlet(s)**
128 MB, 200 MHz
0.0085 EUR per..
Discount: **15%**

Horizontal scaling

1 node(s)

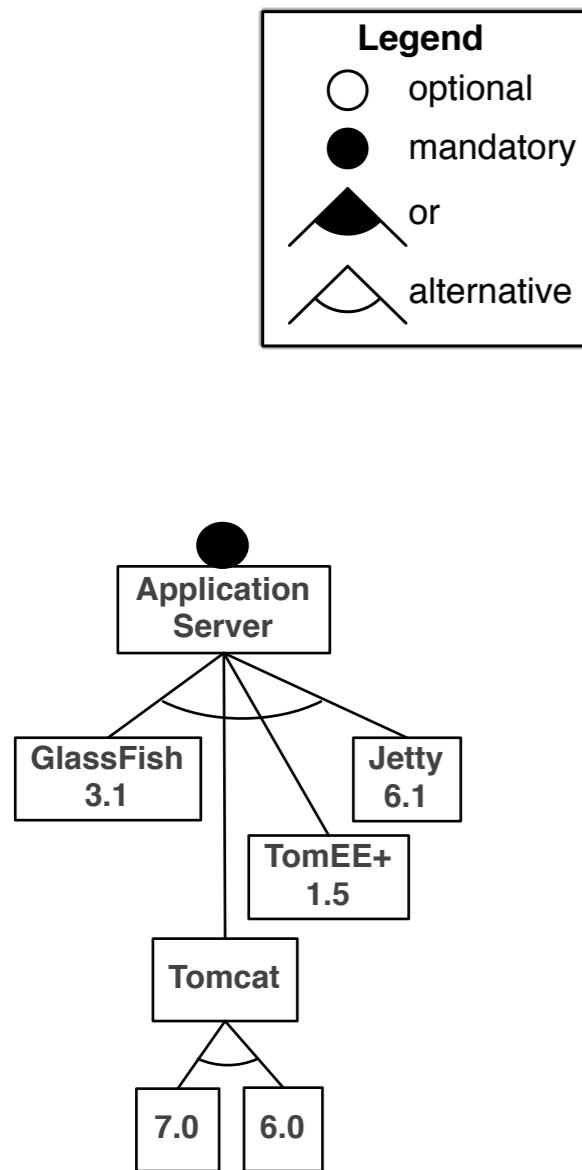
Tomcat 7.0 **Java 7**

High-availability **OFF**

Public IPv4 **OFF**

Environment name
env-2897066

Use Case : Jelastic



JELASTIC CLOUD

PHP **JAVA**

Balancing

Application Servers **ON**

Vertical scaling per node

Reserved **1 cloudlet(s)**
128 MB, 200 MHz
0.0085 EUR per..
Discount: **15%**

Horizontal scaling

1 node(s)

Total Reserved Cloud

You're saving **€1/month** b

Total monthly cost

*Free of charge for the tr

Cost details

Quotas & pricing

Environment name

env-2897066

Application Servers

Tomcat 7.0 **Java 7**

GlassFish 3.1
Jetty 6.1
TomEE+ 1.5
Tomcat 6.0
Tomcat 7.0

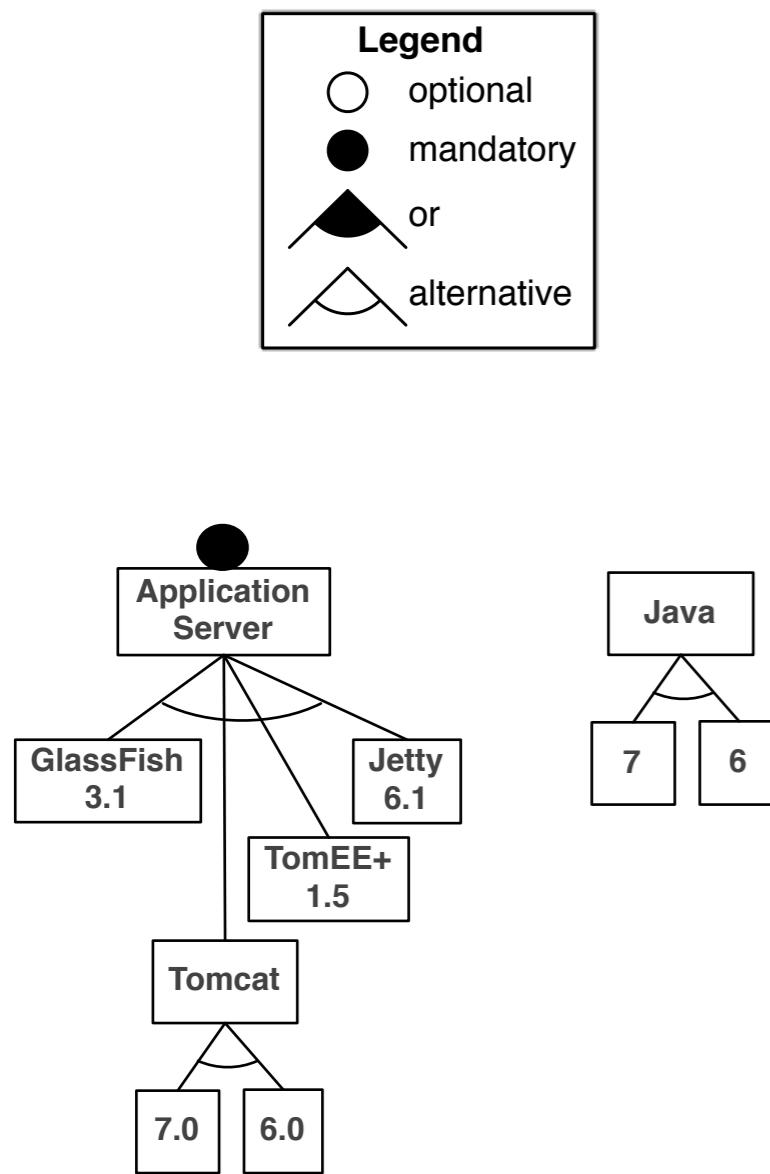
SQL NoSQL

VDS SSL Build

High-availability **OFF**

Public IPv4 **OFF**

Use Case : Jelastic



Jelastic Platform UI

Application Servers (ON)

Vertical scaling per node

Reserved 1 cloudlet(s)
128 MB, 200 MHz
0.0085 EUR per..
Discount: 15%

Horizontal scaling

1 node(s)

High-availability

Public IPv4

Resources (cloudlets)

Reserved cloudlets
Now: 1
Discount: 15%

Total Reserved Cloud

You're saving €1/month b...

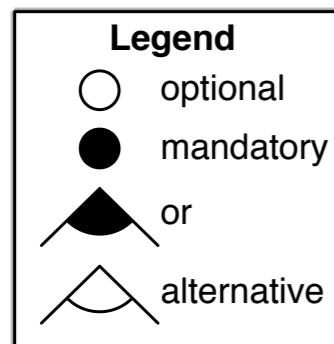
Total monthly cost
*Free of charge for the tra...

Cost details

Quotas & pricing

Environment name
env-2897066

Use Case : Jelastic



JAVA **PHP**

Balancing

Application Servers **ON**

Vertical scaling per node

Reserved **1 cloudlet(s)**
128 MB, 200 MHz
0.0085 EUR per..
Discount: **15%**

Horizontal scaling

1 node(s)

Total Reserved Cloud

You're saving **€1/month** b

Total monthly cost
***Free of charge for the tr**

Cost details

Quotas & pricing

Environment name

env-2897066

Tomcat 7.0 **Java 7**

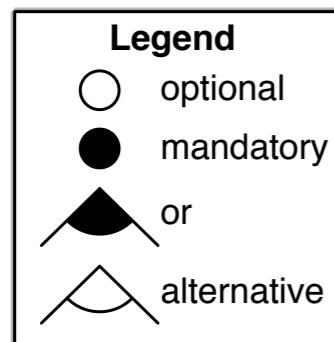
GlassFish 3.1
Jetty 6.1
TomEE+ 1.5
Tomcat 6.0
Tomcat 7.0

VDS SSL Build

High-availability Public IPv4

Detailed description: The screenshot shows the Jelastic interface for creating a Java application environment. It includes a legend for symbols: a circle for optional, a solid circle for mandatory, a triangle for 'or', and a mountain range for 'alternative'. The main area shows a 'Balancing' section with a Tomcat 7.0 icon, Java 7 dropdown, and a list of other Java containers like GlassFish 3.1 and Jetty 6.1. Below this are VDS, SSL, and Build buttons. To the right, 'Application Servers' are turned ON with vertical scaling set to 1 cloudlet (128 MB, 200 MHz, 0.0085 EUR per month, 15% discount). Horizontal scaling is set to 1 node. Other options like High-availability and Public IPv4 are off. On the far right, there's a sidebar with 'Resources' (cloudlets), 'Reserved cloudlets' (1, 15% discount), 'Total Reserved Cloud' (saving €1/month), 'Total monthly cost' (free), 'Cost details', 'Quotas & pricing', and an 'Environment name' field set to 'env-2897066'.

Use Case : Jelastic



The screenshot shows the Jelastic application builder interface. At the top, there are tabs for JAVA and PHP. Below the tabs, there's a diagram of an application stack:

- A blue box labeled "Balancing" contains a yellow cat icon.
- Three cylinders below it represent database layers: Cache, SQL, and NoSQL. The "SQL" cylinder is highlighted with a red border.
- At the bottom of the stack are three buttons: VDS, SSL, and Build.

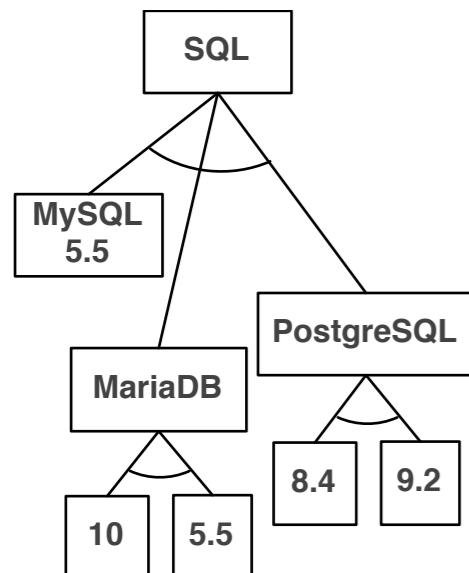
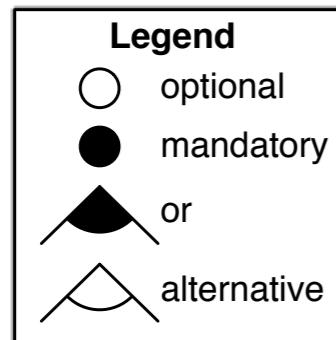
To the right of the stack, there are configuration sections:

- Application Servers**: A switch is set to **ON**. It shows "Vertical scaling per node" with 1 reserved cloudlet (128 MB, 200 MHz, 0.0085 EUR per..) and a 15% discount. A slider for horizontal scaling ranges from 0 to 64, currently at 0.
- Horizontal scaling**: Shows 1 node(s) with a plus/minus button to add more nodes. It includes a preview icon with a yellow cat and dropdown menus for **Tomcat 7.0** and **Java 7**.
- High-availability** and **Public IPv4** options are both turned off.

On the far right, there are sidebar sections:

- Resources (cloudlets)**: Shows 1 reserved cloudlet, a 15% discount, and a progress bar indicating savings.
- Total Reserved Cloud**: Shows monthly cost savings.
- Total monthly cost**: Shows the total monthly cost, noting it's free of charge.
- Cost details** and **Quotas & pricing** links.
- Environment name**: Set to **env-2897066**.

Use Case : Jelastic



JAVA **PHP**

Balancing

Application Servers **ON**

Vertical scaling per node

Reserved 1 cloudlet(s)
128 MB, 200 MHz
0.0085 EUR per..
Discount: 15%

Horizontal scaling

1 node(s)

PostgreSQL 8.4

MariaDB 10.0
MariaDB 5.5
MySQL 5.5
PostgreSQL 8.4
PostgreSQL 9.2

VDS

High-availability **OFF**

Public IPv4 **OFF**

Resources (cloudlet)

Reserved cloudlets
Now: 1
Discount: 15%

Total Reserved Cloud

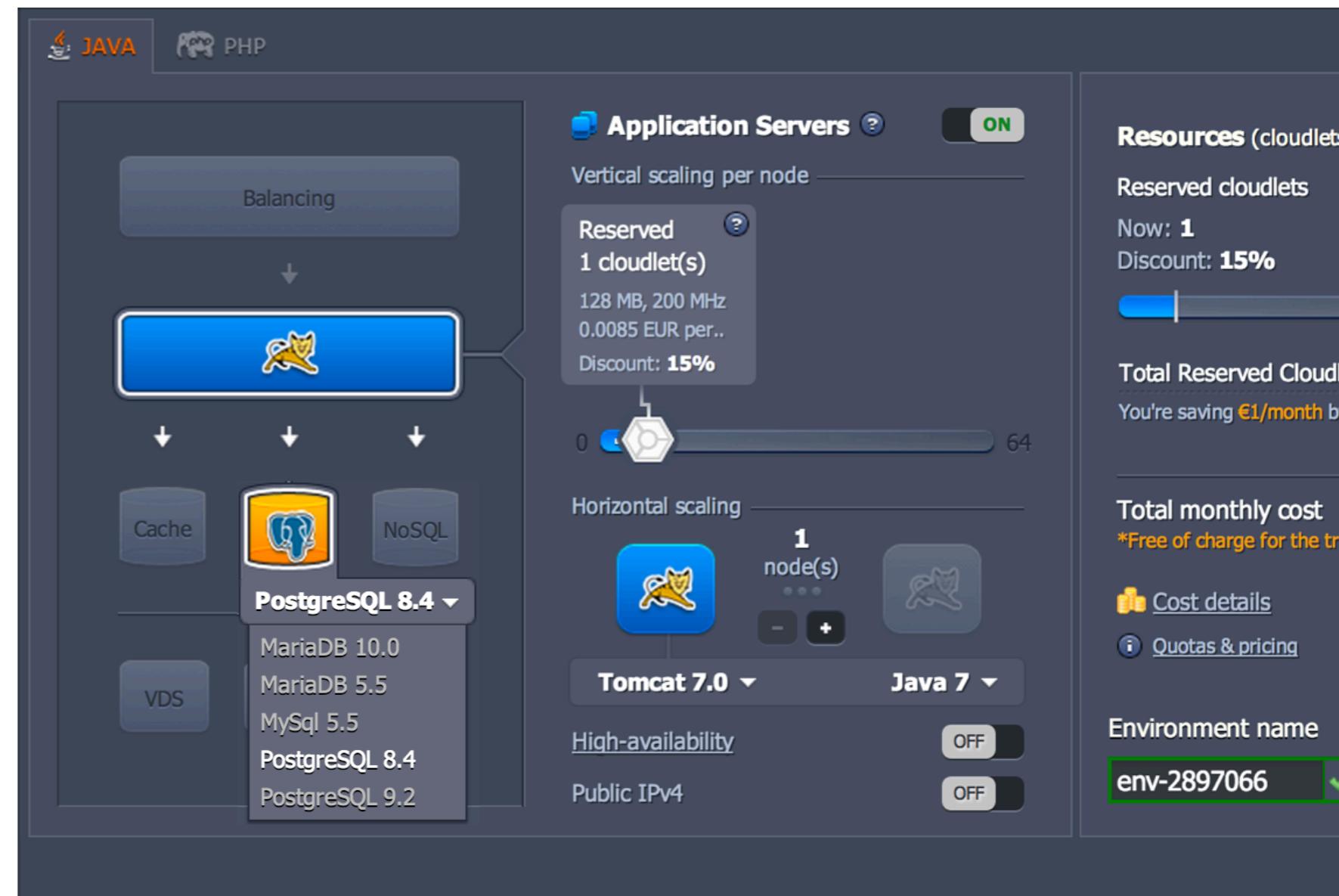
You're saving €1/month b

Total monthly cost
*Free of charge for the tr

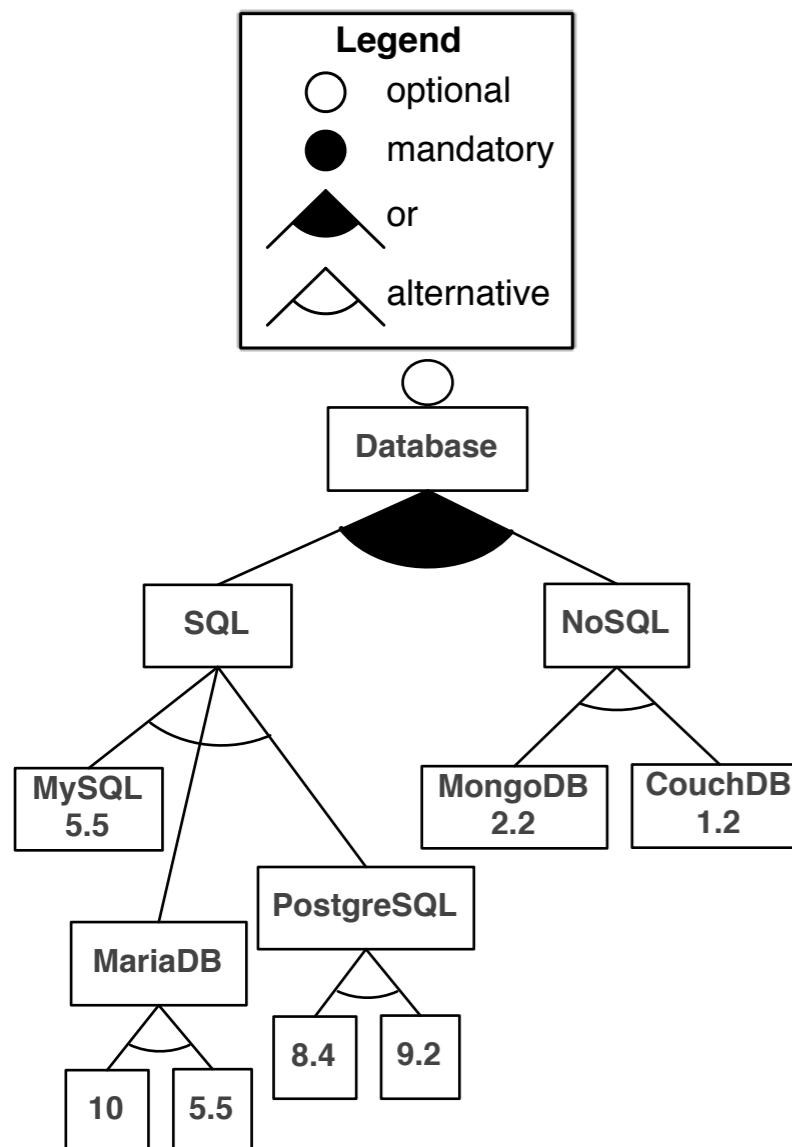
Cost details

Quotas & pricing

Environment name
env-2897066



Use Case : Jelastic



Jelastic Platform Interface

Application Servers **ON**

Vertical scaling per node

Reserved **1 cloudlet(s)**
128 MB, 200 MHz
0.0085 EUR per..
Discount: **15%**

Horizontal scaling

1 node(s)

MongoDB 2.2

CouchDB 1.2

MongoDB 2.2

VDS

Tomcat 7.0

Java 7

High-availability OFF

Public IPv4 OFF

Resources (cloudlet)

Reserved cloudlets
Now: **1**
Discount: **15%**

Total Reserved Cloud
You're saving **€1/month** b...

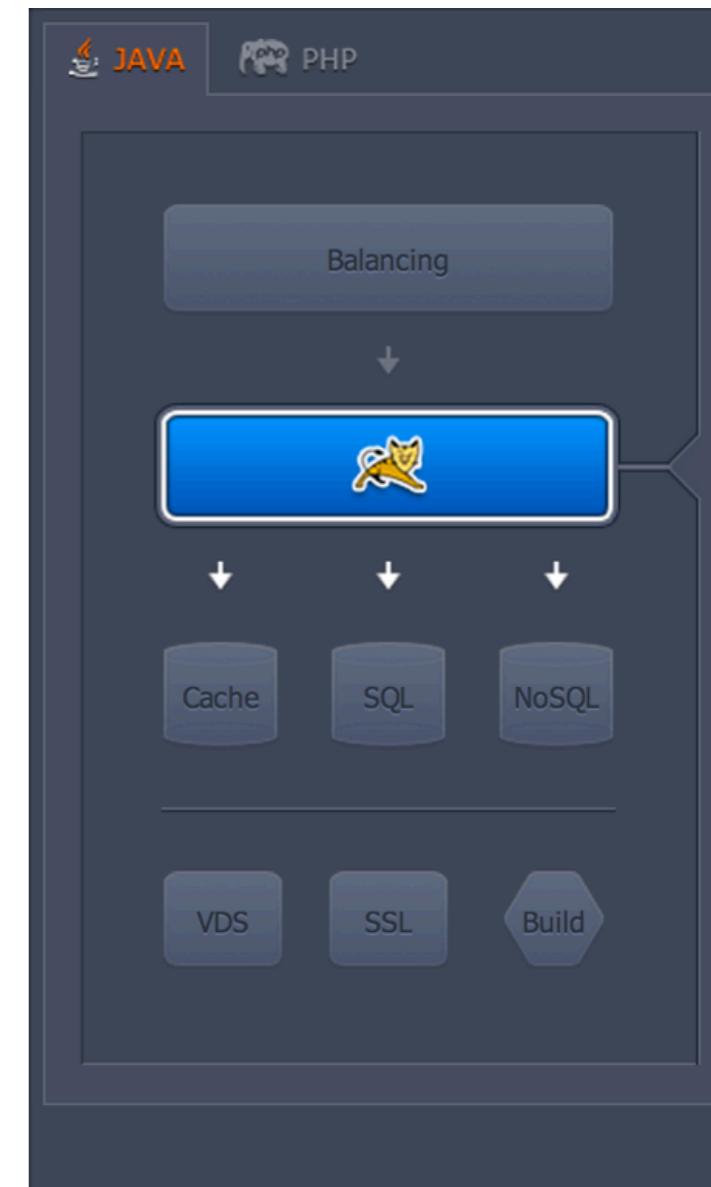
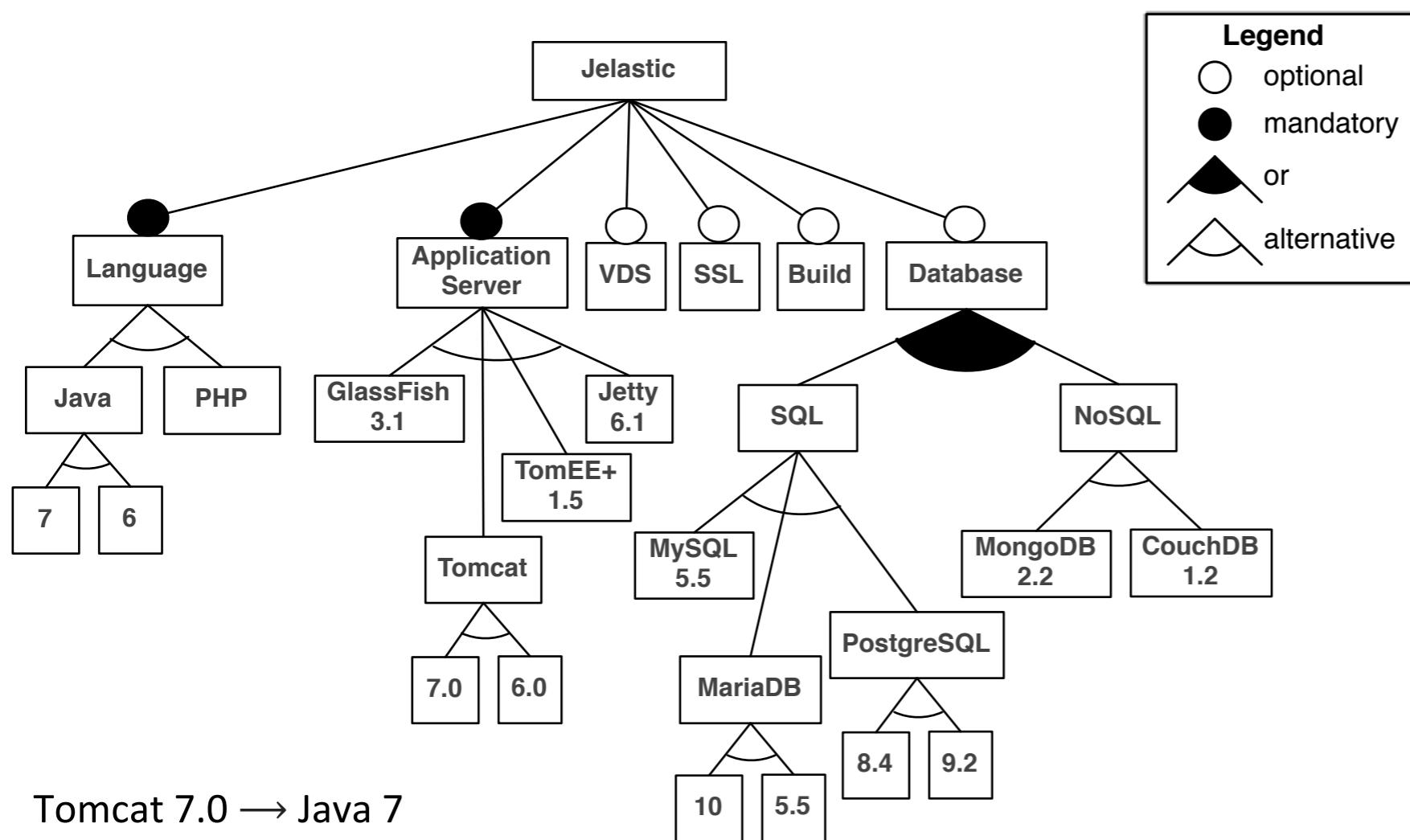
Total monthly cost
***Free of charge for the first month**

Cost details

Quotas & pricing

Environment name
env-2897066

Use Case : Jelastic



Use Case : Jelastic

The screenshot shows the Jelastic interface for deploying a Java application. On the left, a sidebar lists Java and PHP options, with Java selected. The main area displays a stack diagram with a central application server node (Tomcat 7.0) connected to a database node (MySQL) and a cache node (Redis). Below the stack are options for VDS, SSL, and Build.

Application Servers (ON)

Vertical scaling per node

Reserved 1 cloudlet(s)
128 MB, 200 MHz
0.0085 EUR per..
Discount: 15%

Horizontal scaling

0 to 64 nodes

Total monthly cost €6*

*Free of charge for the trial period.

Resources (cloudlets)

Reserved cloudlets

Now: 1
Discount: 15%
Next: 8
Discount: 25%

Total Reserved Cloudlets €6 per month

You're saving €1/month by using Reserved Cloudlets

Cost details

Quotas & pricing

Environment name

env-2897066 .jelastic.dogado.eu

Create

Use Case : Jelastic

The screenshot shows the Jelastic application deployment interface. It has two tabs at the top: **JAVA** (selected) and **PHP**. The main area displays a flowchart of the application architecture:

- Balancing**: A central node connected to three database nodes.
- Cache**, **SQL**, and **NoSQL**: Three separate database nodes.
- VDS**, **SSL**, and **Build**: External services or build-related components.

Application Servers (ON):

- Vertical scaling per node**:
 - Reserved**: 1 cloudlet(s)
128 MB, 200 MHz
0.0085 EUR per..
Discount: 15%
 - A slider for memory from 0 to 64 MB.
- Horizontal scaling**:
 - Two instances of the application server icon, one highlighted with a red box.
 - Buttons for **-** and **+** to manage the number of nodes.
 - Tomcat 7.0** and **Java 7** dropdown menus.
- High-availability** and **Public IPv4** options with OFF buttons.

Resources (cloudlets):

- Reserved cloudlets**: Now: 1, Next: 8. Discount: 15% (green), 25% (green).
- Total Reserved Cloudlets**: €6 per month.
- You're saving** €1/month **by using Reserved Cloudlets**.
- Total monthly cost**: €6* (green). *Free of charge for the trial period.
- Cost details** and **Quotas & pricing** links.

Environment name: env-2897066 .jelastic.dogado.eu

Buttons: **Cancel** and **Create** (with a checkmark icon).

Use Case : Jelastic

The screenshot shows the Jelastic interface for creating a new environment. On the left, a sidebar lists available services: Java (selected), PHP, and MySQL. The main area displays a stack configuration:

- Application Servers:** Enabled (ON). Configuration includes:
 - Vertical scaling per node: Reserved 1 cloudlet(s) (128 MB, 200 MHz, 0.0085 EUR per.., Discount: 15%).
 - Horizontal scaling: 2 node(s).
 - Software stack: Tomcat 7.0 and Java 7.
 - Additional settings: High-availability (OFF), Public IPv4 (OFF).
- Resources (cloudlets):** Reserved cloudlets: Now: 3, Next: 8. Discounts: 15% (current), 25% (next). Total Reserved Cloudlets: €18 per month. Savings: You're saving €3/month by using Reserved Cloudlets.
- Total monthly cost:** €18* → €109*. Note: *Free of charge for the trial period.
- Environment name:** env-2897066.jelastic.dogado.eu (with a checked checkbox).

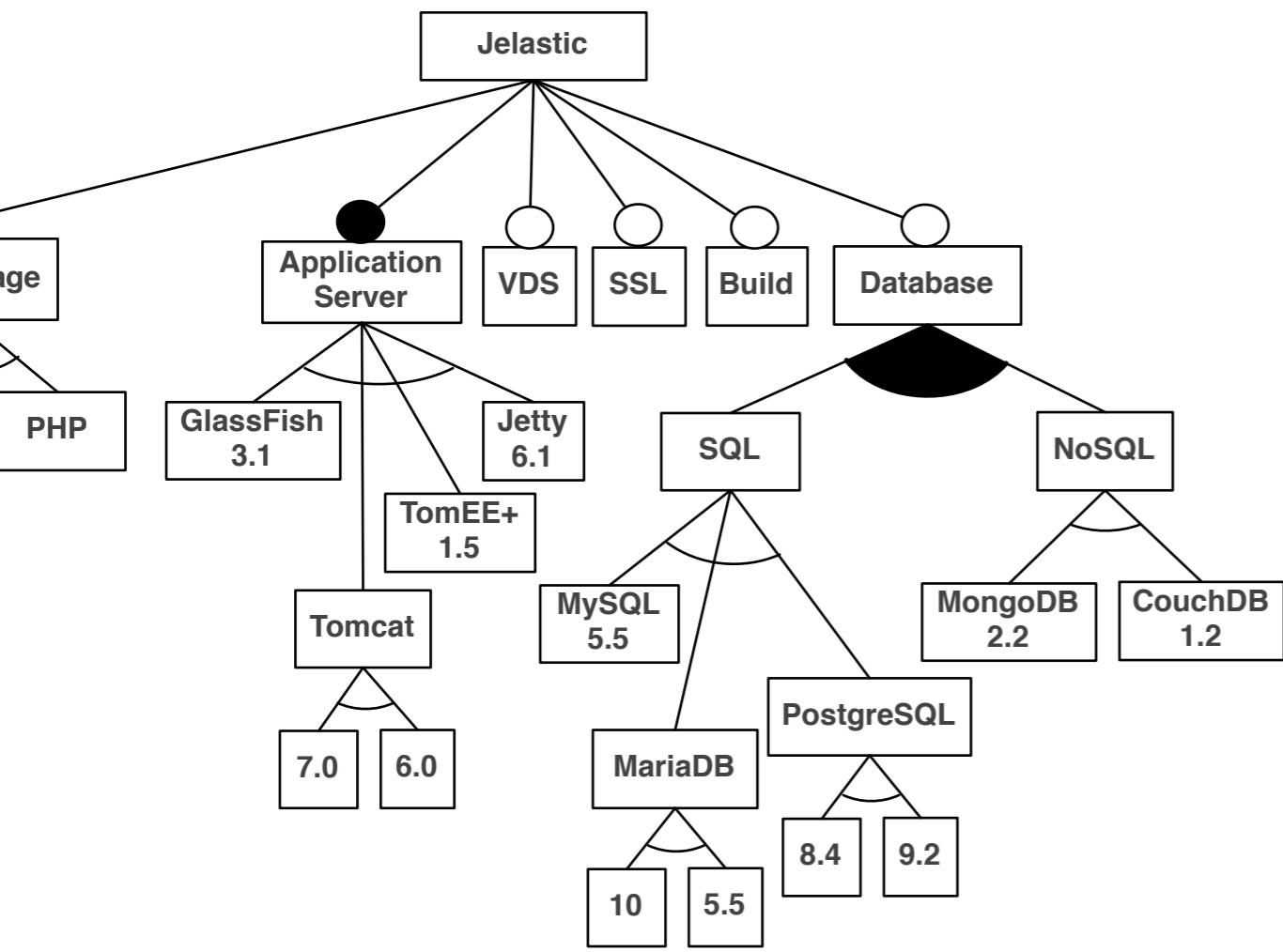
At the bottom right are "Cancel" and "Create" buttons.

Use Case : Jelastic

Load balancer

The screenshot shows the Jelastic application deployment interface. On the left, a stack diagram illustrates the application architecture: Nginx (green box) sits above Tomcat (blue box), which in turn sits above three databases (Cache, SQL, NoSQL). Below the stack are deployment options: VDS, SSL, and Build. On the right, the 'Application Servers' section is active (ON). It shows vertical scaling settings: Reserved 1 cloudlet(s) at 128 MB, 200 MHz, 0.0085 EUR per.., with a 15% discount. Horizontal scaling is set to 2 node(s). Application details include Tomcat 7.0 and Java 7. Under 'Resources (cloudlets)', it shows Reserved cloudlets: Now: 3, Next: 8, Discount: 15%, and Total Reserved Cloudlets at €18 per month, saving €3/month. The total monthly cost is listed as €18* → €109*. Below this are links for Cost details and Quotas & pricing. The environment name is set to env-2897066.jelastic.dogado.eu. At the bottom are 'Cancel' and 'Create' buttons.

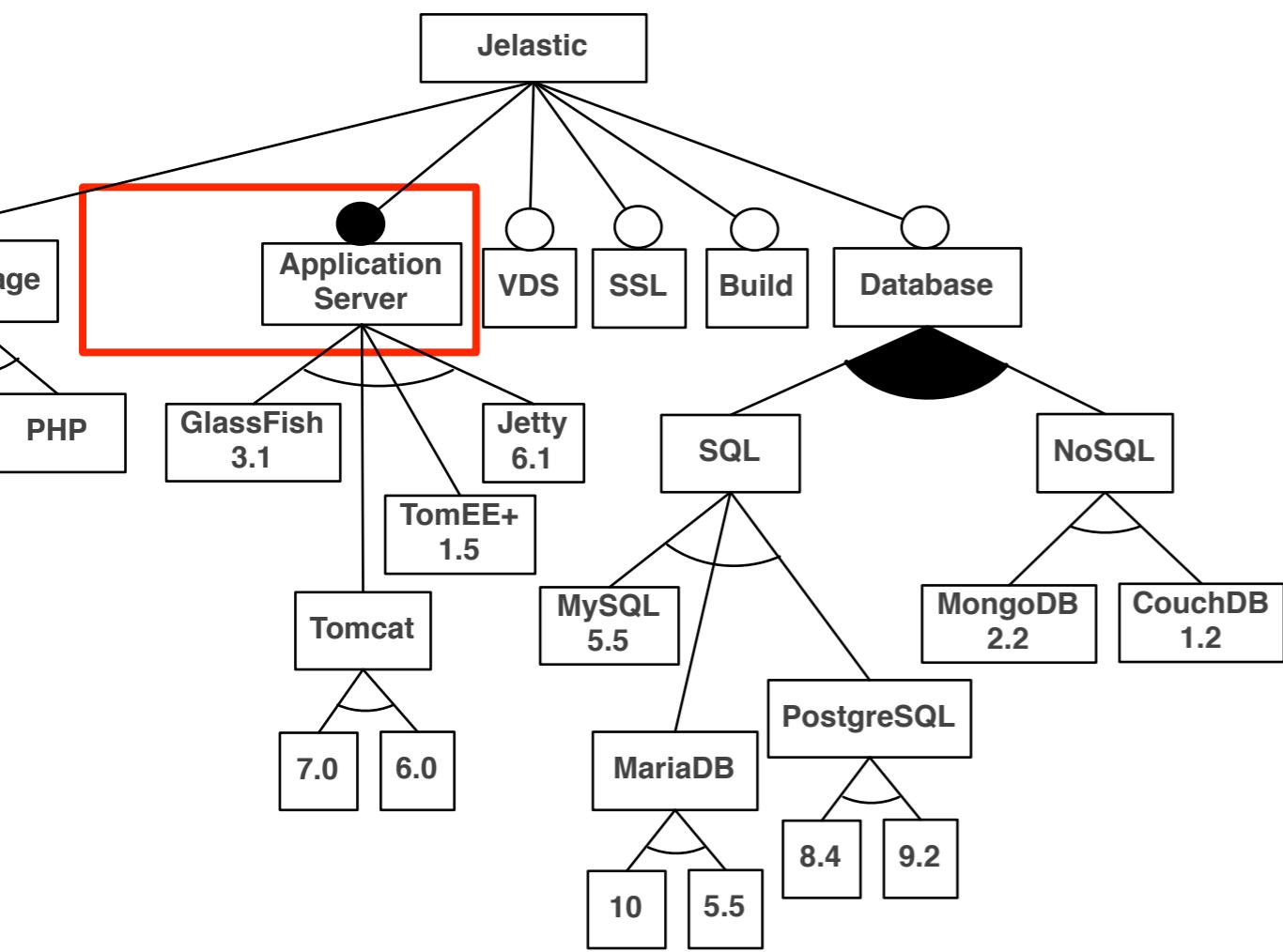
Use Case : Jelastic



Tomcat 7.0 → Java 7

The screenshot shows the Jelastic Cloud Platform interface for configuring a Java application server. The top navigation bar has tabs for JAVA and PHP, with JAVA selected. On the left, there's a sidebar with "Application Servers" and a switch for "Vertical scaling per node". The main area shows a configuration for a Tomcat 7.0 instance. It includes sections for "Vertical scaling per node" (1 cloudlet(s), 128 MB, 200 MHz, 0.0085 EUR per.., Discount: 15%), "Horizontal scaling" (2 node(s)), and "High-availability" (OFF). The configuration includes options for "VDS", "SSL", and "Build". At the bottom, there are buttons for "Tomcat 7.0" and "Java 7".

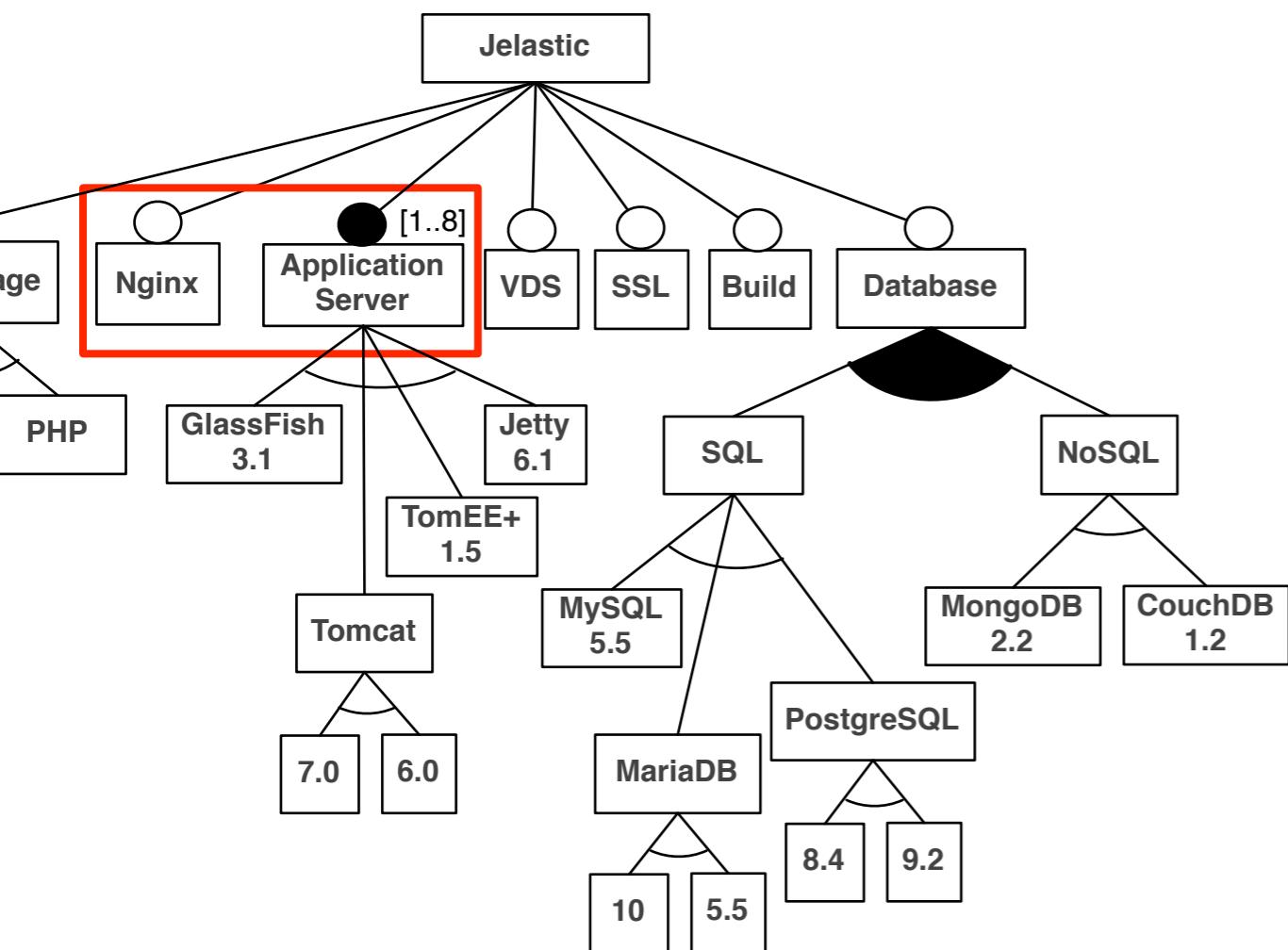
Use Case : Jelastic



Tomcat 7.0 → Java 7

The screenshot shows the Jelastic Cloud Platform interface. On the left, under the "JAVA" tab, a configuration panel for an application server is displayed. It includes sections for "Application Servers", "Vertical scaling per node", and "Horizontal scaling". The "Application Servers" section shows a configuration for "Tomcat 7.0" with "2 node(s)" and a "Java 7" dependency. Below this, there are options for "High-availability" and "Public IPv4". On the right, a detailed view of the Tomcat 7.0 configuration is shown, featuring a green "Nginx" button at the top, followed by a blue "Tomcat" icon, and three smaller icons for "Cache", "SQL", and "NoSQL". At the bottom, there are buttons for "VDS", "SSL", and "Build".

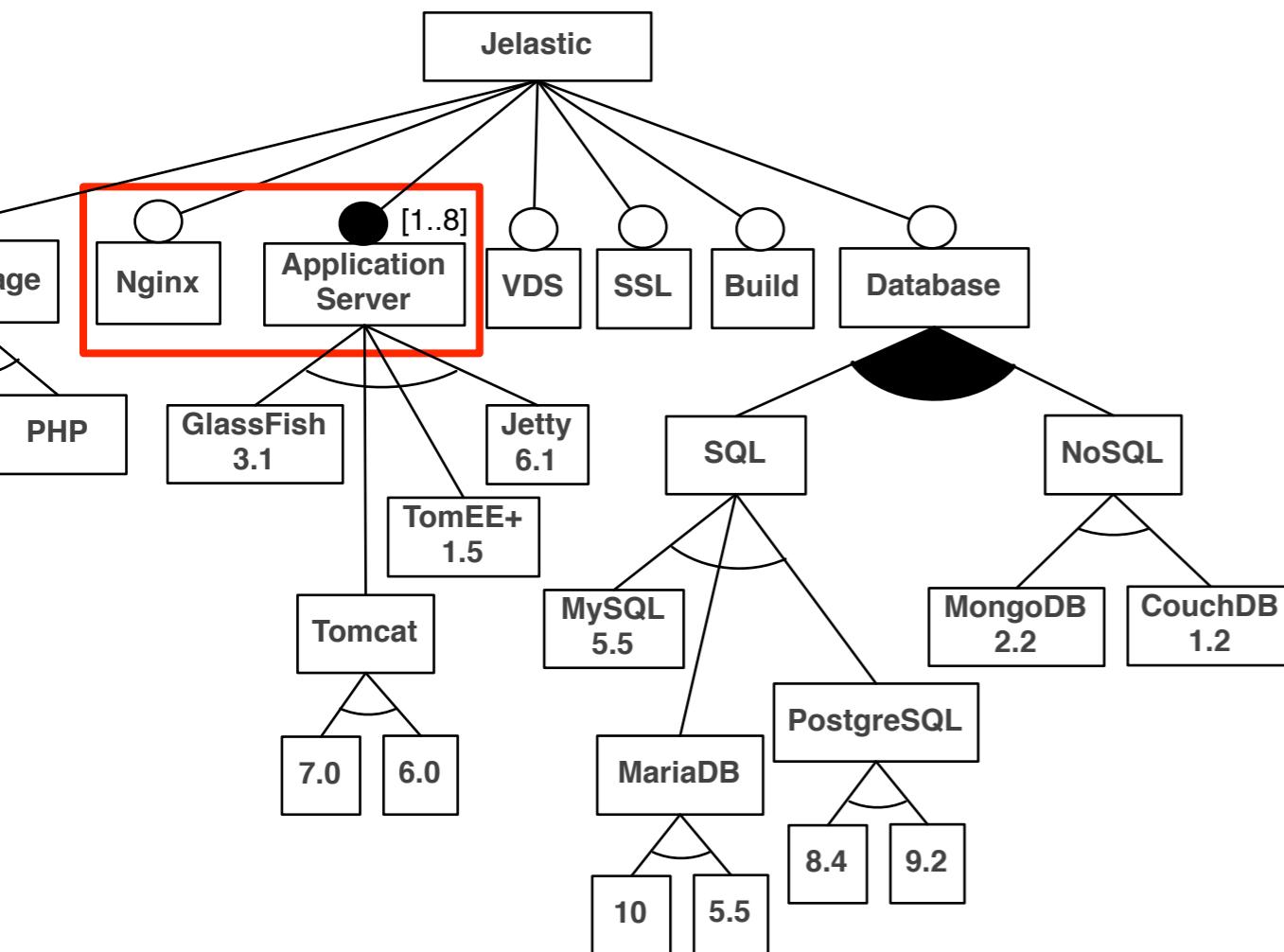
Use Case : Jelastic



Tomcat 7.0 → Java 7

The screenshot shows the Jelastic Cloud Platform interface. The top navigation bar includes tabs for **JAVA** and **PHP**. The main area displays a configuration for a Java application. It includes sections for **Application Servers** (vertical scaling per node), **Vertical scaling per node** (1 cloudlet(s), 128 MB, 200 MHz, 0.0085 EUR per.., Discount: 15%), and **Horizontal scaling** (2 node(s)). The application server configuration shows **Tomcat 7.0** selected. Other options like **Java 7** and **High-availability** are also visible.

Use Case : Jelastic



Tomcat 7.0 → Java 7

Application Server >= 2 → Nginx

The screenshot shows the Jelastic cloud interface for managing an application. The 'JAVA' tab is selected. The application structure is shown as a tree:

- Nginx** (highlighted in green)
- Cat icon** (highlighted in blue)
- Cache**, **SQL**, **NoSQL** (under the cat icon)
- VDS**, **SSL**, **Build** (under the Nginx node)

Application Servers section (Vertical scaling per node):

- Reserved: 1 cloudlet(s), 128 MB, 200 MHz, 0.0085 EUR per.., Discount: 15%
- Horizontal scaling: 2 node(s) (with a minus and plus button)

High-availability and **Public IPv4** options are also visible.

Use Case : Jelastic

The screenshot shows the Jelastic interface for creating a new application environment. The left sidebar has tabs for JAVA (selected) and PHP. The main area displays a flowchart of the application architecture: **Balancing** (represented by a yellow cat icon) connects to **Vertical scaling per node** (represented by a yellow cat icon), which then connects to **Horizontal scaling** (represented by a yellow cat icon). **Vertical scaling per node** includes options for **Reserved 1 cloudlet(s)** (128 MB, 200 MHz, 0.0085 EUR per.., Discount: 15%) and **Horizontal scaling** includes **Tomcat 7.0** and **Java 7**. Below these are **High-availability** and **Public IPv4** settings. The right side shows **Resources (cloudlets)** with **Reserved cloudlets** (Now: 1, Next: 8, Discount: 15%, Discount: 25%), **Total Reserved Cloudlets** (€6 per month), and a note about saving €1/month. It also shows **Total monthly cost** (€6*) and links for **Cost details** and **Quotas & pricing**. The **Environment name** field contains "env-2897066 .jelastic.dogado.eu". At the bottom are **Cancel** and **Create** buttons.

JAVA **PHP** **X**

Application Servers **ON**

Vertical scaling per node

Reserved 1 cloudlet(s)
128 MB, 200 MHz
0.0085 EUR per..
Discount: 15%

Horizontal scaling

Tomcat 7.0 **Java 7**

High-availability **OFF**

Public IPv4 **OFF**

Resources (cloudlets)

Reserved cloudlets

Now: 1 Next: 8
Discount: 15% Discount: 25%

Total Reserved Cloudlets **€6 per month**

You're saving **€1/month** by using Reserved Cloudlets

Total monthly cost **€6***
*Free of charge for the trial period.

Cost details **Quotas & pricing**

Environment name

env-2897066 .jelastic.dogado.eu

Create

Use Case : Jelastic

The screenshot shows the Jelastic interface for creating a new application environment. The left sidebar has tabs for JAVA (selected) and PHP. The main area shows a flowchart: Balancing → Application Server (represented by a cat icon) → Cache, SQL, NoSQL databases. Below this are VDS, SSL, and Build options. On the right, the 'Application Servers' section is active, showing 'Vertical scaling per node' (1 cloudlet reserved, 15% discount) and 'Horizontal scaling' (1 node selected, highlighted with a red box). Configuration options include Tomcat 7.0, Java 7, High-availability, and Public IPv4. To the right, a 'Resources (cloudlets)' panel is highlighted with a red box, showing 'Reserved cloudlets' (Now: 1, Next: 8, Discount: 15%, 25%), 'Total Reserved Cloudlets' (€6 per month), and a note about saving €1/month. It also shows 'Total monthly cost' (€6*) and a note about being free of charge for the trial period. At the bottom, there are 'Cost details' and 'Quotas & pricing' links, an 'Environment name' field containing 'env-2897066 .jelastic.dogado.eu', and 'Cancel' and 'Create' buttons.

Application Servers ON

Vertical scaling per node

Reserved 1 cloudlet(s)
128 MB, 200 MHz
0.0085 EUR per..
Discount: 15%

Horizontal scaling

1 node(s)

Tomcat 7.0 **Java 7**

High-availability OFF

Public IPv4 OFF

Resources (cloudlets)

Reserved cloudlets

Now: 1 Next: 8
Discount: 15% Discount: 25%

Total Reserved Cloudlets €6 per month

You're saving €1/month by using Reserved Cloudlets

Total monthly cost price €6*

*Free of charge for the trial period.

Cost details

Quotas & pricing

Environment name

env-2897066 .jelastic.dogado.eu

Cancel Create

Use Case : Jelastic

The screenshot shows the Jelastic application creation interface for a Java application. The left sidebar has tabs for JAVA (selected) and PHP. The main area shows a stack configuration with NGINX at the top, followed by a Tomcat node, and then Cache, SQL, and NoSQL databases. At the bottom are VDS, SSL, and Build options. On the right, the 'Application Servers' section is set to 'ON'. It includes 'Vertical scaling per node' (Reserved 1 cloudlet(s), 128 MB, 200 MHz, 0.0085 EUR per.., Discount: 15%) and 'Horizontal scaling' (2 node(s)). Below these are 'High-availability' and 'Public IPv4' options. A red box highlights the 'Resources (cloudlets)' section, which shows 'Reserved cloudlets' (Now: 3, Next: 8, Discount: 15%, Discount: 25%), 'Total Reserved Cloudlets' (€18 per month), and a note about saving €3/month. It also shows 'Total monthly cost' (€18* → €109*) and a note about the trial period being free of charge. Other sections include 'Cost details' and 'Quotas & pricing'. The environment name is set to 'env-2897066.jelastic.dogado.eu'. Buttons at the bottom are 'Cancel' and 'Create'.

Resources (cloudlets)

Reserved cloudlets

Now: 3 Next: 8
Discount: 15% Discount: 25%

Total Reserved Cloudlets €18 per month
You're saving €3/month by using Reserved Cloudlets

Total monthly cost €18* → €109*

*Free of charge for the trial period.

Environment name

env-2897066.jelastic.dogado.eu

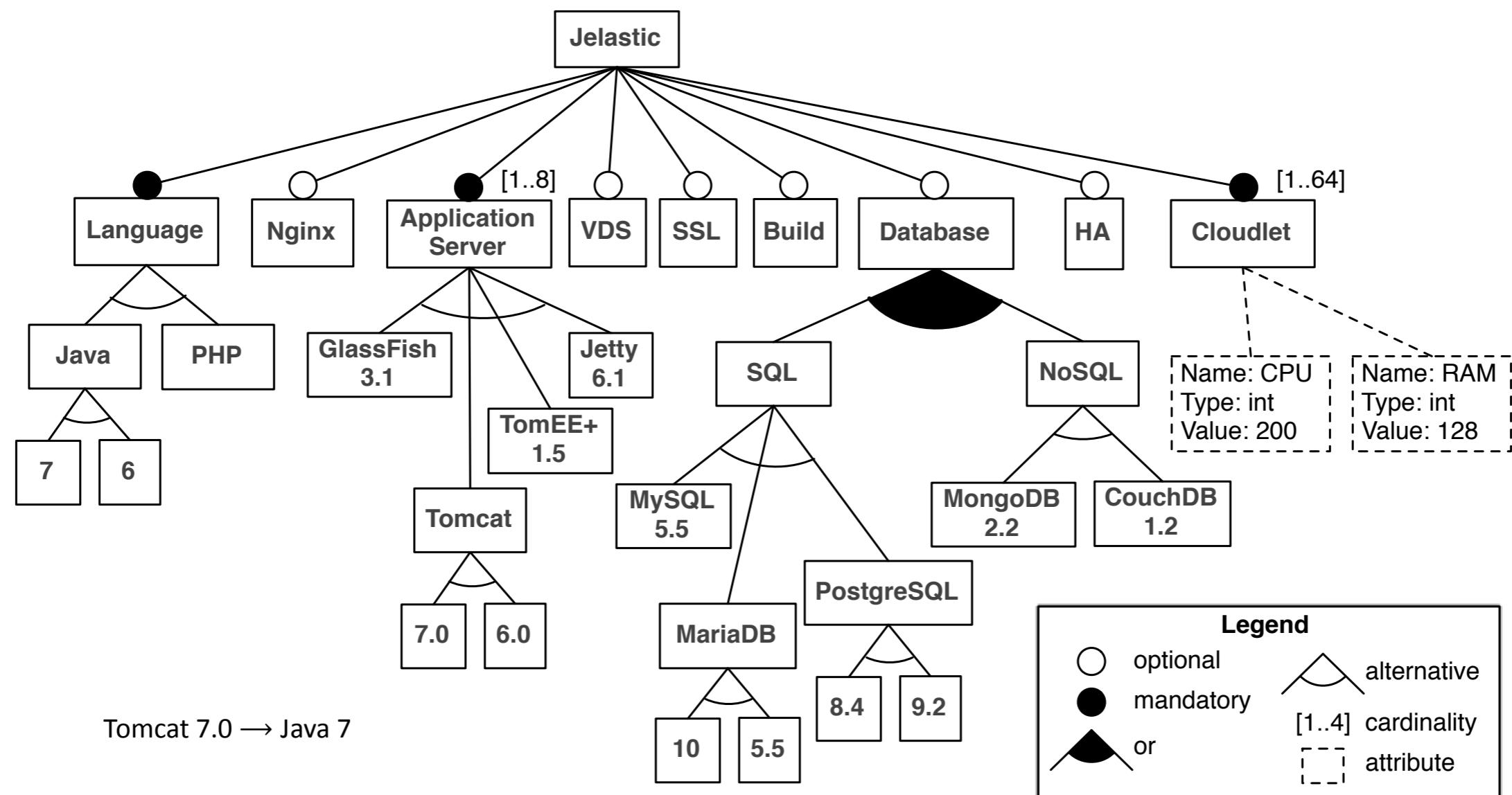
Cancel Create

Use Case : Jelastic

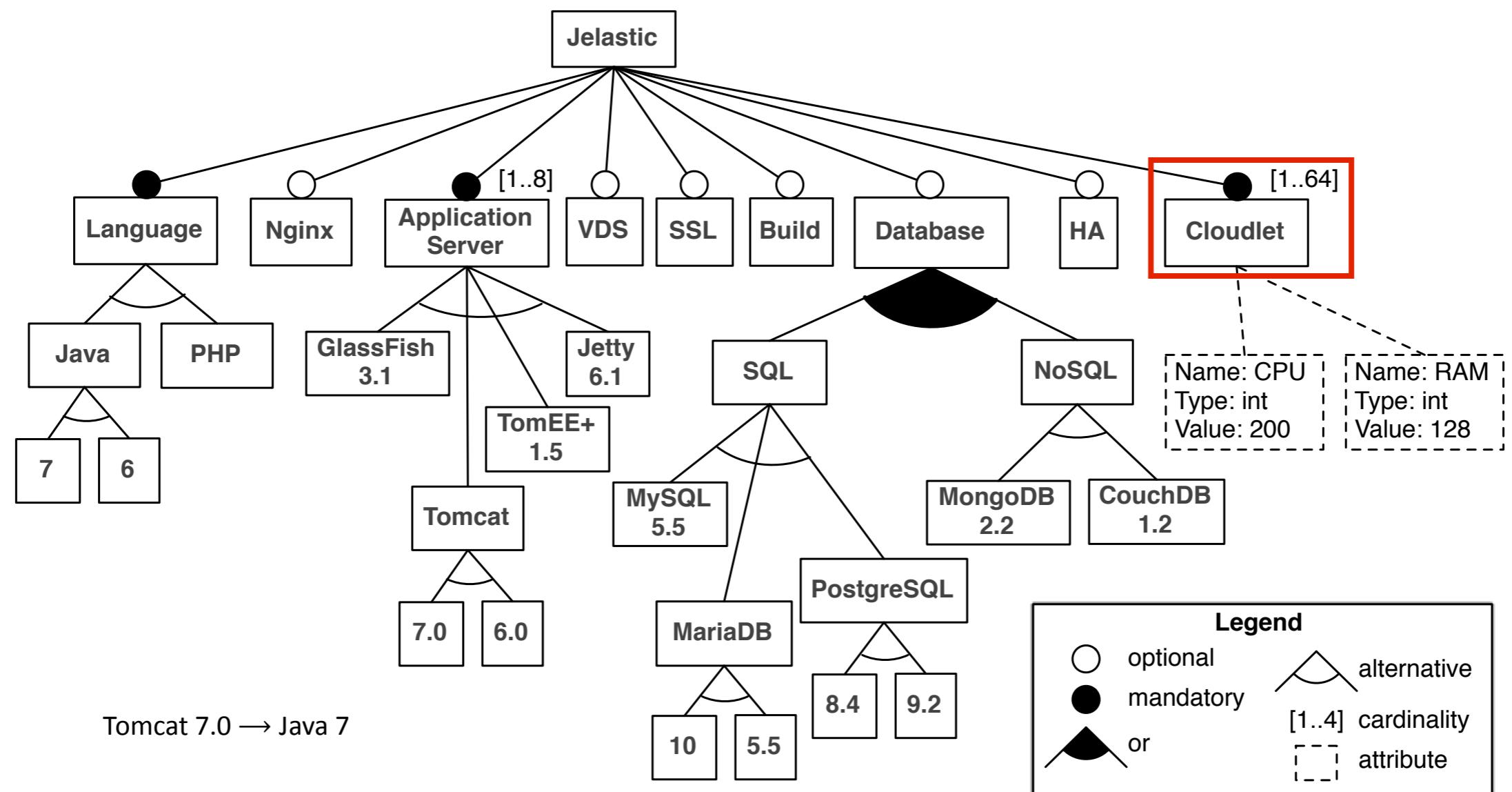
The screenshot shows the Jelastic interface for deploying a Java application. The left sidebar has tabs for JAVA (selected) and PHP. The main area shows a stack configuration with NGINX at the top, followed by a Tomcat node, and then Cache, SQL, and NoSQL databases. At the bottom are VDS, SSL, and Build options. On the right, the 'Application Servers' section is set to 'ON'. It includes 'Vertical scaling per node' (Reserved 1 cloudlet(s), 128 MB, 200 MHz, 0.0085 EUR per.., Discount: 15%) and 'Horizontal scaling' (2 node(s)). Below these are 'Tomcat 7.0' and 'Java 7' dropdowns, and 'High-availability' and 'Public IPv4' checkboxes. A red box highlights the 'Resources (cloudlets)' section, which shows 'Reserved cloudlets' (Now: 3, Next: 8, Discount: 15%, Discount: 25%), 'Total Reserved Cloudlets' (€18 per month), and a note about saving €3/month. It also shows the 'Total monthly cost' (€18* → €109*) and a note about the trial period being free of charge. The environment name is set to 'env-2897066.jelastic.dogado.eu'. Buttons at the bottom are 'Cancel' and 'Create'.

for each (Tomcat) → 1 Cloudlet

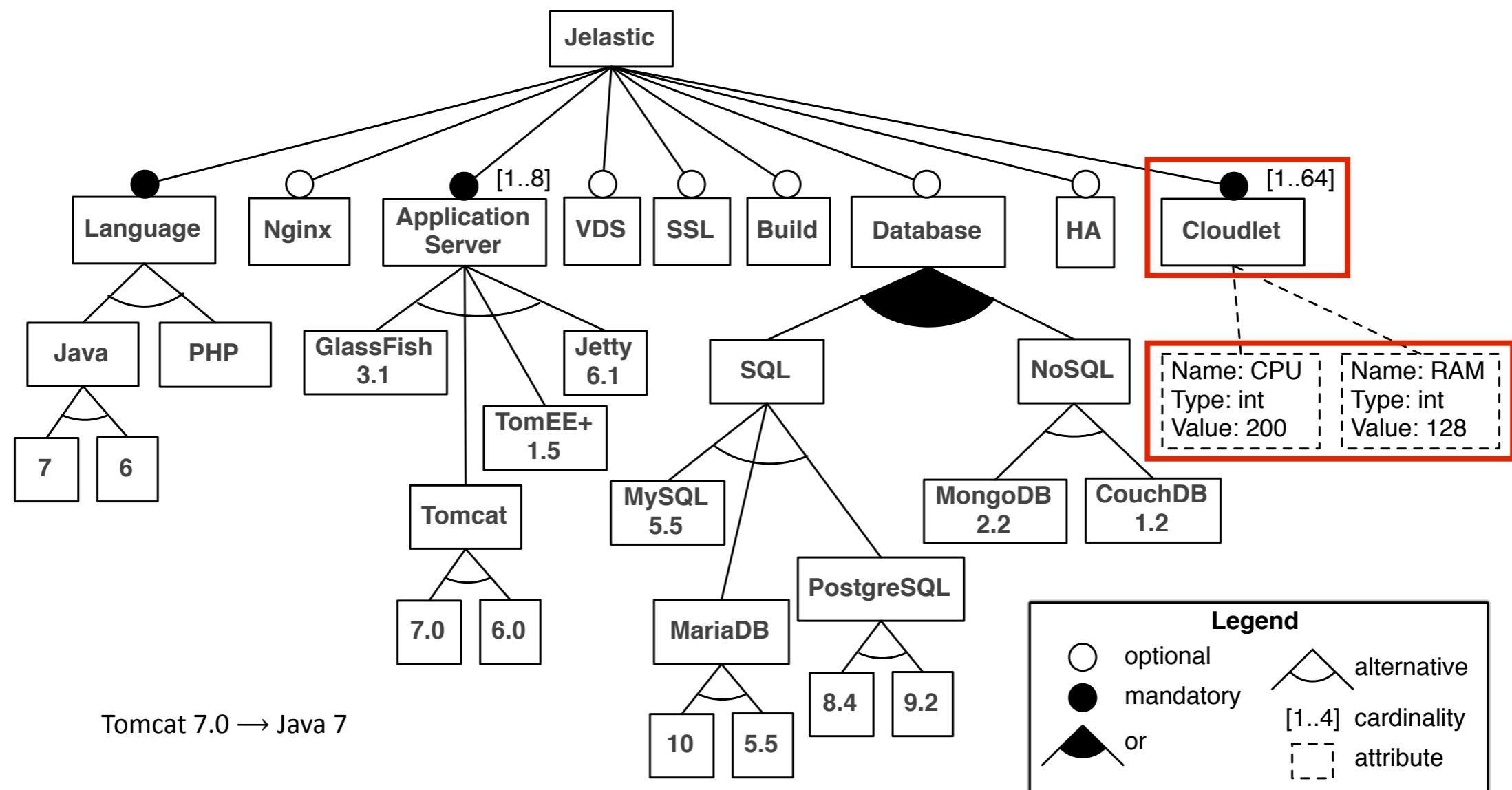
Use Case : Jelastic



Use Case : Jelastic



Use Case : Jelastic



Configuration Constraints

- Boolean constraints

Tomcat 7 → Java 7

Configuration Constraints

- Boolean constraints

Tomcat 7 → Java 7

- Constraints on feature instances

ApplicationServer >= 2 → Nginx

Tomcat → 1 Cloudlet

Configuration Constraints

- Boolean constraints

Tomcat 7 → Java 7

- Constraints on feature instances

ApplicationServer >= 2 → Nginx

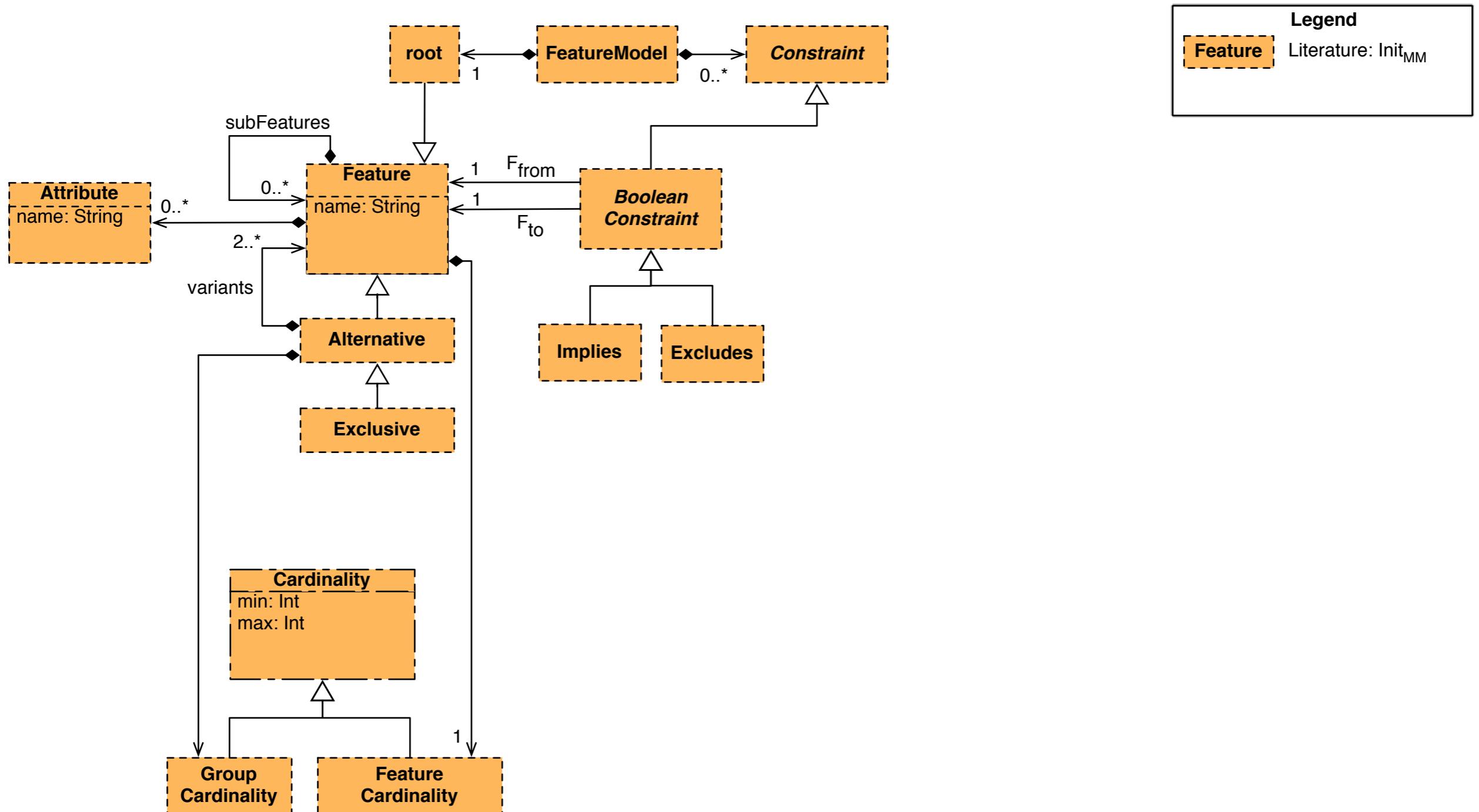
Tomcat → 1 Cloudlet

- Constraints on attribute values

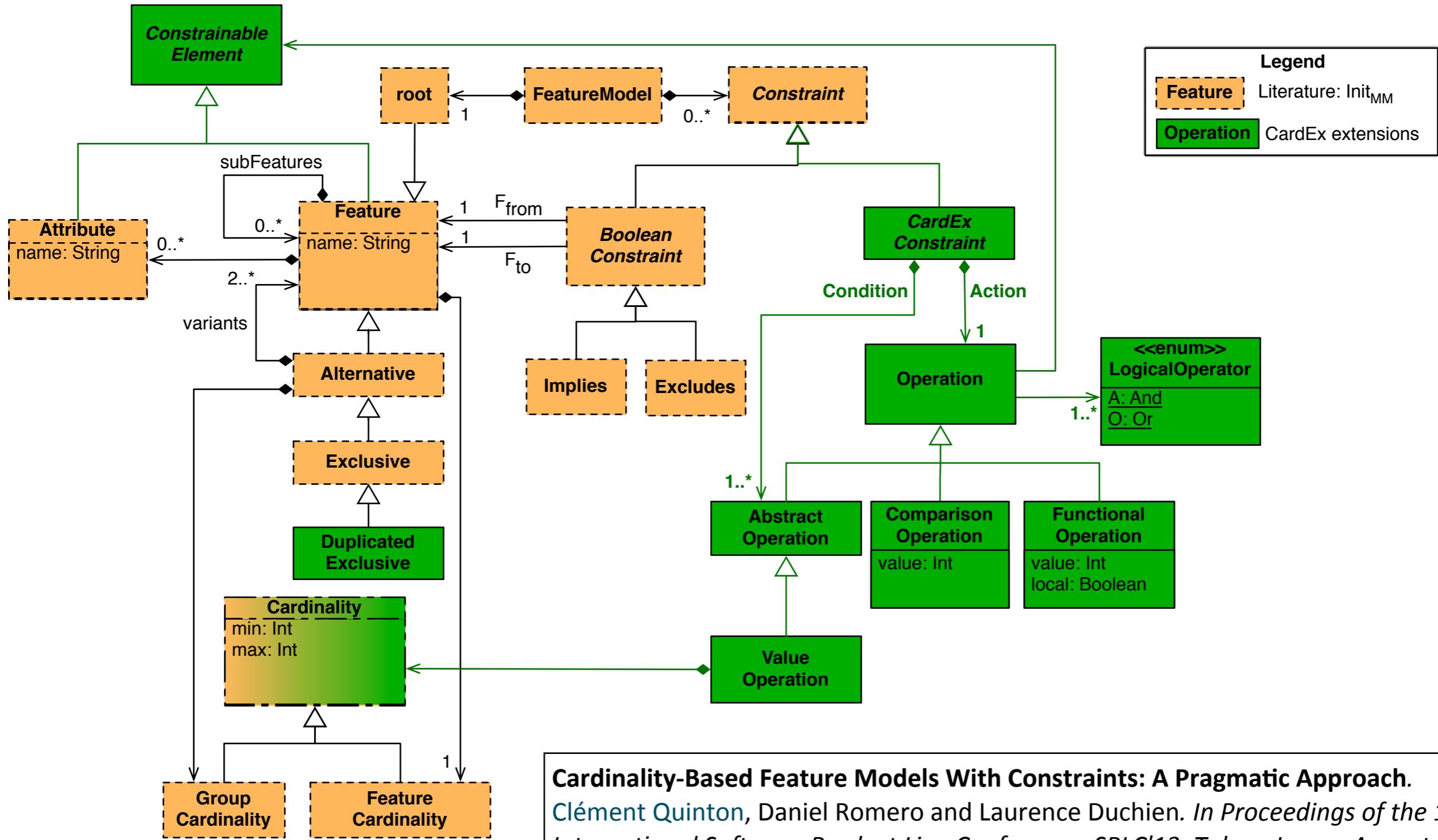
Server.size = S → CPU Core.size = 500

[1,2] RAM.size → Disk.size = 10

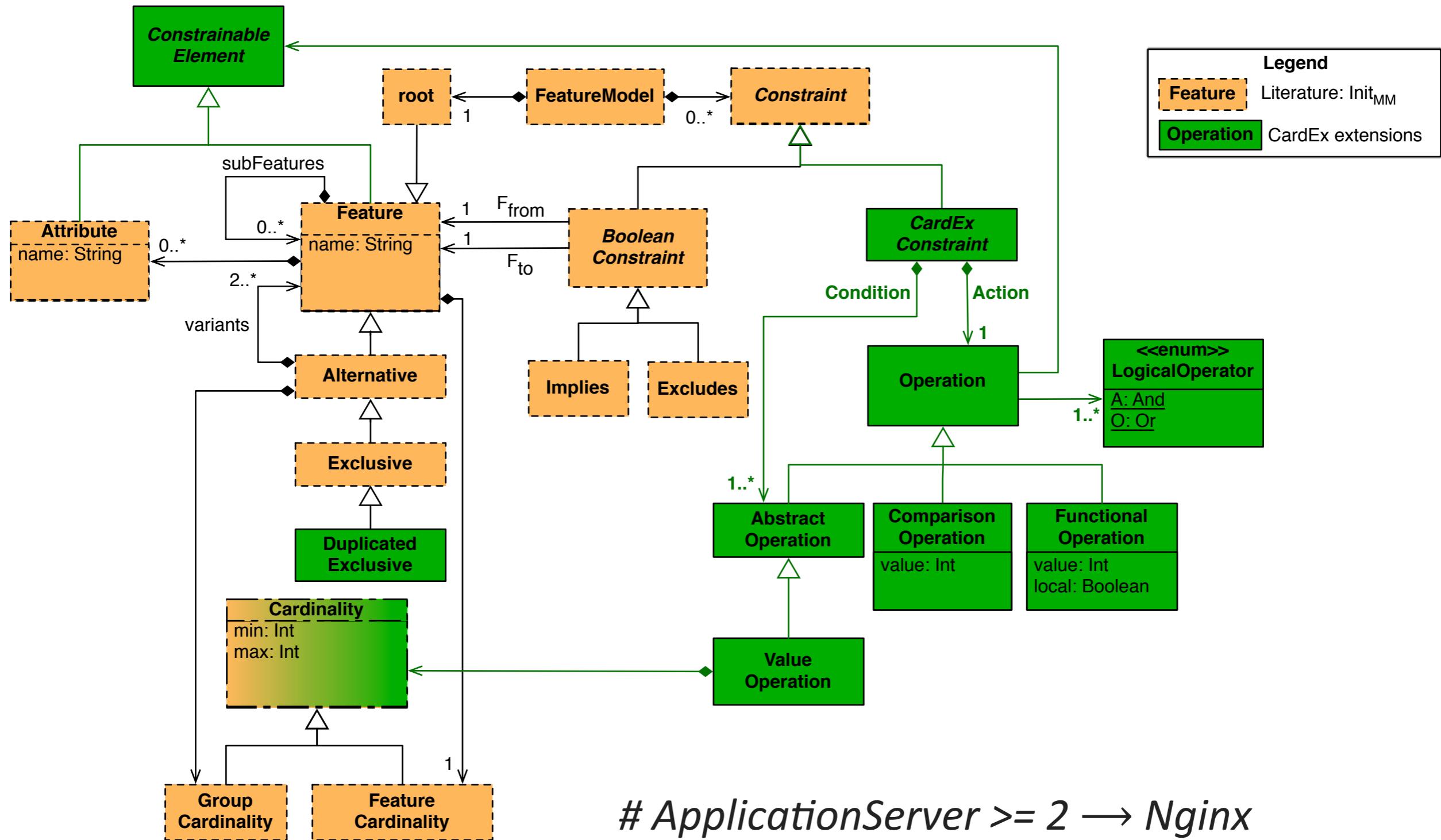
Feature Metamodel



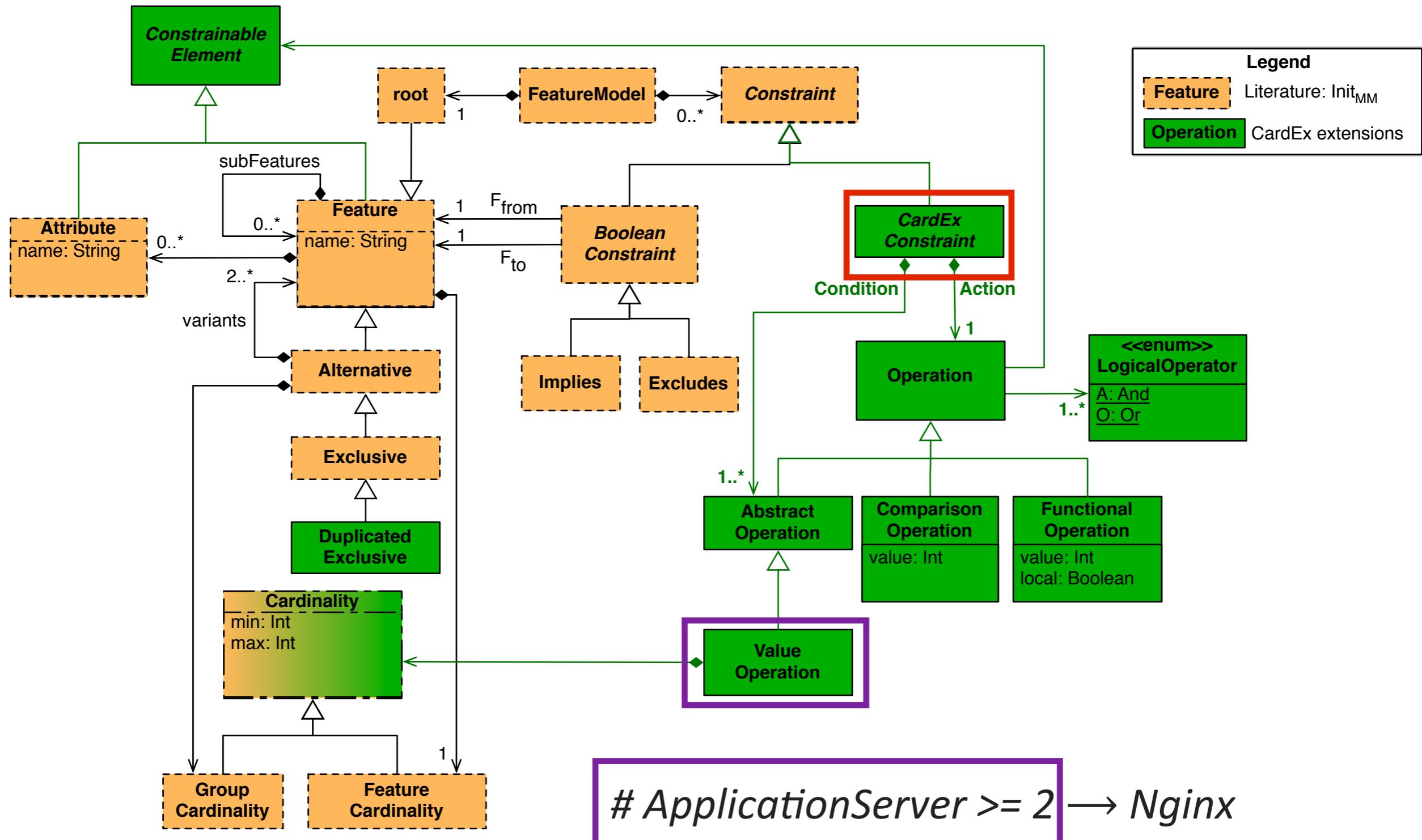
Feature Metamodel



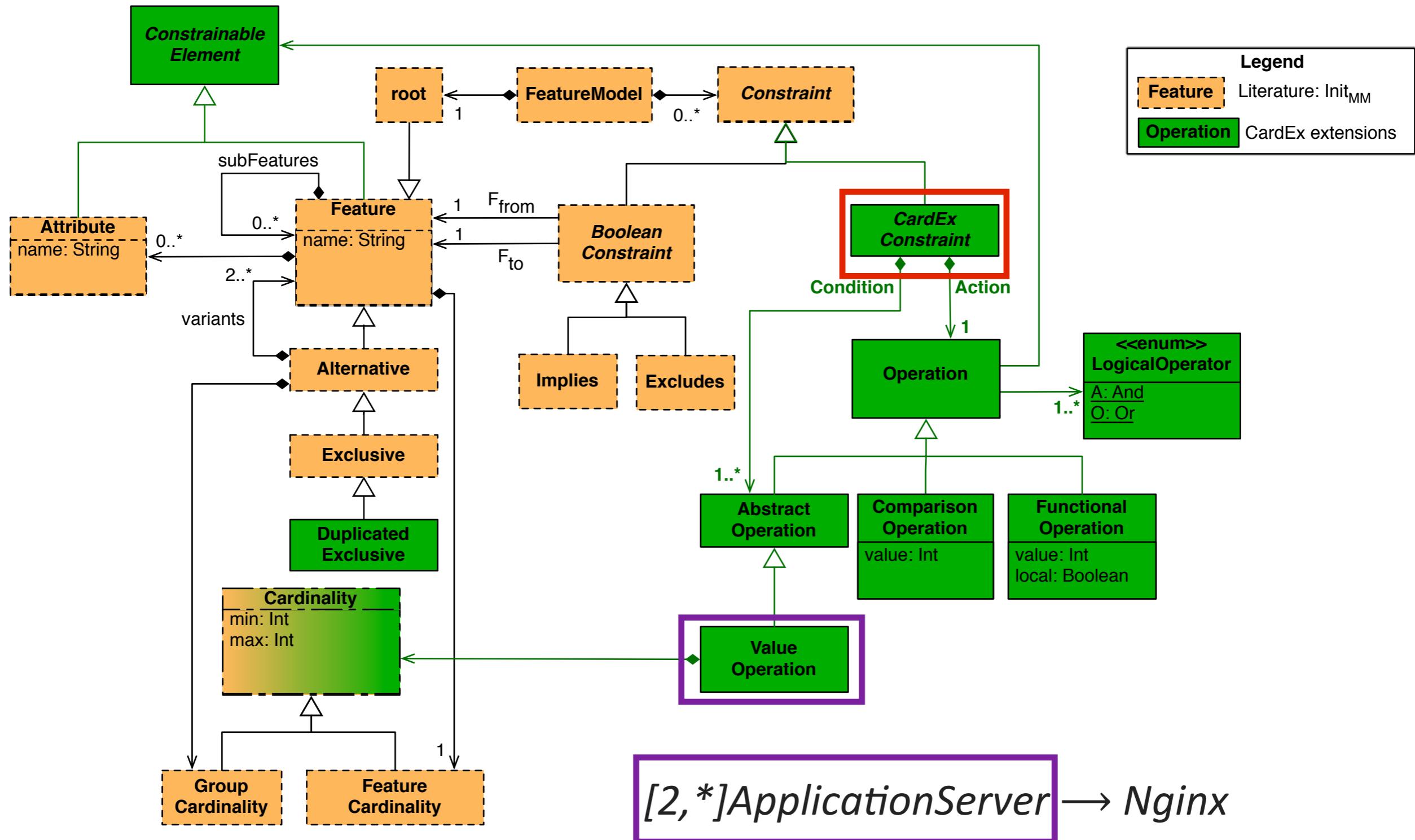
Feature Metamodel



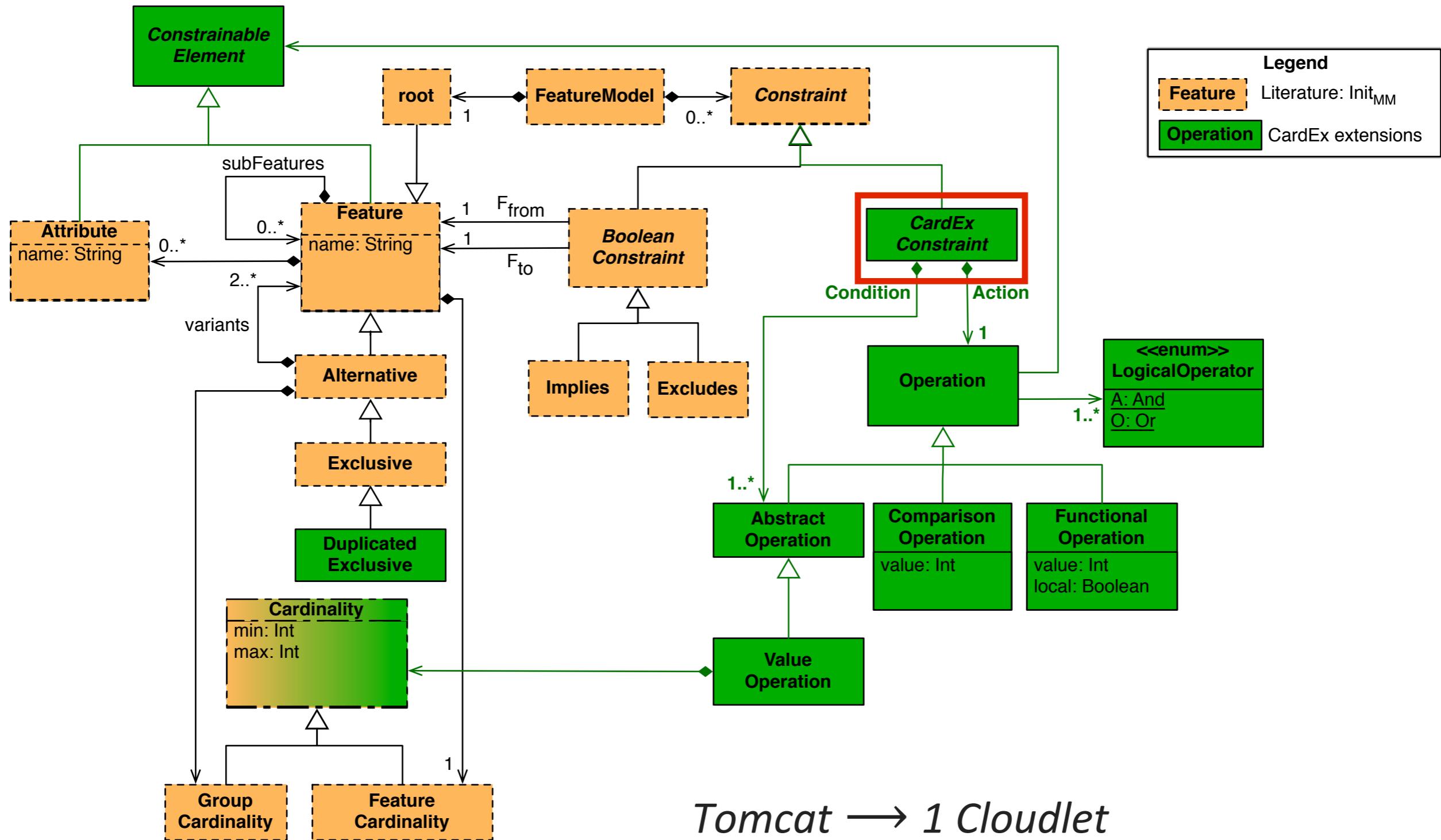
Feature Metamodel



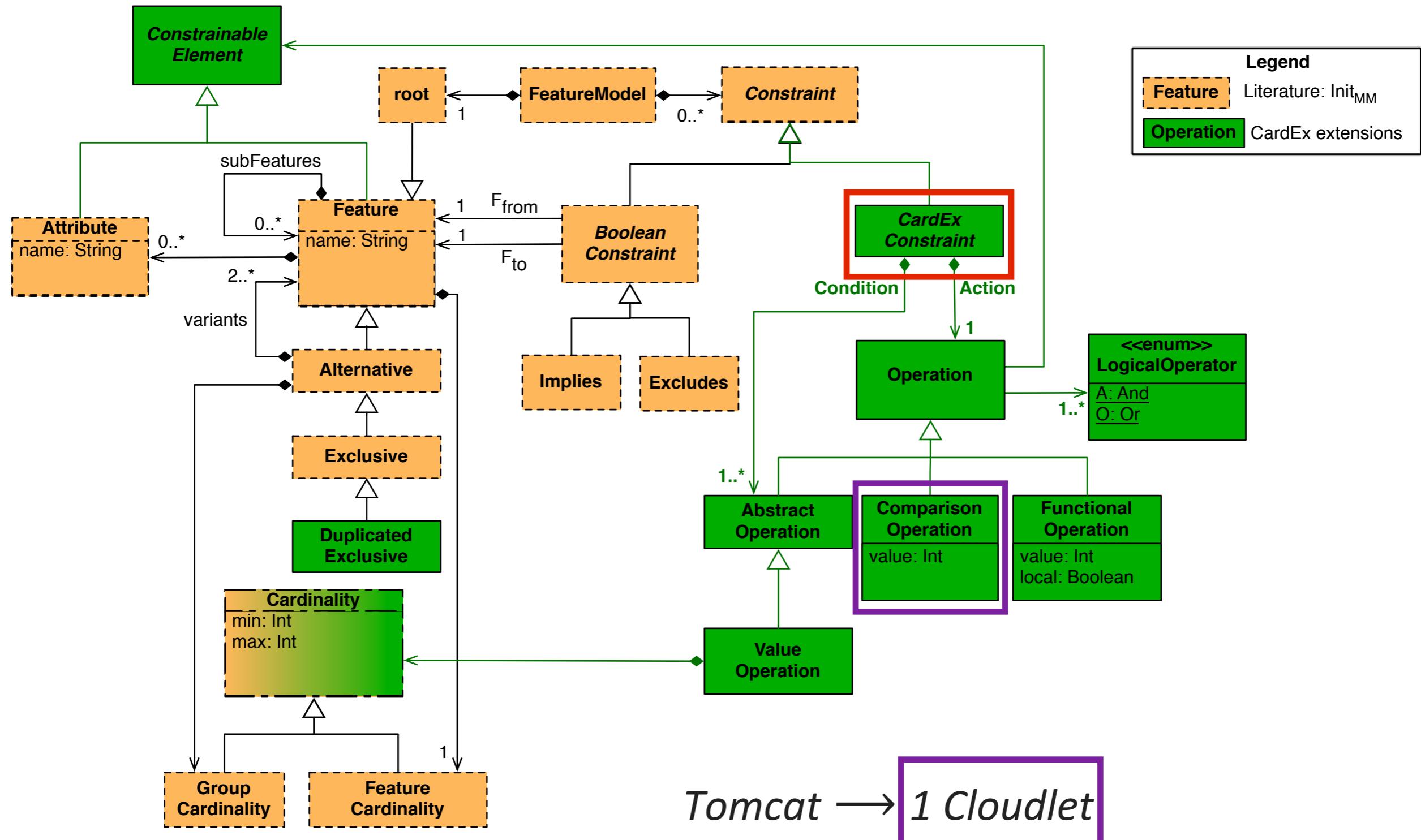
Feature Metamodel



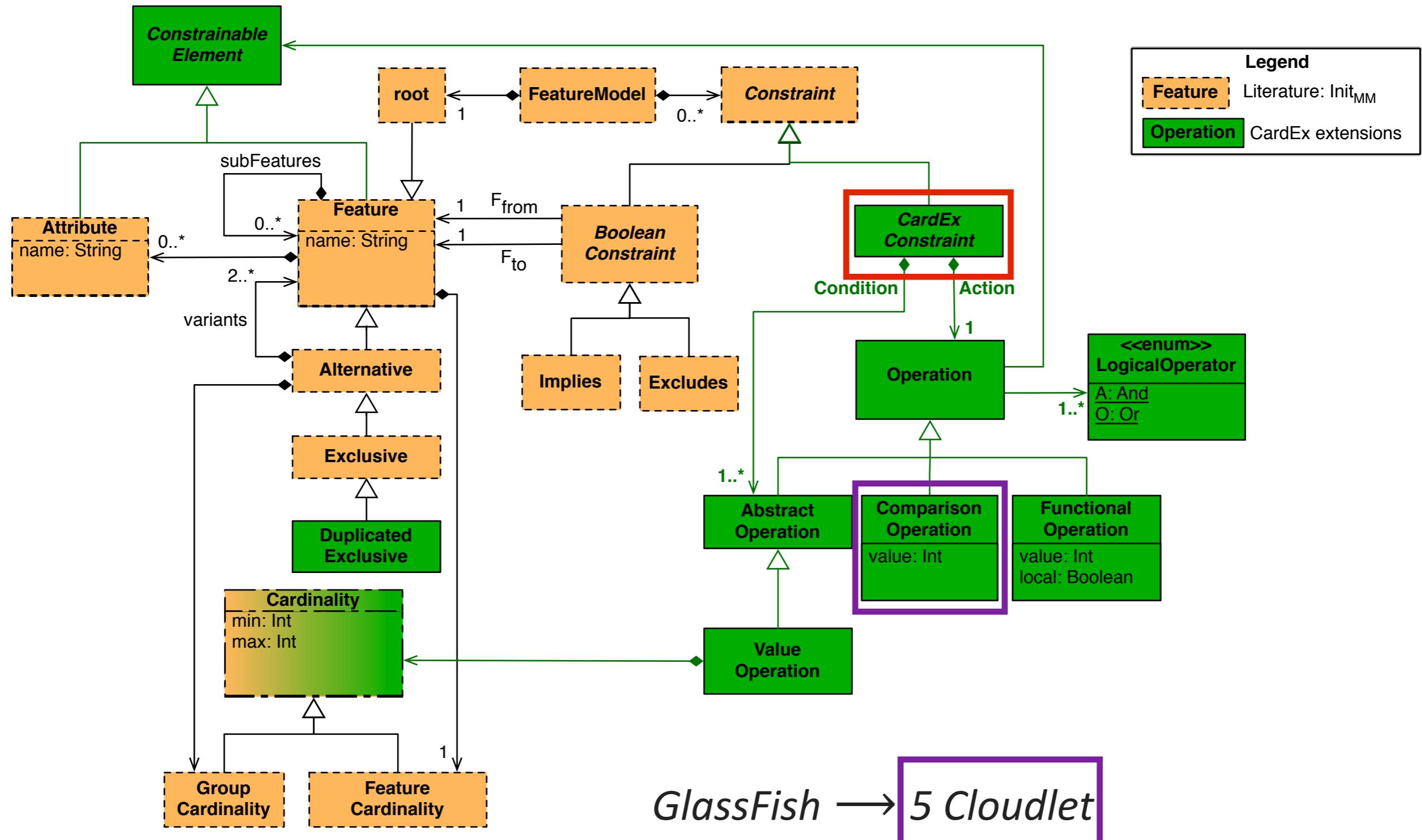
Feature Metamodel



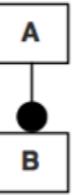
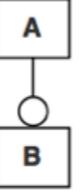
Feature Metamodel



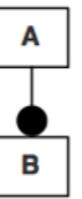
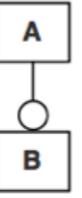
Feature Metamodel



Reasoning

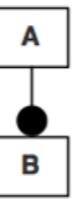
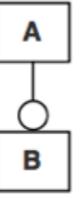
Relation	Feature model notation
Mandatory	
Optional	
Feature Cardinality	
Or-group	
Alternative-group	
Group cardinality	
Implies	$A \rightarrow B$
Excludes	$A \rightarrow \neg B$

Reasoning with Constraints Satisfaction Problems

Relation	Feature model notation	Constraint
Mandatory		$B = A$
Optional		$\text{ifThen}(A = 0; B = 0;)$
Feature Cardinality		$\text{ifThenElse}(A = 0; B = 0; B \in \{m, n\})$
Or-group		$\text{ifThenElse}(A > 0; \text{sum}(B, C) \geq 1; \text{sum}(B, C) = 0;)$
Alternative-group		$\text{ifThenElse}(A > 0; \text{sum}(B, C) = 1; \text{sum}(B, C) = 0;)$
Group cardinality		$\text{ifThenElse}(A > 0; \text{sum}(B, C) \in \{m, n\}; \text{sum}(B, C) = 0;)$
Implies	$A \rightarrow B$	$\text{ifThen}(A > 0; B > 0;)$
Excludes	$A \rightarrow \neg B$	$\text{ifThen}(A > 0; B = 0;)$

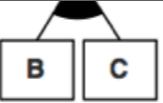
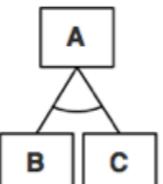
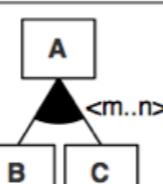


Reasoning with Constraints Satisfaction Problems

Relation	Feature model notation	Constraint
Mandatory		$B = A$
Optional		$\text{ifThen}(A = 0; B = 0;)$
Feature Cardinality		$\text{ifThenElse}(A = 0; B = 0; B \in \{m, n\})$
Or-group		$\text{ifThenElse}(A > 0; \text{sum}(B, C) \geq 1; \text{sum}(B, C) = 0;)$
Alternative-group		$\text{ifThenElse}(A > 0; \text{sum}(B, C) = 1; \text{sum}(B, C) = 0;)$
Group cardinality		$\text{ifThenElse}(A > 0; \text{sum}(B, C) \in \{m, n\}; \text{sum}(B, C) = 0;)$
Implies	$A \rightarrow B$	$\text{ifThen}(A > 0; B > 0;)$
Excludes	$A \rightarrow \neg B$	$\text{ifThen}(A > 0; B = 0;)$



Reasoning with Constraints Satisfaction Problems

Or-group		<code>ifThenElse(A > 0; sum(B, C) ≥ 1; sum(B, C) = 0;)</code>
Alternative-group		<code>ifThenElse(A > 0; sum(B, C) = 1; sum(B, C) = 0;)</code>
Group cardinality		<code>ifThenElse(A > 0; sum(B, C) ∈ {m, n}; sum(B, C) = 0;)</code>
Implies	$A \rightarrow B$	<code>ifThen(A > 0; B > 0;)</code>
Excludes	$A \rightarrow \neg B$	<code>ifThen(A > 0; B = 0;)</code>
Value	$[i, j]A \rightarrow [m, n]B$	<code>ifThen(A ∈ {i, j}; B ∈ {m, n};)</code>
Comparison	$A \rightarrow nB$	<code>ifThen(A > 0; B ≥ n × A;)</code>
Functional	$mA \wedge nB \rightarrow C \geq k$	<code>ifThen(and(A = m, B = n); C ≥ k;)</code>

In Practice

In Practice

Cloud	Type	Total	Features			Attributes	Total	Constraints	
			F_{card}	F_{attr}				C_{card}	C_{attr}
Amazon EC2	IaaS	23	2	2		5	28	9	18
Cloudbees	PaaS	23	2	1		4	12	3	9
Dotcloud	PaaS	34	4	3		6	21	6	17
GoGrid	IaaS	14	3	4		10	21	7	21
Google AE	PaaS	23	1	5		13	10	0	10
Heroku	PaaS	42	1	11		20	7	0	3
Jelastic	PaaS	31	3	1		2	12	10	0
OpenShift	PaaS	29	1	2		7	18	2	15
Pagoda Box	IaaS/PaaS	28	5	5		9	8	4	8
Windows Azure	IaaS/PaaS	31	6	12		29	46	0	46

SALOON: A Platform for Selecting and Configuring Cloud Environments.

Clément Quinton, Daniel Romero and Laurence Duchien.

In Software: Practice and Experience journal (SPE). Accepted with minor revisions, September 2014.

In Practice

Cloud	Type	Total	Features			Attributes	Total	Constraints	
			F_{card}	F_{attr}				C_{card}	C_{attr}
Amazon EC2	IaaS	23	2	2		5	28	9	18
Cloudbees	PaaS	23	2	1		4	12	3	9
Dotcloud	PaaS	34	4	3		6	21	6	17
GoGrid	IaaS	14	3	4		10	21	7	21
Google AE	PaaS	23	1	5		13	10	0	10
Heroku	PaaS	42	1	11		20	7	0	3
Jelastic	PaaS	31	3	1		2	12	10	0
OpenShift	PaaS	29	1	2		7	18	2	15
Pagoda Box	IaaS/PaaS	28	5	5		9	8	4	8
Windows Azure	IaaS/PaaS	31	6	12		29	46	0	46

Average
19

SALOON: A Platform for Selecting and Configuring Cloud Environments.

Clément Quinton, Daniel Romero and Laurence Duchien.

In Software: Practice and Experience journal (SPE). Accepted with minor revisions, September 2014.

Research Goals

- Manage Cloud variability
- Guarantee environment independance
- Provide a flexible solution
- Deliver an automated support
- Maintain consistency

Research Goals

- Manage Cloud variability
- Guarantee environment independance
- Provide a flexible solution
- Deliver an automated support
- Maintain consistency



Agenda

I. Introduction

II. Contributions

- Cloud environments variability modeling
- SALOON
- Consistency checking for evolving Cloud models

III. Conclusion and Perspectives

Selecting & Configuring Cloud Environments



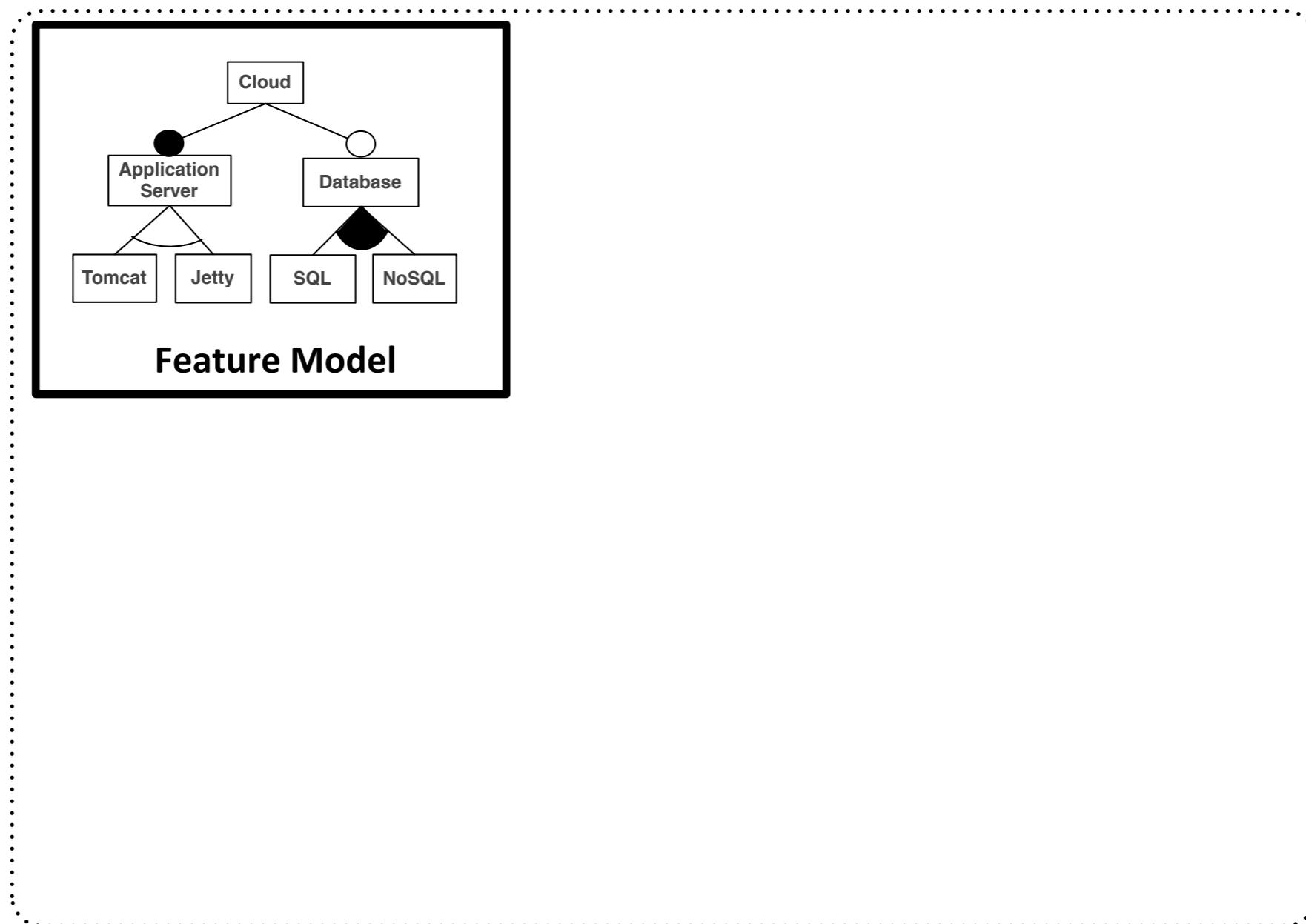
Selecting & Configuring Cloud Environments



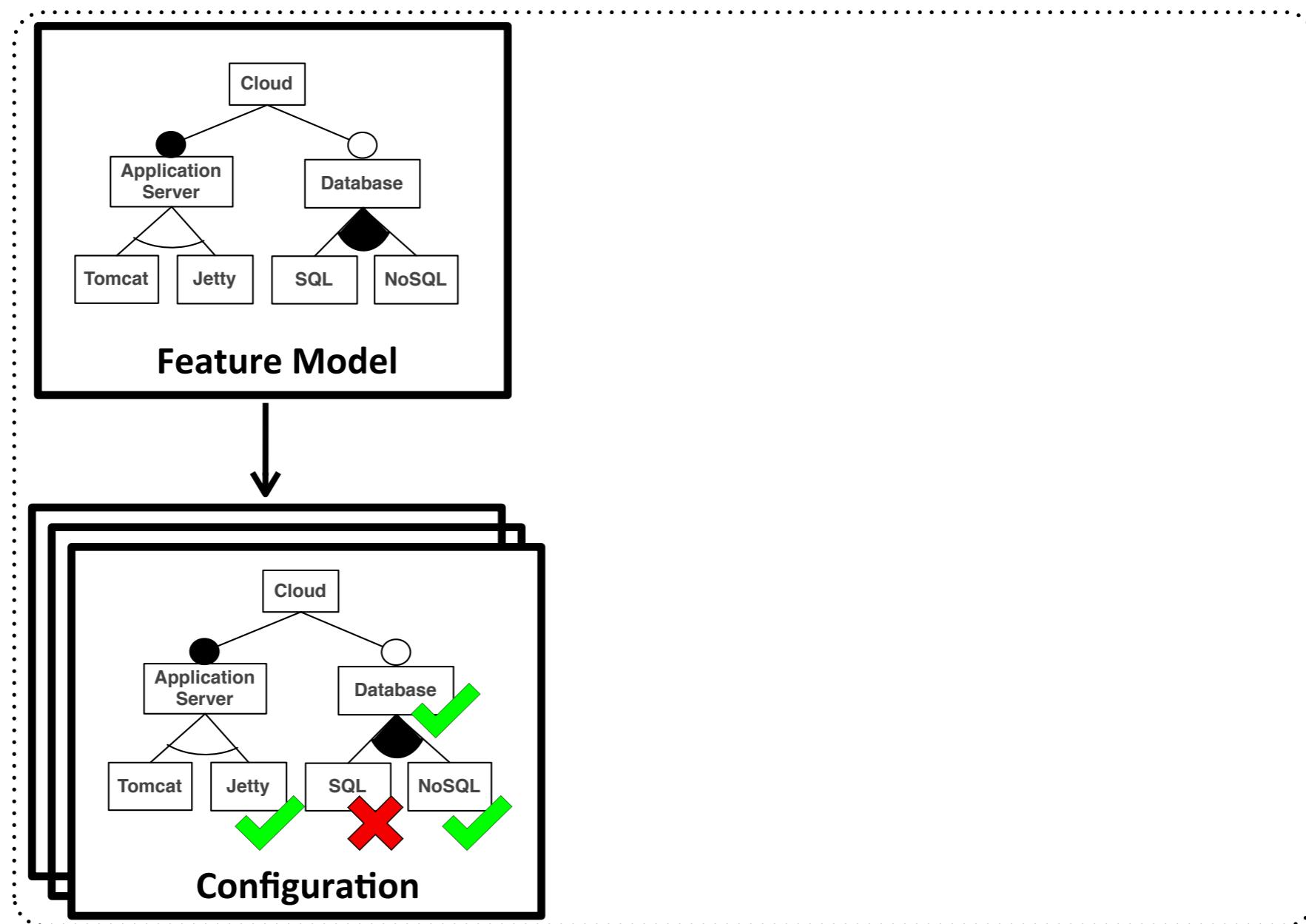
Selecting & Configuring Cloud Environments



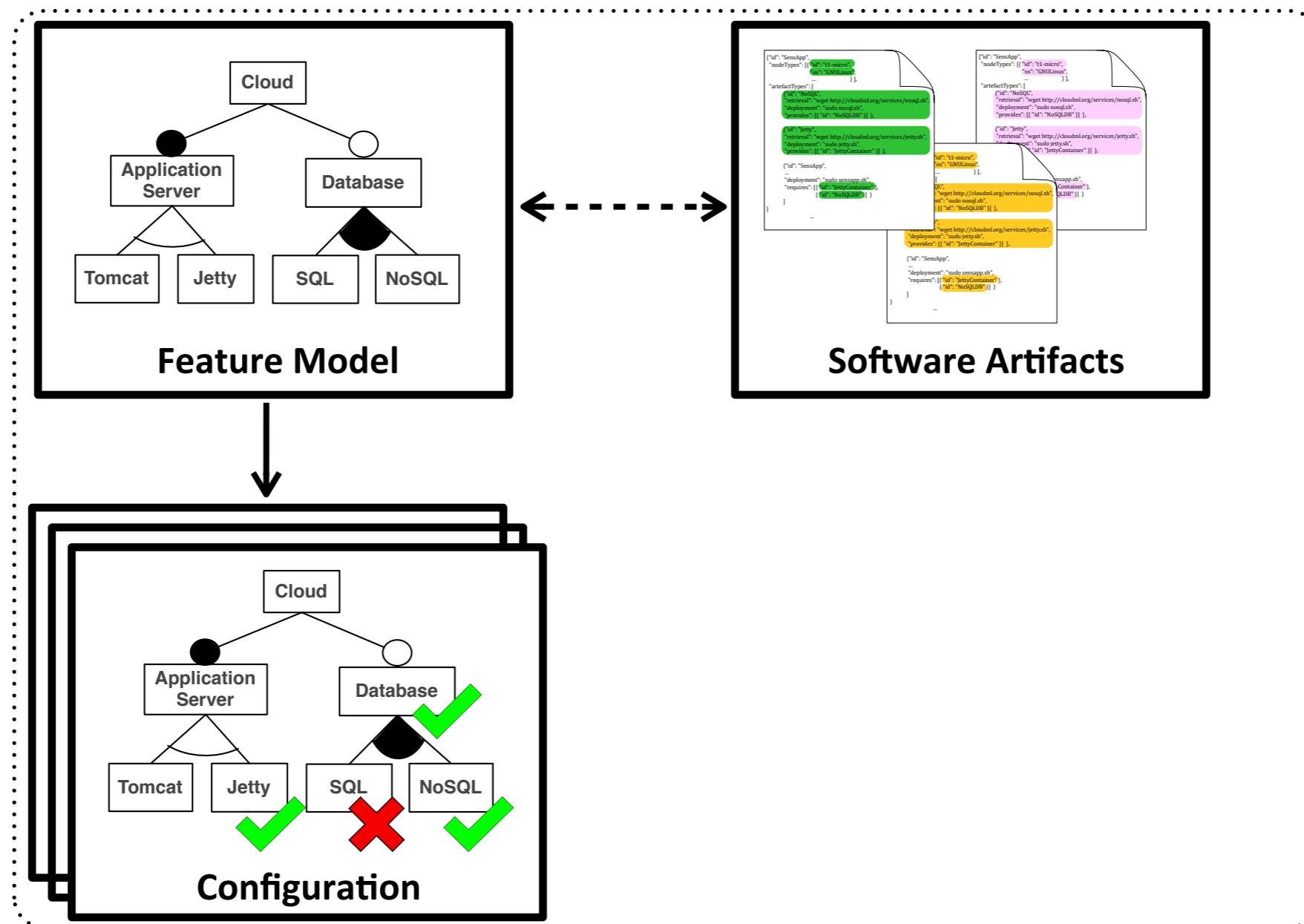
Selecting & Configuring Cloud Environments



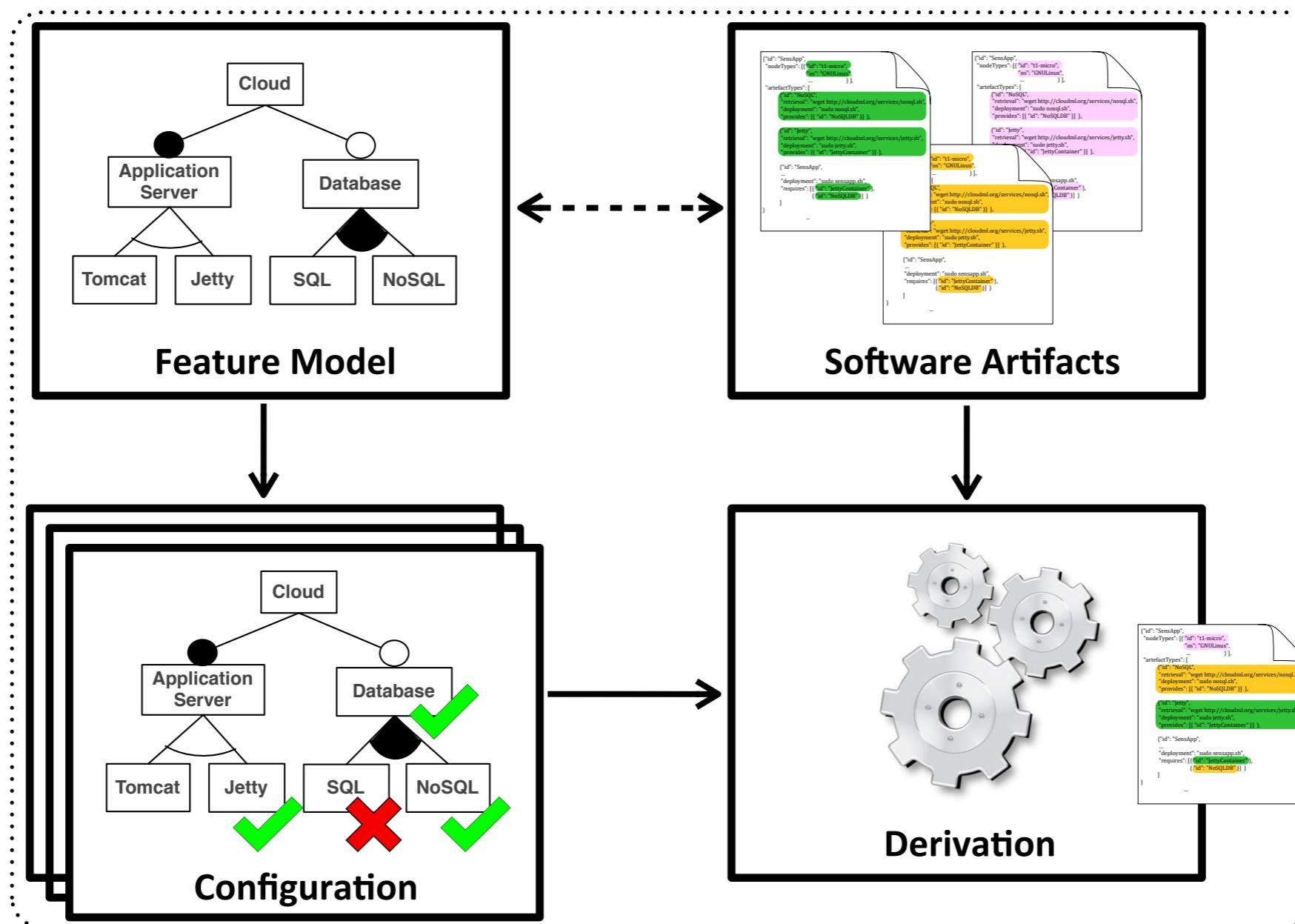
Selecting & Configuring Cloud Environments



Selecting & Configuring Cloud Environments



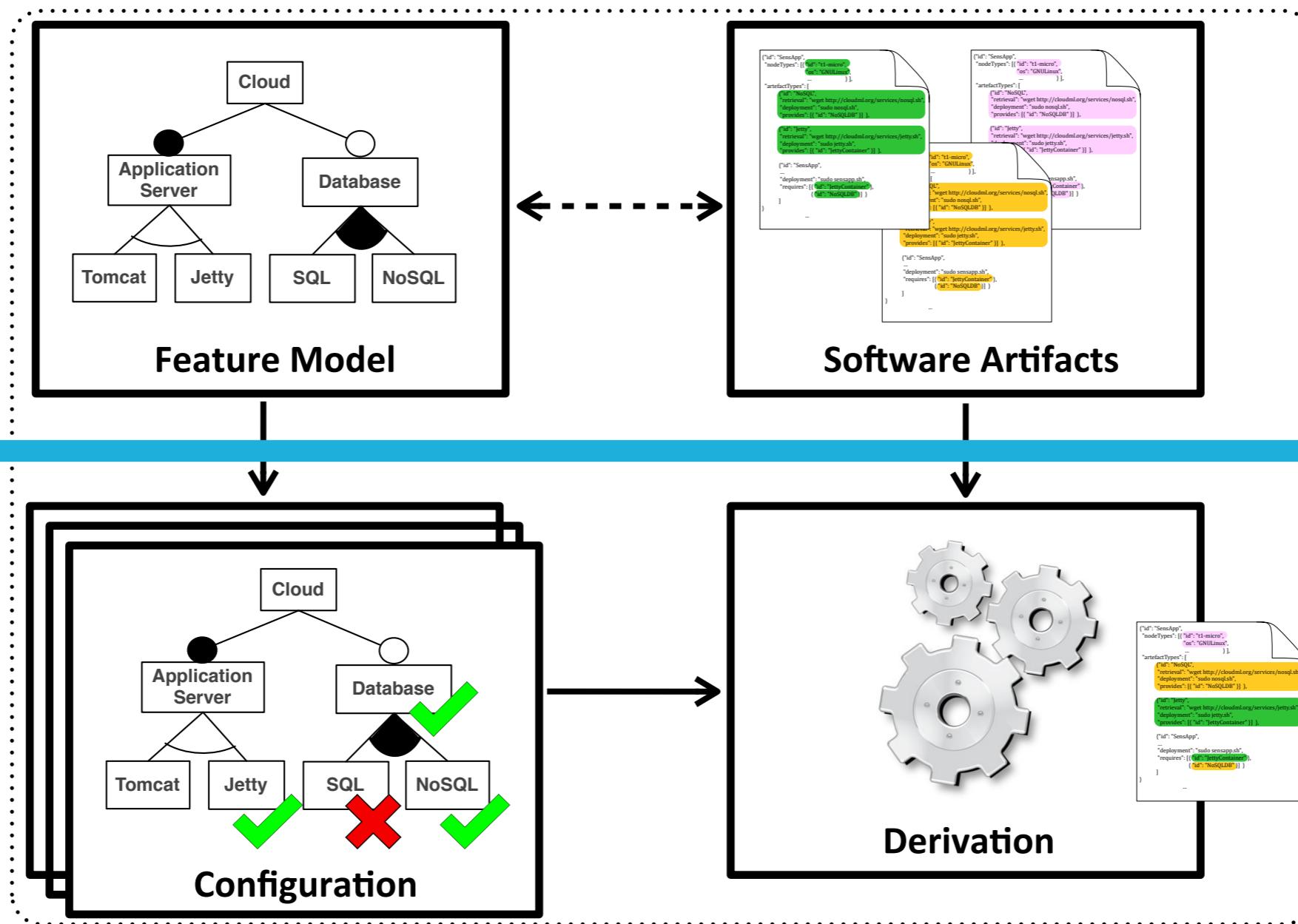
Selecting & Configuring Cloud Environments



Selecting & Configuring Cloud Environments

SoftwAre product Lines

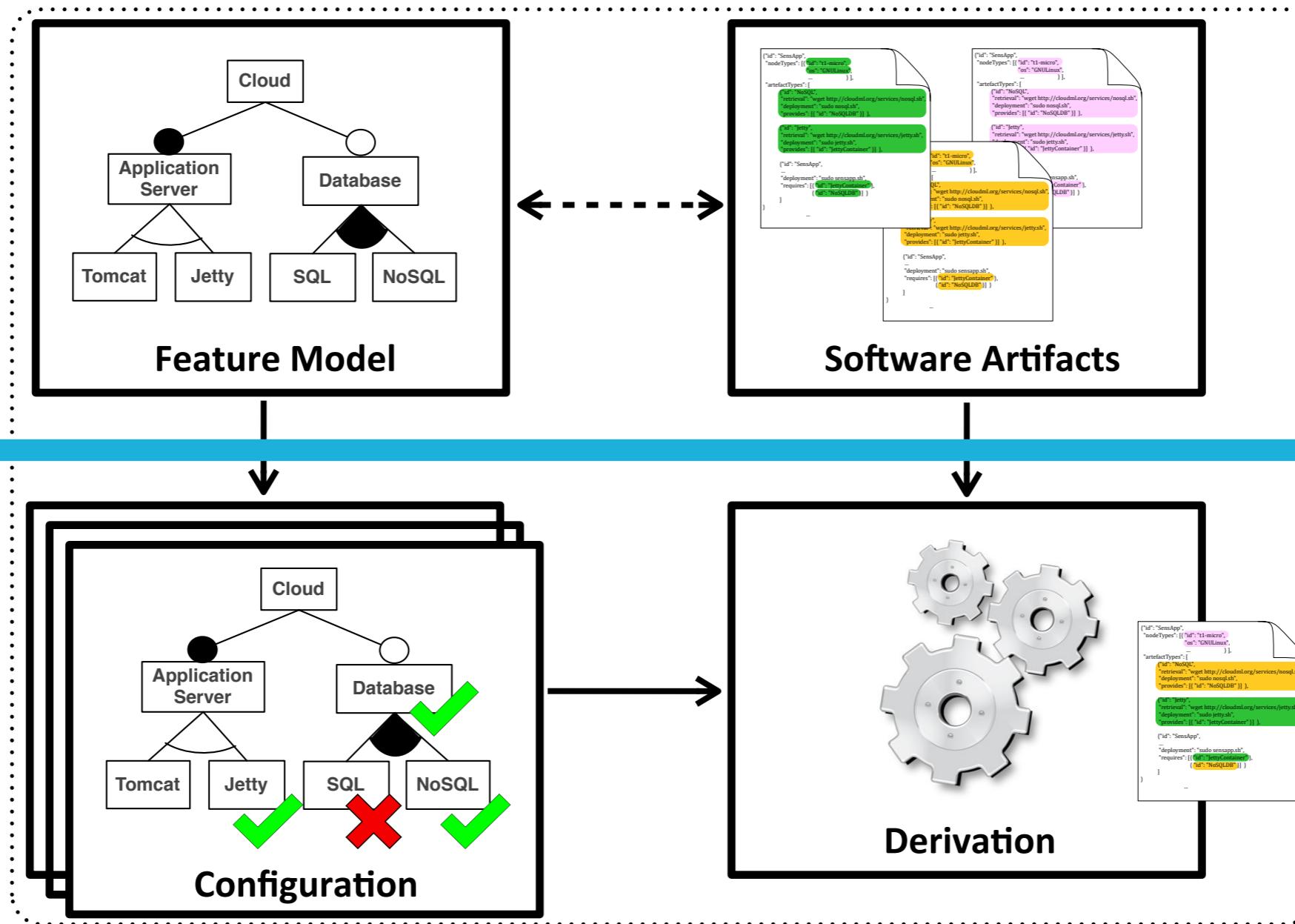
Domain
Engineering



Selecting & Configuring Cloud Environments

SoftwAre product Lines for cOud cOmputiNg

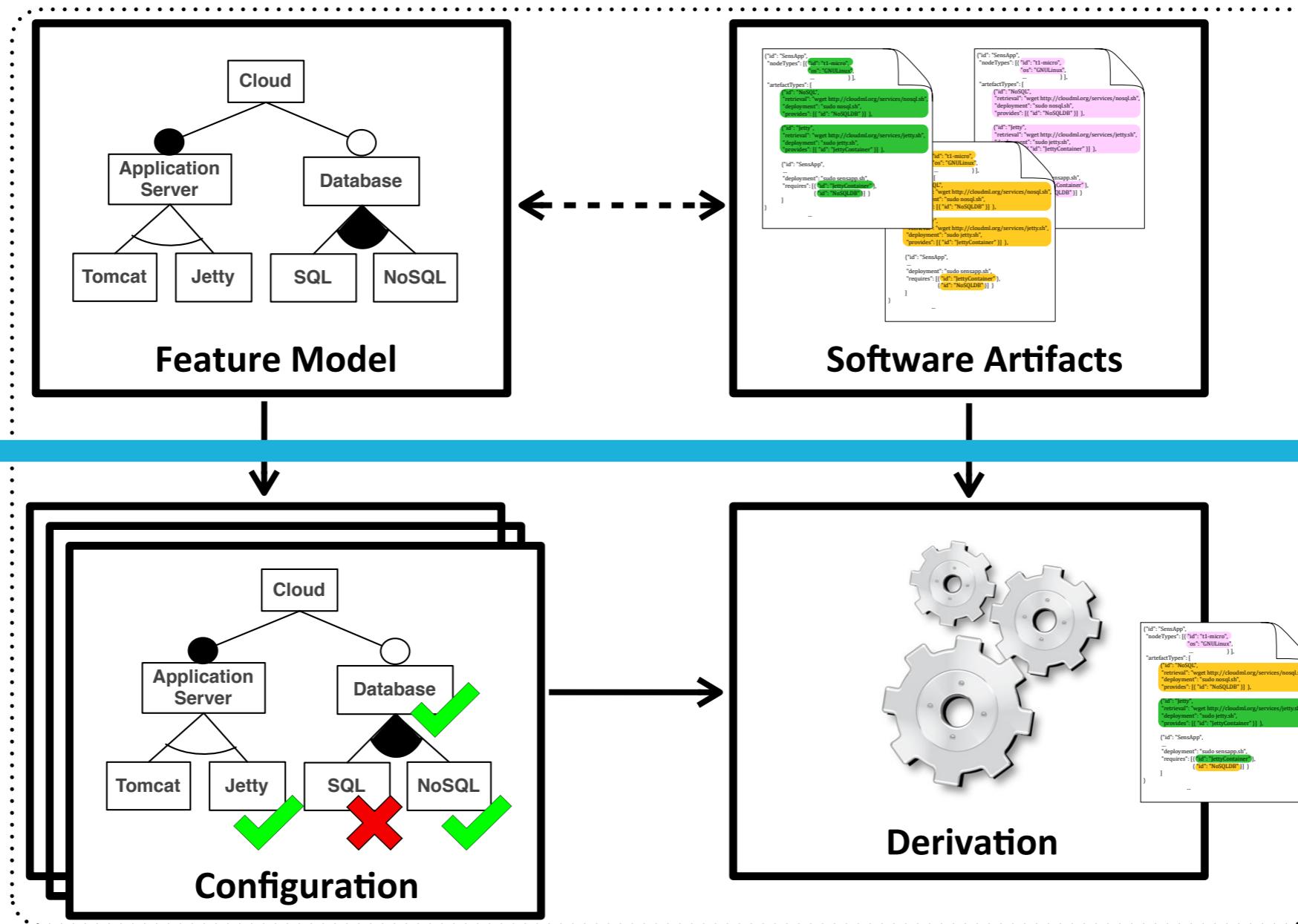
Domain
Engineering



Selecting & Configuring Cloud Environments

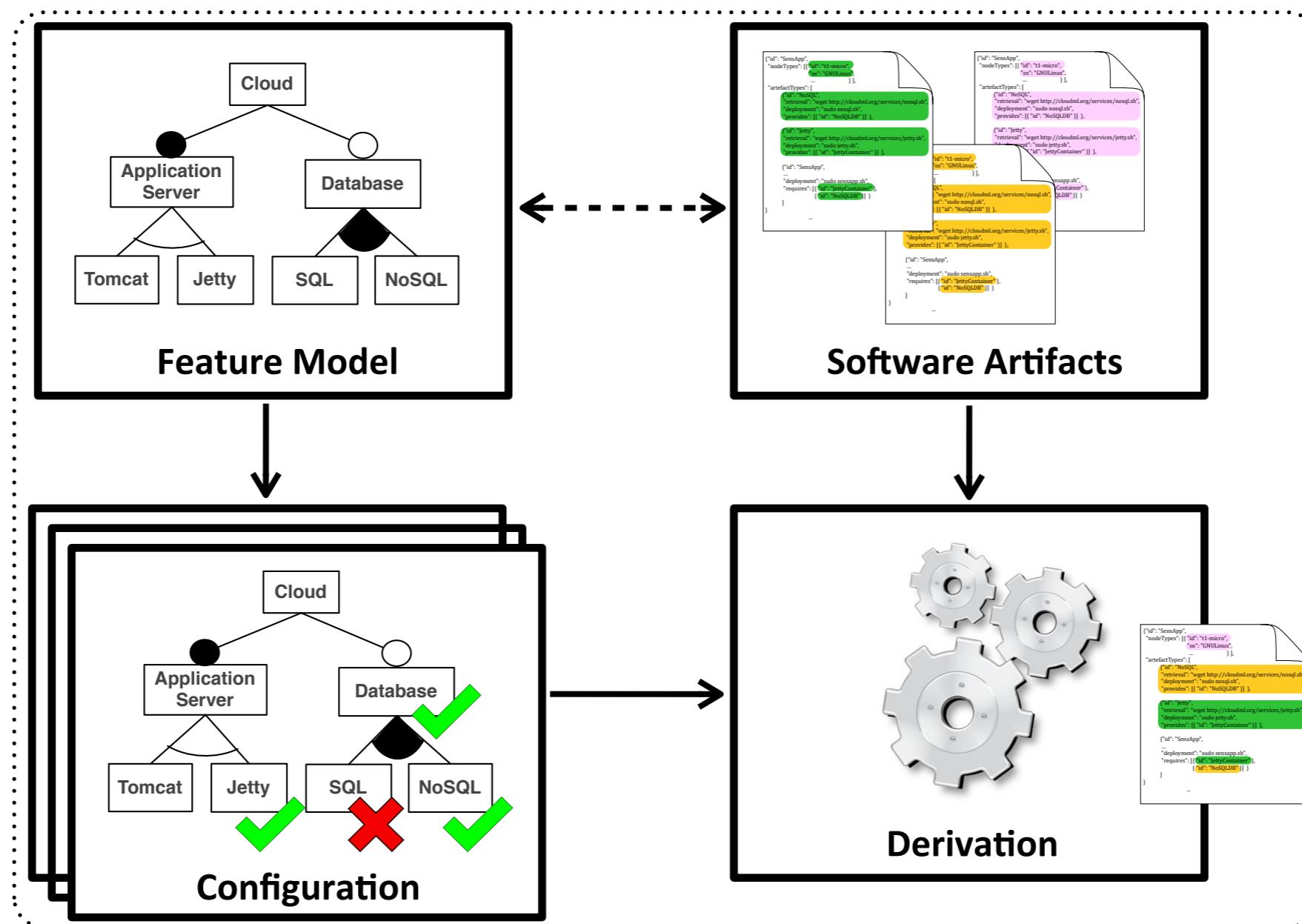
SALOON : SoftwAre product Lines for cLOUD cOmputiNg

Domain
Engineering



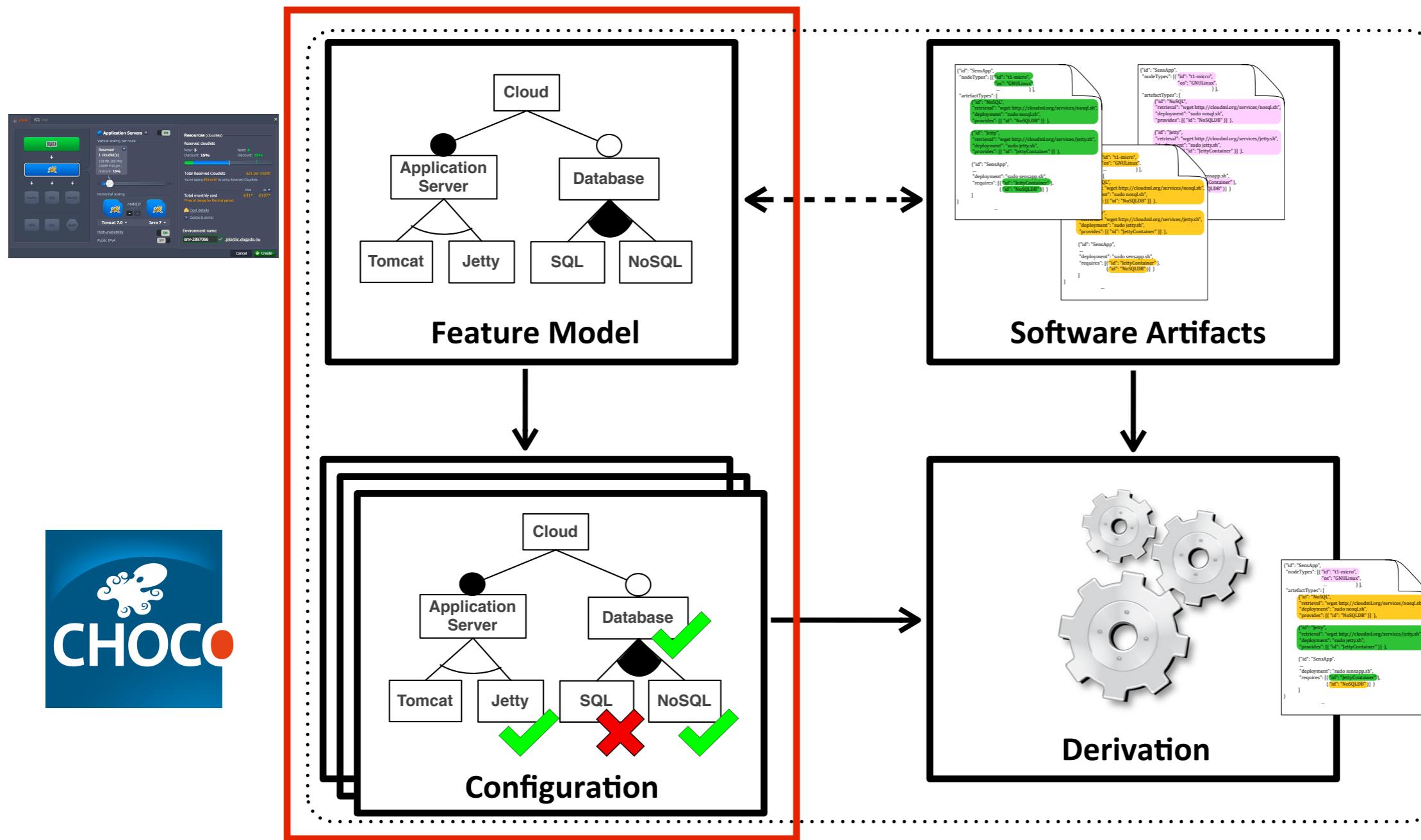
Selecting & Configuring Cloud Environments

SALOON : SoftwAre product Lines for cLOUD cOmputiNg



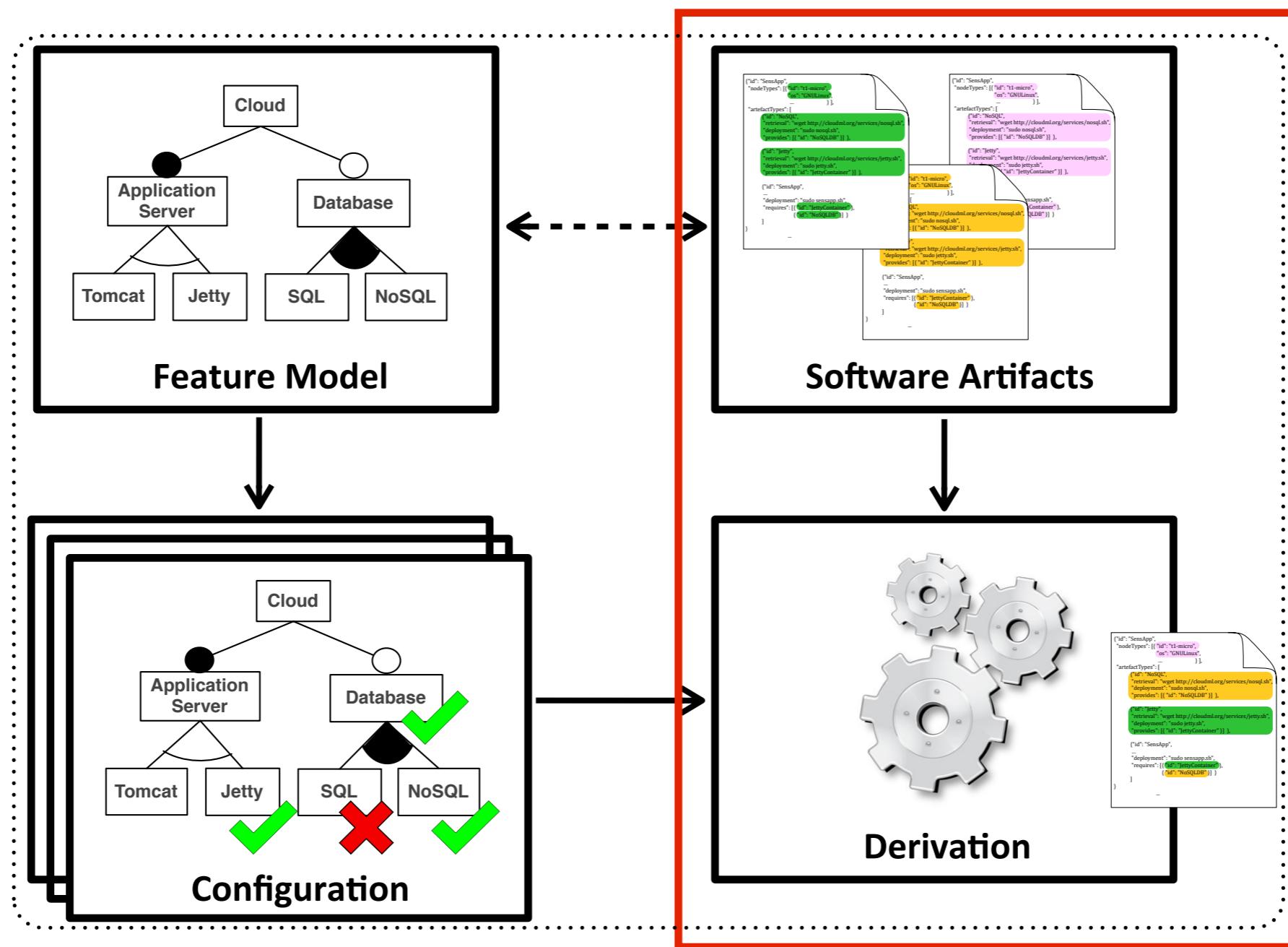
Selecting & Configuring Cloud Environments

SALOON : SoftwAre product Lines for cLOUD cOmputiNg

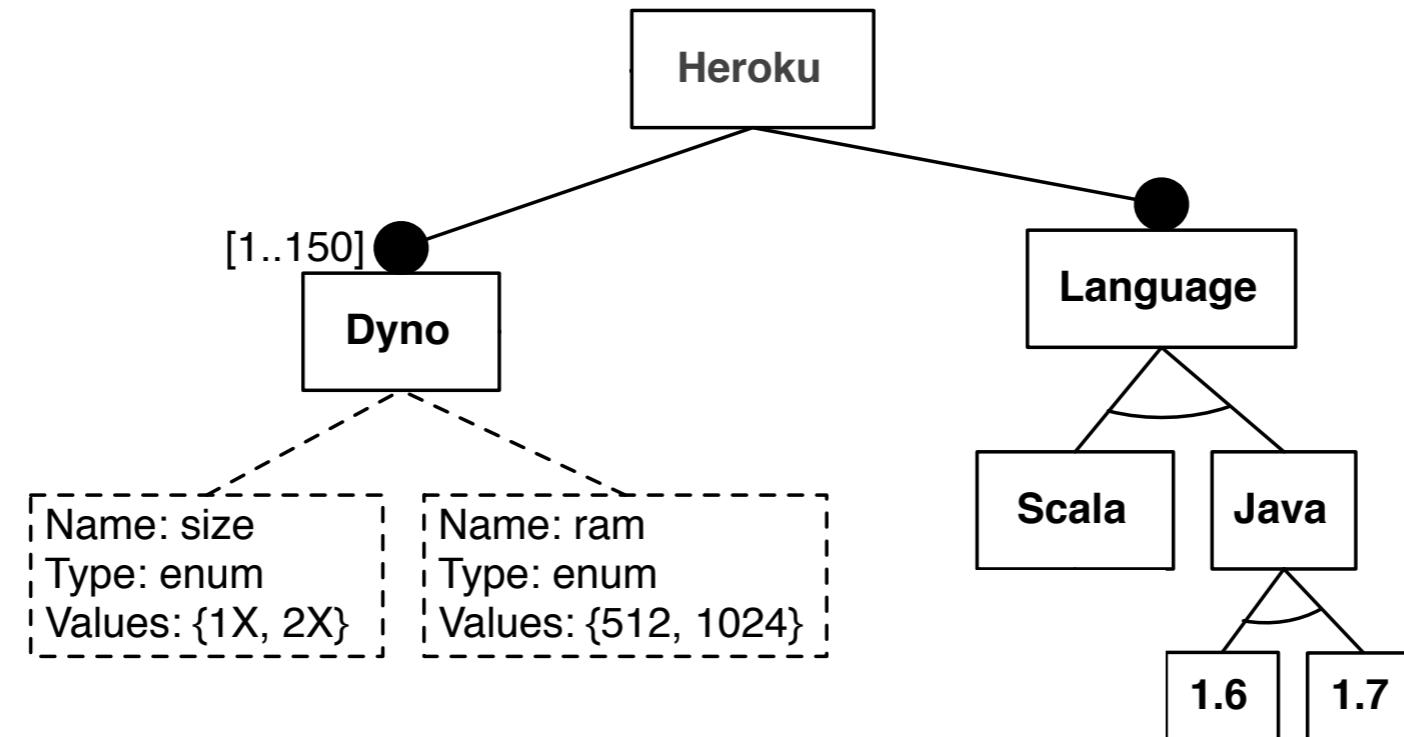


Selecting & Configuring Cloud Environments

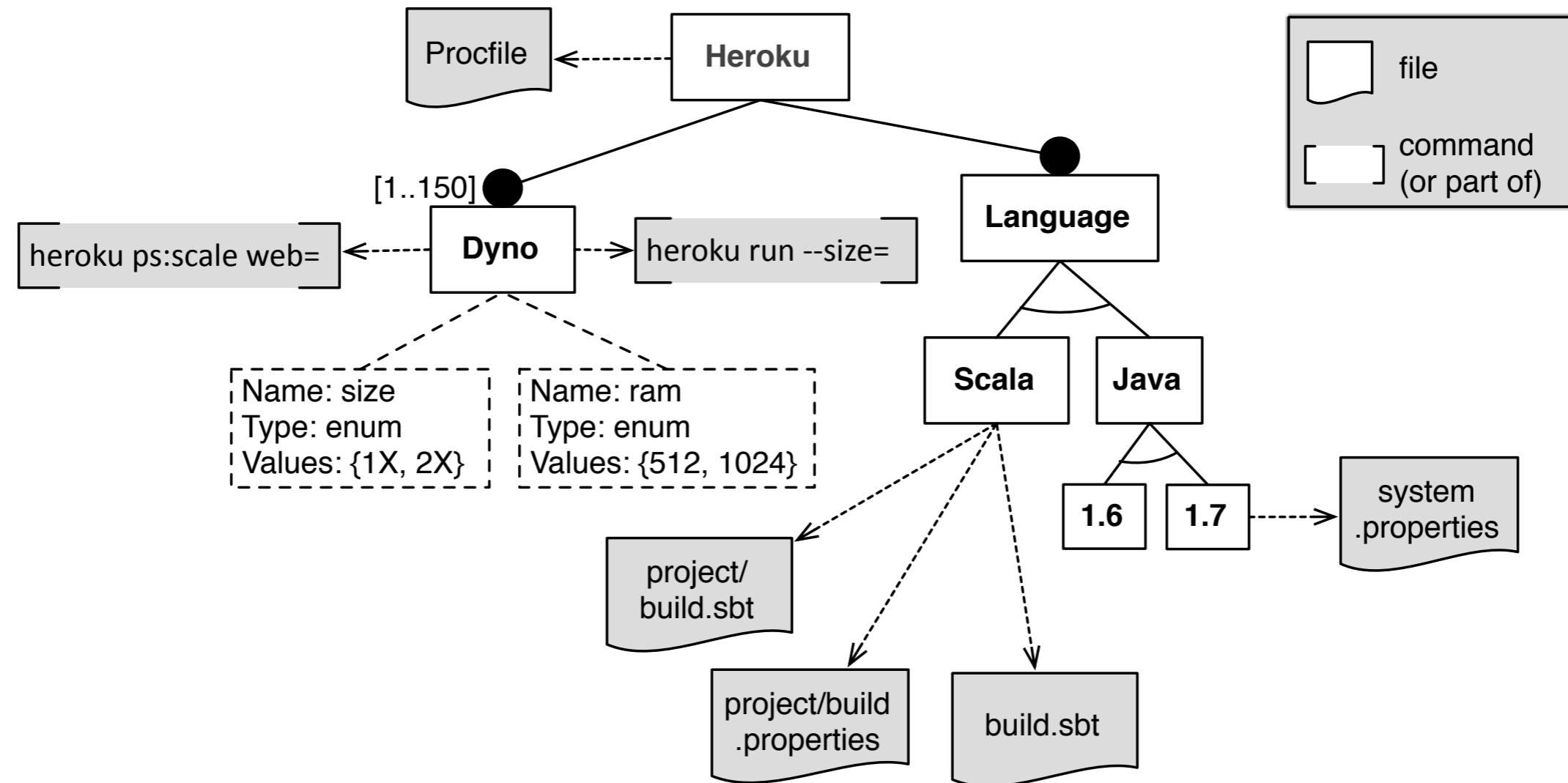
SALOON : SoftwAre product Lines for cLOUD cOmputiNg



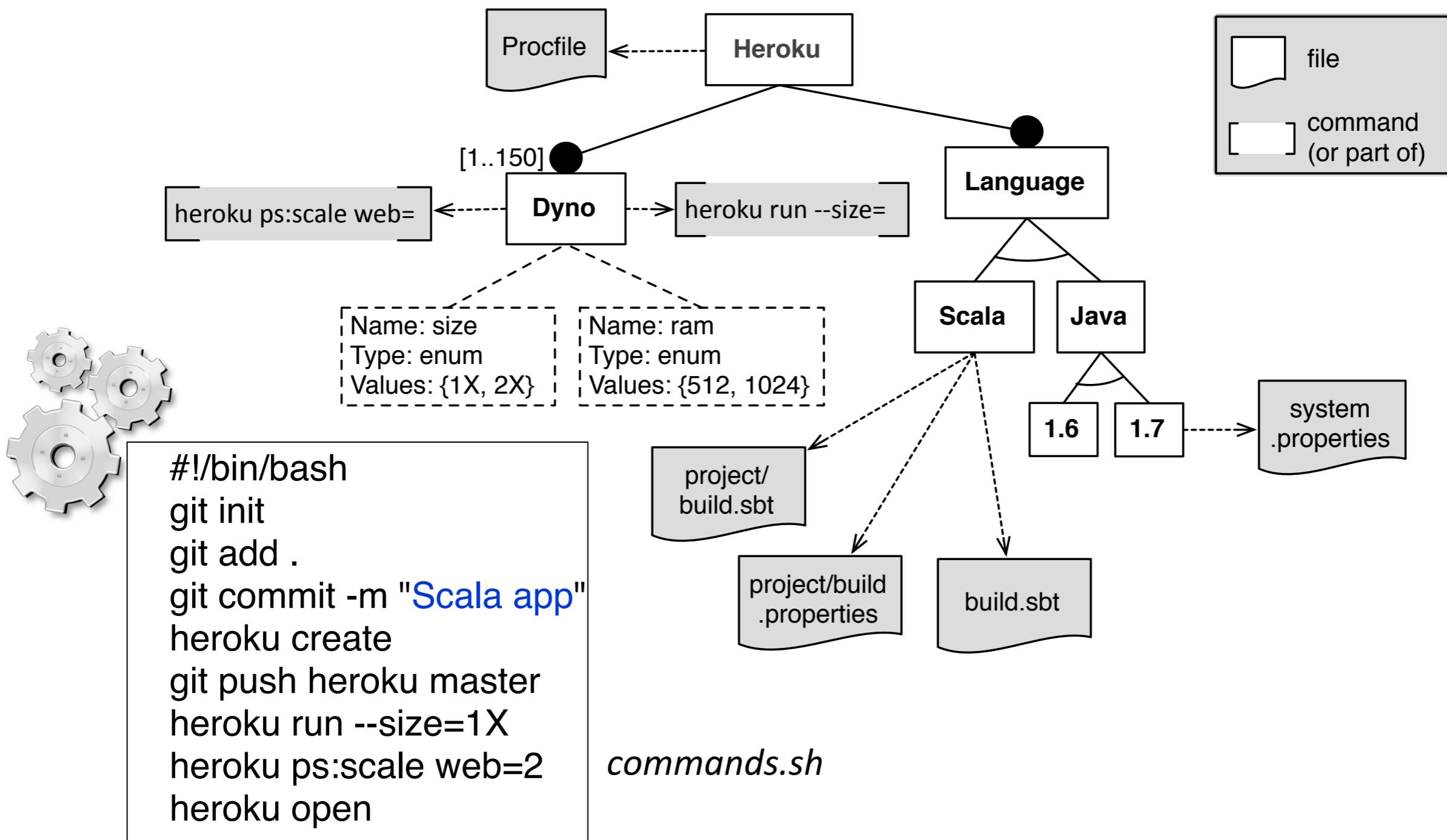
Software Artifacts as Configuration Files & Scripts



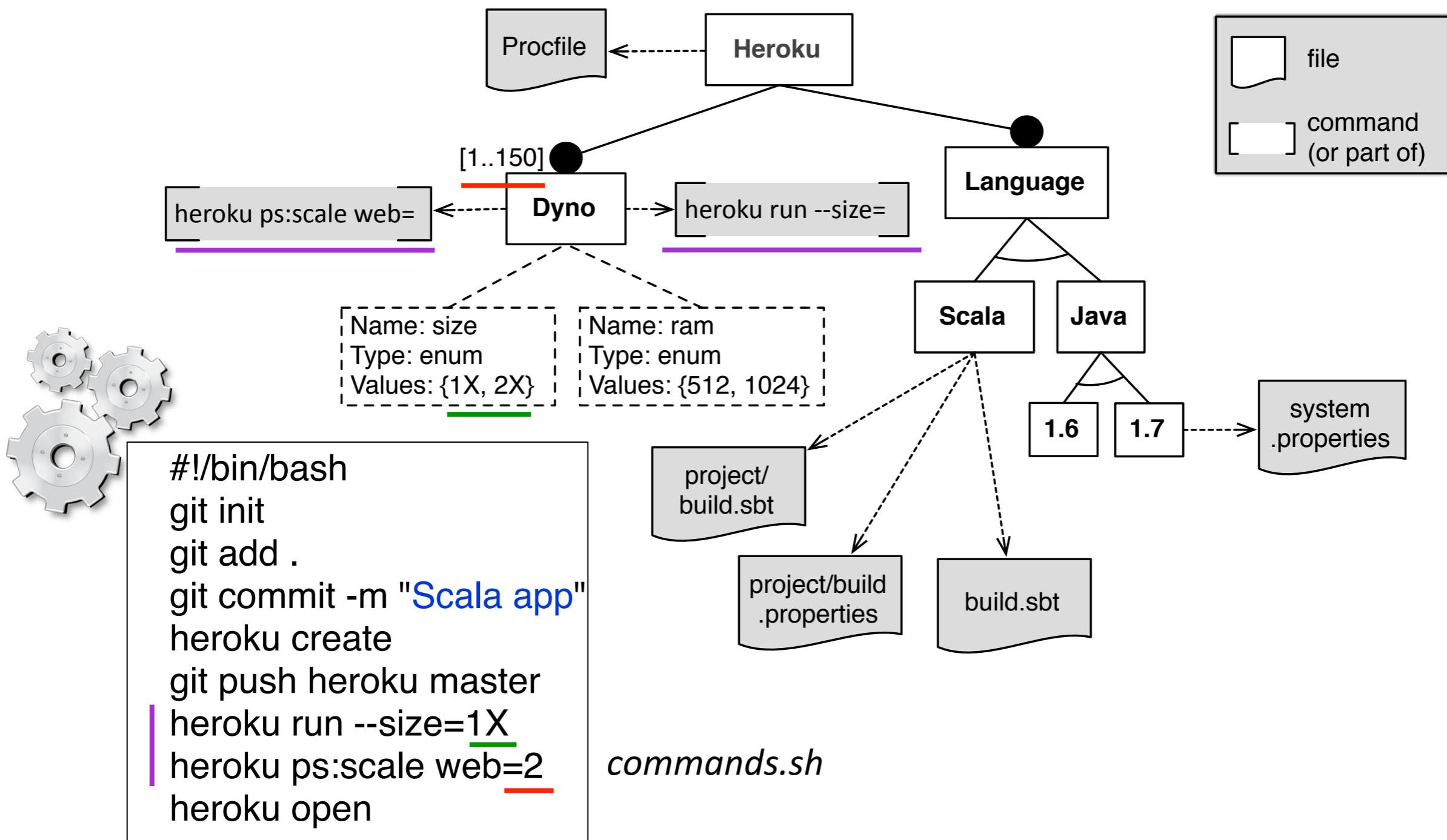
Software Artifacts as Configuration Files & Scripts



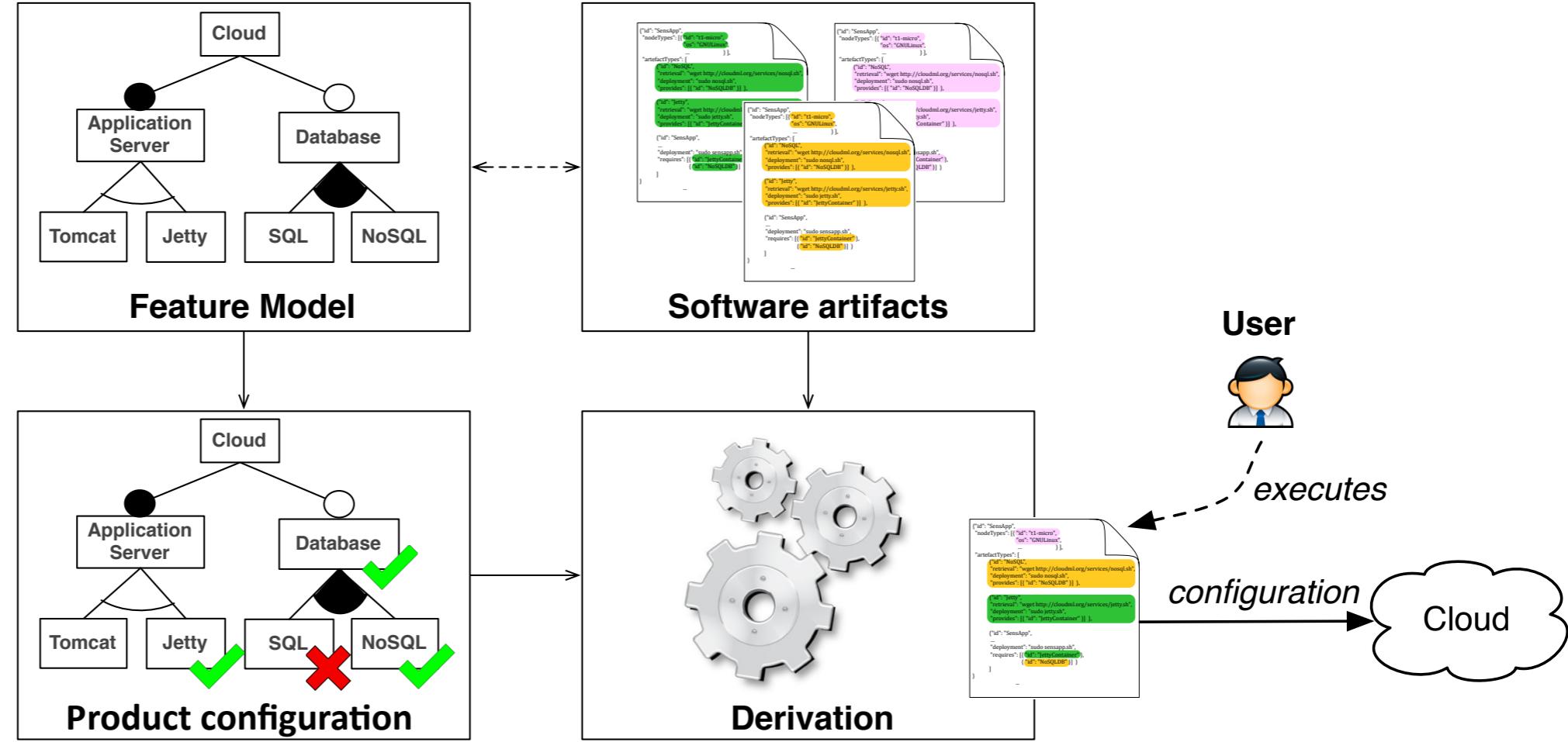
Software Artifacts as Configuration Files & Scripts



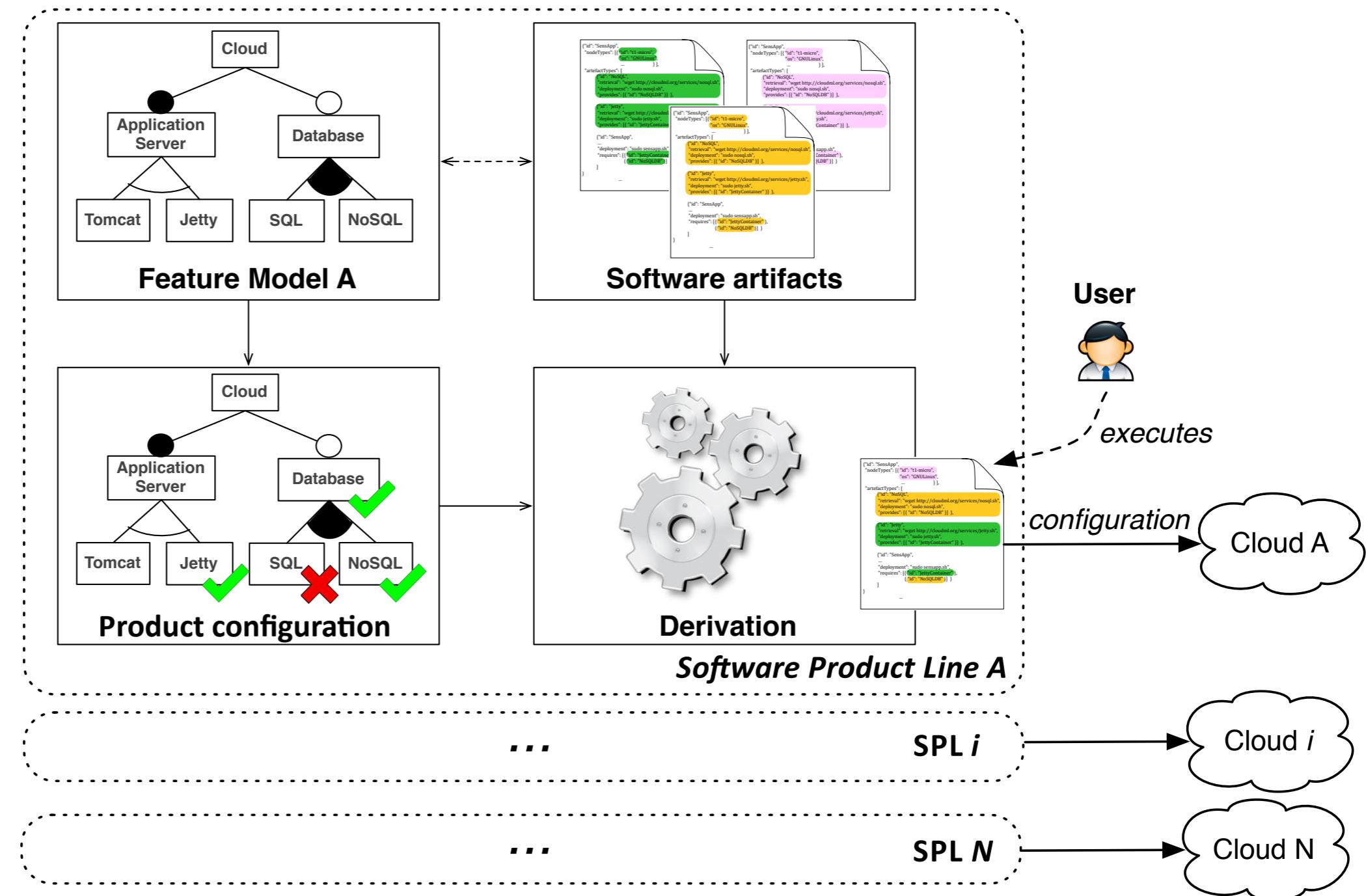
Software Artifacts as Configuration Files & Scripts



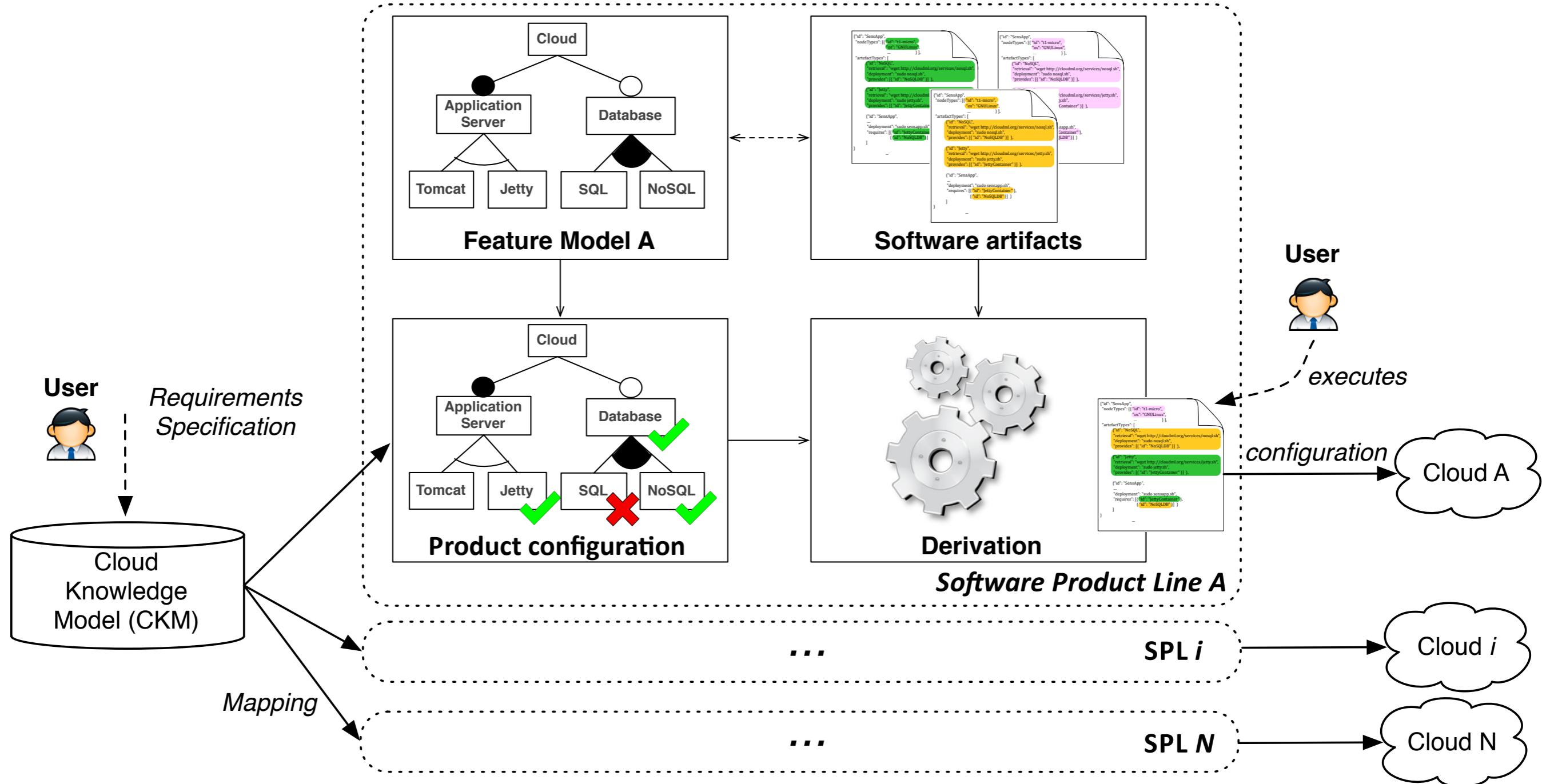
SALOON Overview



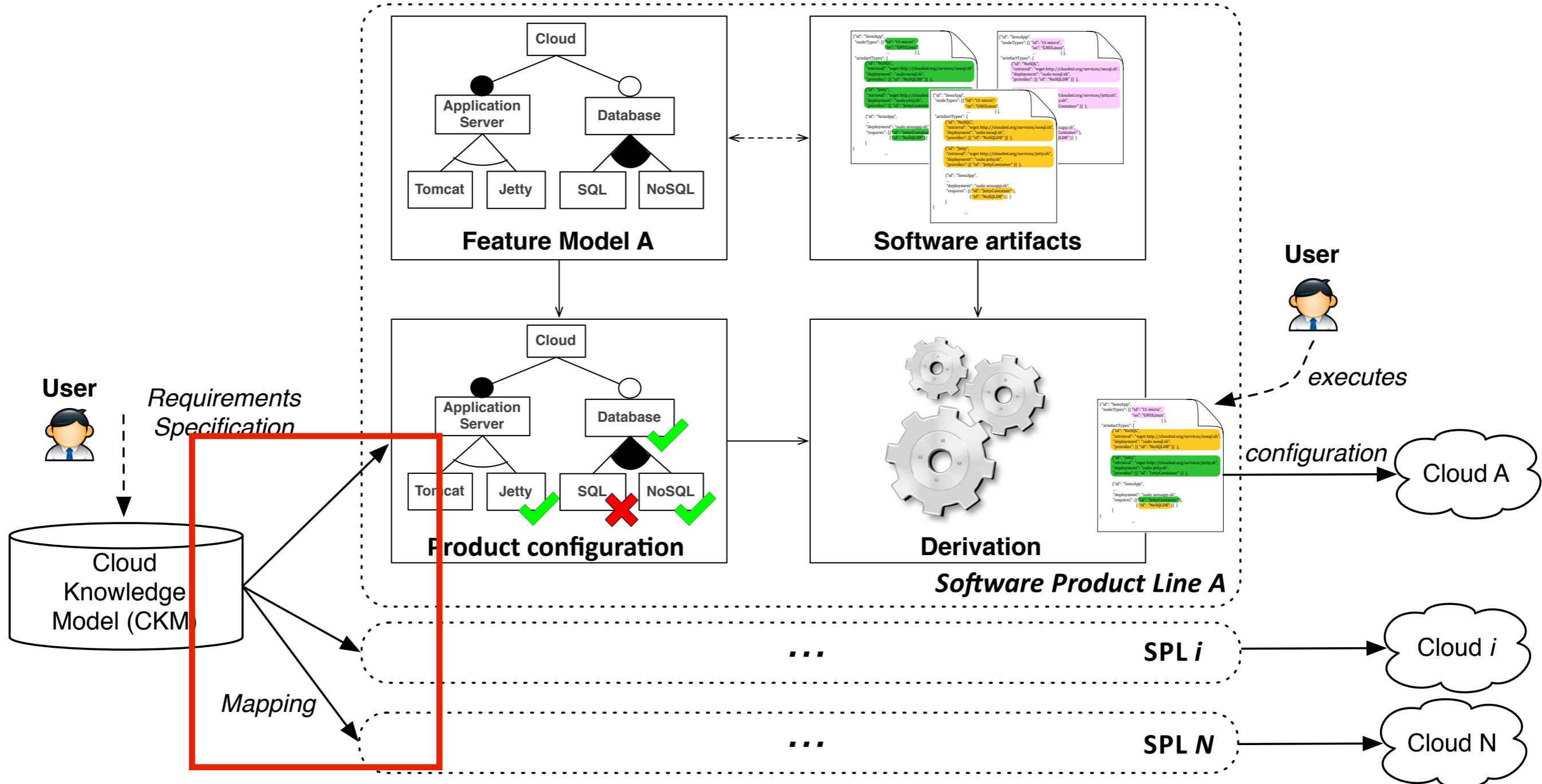
SALOON Overview



SALOON Overview

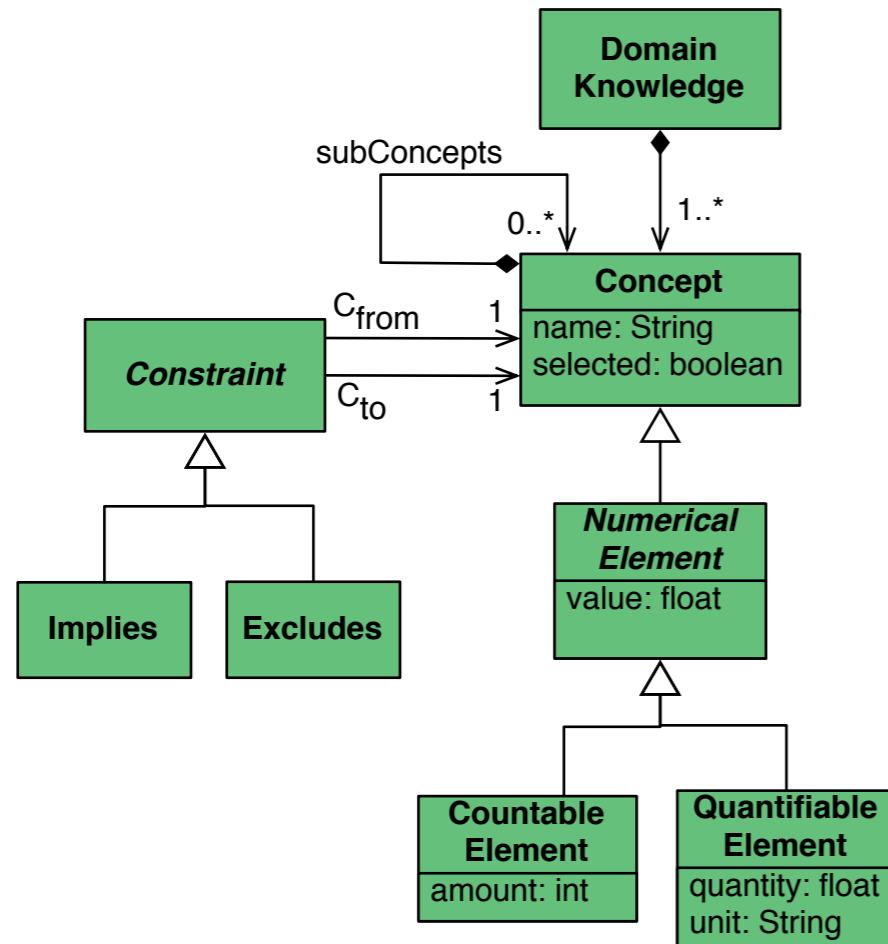


SALOON Overview

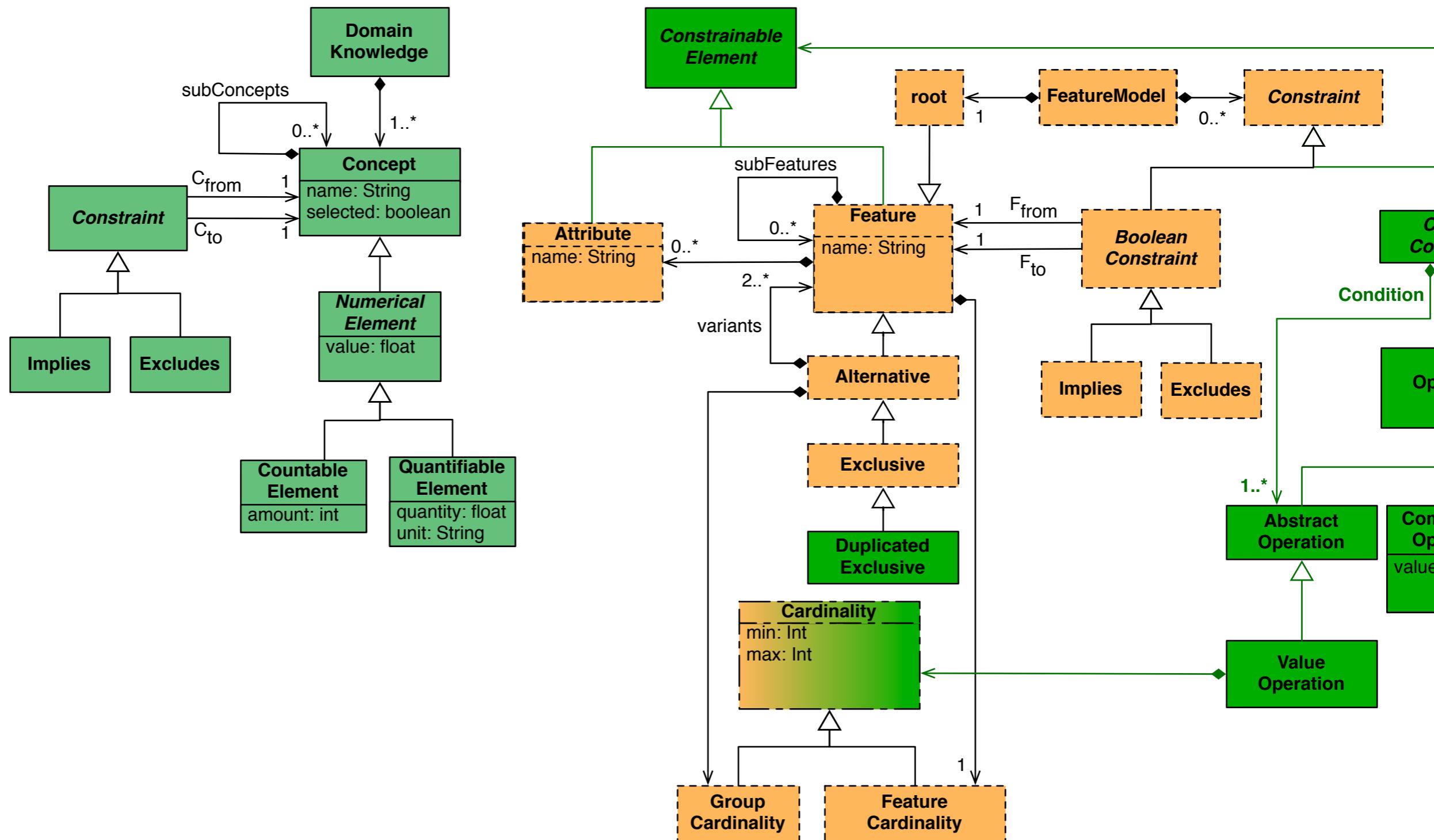


Mappings

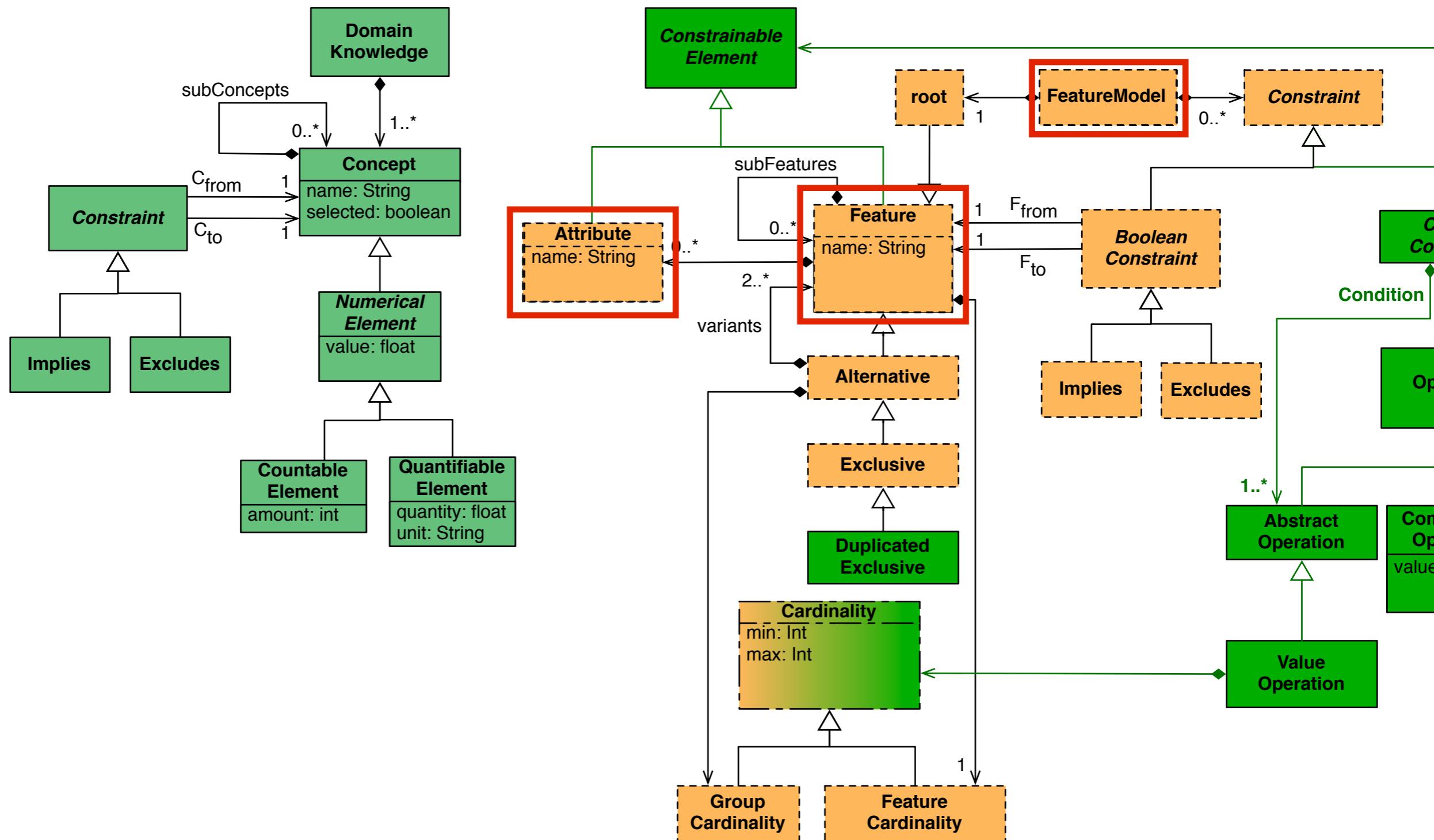
Mappings



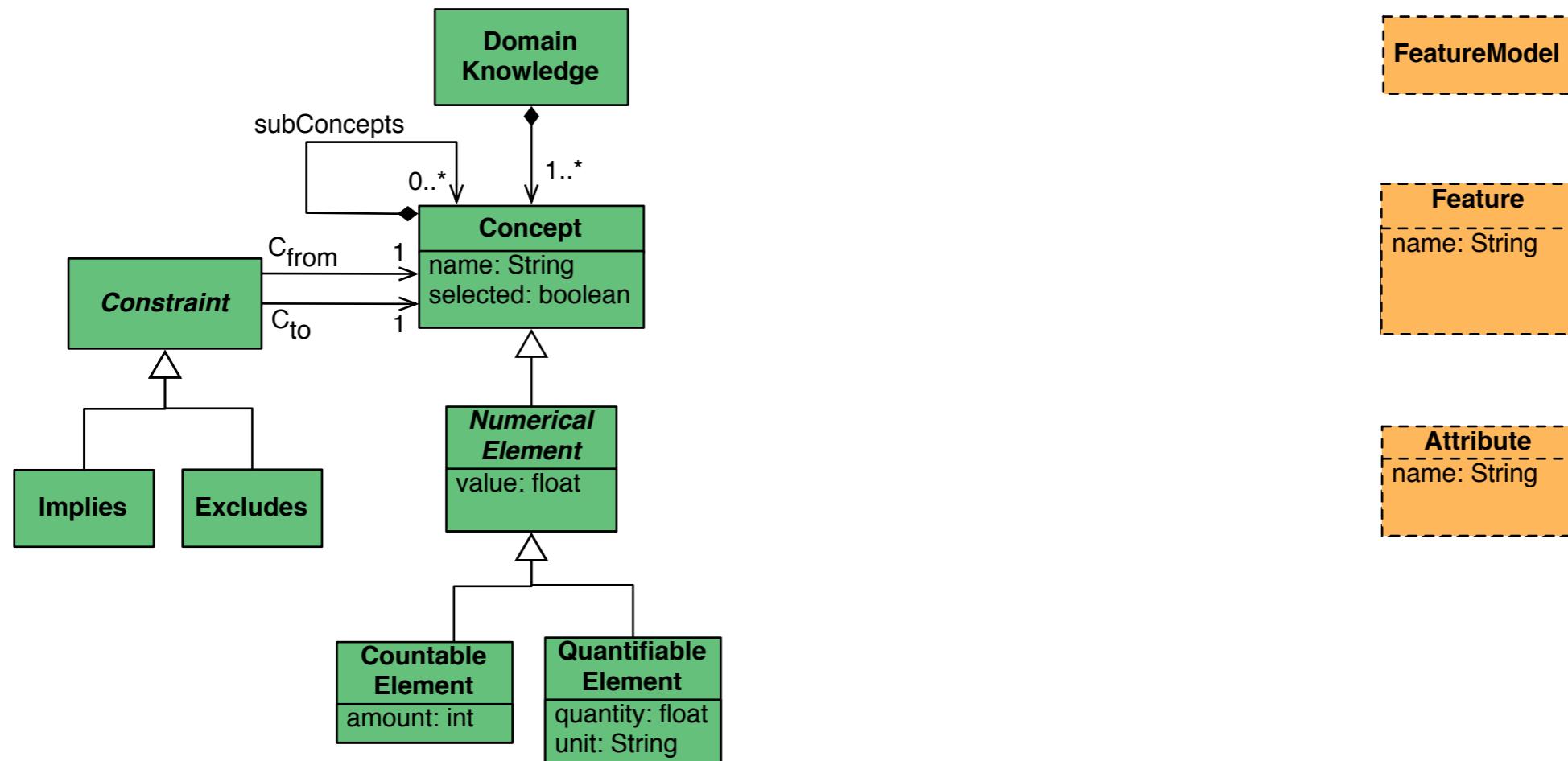
Mappings



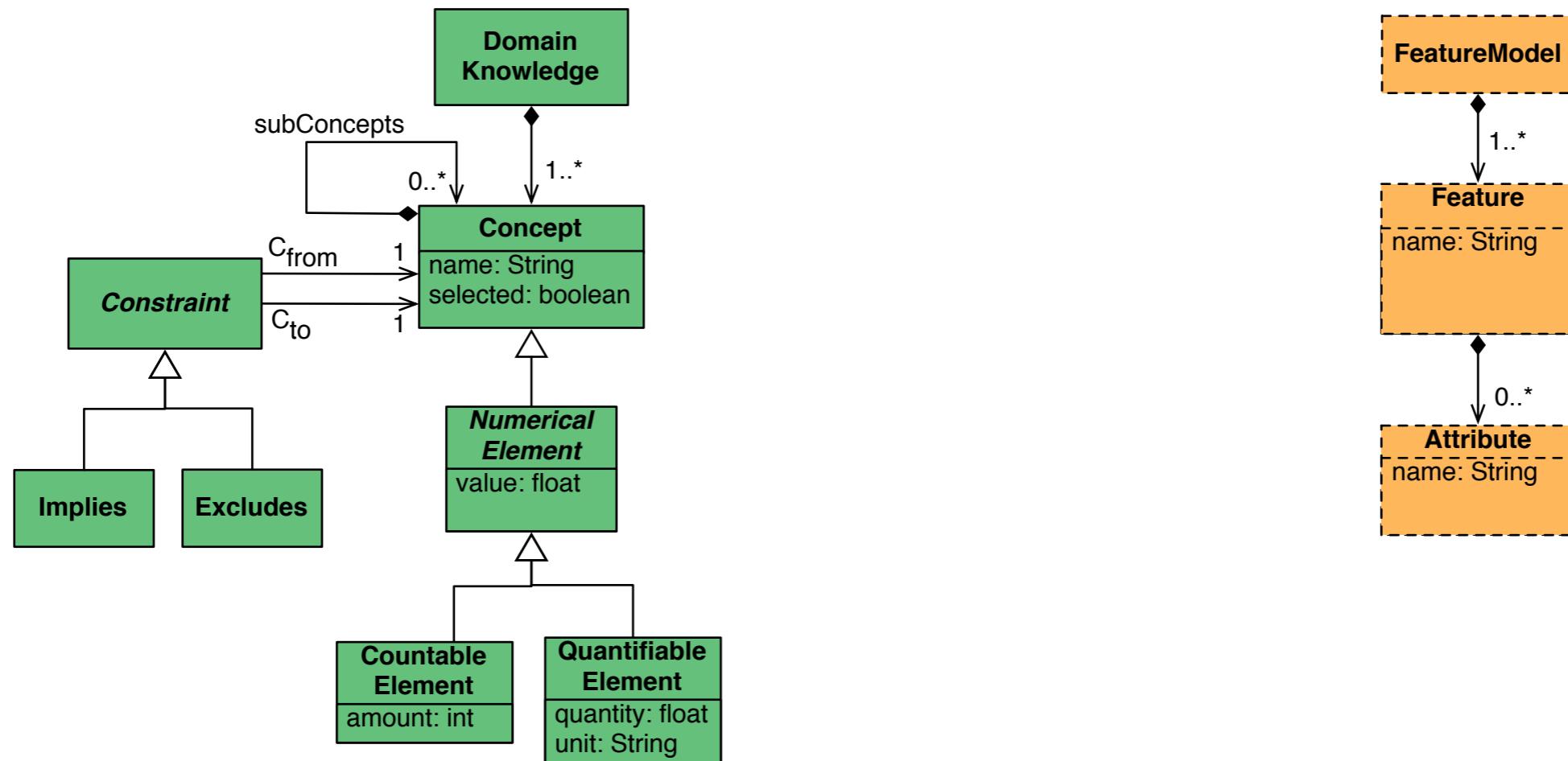
Mappings



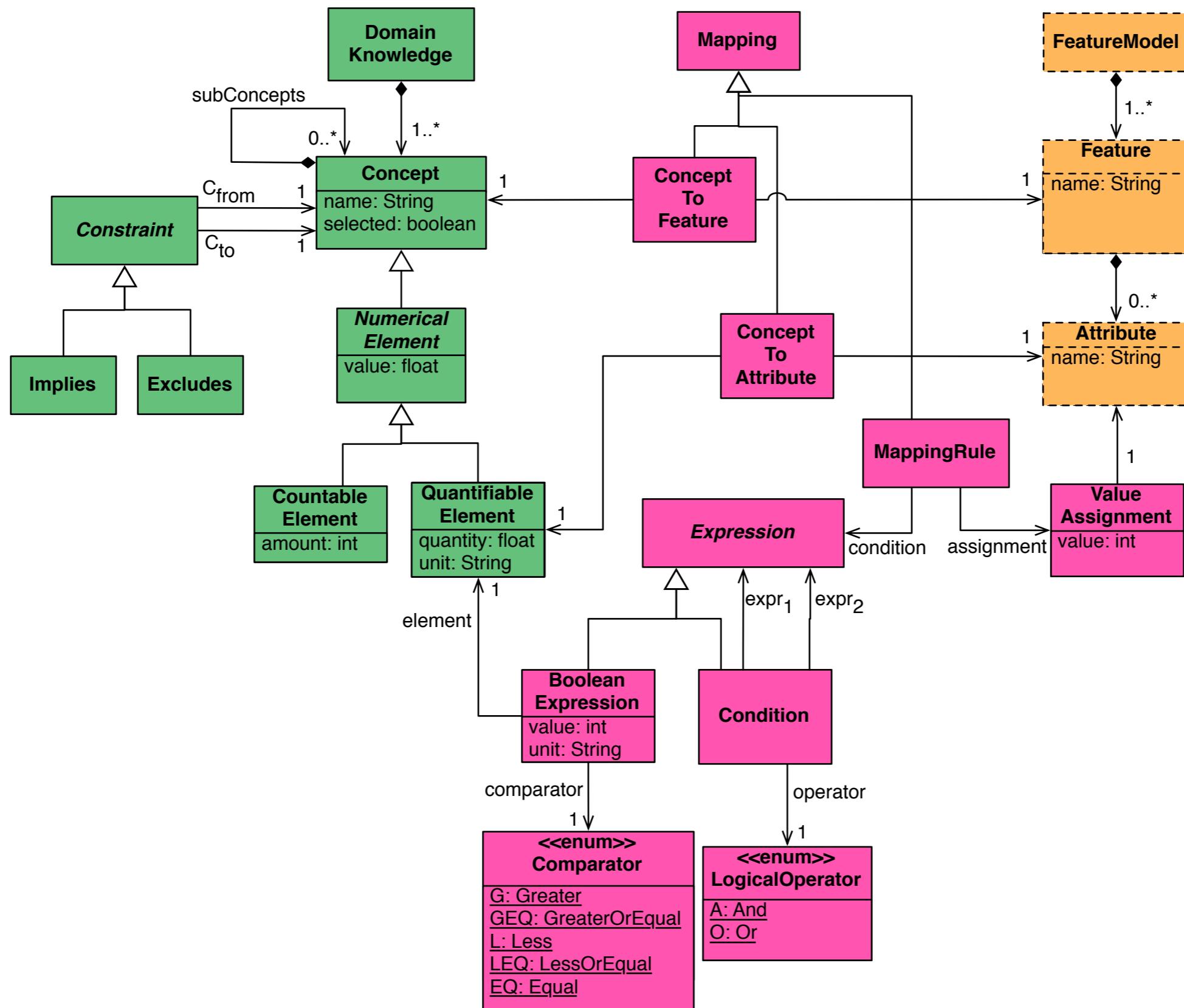
Mappings



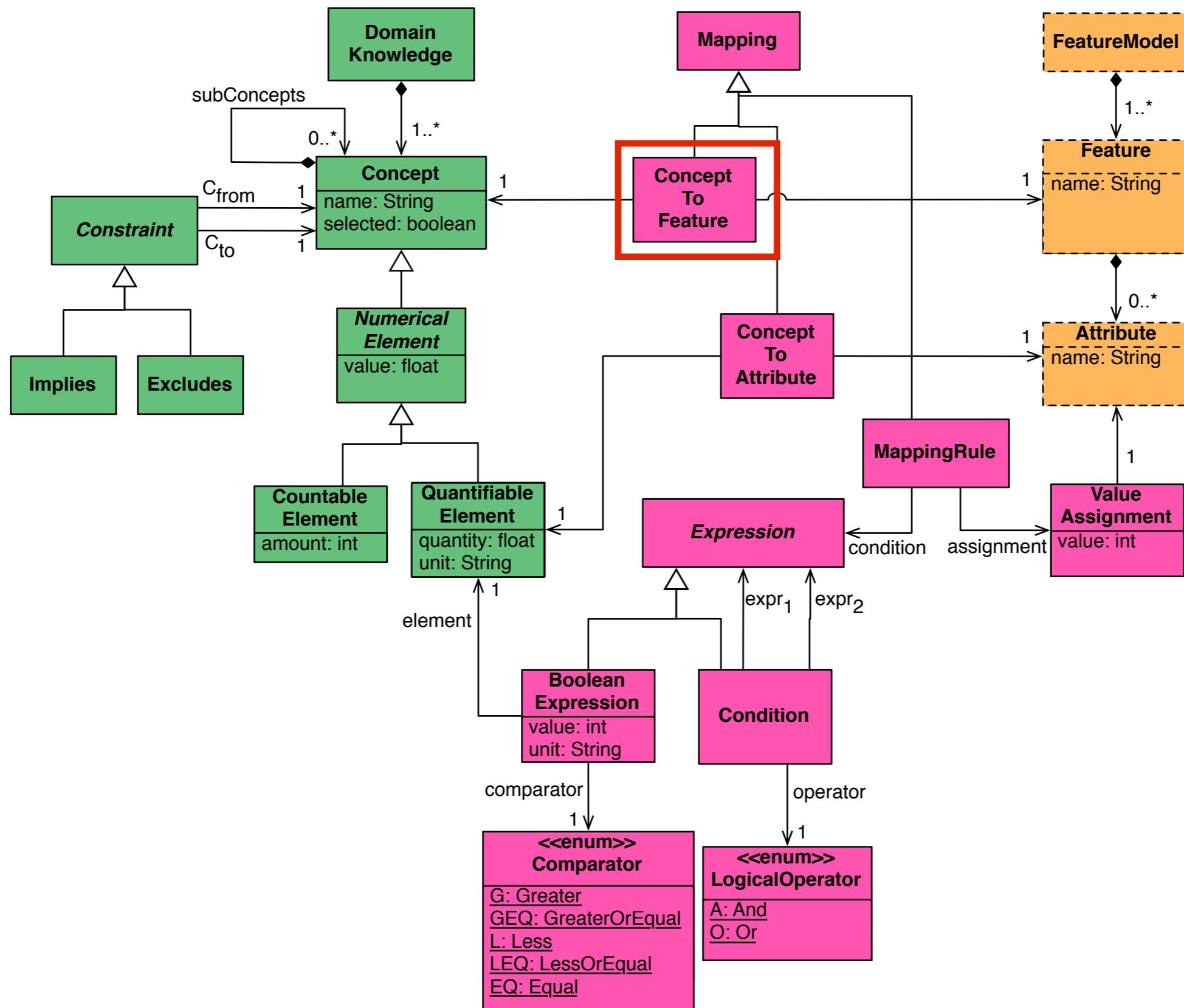
Mappings



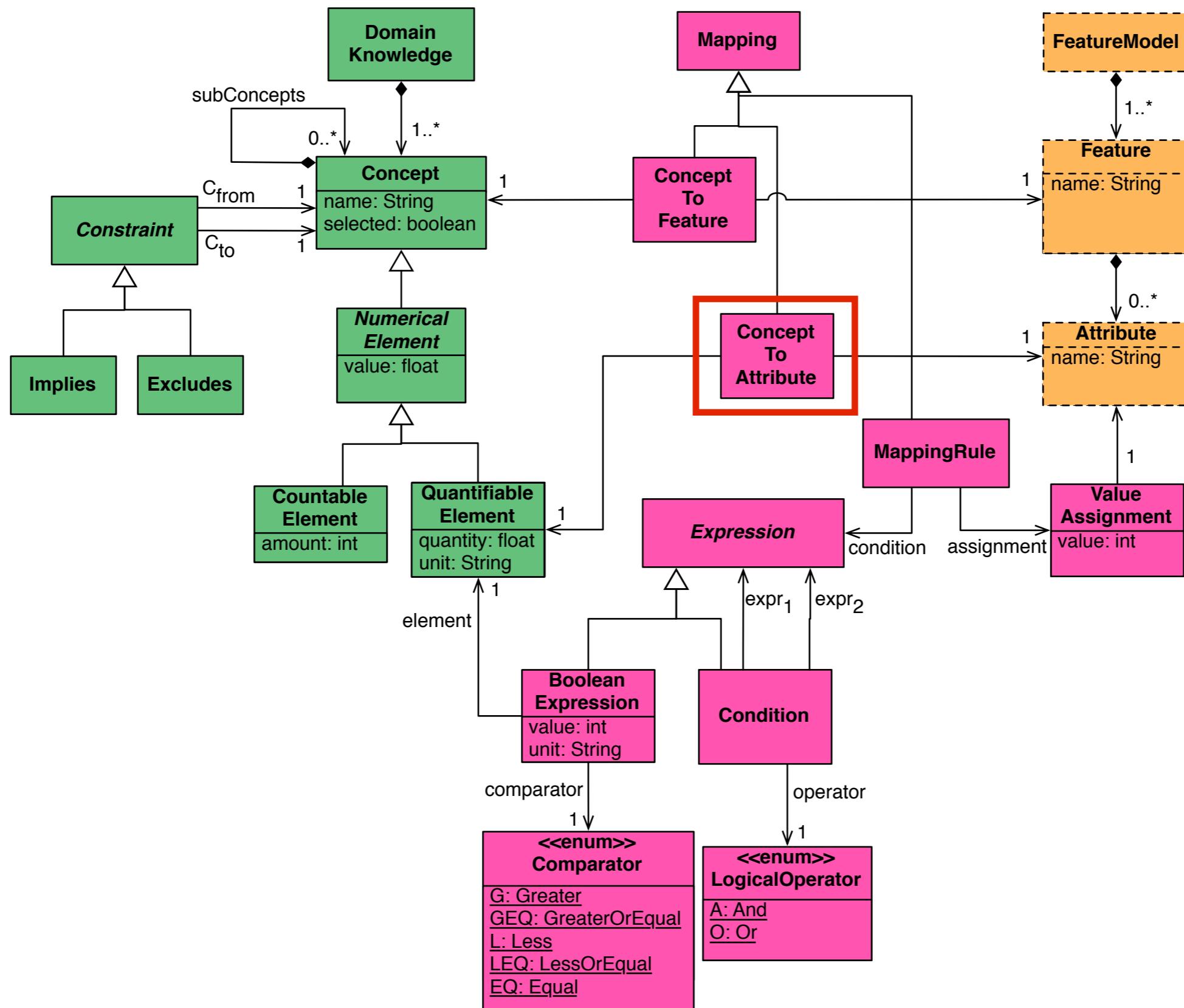
Mappings



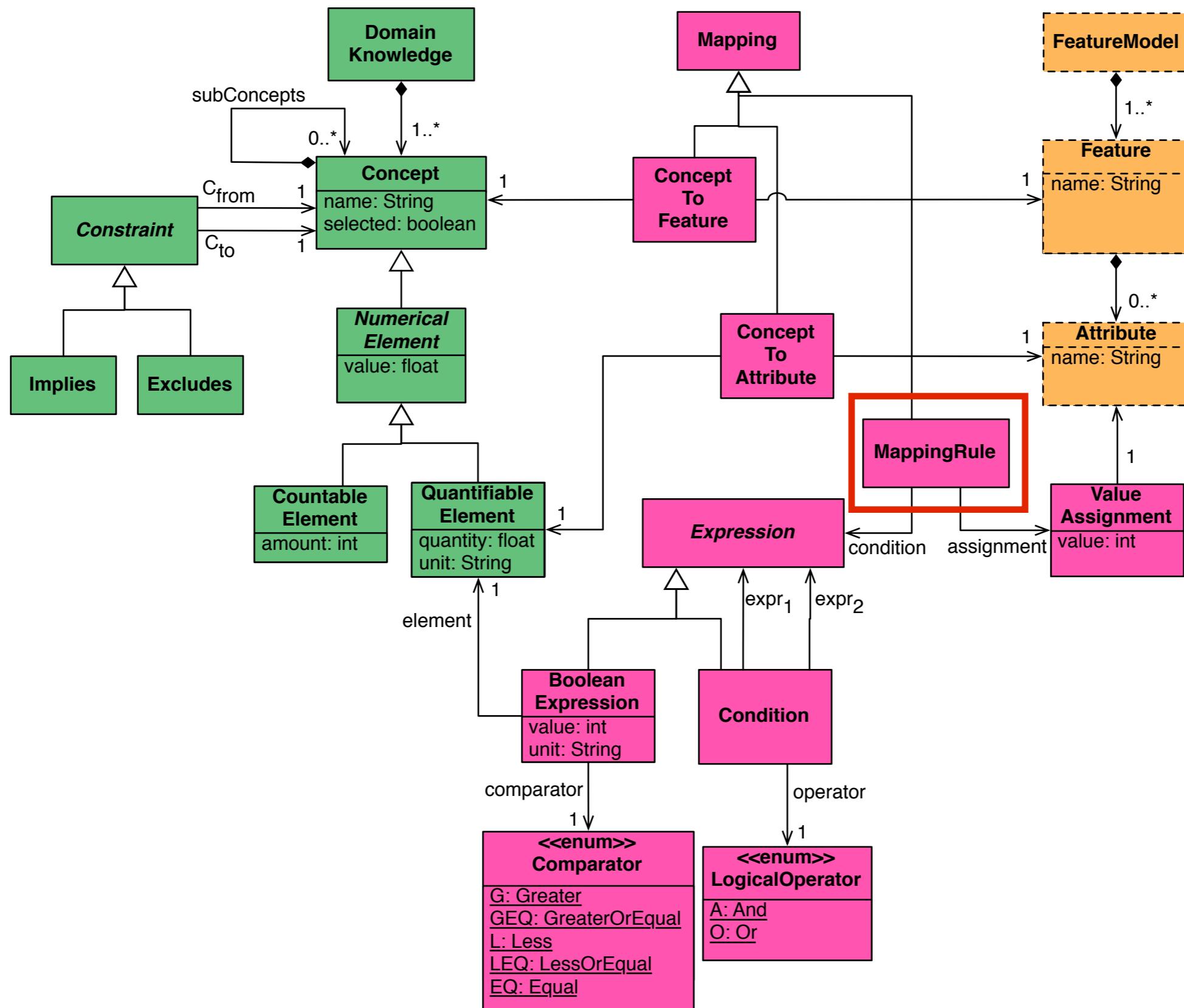
Mappings



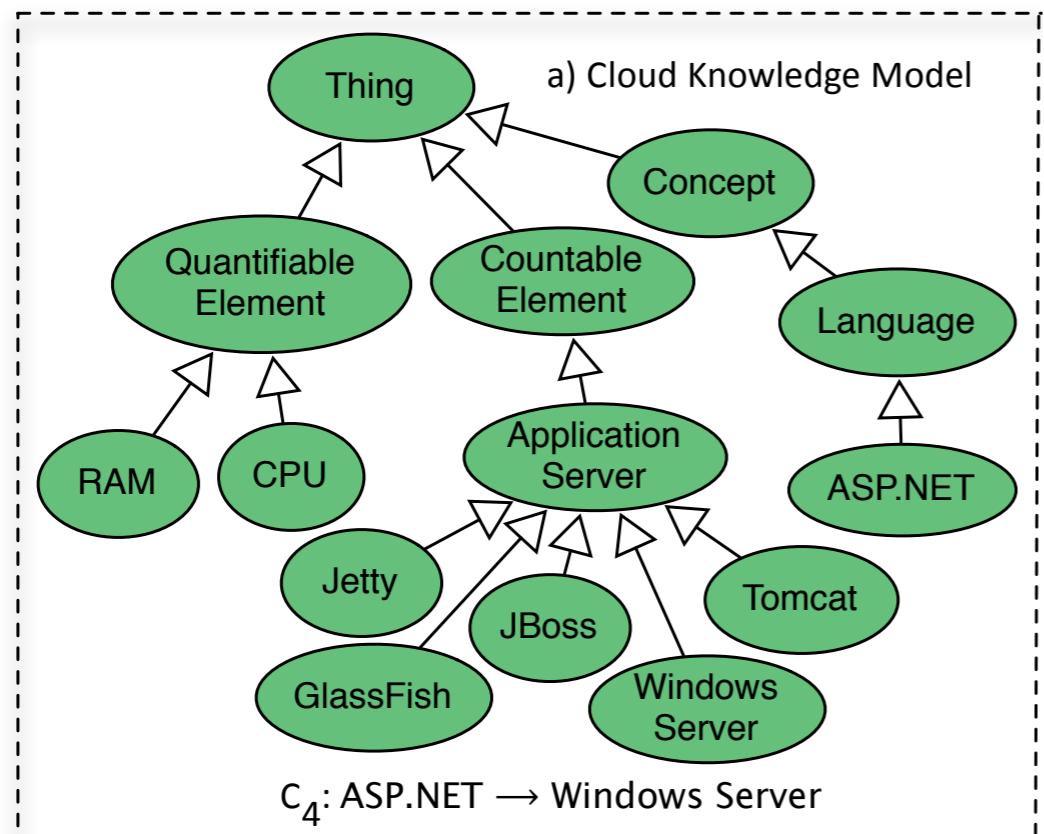
Mappings



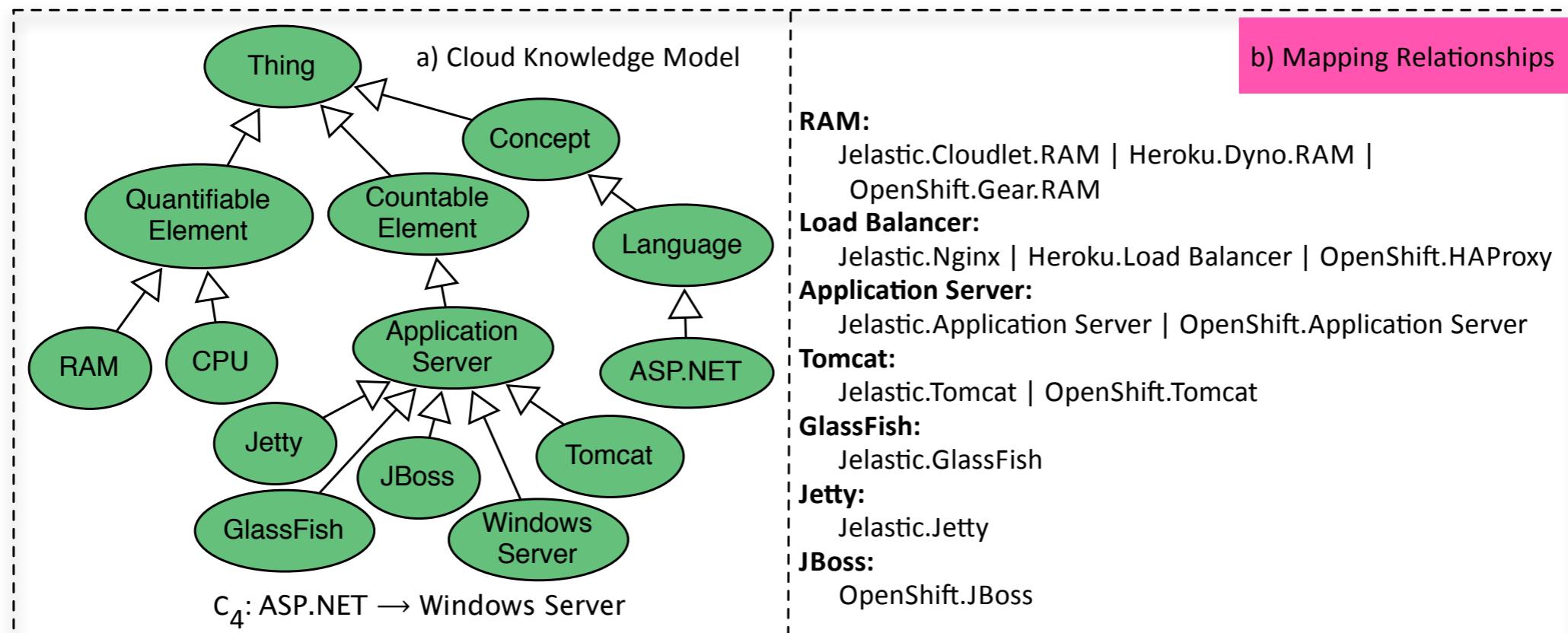
Mappings



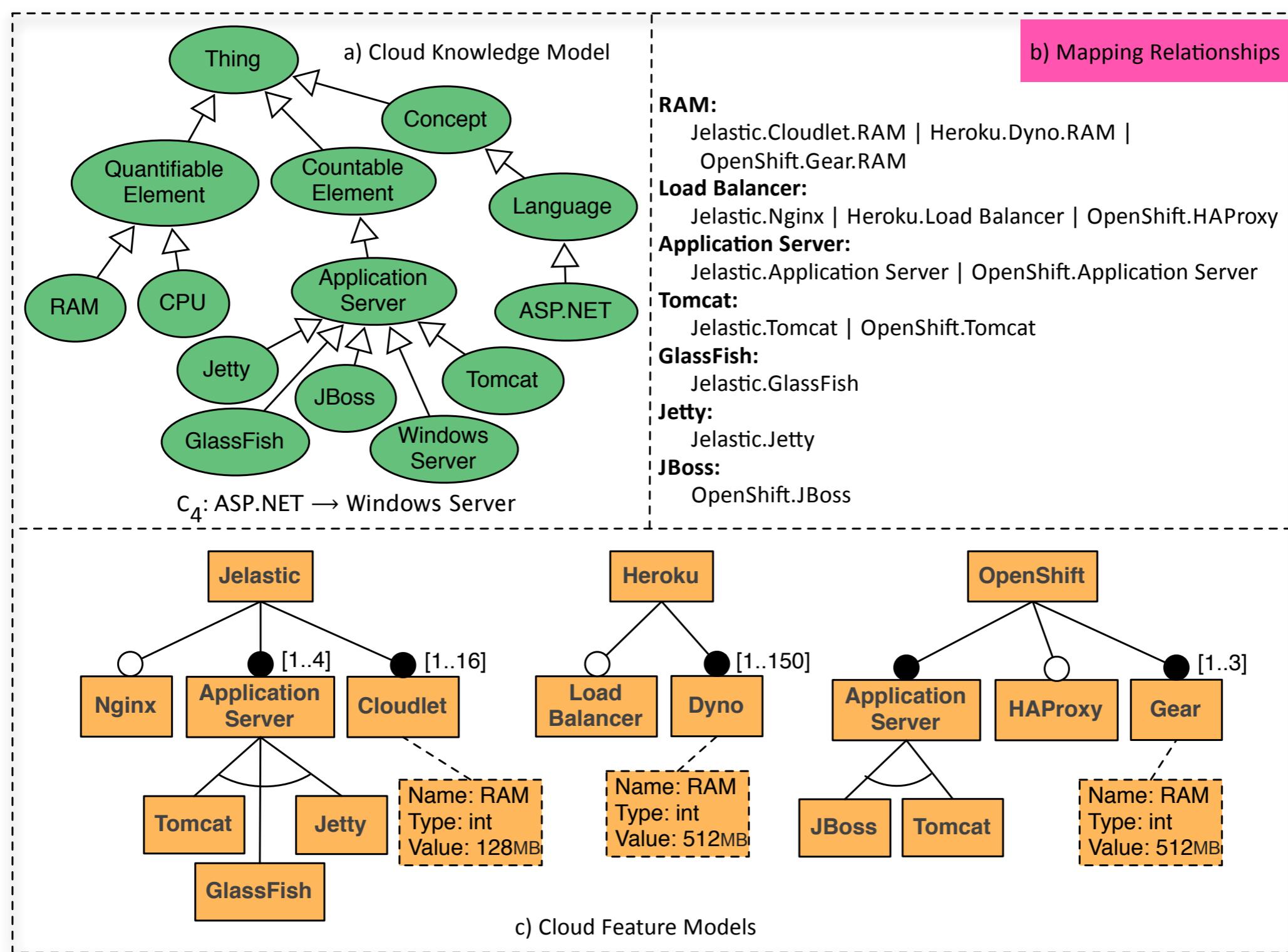
Mappings



Mappings

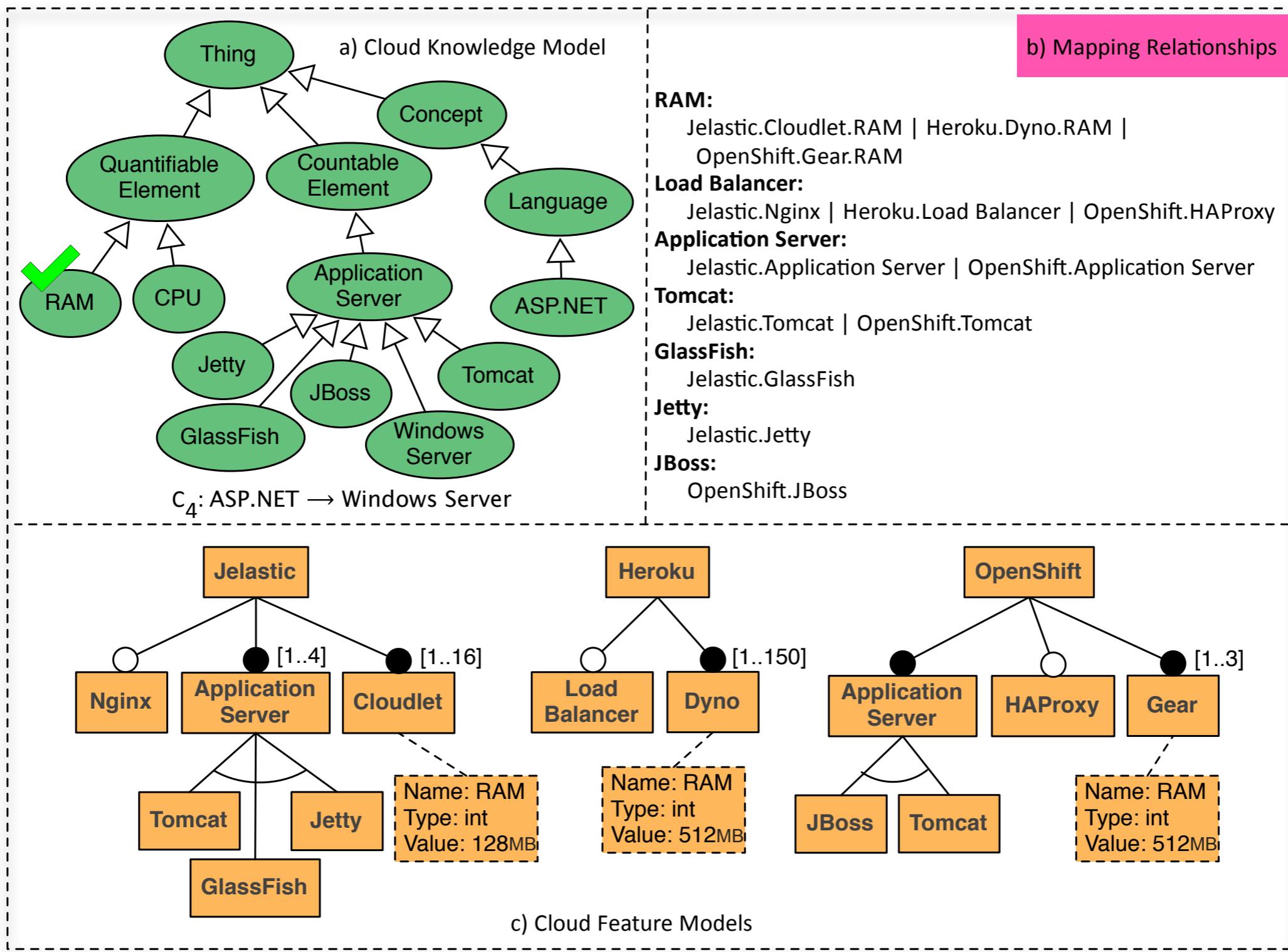


Mappings



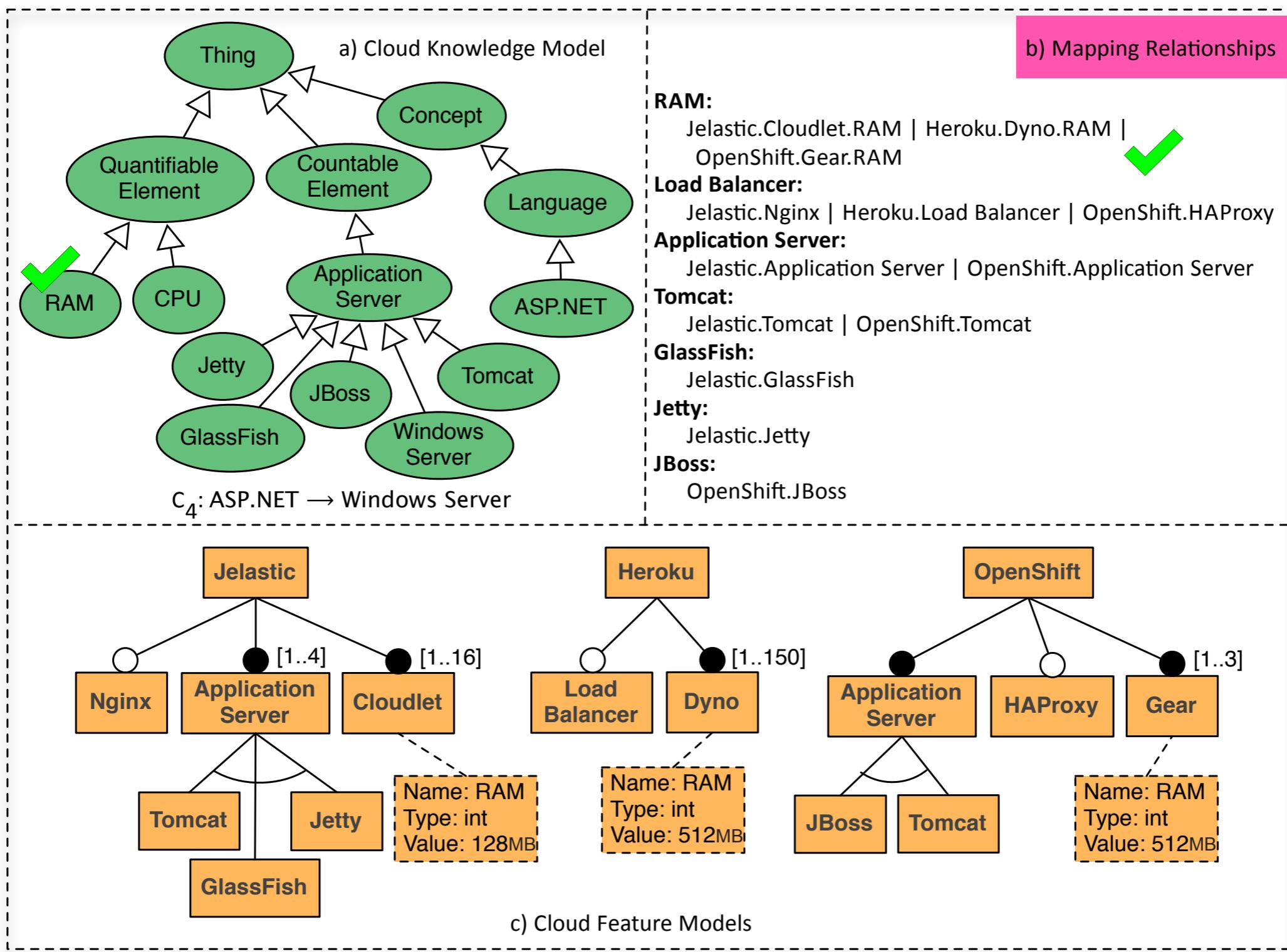
Mappings

2 GB



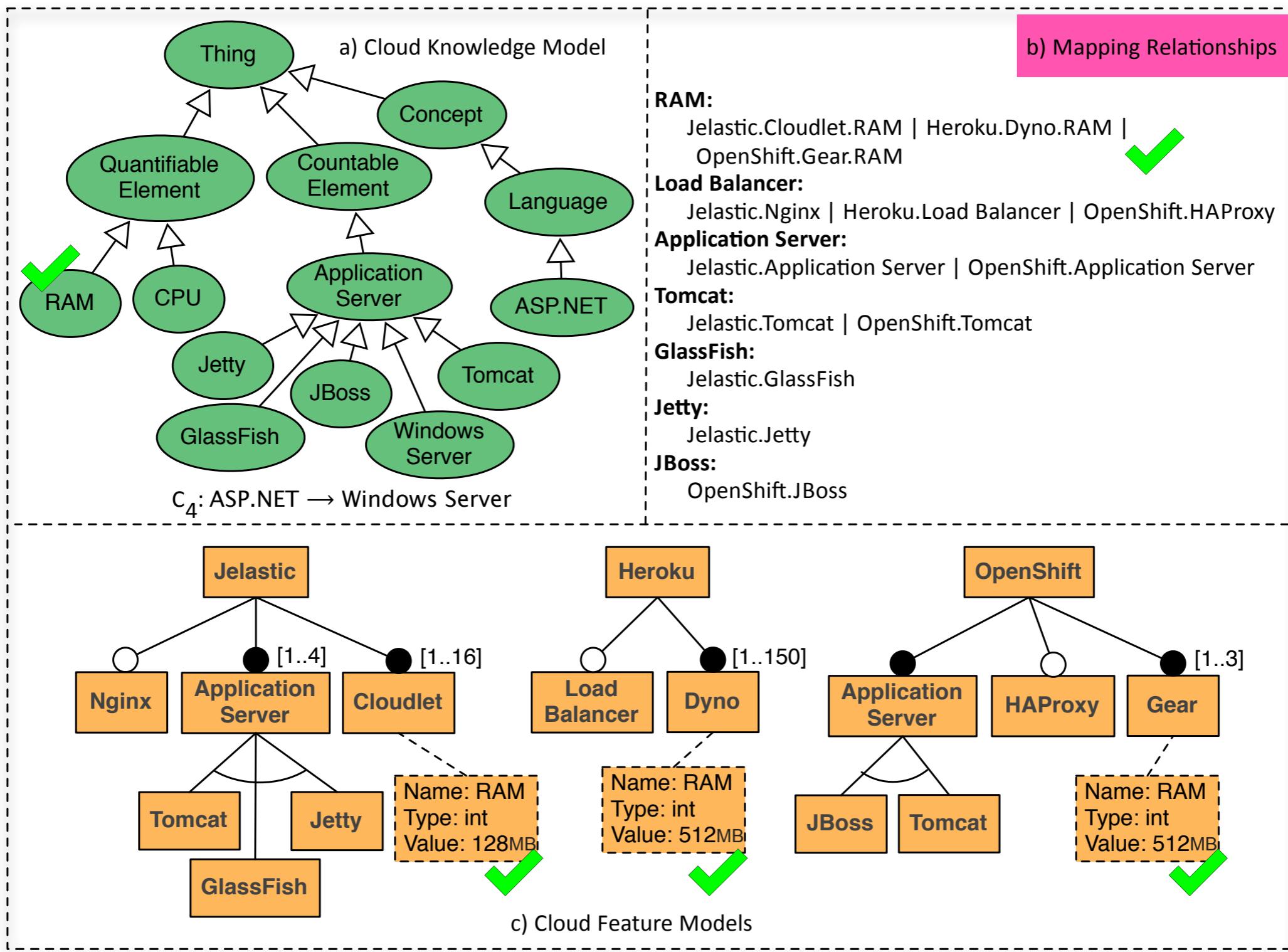
Mappings

2 GB



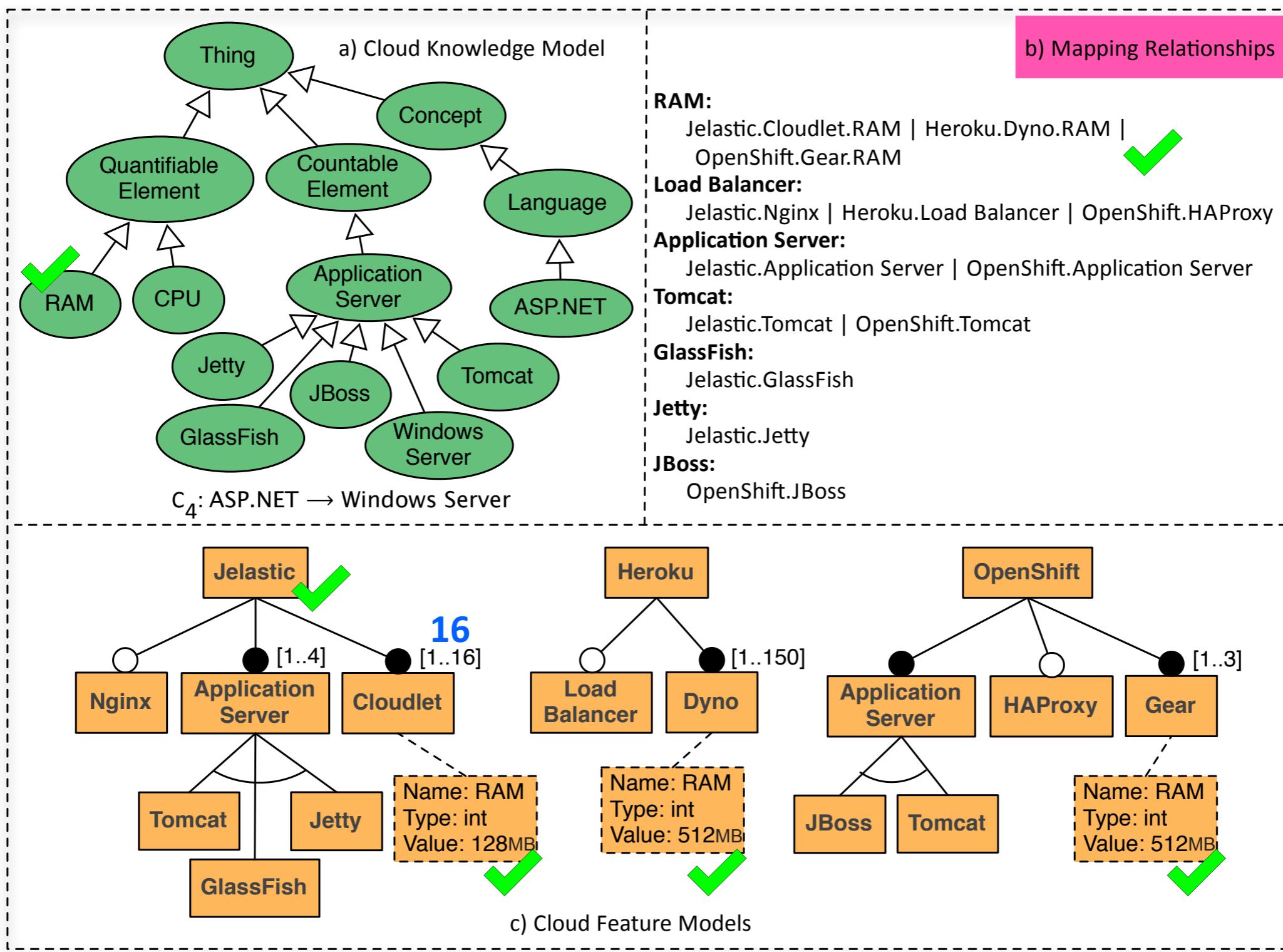
Mappings

2 GB



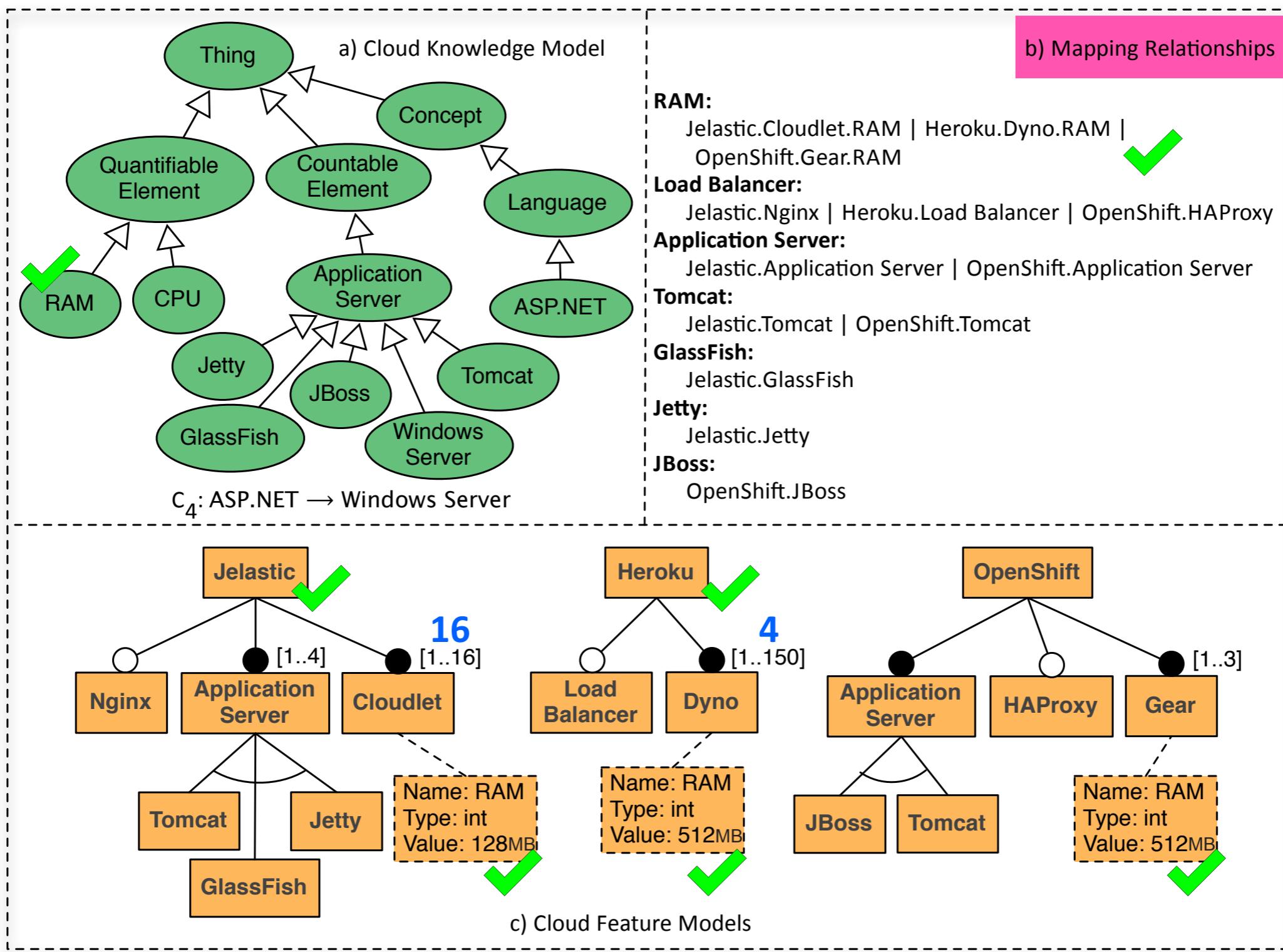
Mappings

2 GB



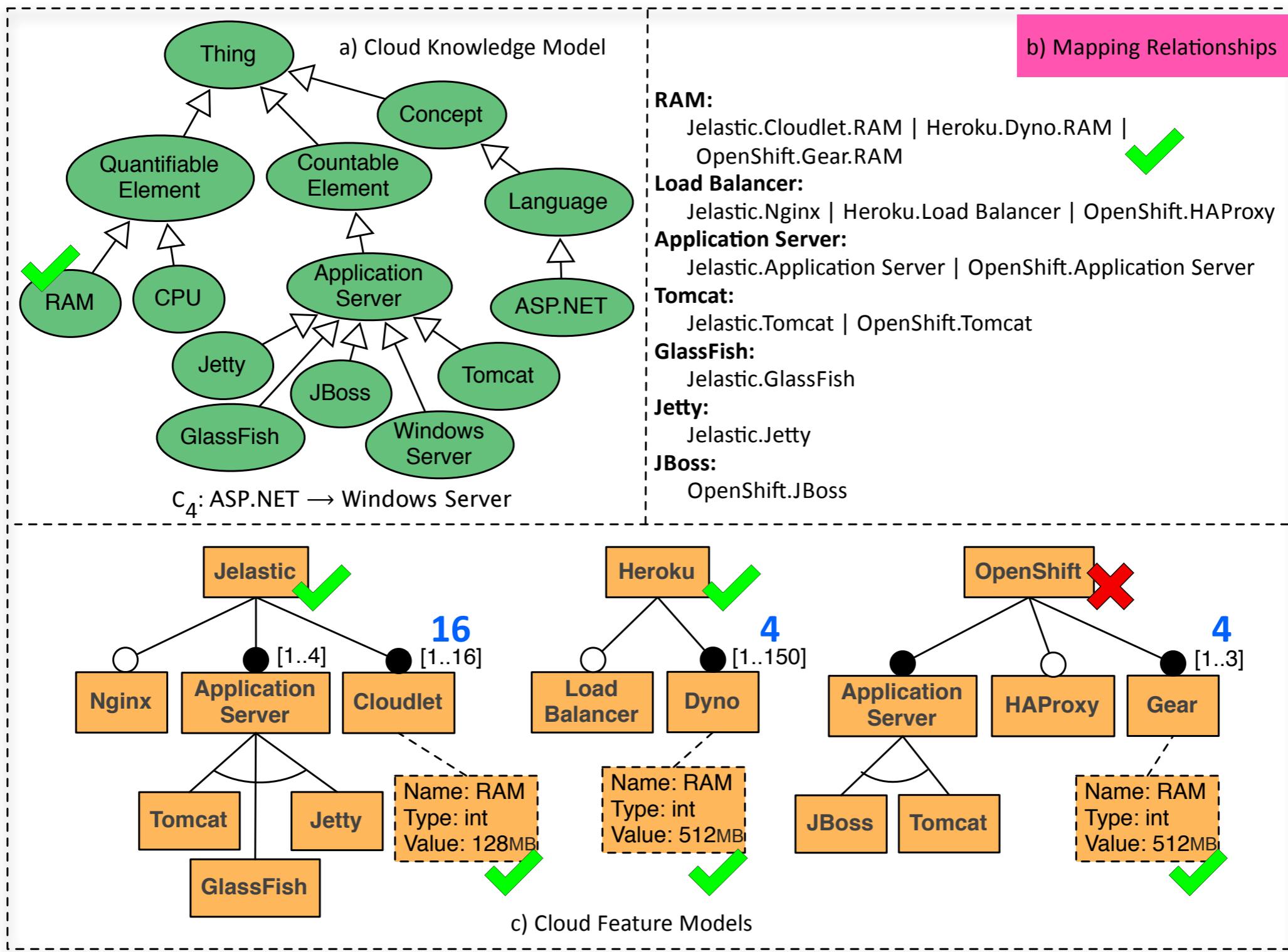
Mappings

2 GB

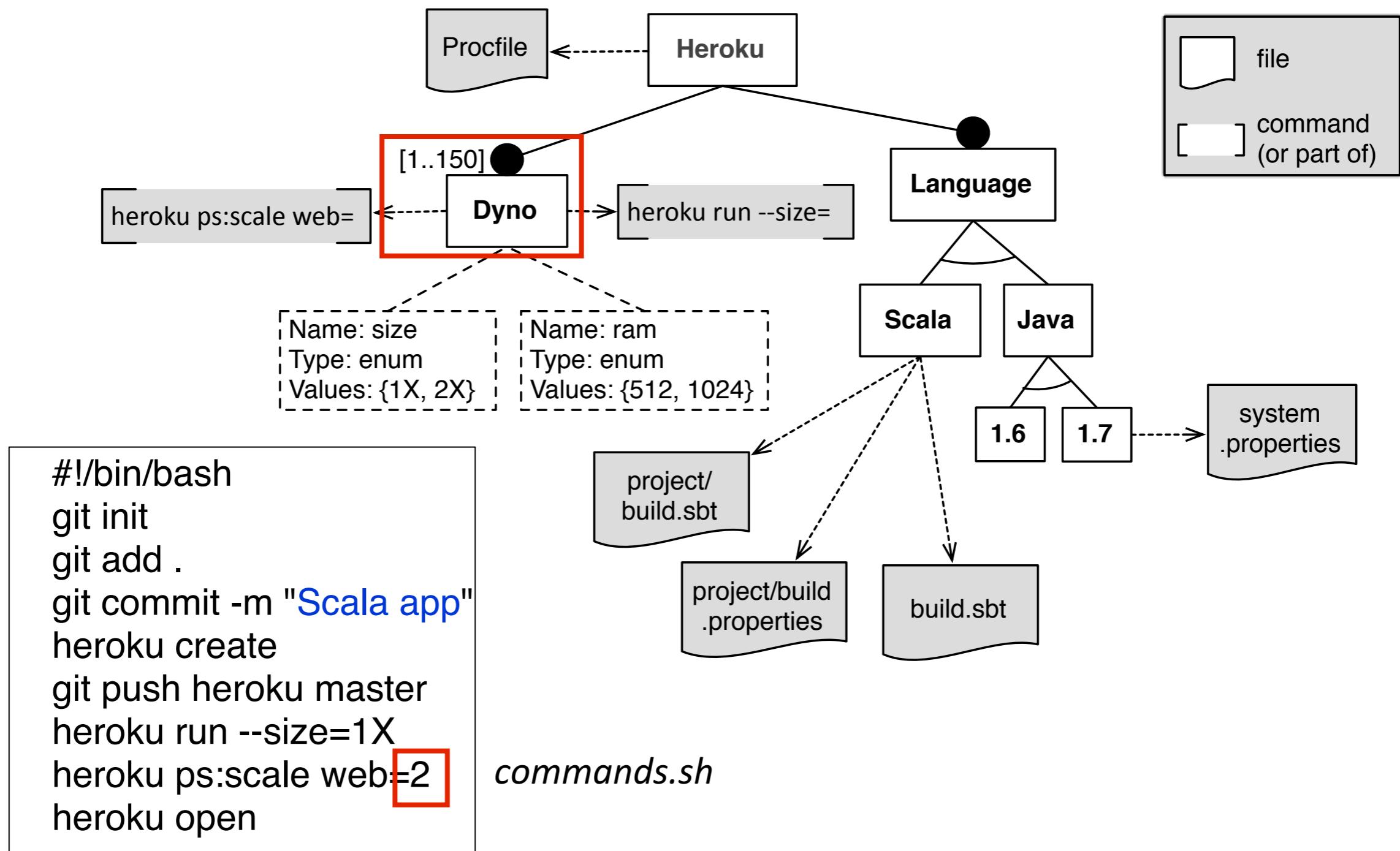


Mappings

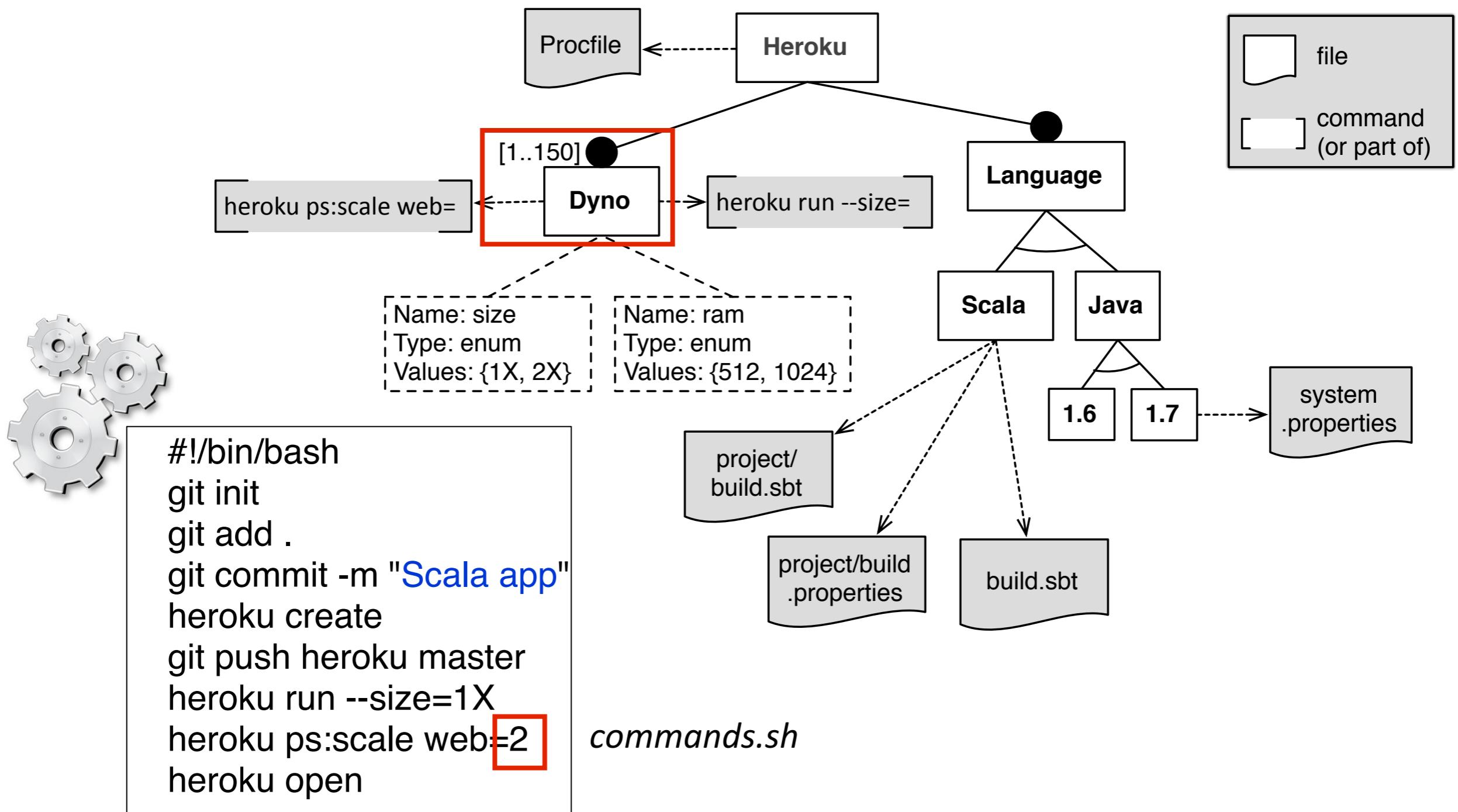
2 GB



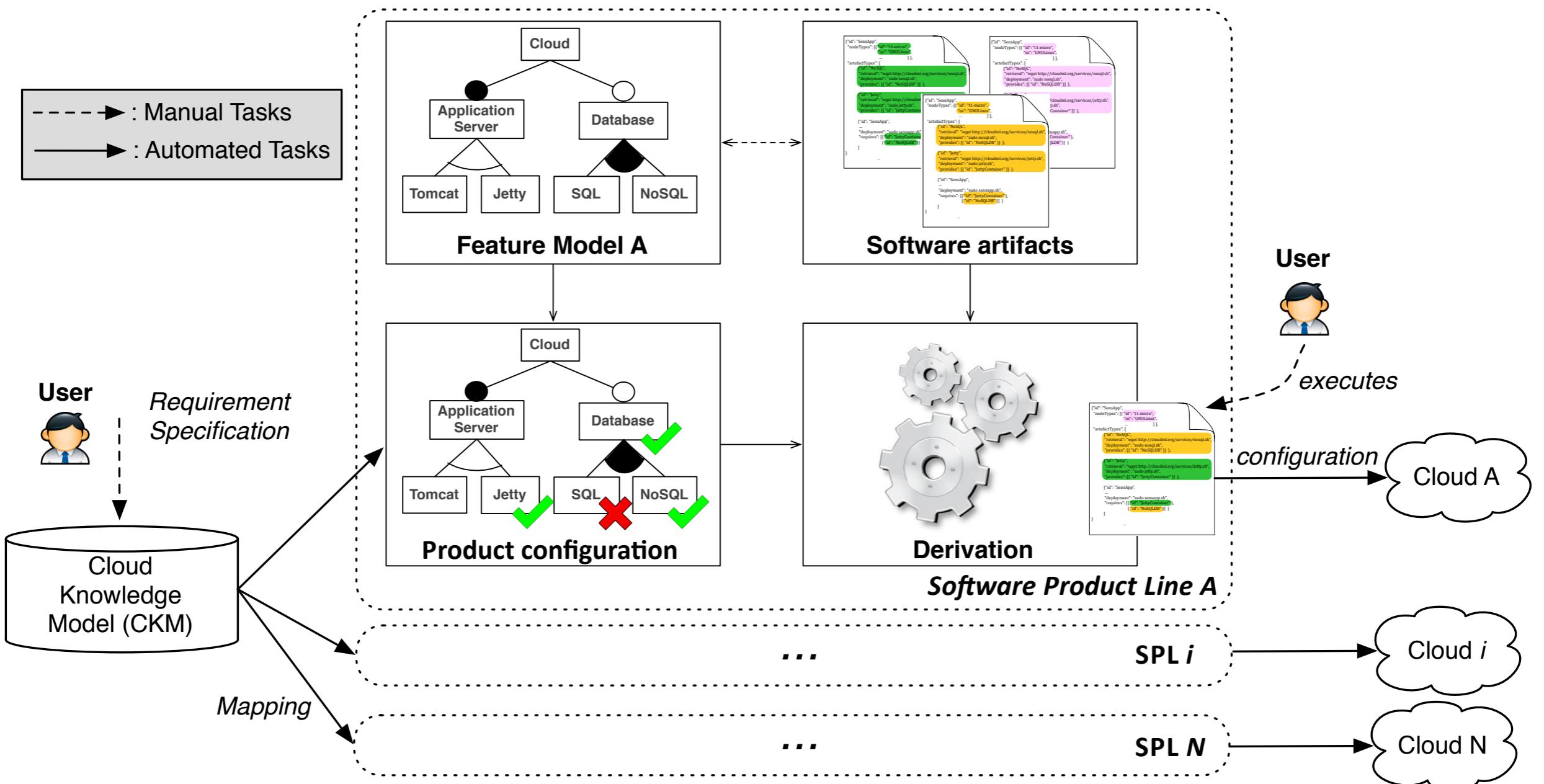
Mappings



Mappings



SALOON : Summary



Automated Selection and Configuration of Cloud Environments Using Software Product Lines Principles.
 Clément Quinton, Daniel Romero and Laurence Duchien. In Proceedings of the 7th IEEE International Conference on Cloud Computing, CLOUD'14. Anchorage, Alaska, July 2014.

Evaluation

SALOON: A Platform for Selecting and Configuring Cloud Environments.

Clément Quinton, Daniel Romero and Laurence Duchien.

In Software: Practice and Experience journal (SPE). Accepted with minor revisions, September 2014.

Evaluation

- Scalability
 - Find a valid configuration
 - From requirements specification

SALOON: A Platform for Selecting and Configuring Cloud Environments.

Clément Quinton, Daniel Romero and Laurence Duchien.

In Software: Practice and Experience journal (SPE). Accepted with minor revisions, September 2014.

Evaluation

- Scalability
 - Find a valid configuration
 - From requirements specification
- Practicality
 - Reliability
 - Ease of use

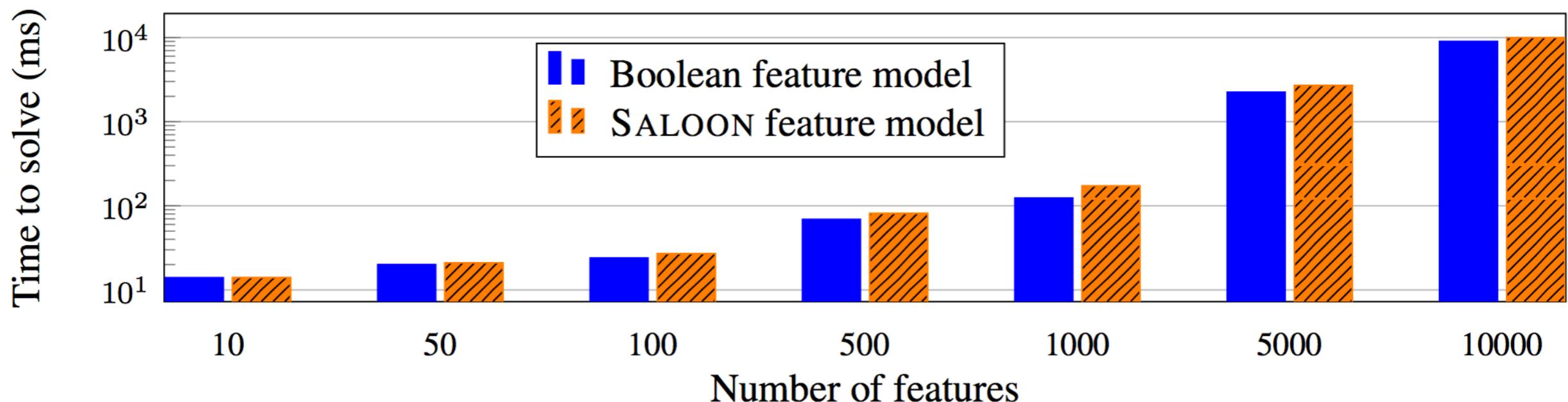
SALOON: A Platform for Selecting and Configuring Cloud Environments.

Clément Quinton, Daniel Romero and Laurence Duchien.

In Software: Practice and Experience journal (SPE). Accepted with minor revisions, September 2014.

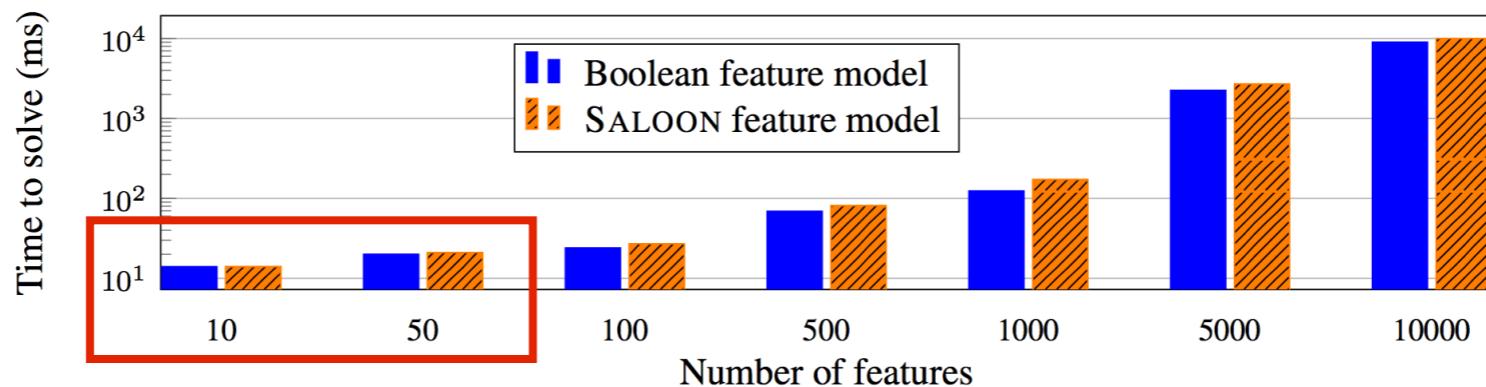
Scalability : Experiment #1

CardEx constraints overhead



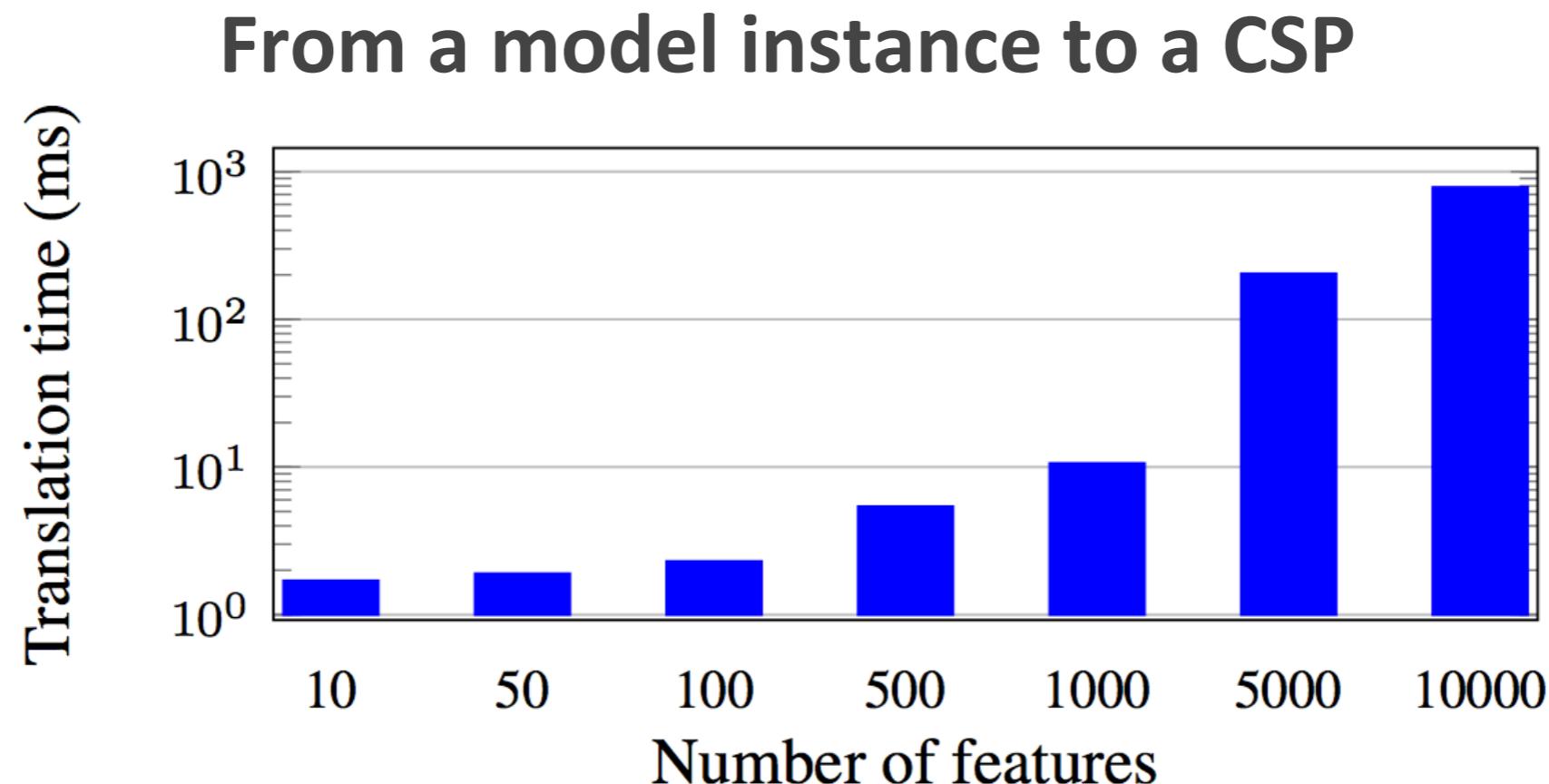
Scalability : Experiment #1

CardEx constraints overhead

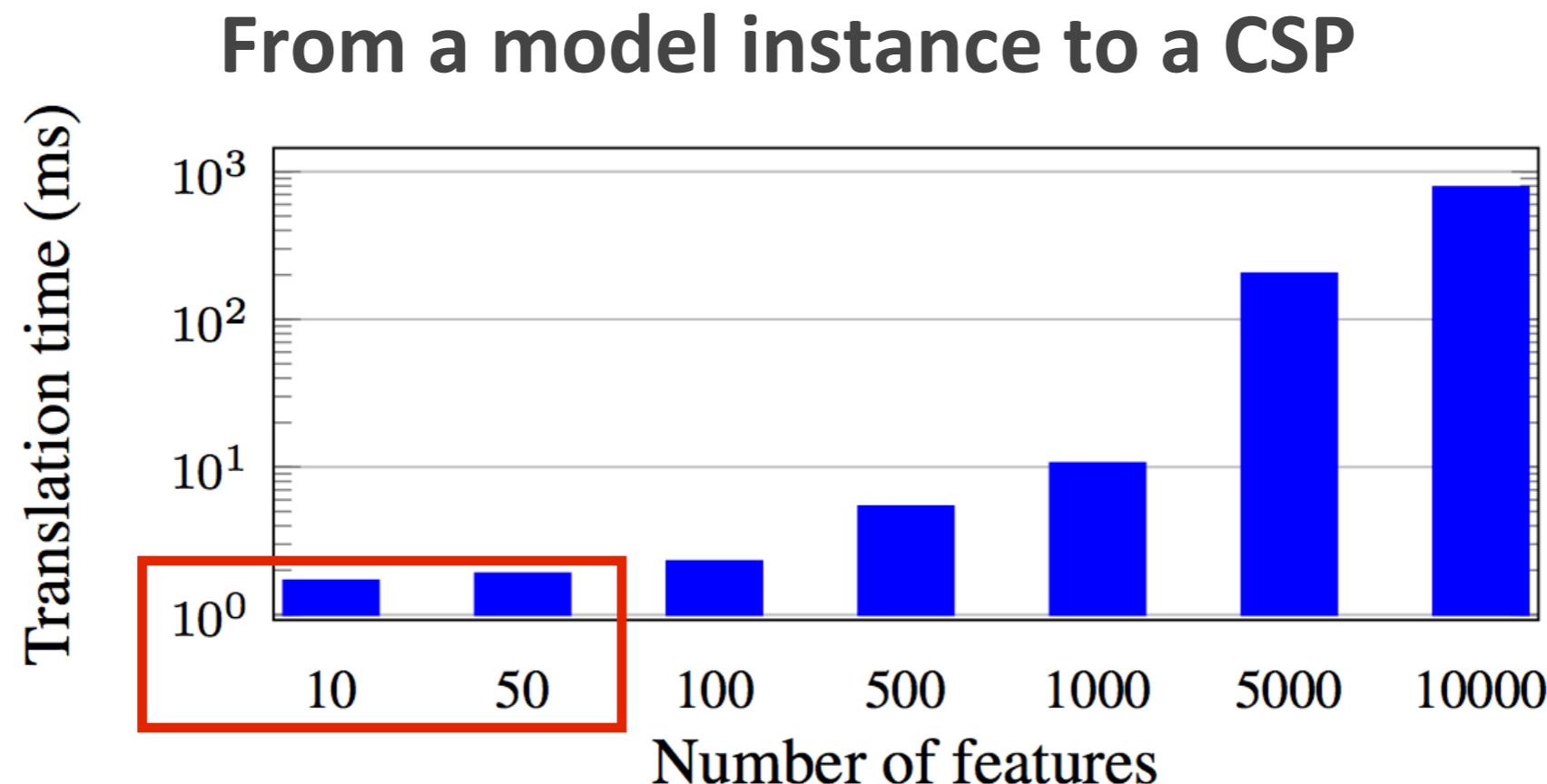


Cloud	Type	Total	Features			Constraints		
			F_{card}	F_{attr}	Attributes	Total	C_{card}	C_{attr}
Amazon EC2	IaaS	23	2	2	5	28	9	18
Cloudbees	PaaS	23	2	1	4	12	3	9
Dotcloud	PaaS	34	4	3	6	21	6	17
GoGrid	IaaS	14	3	4	10	21	7	21
Google AE	PaaS	23	1	5	13	10	0	10
Heroku	PaaS	42	1	11	20	7	0	3
Jelastic	PaaS	31	3	1	2	12	10	0
OpenShift	PaaS	29	1	2	7	18	2	15
Pagoda Box	IaaS/PaaS	28	5	5	9	8	4	8
Windows Azure	IaaS/PaaS	31	6	12	29	46	0	46

Scalability : Experiments #2 and #3

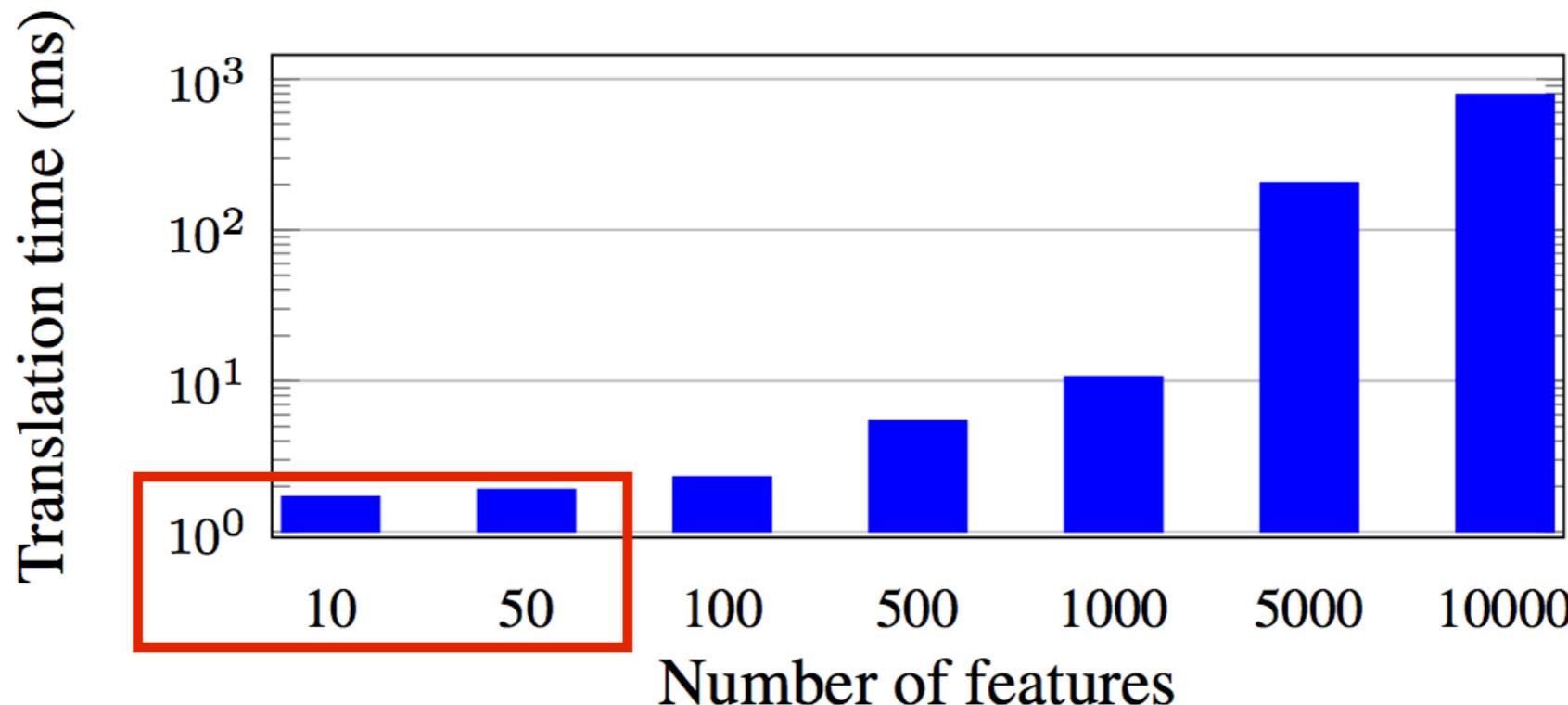


Scalability : Experiments #2 and #3



Scalability : Experiments #2 and #3

From a model instance to a CSP



From a set of requirements to a configuration

# Features < 100	Nb models	10	50	100	200
	Time (s)	1,3	2,8	3,3	4,4

Practicality : Experiment #1

Practicality : Experiment #1

Task: «*Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application*»

Practicality : Experiment #1

Task: «*Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application*»

Method: Git (G), Eclipse plugin (P) or Web interface (W)

Practicality : Experiment #1

Task: «Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application»

Method: Git (G), Eclipse plugin (P) or Web interface (W)

Participant	1	2	3	4	5	6	7	8	9	10
Time (min)	26	19	32	26	48	60	17	-	23	28
Method	G	G	G	P	P	G	G	G	W	G
Experience	2	4	2	3	3	1	4	1	3	2
App running	✓	✓	-	✓	✓	-	-	-	✓	-

Practicality : Experiment #1

Task: «Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application»

Method: Git (G), Eclipse plugin (P) or Web interface (W)

Participant	1	2	3	4	5	6	7	8	9	10
Time (min)	26	19	32	26	48	60	17	-	23	28
Method	G	G	G	P	P	G	G	G	W	G
Experience	2	4	2	3	3	1	4	1	3	2
App running	✓	✓	-	✓	✓	-	-	-	✓	-

50% fail

Practicality : Experiment #1

Task: «Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application»

Method: Git (G), Eclipse plugin (P) or Web interface (W)

Participant	1	2	3	4	5	6	7	8	9	10
Time (min)	26	19	32	26	48	60	17	-	23	28
Method	G	G	G	P	P	G	G	G	W	G
Experience	2	4	2	3	3	1	4	1	3	2
App running	✓	✓	-	✓	✓	-	-	-	✓	-

50% fail

Practicality : Experiment #1

Task: «Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application»

Method: Git (G), Eclipse plugin (P) or Web interface (W)

Participant	1	2	3	4	5	6	7	8	9	10
Time (min)	26	19	32	26	48	60	17	-	23	28
Method	G	G	G	P	P	G	G	G	W	G
Experience	2	4	2	3	3	1	4	1	3	2
App running	✓	✓	-	✓	✓	-	-	-	✓	-

50% fail

Practicality : Experiment #1

Task: «Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application»

Method: Git (G), Eclipse plugin (P) or Web interface (W)

Participant	1	2	3	4	5	6	7	8	9	10	
Time (min)	26	19	32	26	48	60	17	-	23	28	Avg: +28
Method	G	G	G	P	P	G	G	G	W	G	
Experience	2	4	2	3	3	1	4	1	3	2	
App running	✓	✓	-	✓	✓	-	-	-	✓	-	

Practicality : Experiment #2

Task: «Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application»

Method: SALOON

Participant	1	2	3	4	5	6	7	8
Time (min)	4	6	7	5	8	11	12	7
Method		SALOON				SALOON		
Experience	4	1	4	1	2	2	2	3
App running	✓	✓	✓	✓	✓	✓	✓	✓

Practicality : Experiment #2

Task: «Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application»

Method: SALOON

Participant	1	2	3	4	5	6	7	8	New users
Time (min)	4	6	7	5	8	11	12	7	
Method			SALOON			SALOON			
Experience	4	1	4	1	2	2	2	3	
App running	✓	✓	✓	✓	✓	✓	✓	✓	

Practicality : Experiment #2

Task: «Configure an Heroku environment and add a PostgreSQL support, then deploy the provided application»

Method: SALOON

Participant	1	2	3	4	5	6	7	8	
Time (min)	4	6	7	5	8	11	12	7	Average: 8
Method		SALOON				SALOON			
Experience	4	1	4	1	2	2	2	3	
App running	✓	✓	✓	✓	✓	✓	✓	✓	

Practicality

Participant	1	2	3	4	5	6	7	8	9	10
Time (min)	26	19	32	26	48	60	17	-	23	28
Method	G	G	G	P	P	G	G	G	W	G
Experience	2	4	2	3	3	1	4	1	3	2
App running	✓	✓	-	✓	✓	-	-	-	✓	-

Participant	1	2	3	4	5	6	7	8
Time (min)	4	6	7	5	8	11	12	7
Method			SALOON			SALOON		
Experience	4	1	4	1	2	2	2	3
App running	✓	✓	✓	✓	✓	✓	✓	✓

Practicality

Participant	1	2	3	4	5	6	7	8	9	10
Time (min)	26	19	32	26	48	60	17	-	23	28
Method	G	G	G	P	P	G	G	G	W	G
Experience	2	4	2	3	3	1	4	1	3	2
App running	✓	✓	-	✓	✓	-	-	-	✓	-

Participant	1	2	3	4	5	6	7	8
Time (min)	4	6	7	5	8	11	12	7
Method	SALOON					SALOON		
Experience	4	1	4	1	2	2	2	3
App running	✓	✓	✓	✓	✓	✓	✓	✓

- Decrease time (72%)
- Improve reliability

Research Goals

- Manage Cloud variability
- Guarantee environment independance
- Provide a flexible solution
- Deliver an automated support
- Maintain consistency



Research Goals

- Manage Cloud variability
- Guarantee environment independance
- Provide a flexible solution
- Deliver an automated support
- Maintain consistency



Research Goals

- Manage Cloud variability 
- Guarantee environment independance 
- Provide a flexible solution 
- Deliver an automated support
- Maintain consistency

Research Goals

- Manage Cloud variability 
- Guarantee environment independance 
- Provide a flexible solution 
- Deliver an automated support 
- Maintain consistency

Agenda

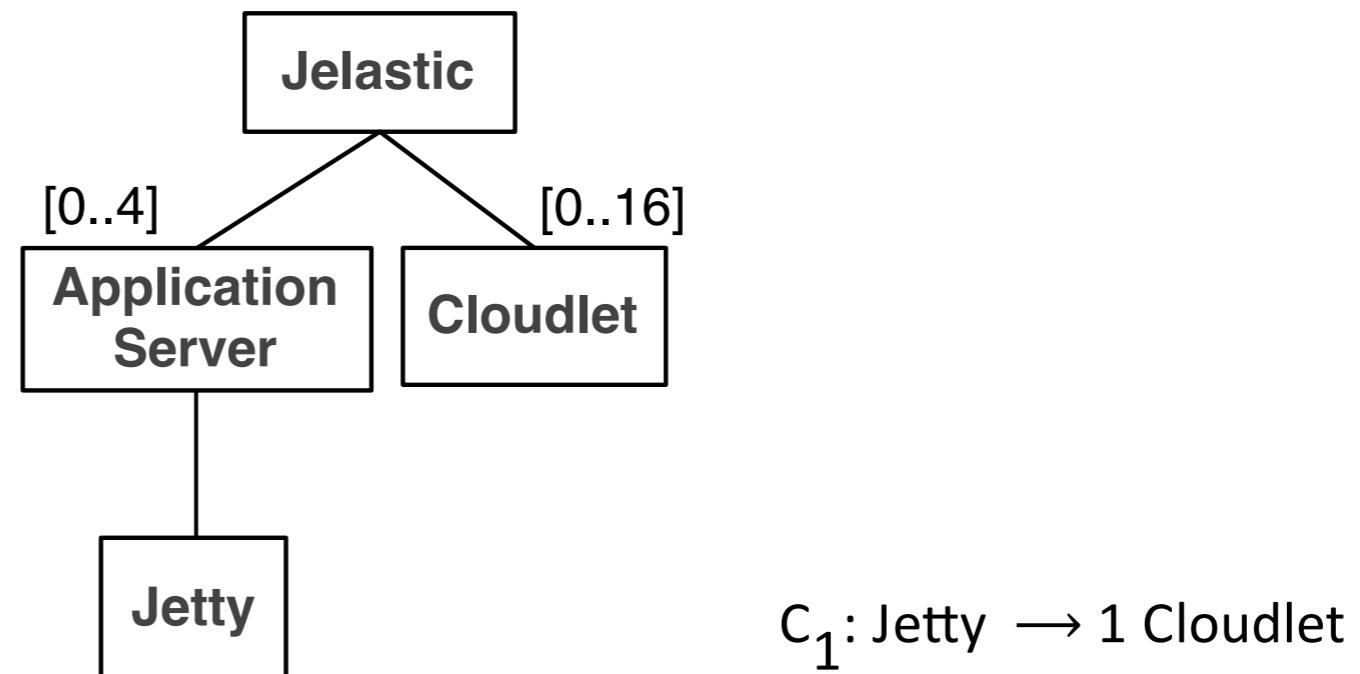
I. Introduction

II. Contributions

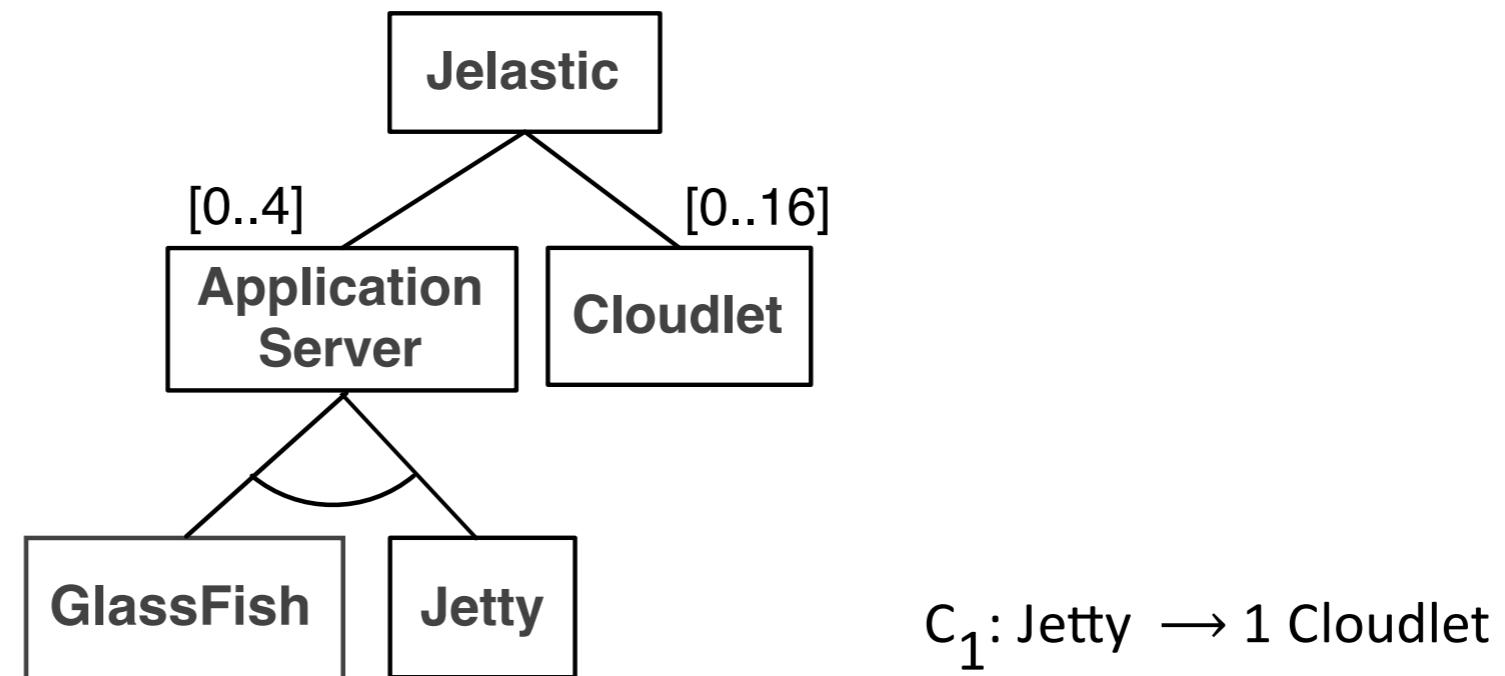
- Cloud environments variability modeling
- SALOON
- Consistency checking for evolving Cloud models

III. Conclusion and Perspectives

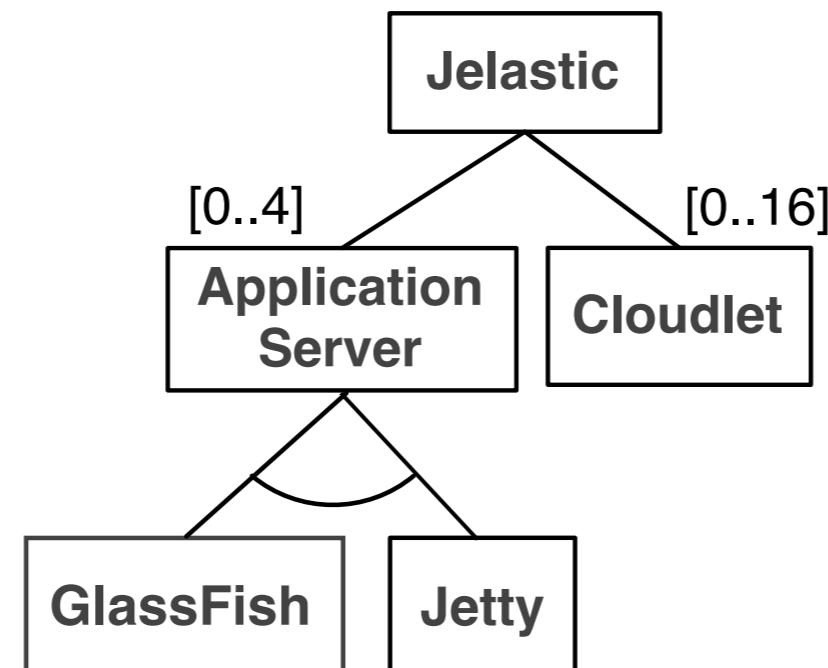
Motivating Example



Motivating Example



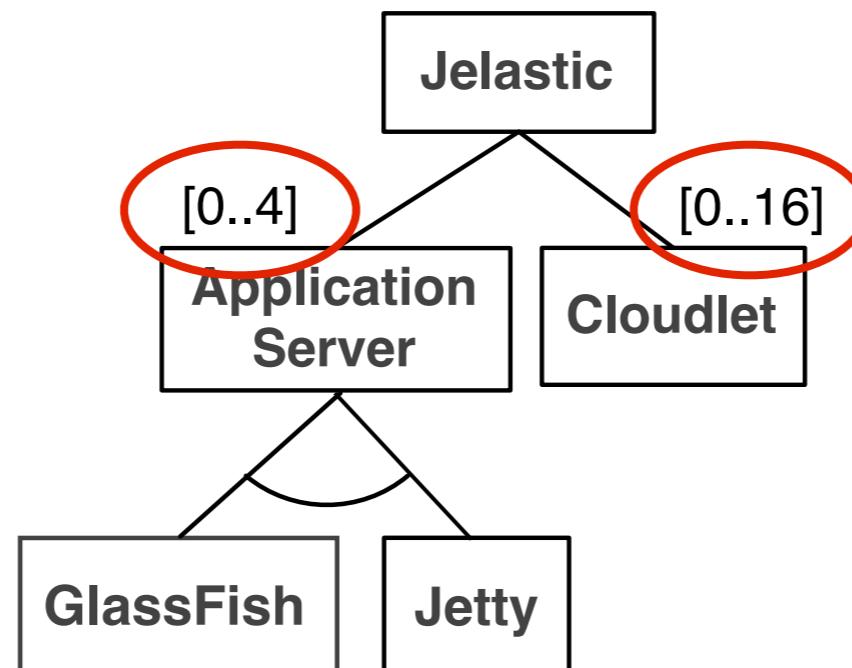
Motivating Example



$C_1: \text{Jetty} \rightarrow 1 \text{ Cloudlet}$

$C_2: \text{GlassFish} \rightarrow 5 \text{ Cloudlet}$

Motivating Example

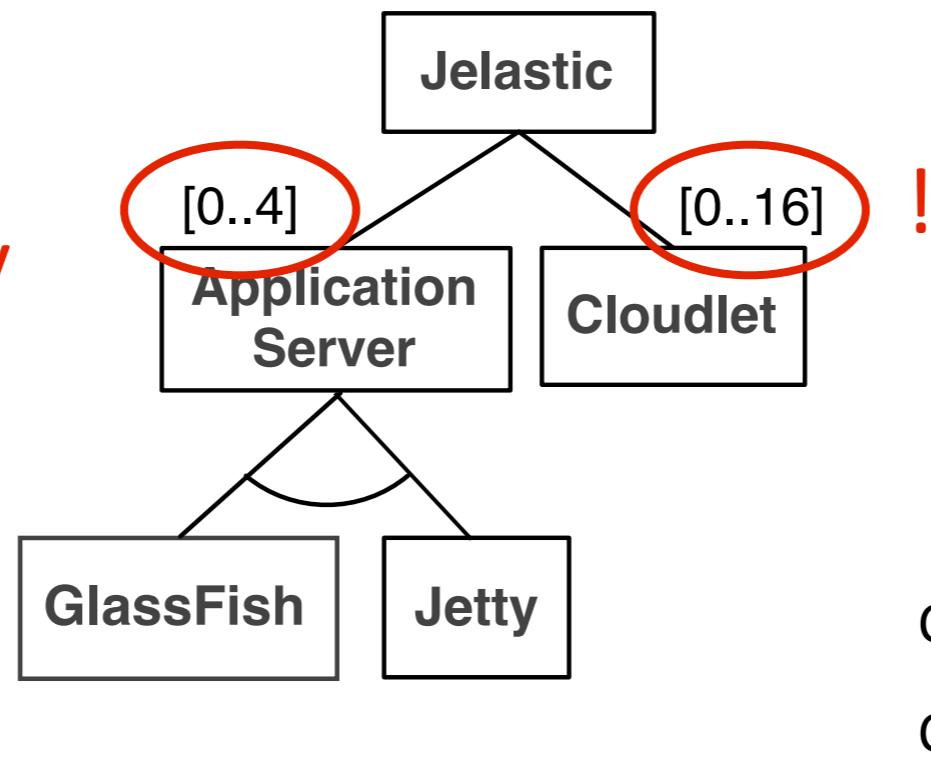


$C_1: \text{Jetty} \rightarrow 1 \text{ Cloudlet}$

$C_2: \text{GlassFish} \rightarrow 5 \text{ Cloudlet}$

Motivating Example

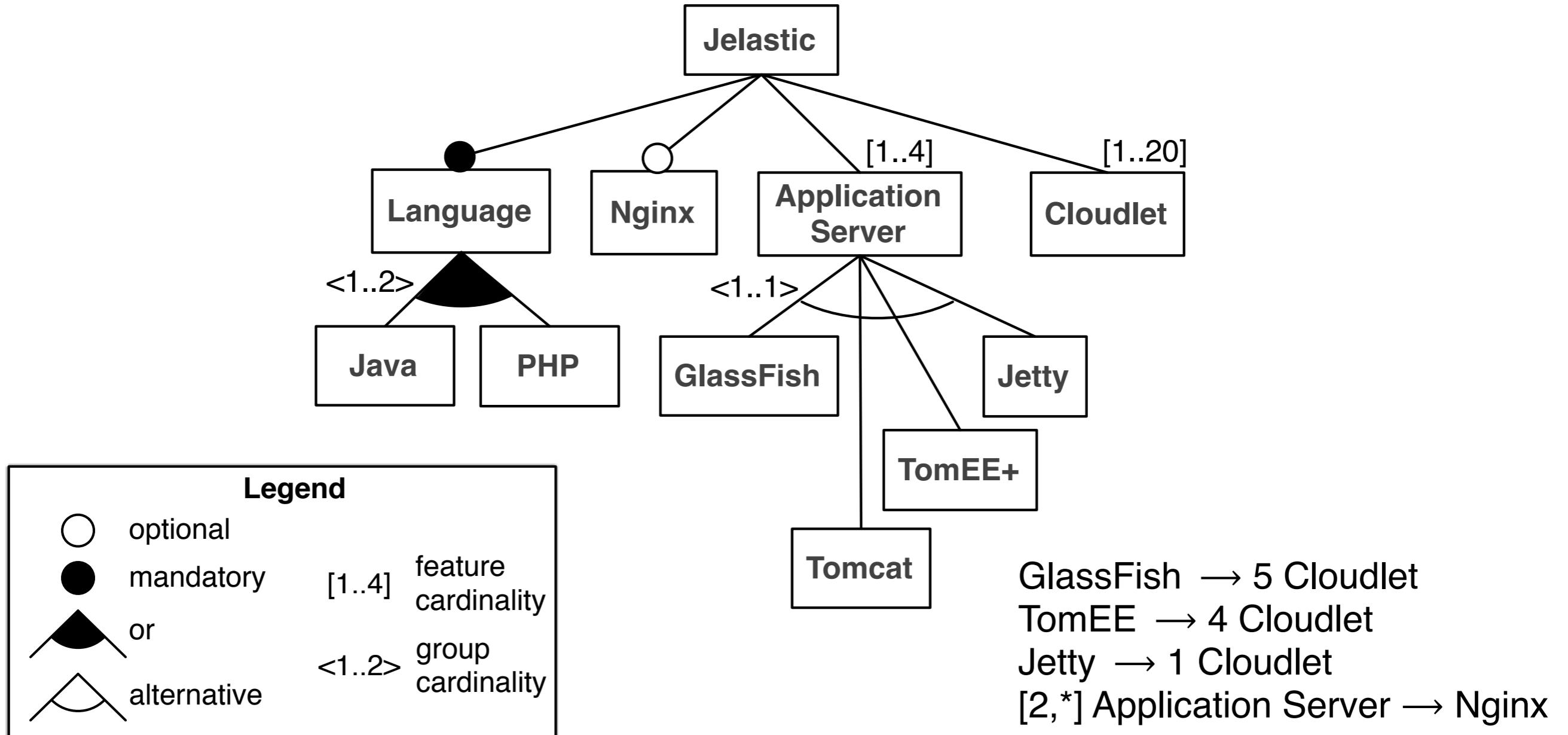
Range inconsistency



$C_1: \text{Jetty} \rightarrow 1 \text{ Cloudlet}$

$C_2: \text{GlassFish} \rightarrow 5 \text{ Cloudlet}$

Evolving Cardinality-based Feature Models

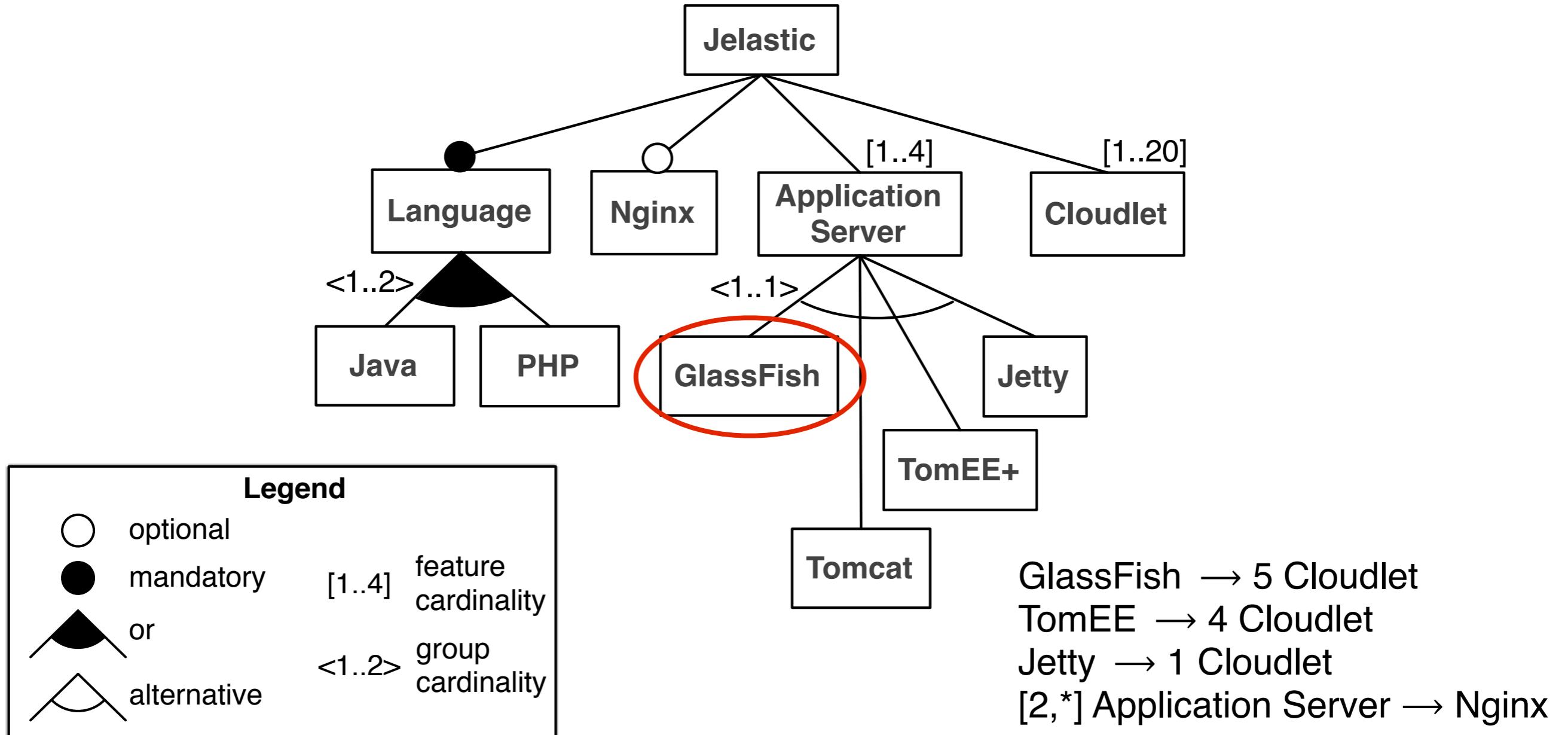


Consistency Checking for the Evolution of Cardinality-based Feature Models.

Clément Quinton, Andreas Pleuss, Daniel Le Berre, Laurence Duchien and Goetz Botterweck.

In Proceedings of the 18th International Software Product Line Conference, SPLC'14. Florence, Italy, Septembre 2014.

Evolving Cardinality-based Feature Models

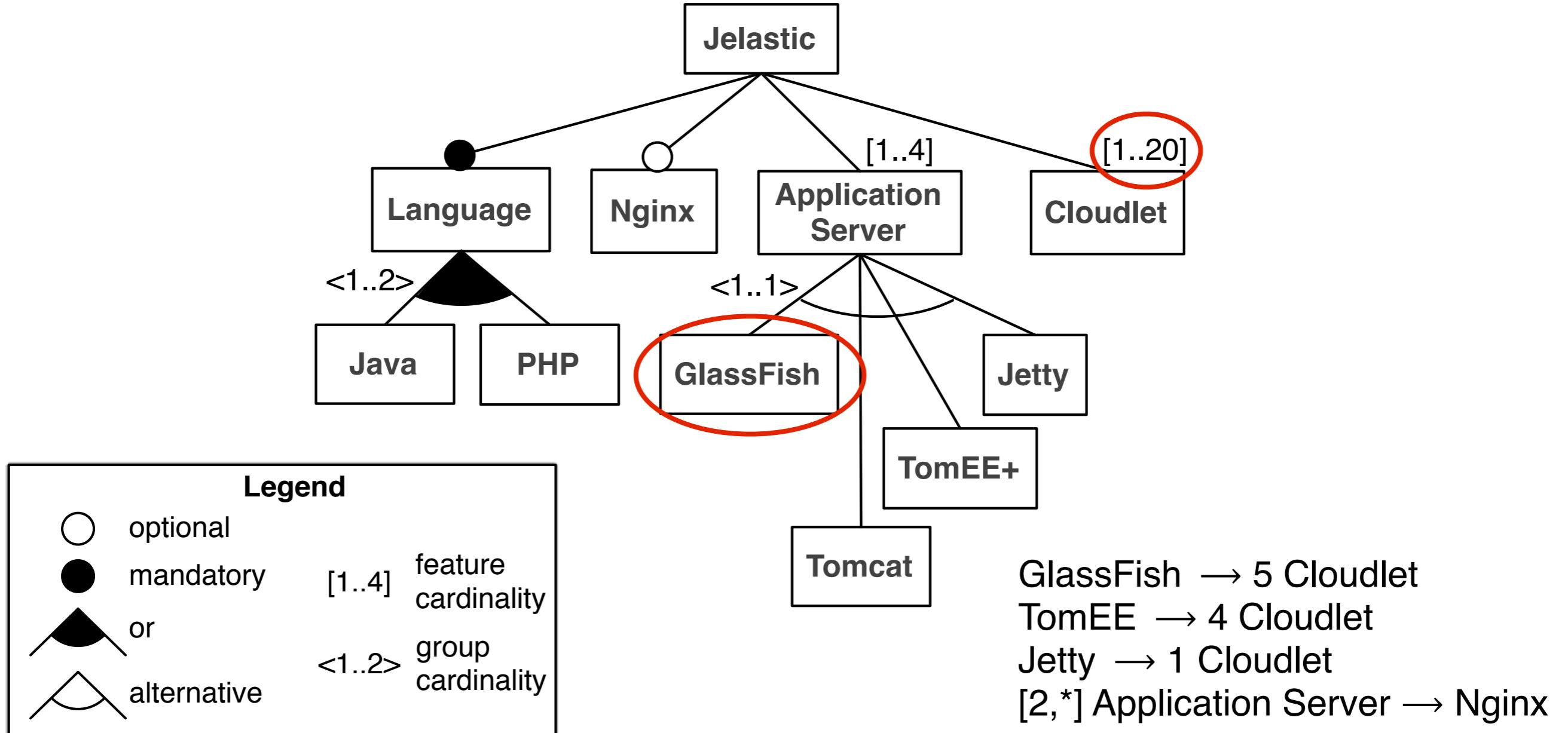


Consistency Checking for the Evolution of Cardinality-based Feature Models.

Clément Quinton, Andreas Pleuss, Daniel Le Berre, Laurence Duchien and Goetz Botterweck.

In Proceedings of the 18th International Software Product Line Conference, SPLC'14. Florence, Italy, Septembre 2014.

Evolving Cardinality-based Feature Models

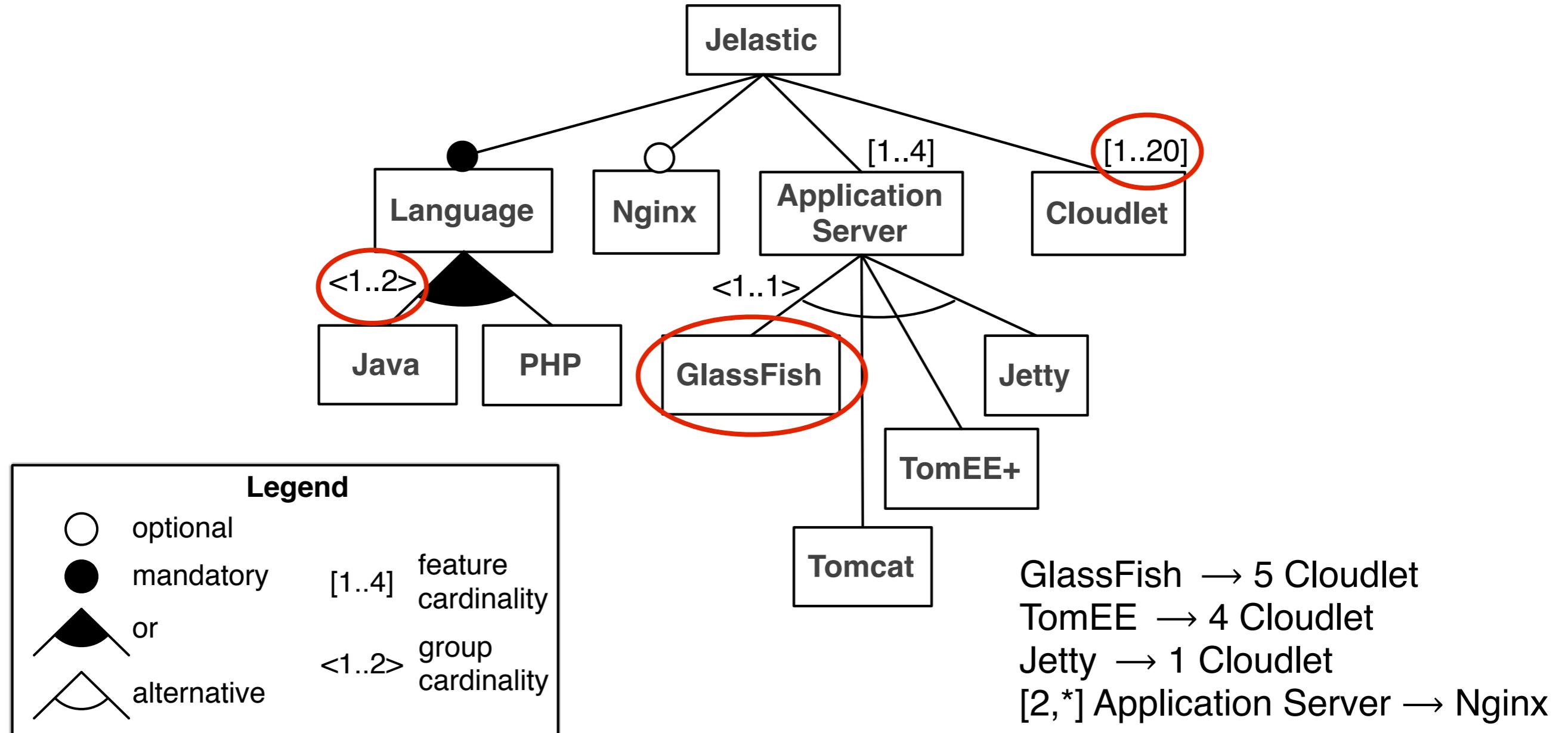


Consistency Checking for the Evolution of Cardinality-based Feature Models.

Clément Quinton, Andreas Pleuss, Daniel Le Berre, Laurence Duchien and Goetz Botterweck.

In Proceedings of the 18th International Software Product Line Conference, SPLC'14. Florence, Italy, Septembre 2014.

Evolving Cardinality-based Feature Models

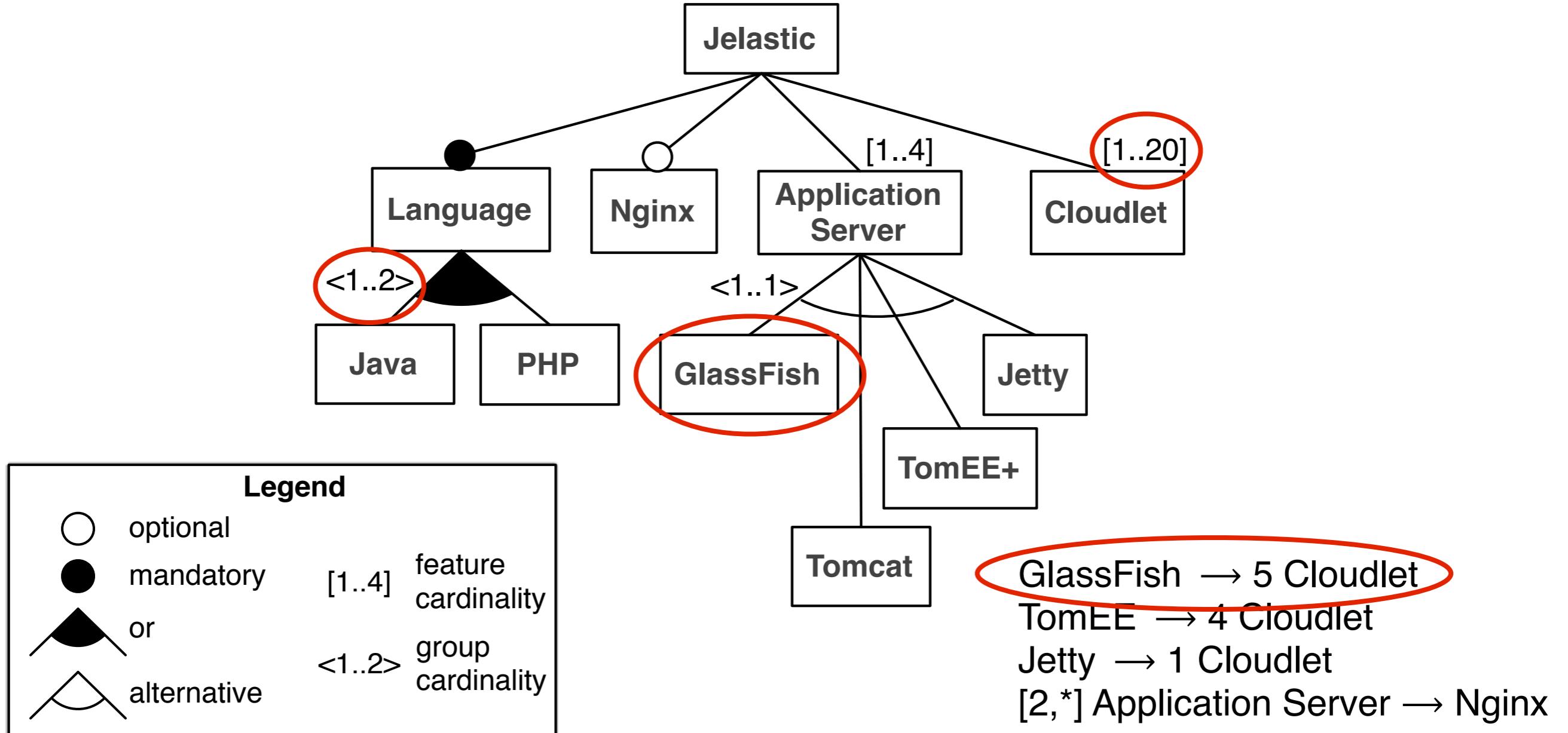


Consistency Checking for the Evolution of Cardinality-based Feature Models.

Clément Quinton, Andreas Pleuss, Daniel Le Berre, Laurence Duchien and Goetz Botterweck.

In Proceedings of the 18th International Software Product Line Conference, SPLC'14. Florence, Italy, Septembre 2014.

Evolving Cardinality-based Feature Models



Consistency Checking for the Evolution of Cardinality-based Feature Models.

Clément Quinton, Andreas Pleuss, Daniel Le Berre, Laurence Duchien and Goetz Botterweck.

In Proceedings of the 18th International Software Product Line Conference, SPLC'14. Florence, Italy, Septembre 2014.

Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature				
Feature Cardinality				
Group Cardinality				
Constraint				

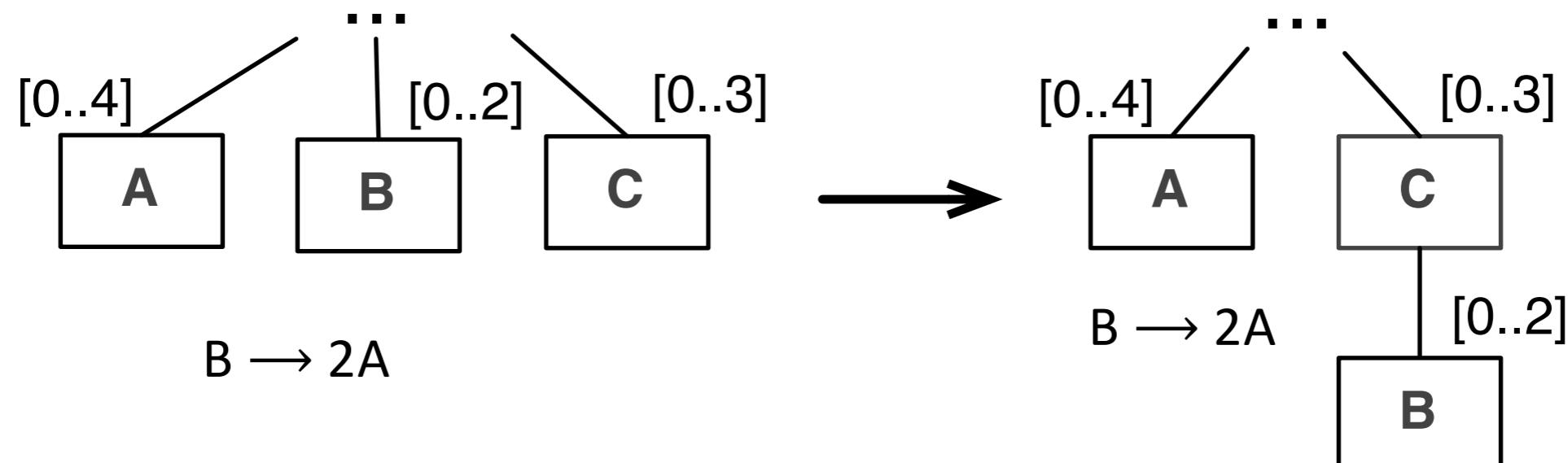
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality				
Group Cardinality				
Constraint				

as leaf

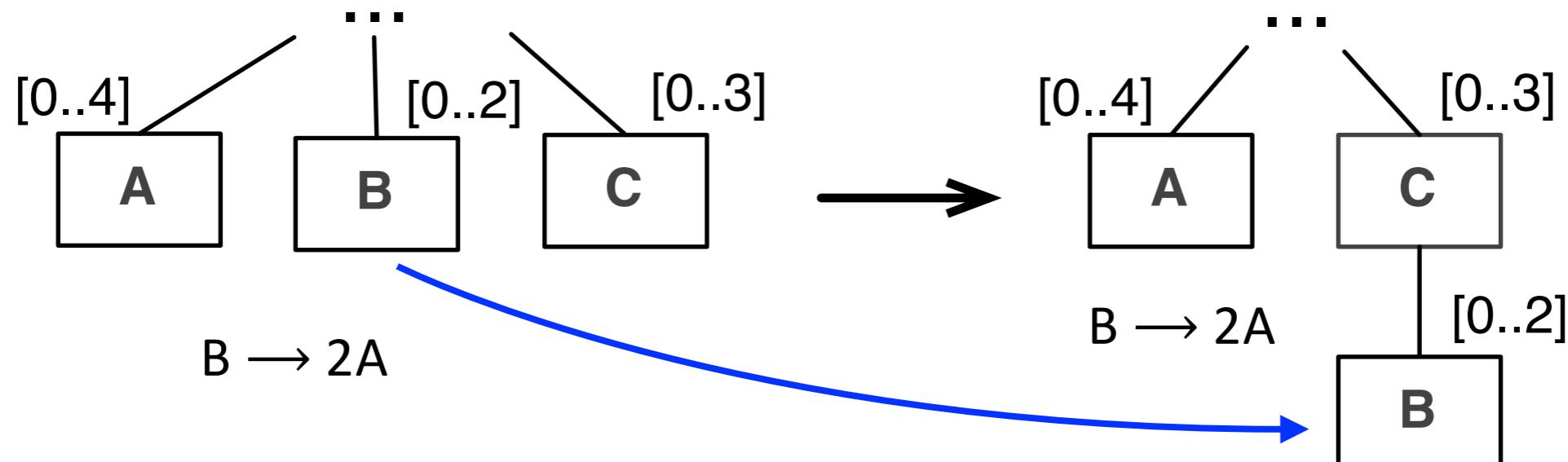
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality				
Group Cardinality				
Constraint				



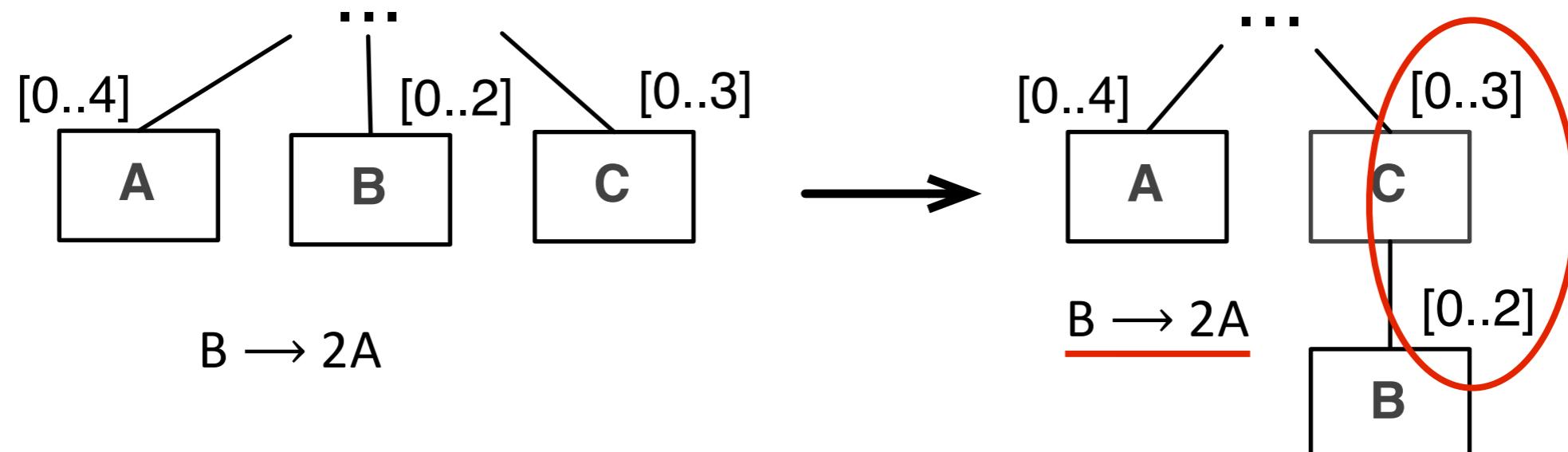
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality				
Group Cardinality				
Constraint				



Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality				
Group Cardinality				
Constraint				



Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality				
Group Cardinality				
Constraint				

Evolving Cardinality-based Feature Models

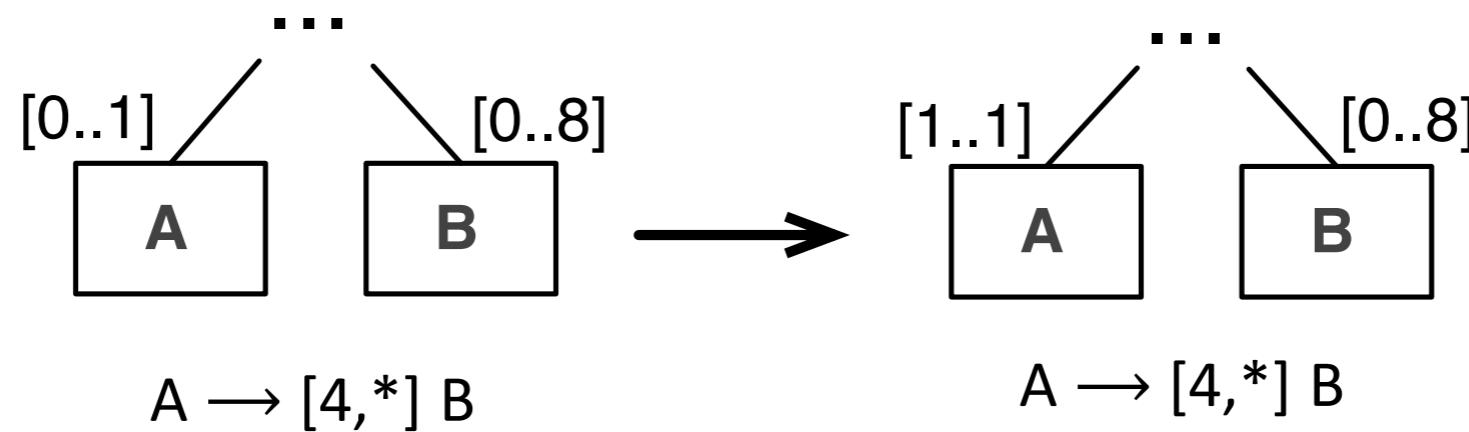
	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality				
Group Cardinality				
Constraint				

Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality				
Constraint				

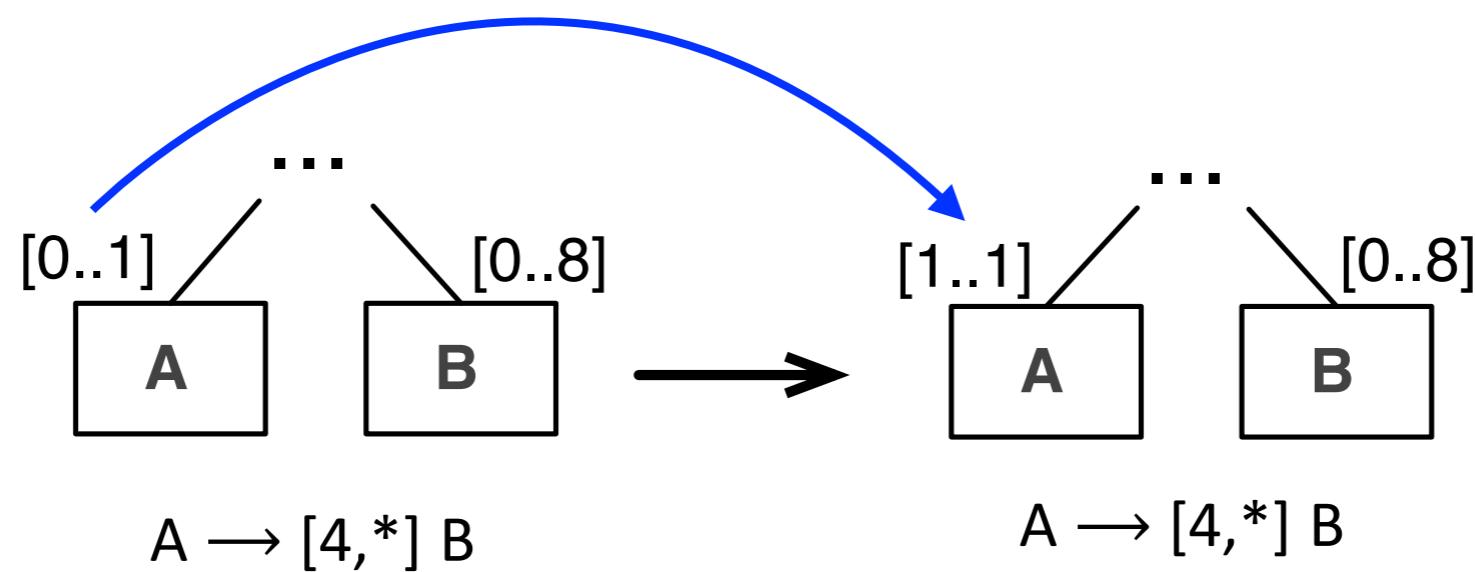
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality				
Constraint				



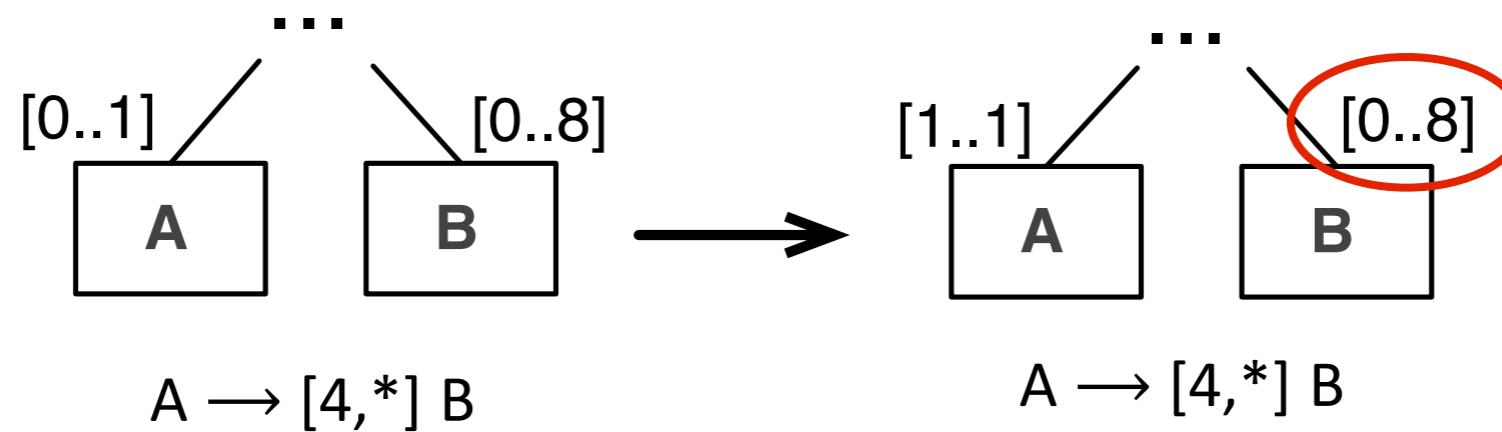
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality				
Constraint				



Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality				
Constraint				



Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality				
Constraint				

Evolving Cardinality-based Feature Models

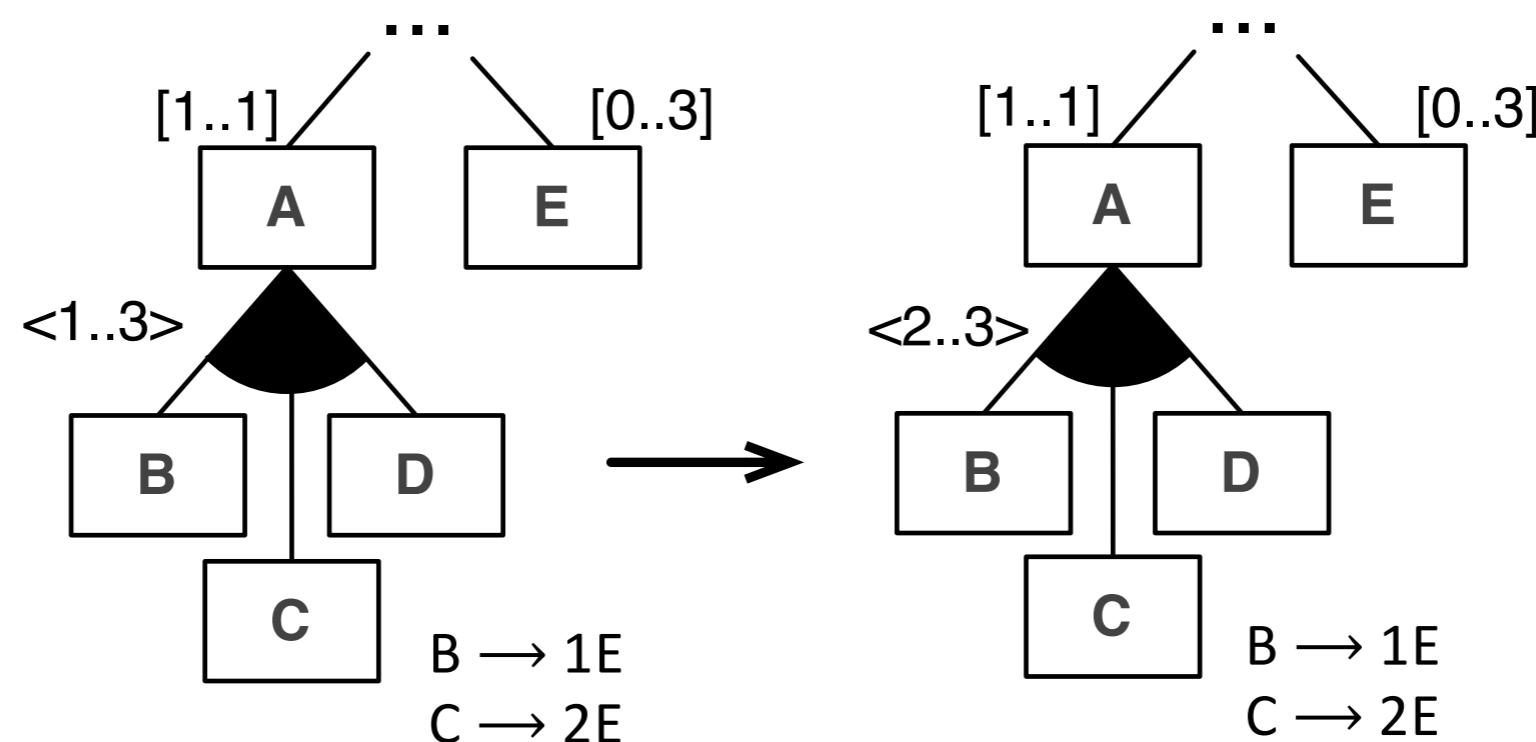
	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	
Constraint				

Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint				

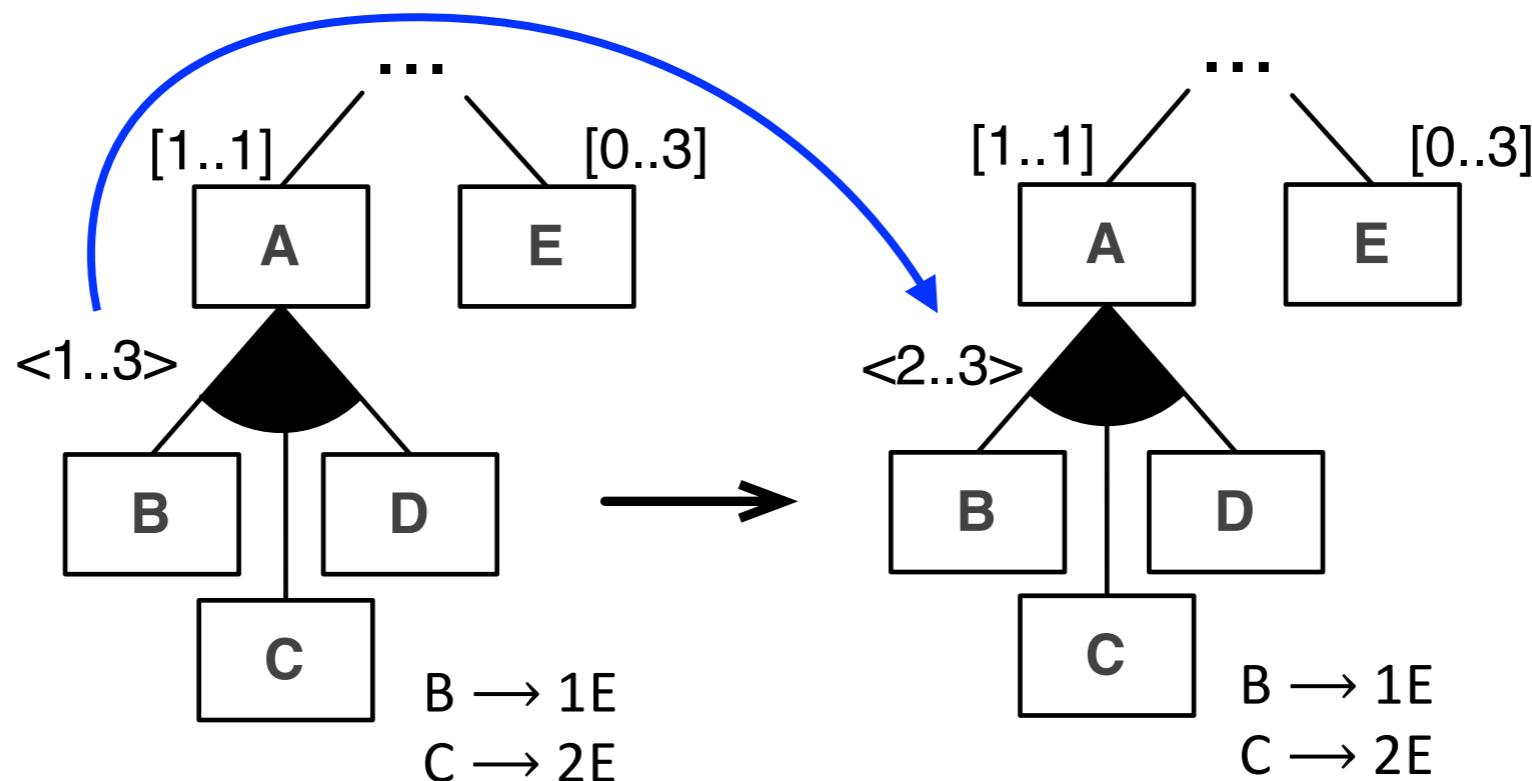
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint				



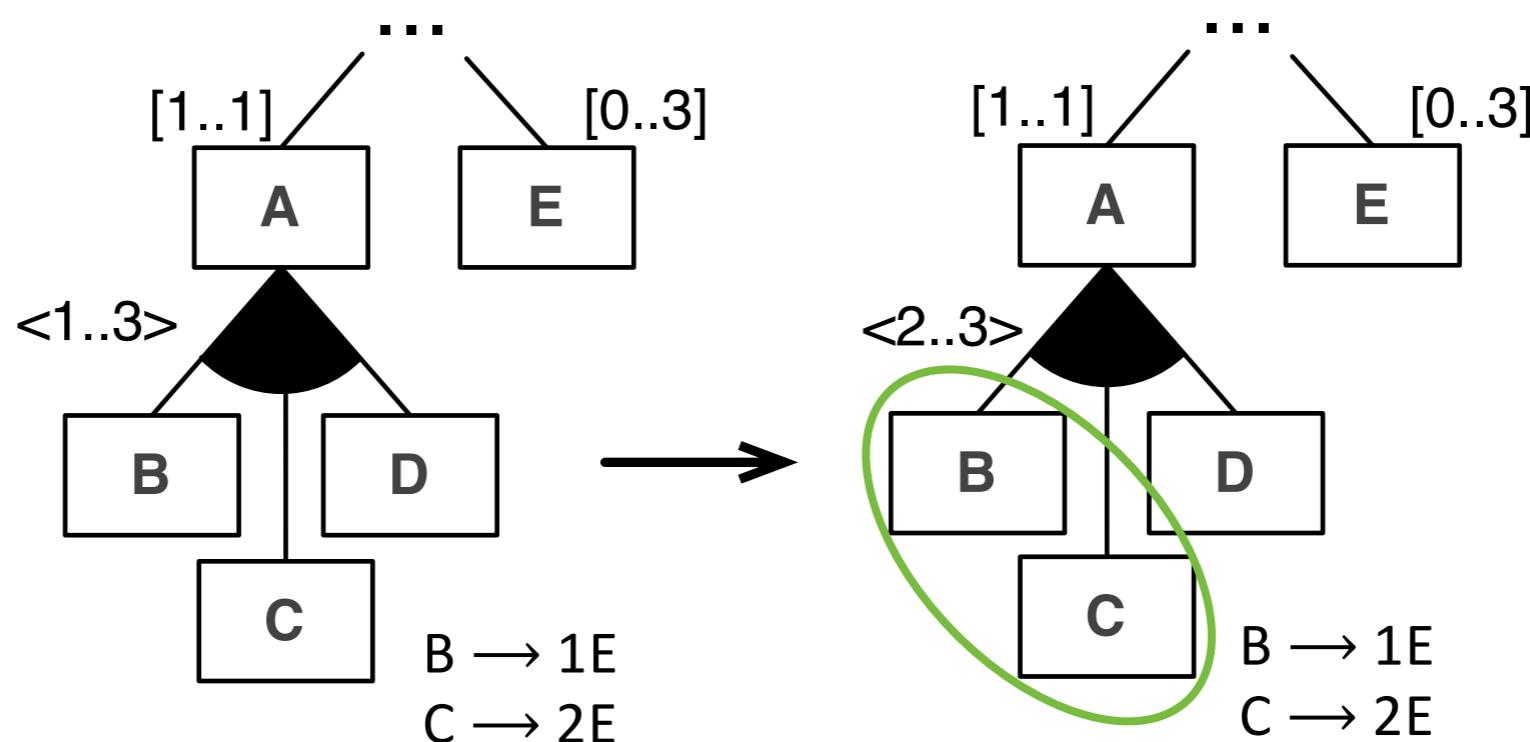
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint				



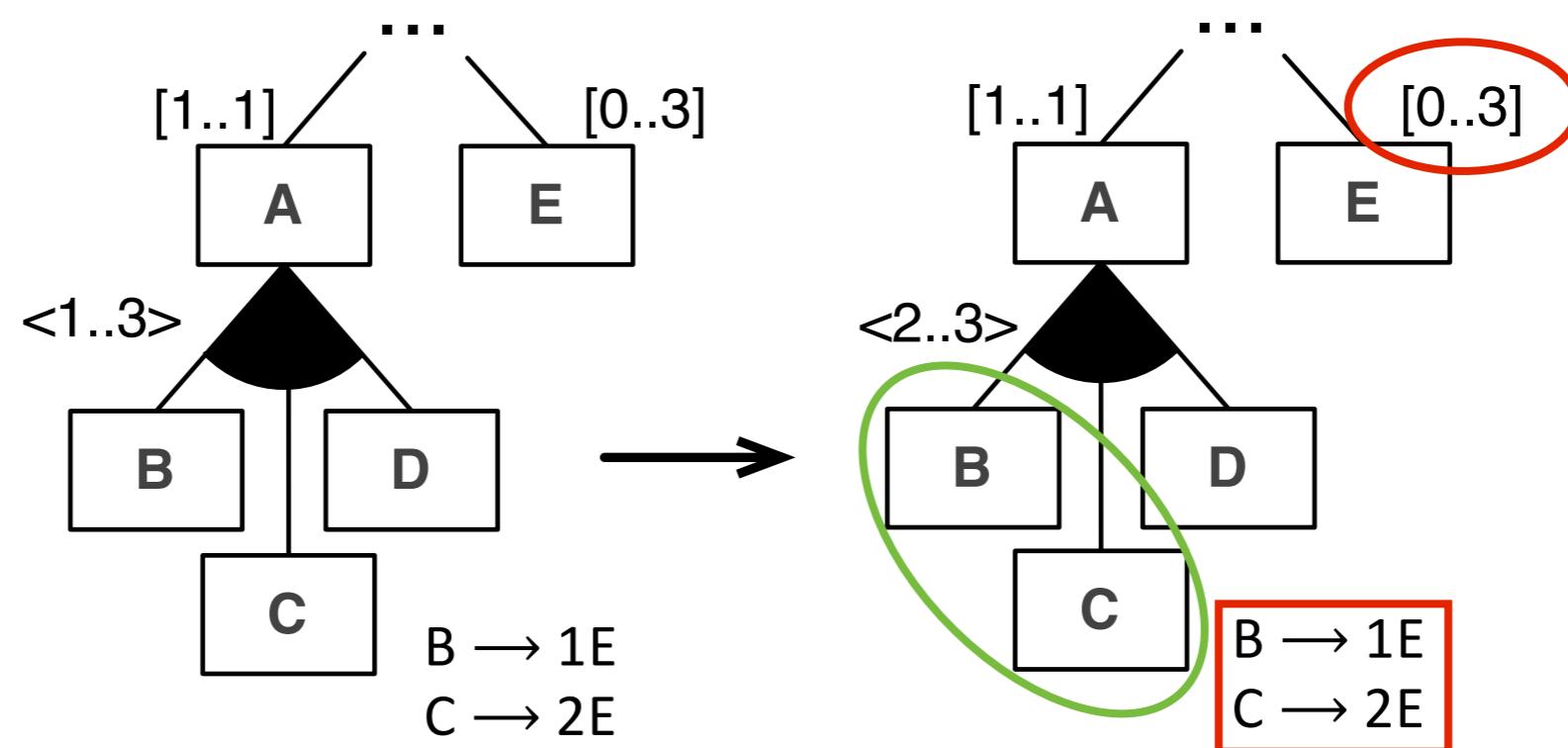
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint				



Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint				



Evolving Cardinality-based Feature Models

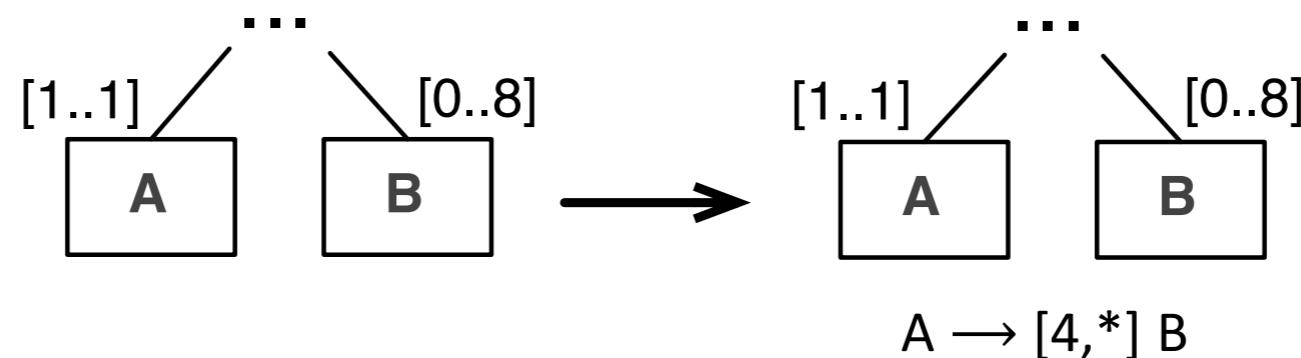
	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint				

Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!			

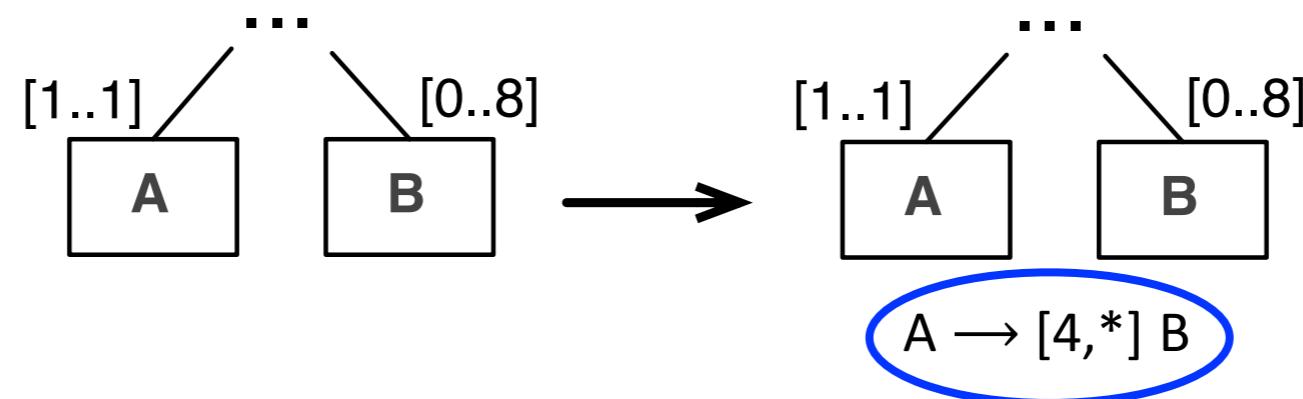
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!			



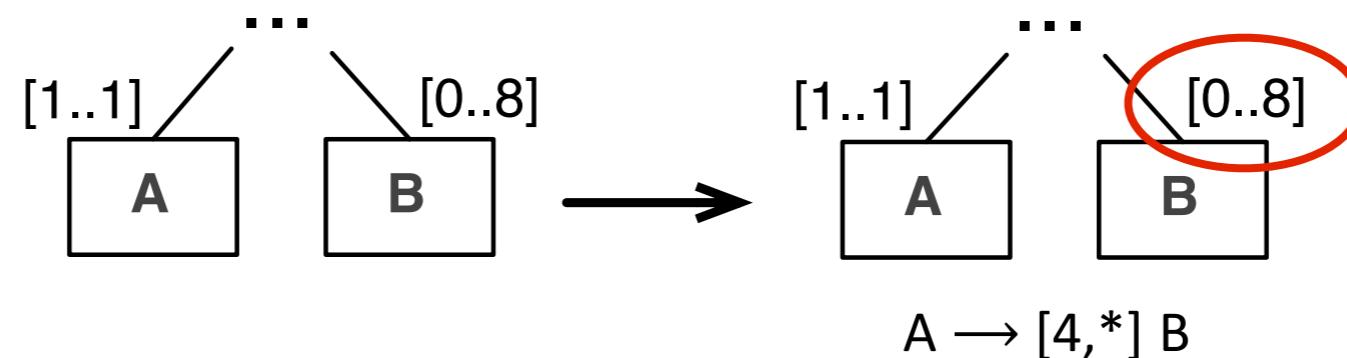
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!			



Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!			



$A \rightarrow [4,*] B$

Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!			

Evolving Cardinality-based Feature Models

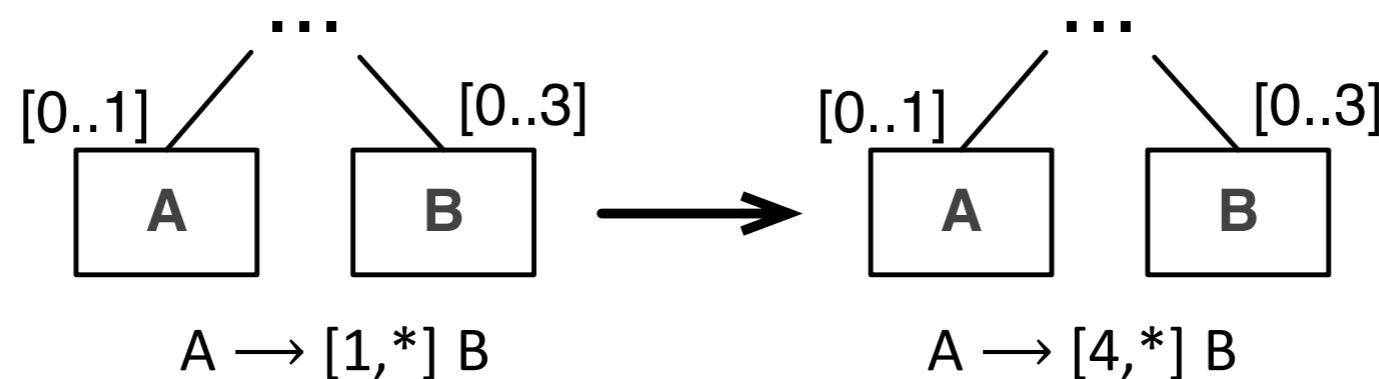
	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	

Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	NA

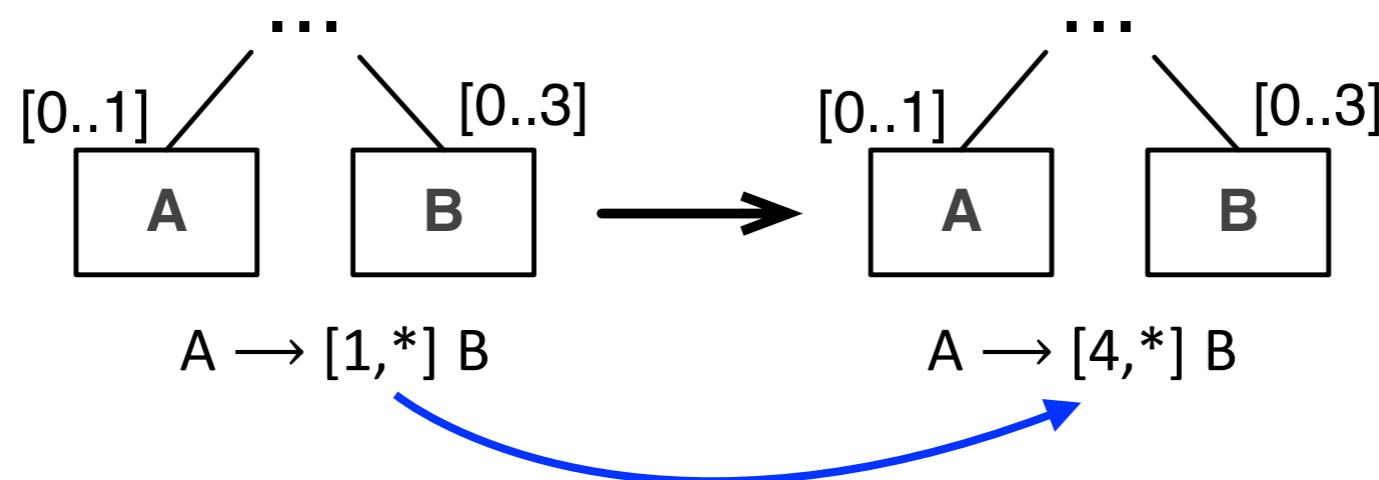
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	NA



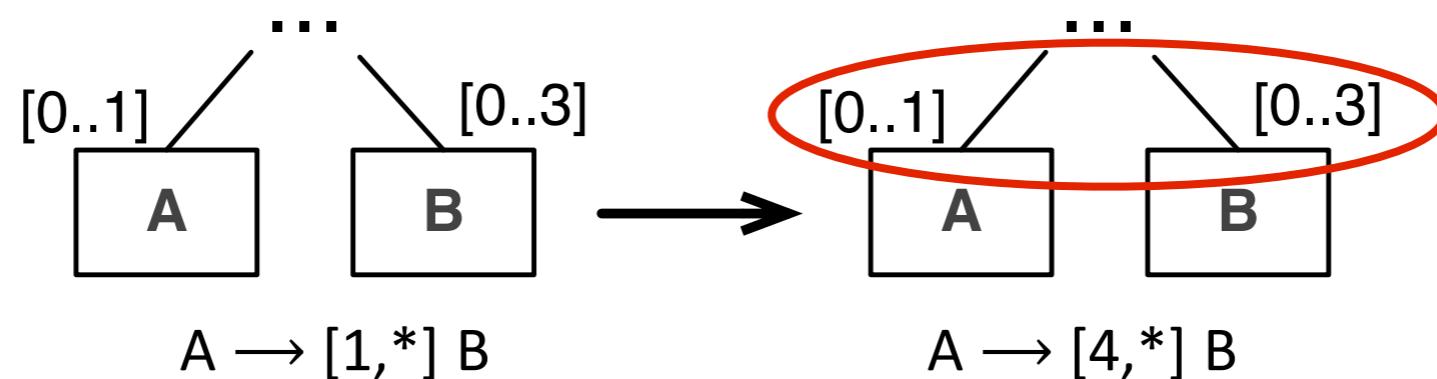
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	NA



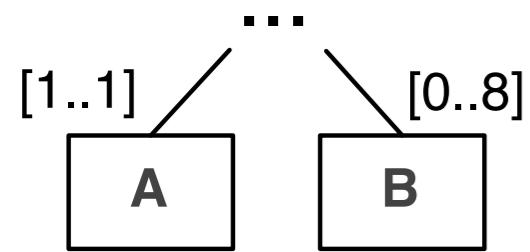
Evolving Cardinality-based Feature Models

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	NA

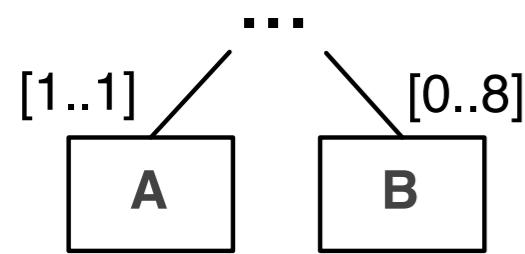


Cardinality-based Feature Model Consistency

Cardinality-based Feature Model Consistency

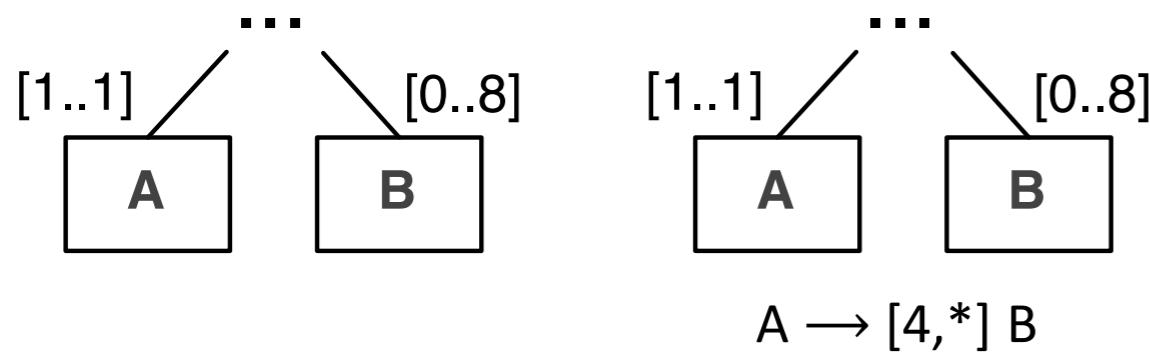


Cardinality-based Feature Model Consistency



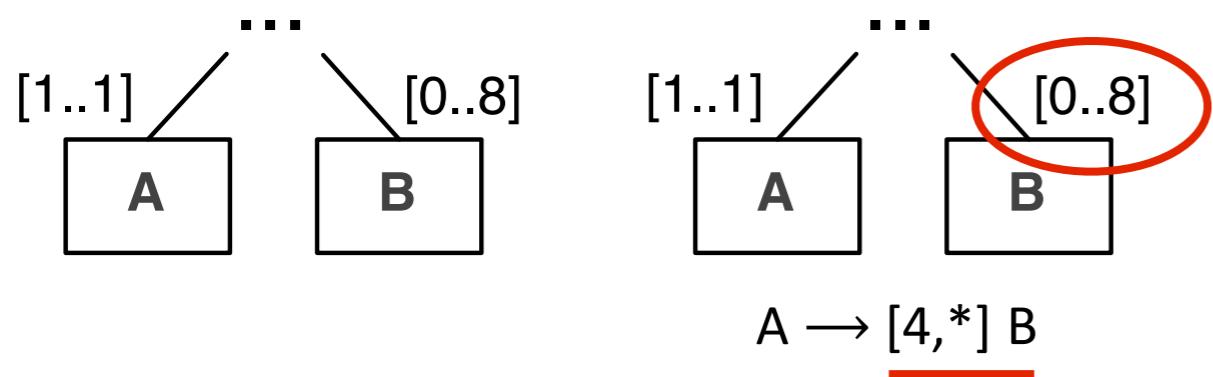
Local Range
Consistent

Cardinality-based Feature Model Consistency



Local Range
Consistent

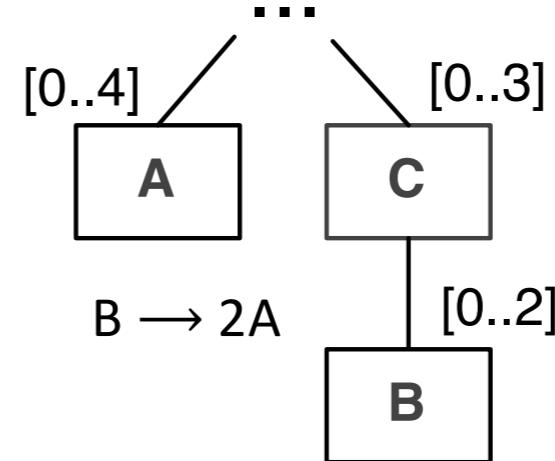
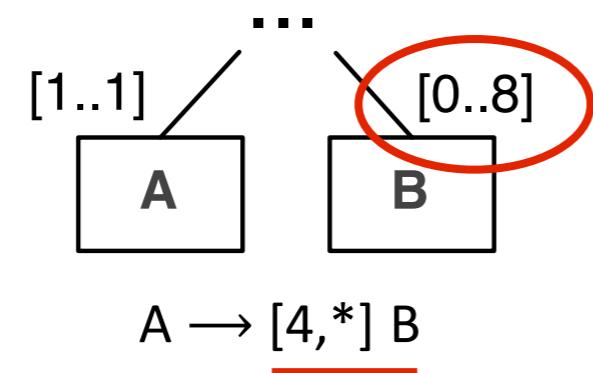
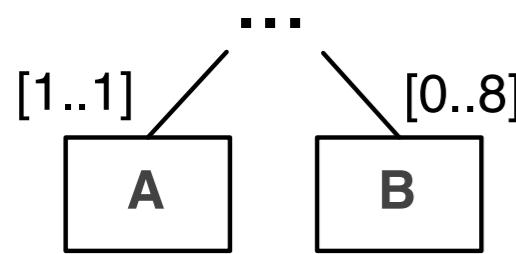
Cardinality-based Feature Model Consistency



Local Range
Consistent

Local Range
Inconsistent

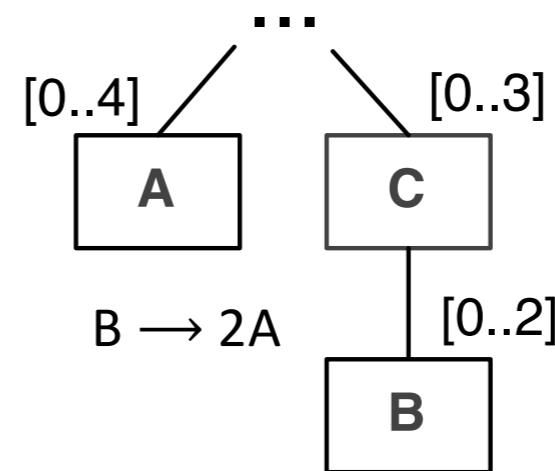
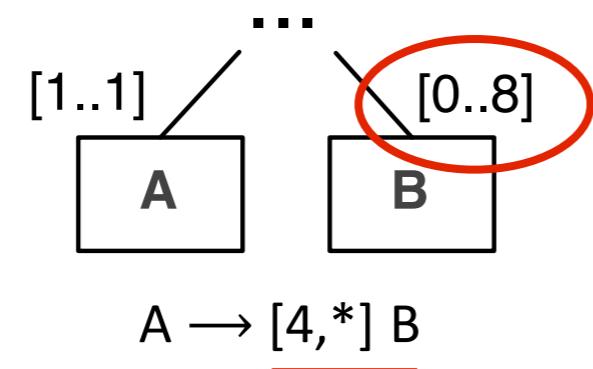
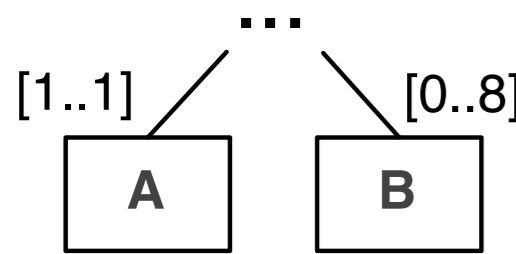
Cardinality-based Feature Model Consistency



Local Range
Consistent

Local Range
Inconsistent

Cardinality-based Feature Model Consistency

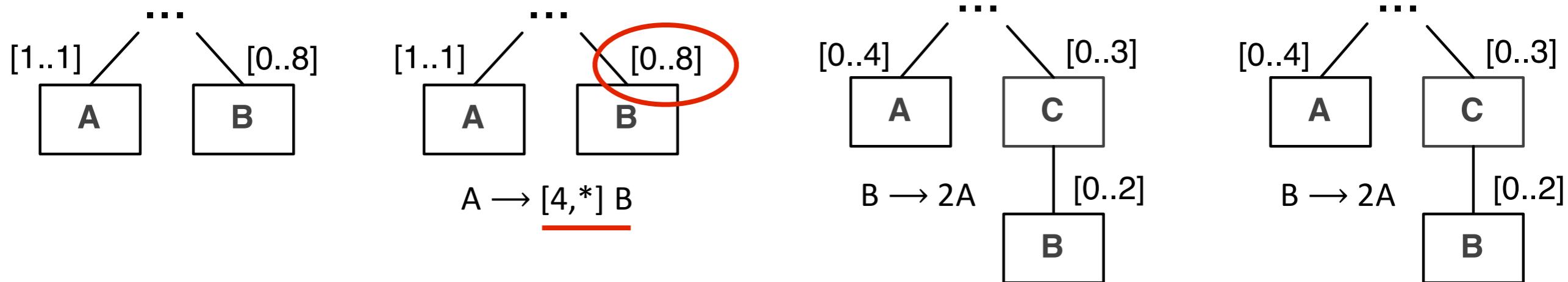


Local Range
Consistent

Local Range
Inconsistent

Local Range
Consistent

Cardinality-based Feature Model Consistency

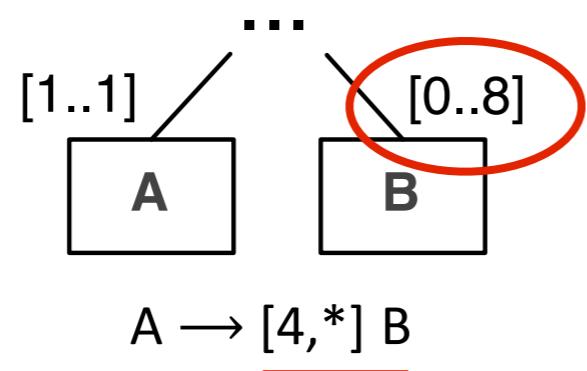
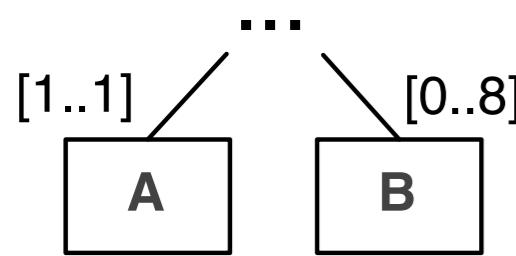


Local Range
Consistent

Local Range
Inconsistent

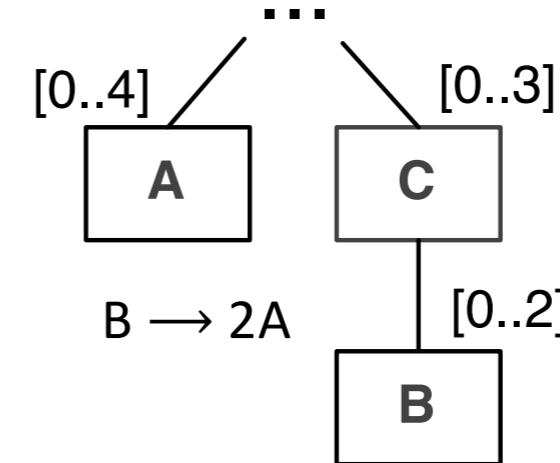
Local Range
Consistent

Cardinality-based Feature Model Consistency

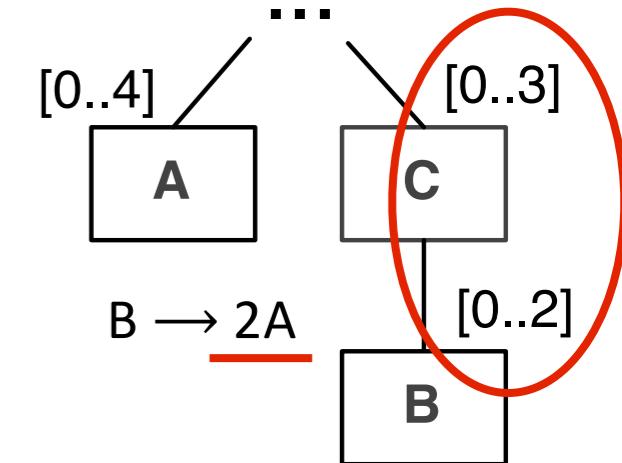


Local Range
Consistent

Local Range
Inconsistent

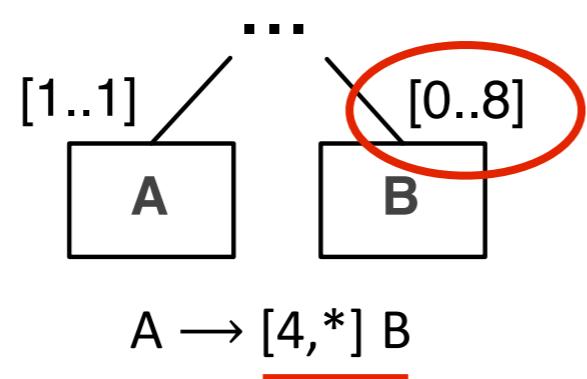
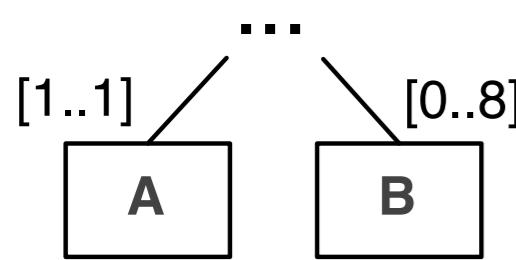


Local Range
Consistent



Global Range
Inconsistent

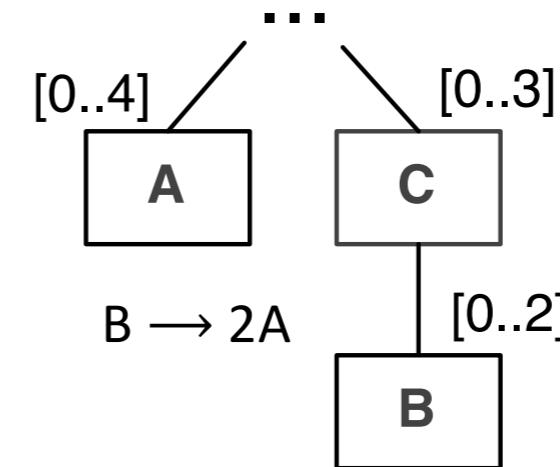
Cardinality-based Feature Model Consistency



$A \rightarrow [4, *] B$

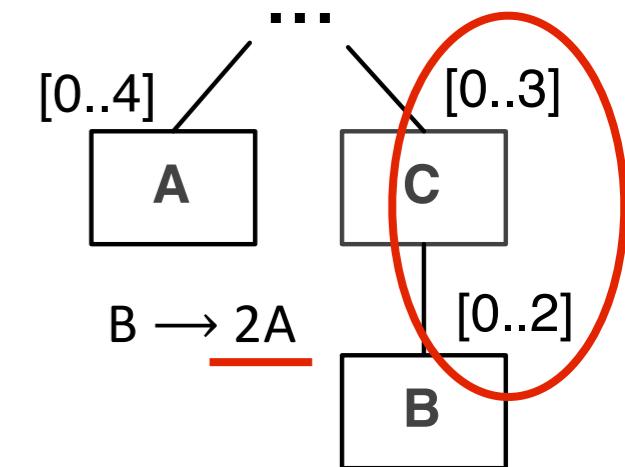
Local Range
Consistent

Local Range
Inconsistent



$B \rightarrow 2A$

Local Range
Consistent

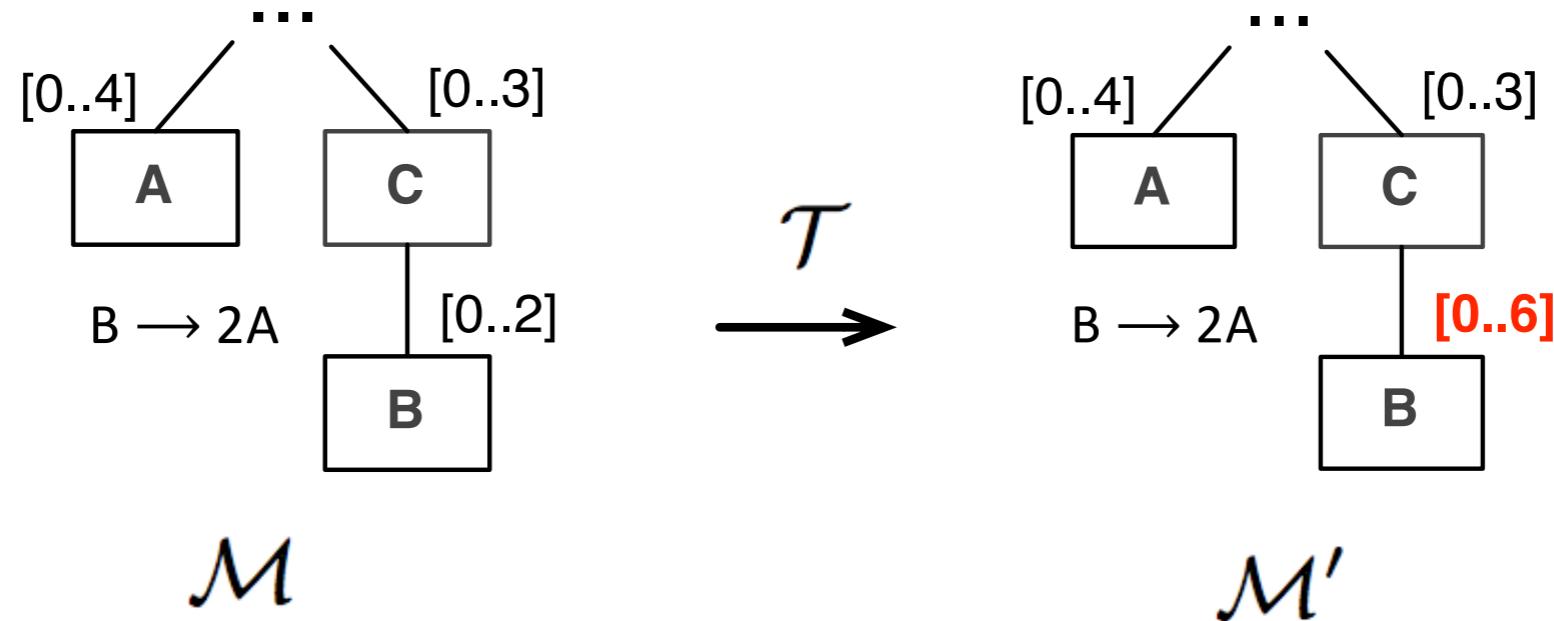


$B \rightarrow 2A$

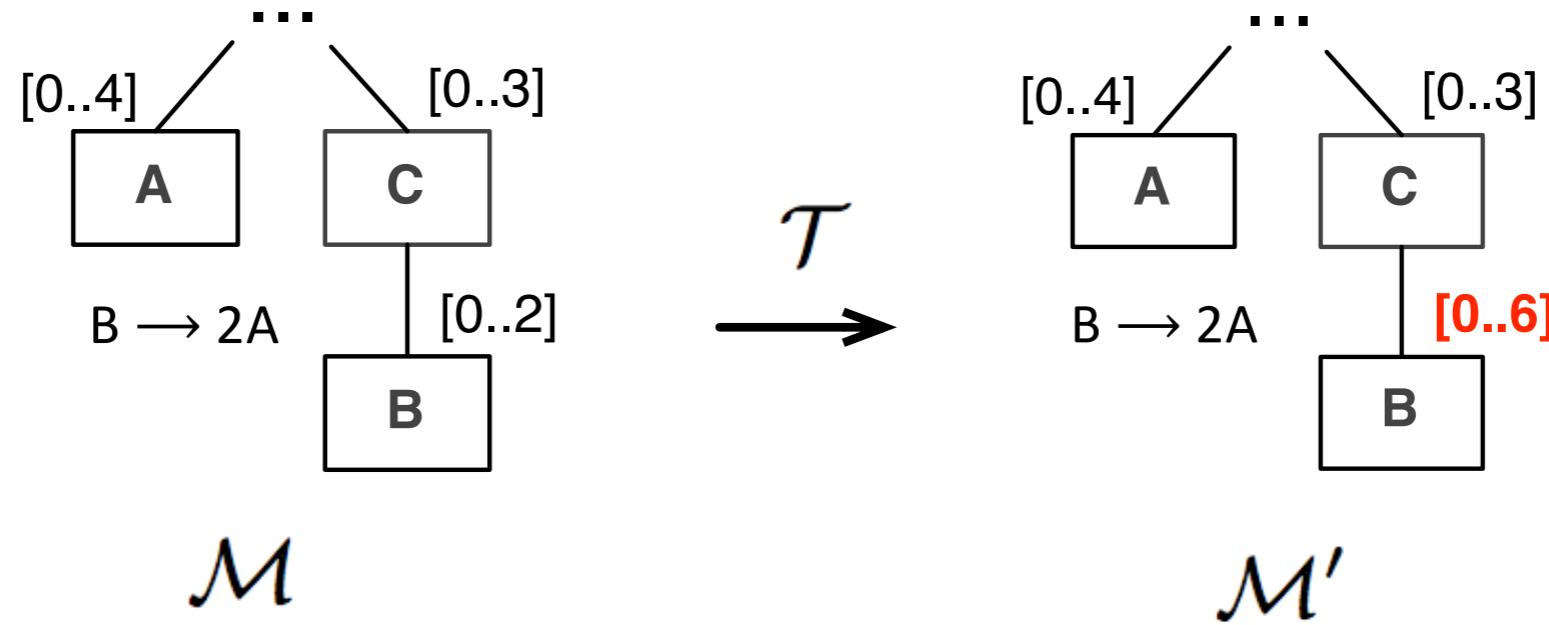
Global Range
Inconsistent

Consistent = Local Range Consistent + Global Range Consistent

Relation Between Consistencies

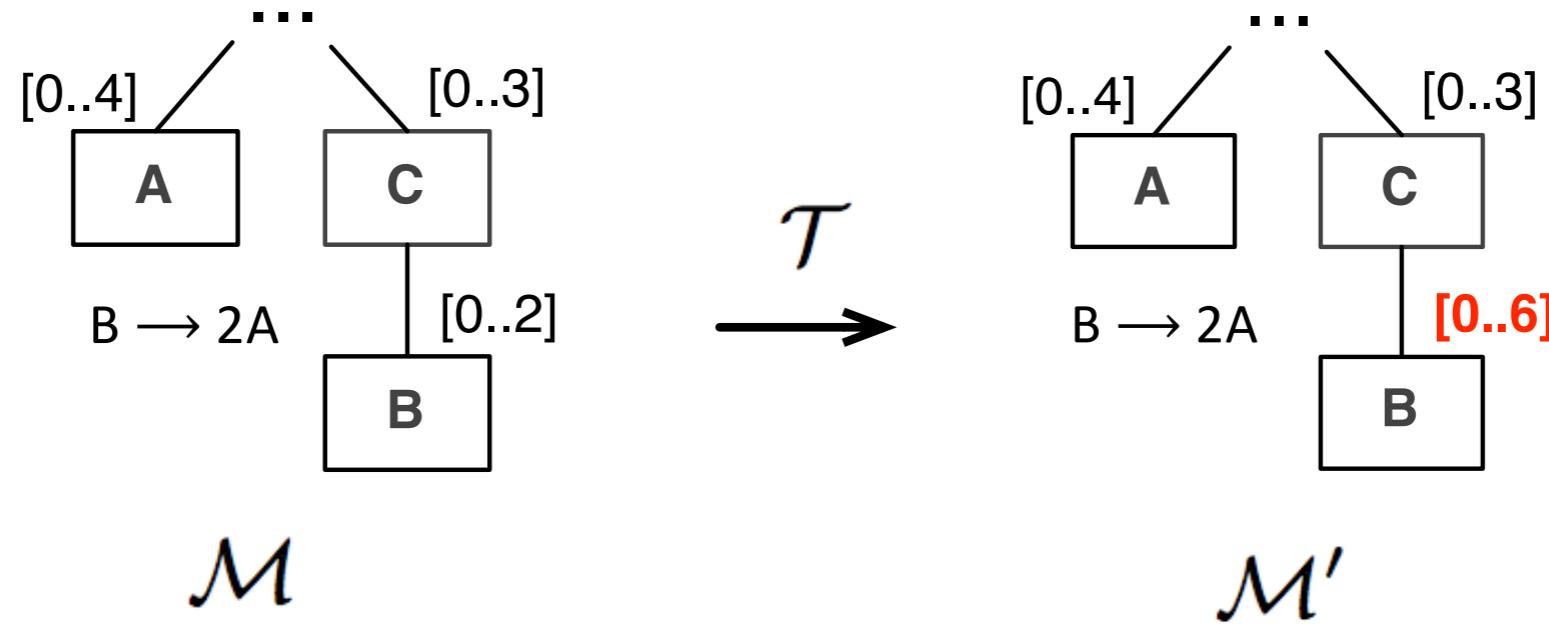


Relation Between Consistencies



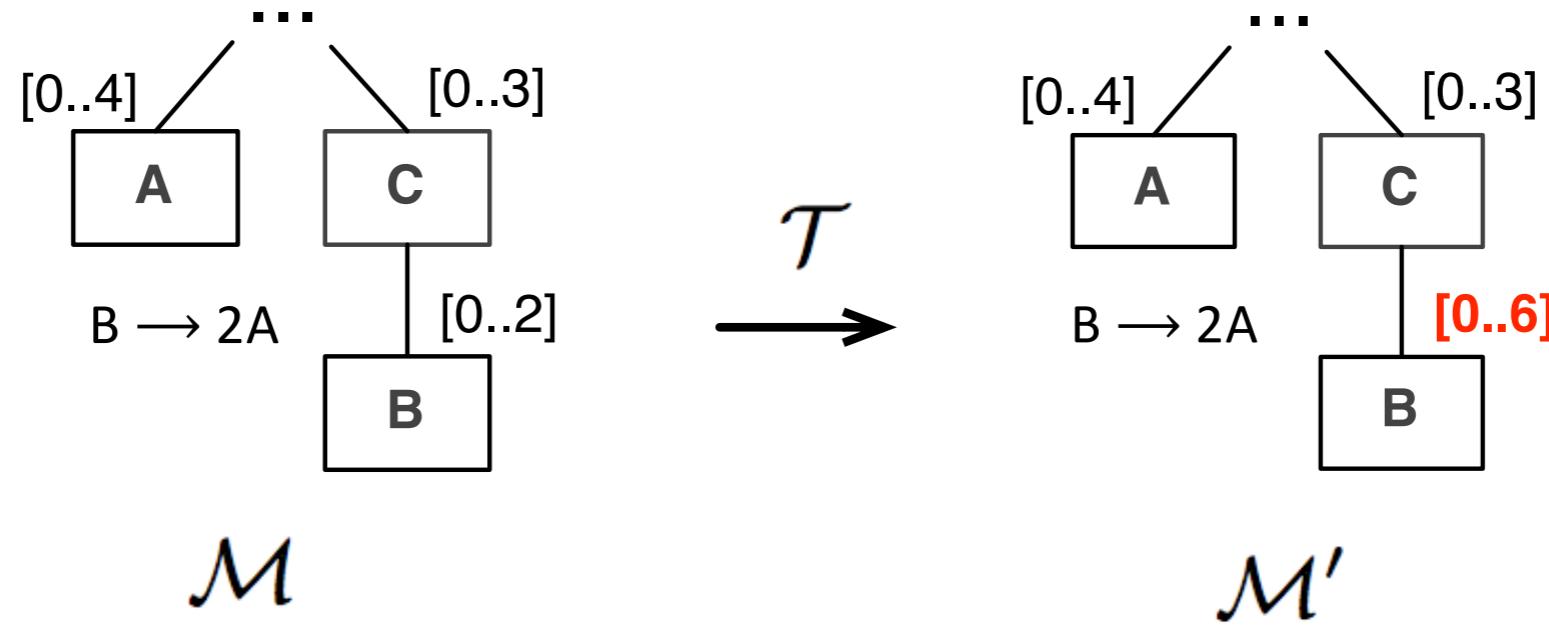
$$\mathcal{T} : \mathcal{M} = (\mathcal{F}, \omega, \varphi) \rightarrow \mathcal{M}' =$$

Relation Between Consistencies



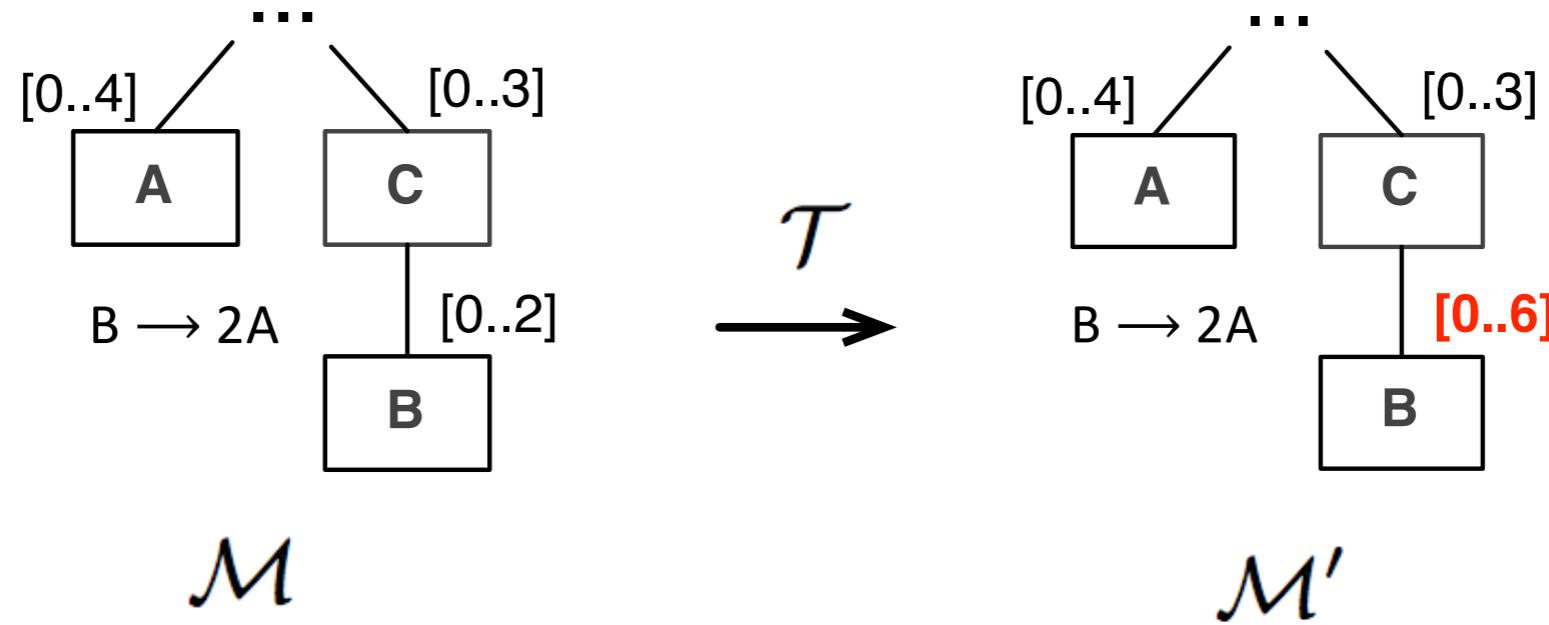
$$\mathcal{T} : \mathcal{M} = (\mathcal{F}, \omega, \varphi) \rightarrow \mathcal{M}' = (\mathcal{F},$$

Relation Between Consistencies



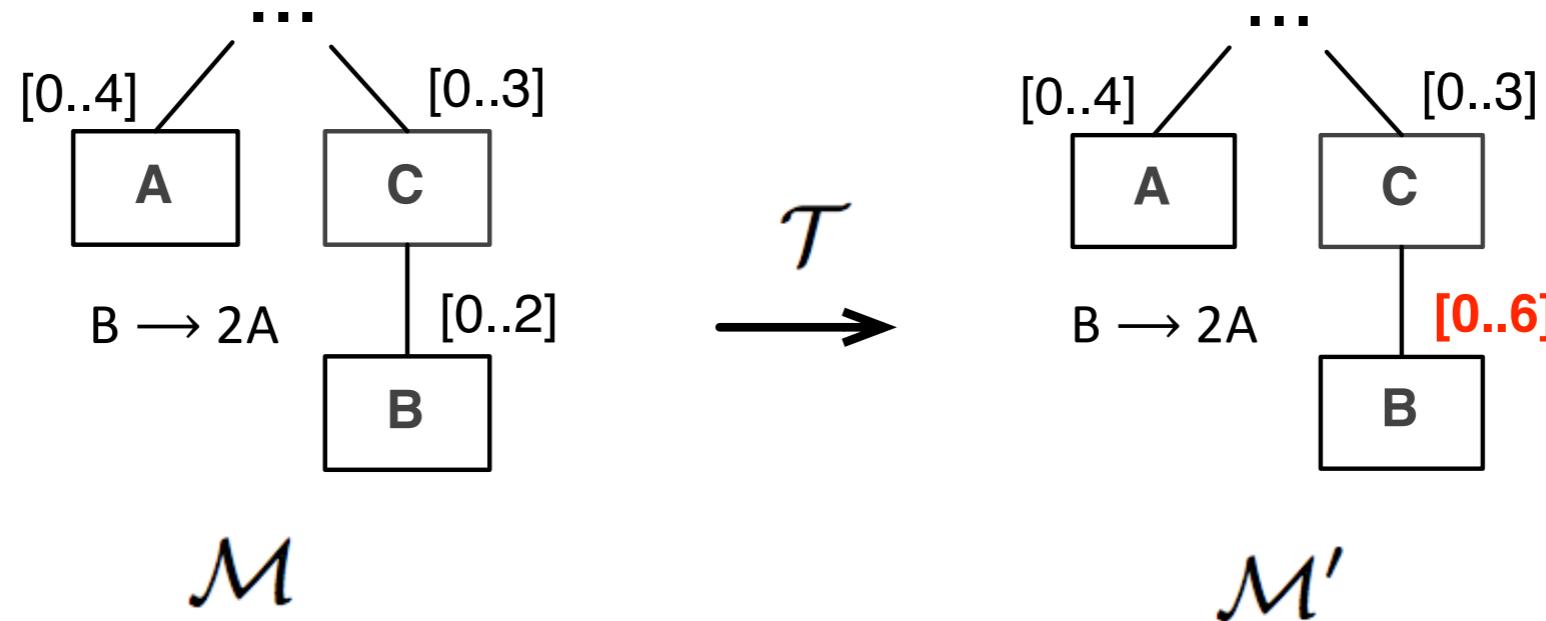
$$\mathcal{T} : \mathcal{M} = (\mathcal{F}, \omega, \varphi) \rightarrow \mathcal{M}' = (\mathcal{F}, \quad , \varphi)$$

Relation Between Consistencies



$$\mathcal{T} : \mathcal{M} = (\mathcal{F}, \omega, \varphi) \rightarrow \mathcal{M}' = (\mathcal{F}, \omega', \varphi')$$

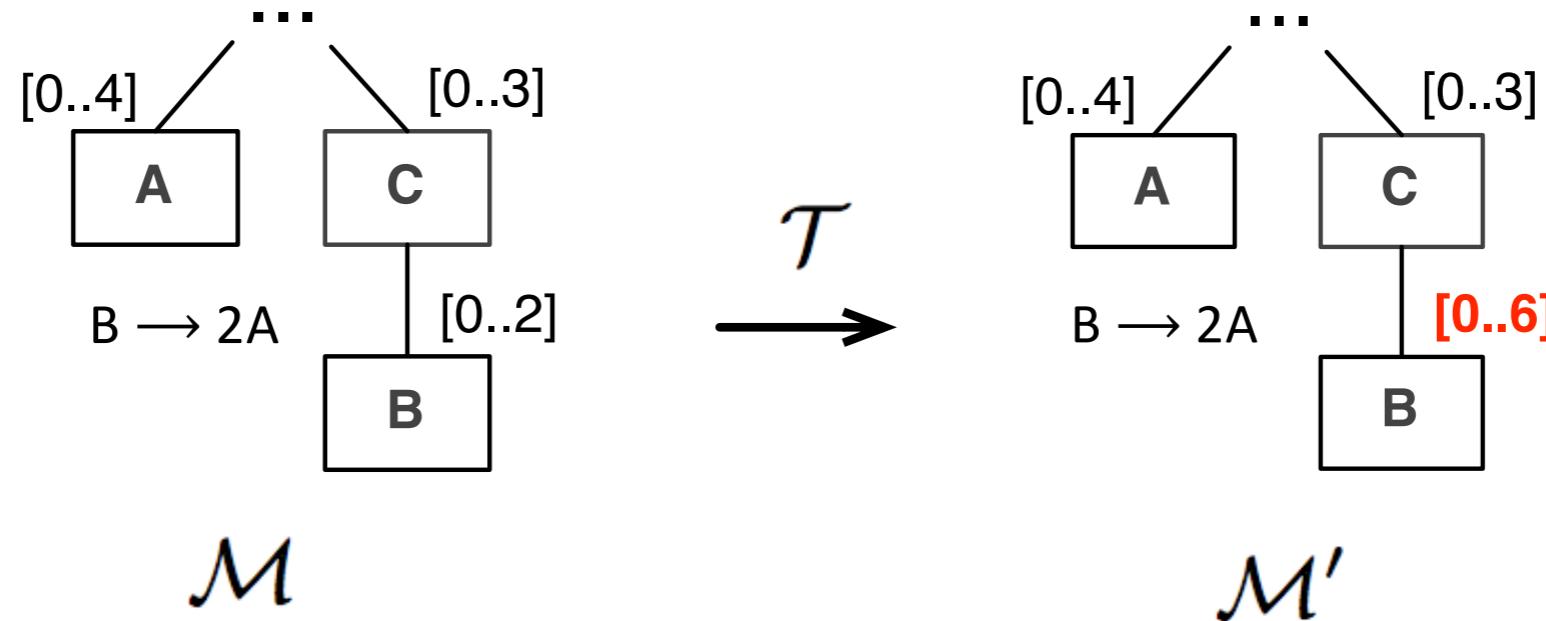
Relation Between Consistencies



$$\mathcal{T} : \mathcal{M} = (\mathcal{F}, \omega, \varphi) \rightarrow \mathcal{M}' = (\mathcal{F}, \omega', \varphi)$$

$$\begin{aligned}\omega'(f) &= \omega(f) \text{ iff } \text{parent}(f) = \emptyset \\ &= \omega(f) \times \omega'(\text{parent}(f))\end{aligned}$$

Relation Between Consistencies



Property: checking the global range consistency of a feature model \mathcal{M} is equivalent to checking the local range consistency of $\mathcal{T}(\mathcal{M})$.

Global Inverse Consistency in CSP

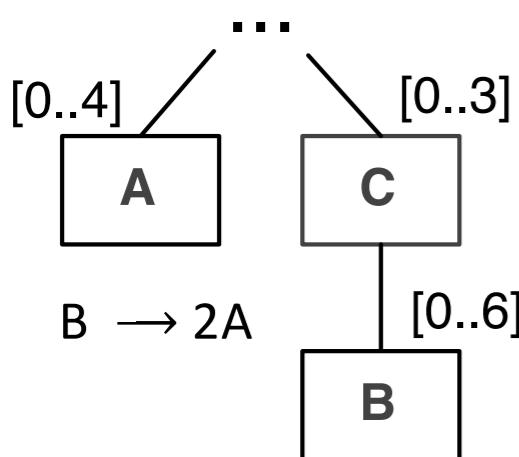
A CSP is Global Inverse Consistent *iff* for every value in the domain of its variables, there exists a solution of the CSP with such value.

[Bessiere *et al.*, CP'13]

Global Inverse Consistency in CSP

A CSP is Global Inverse Consistent *iff* for every value in the domain of its variables, there exists a solution of the CSP with such value.

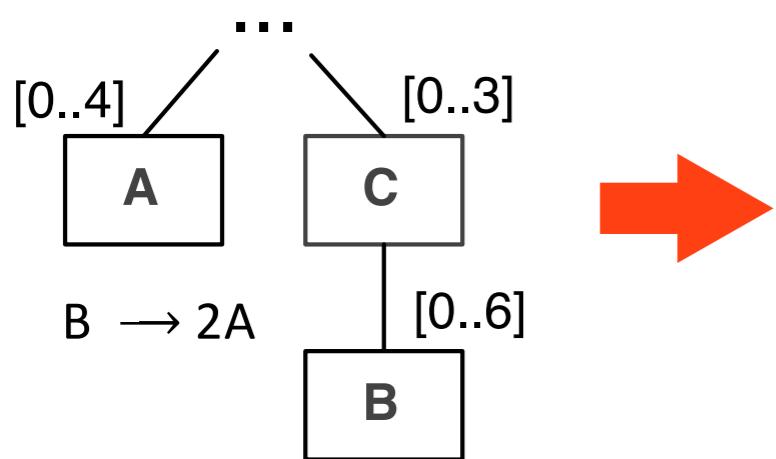
[Bessiere *et al.*, CP'13]



Global Inverse Consistency in CSP

A CSP is Global Inverse Consistent *iff* for every value in the domain of its variables, there exists a solution of the CSP with such value.

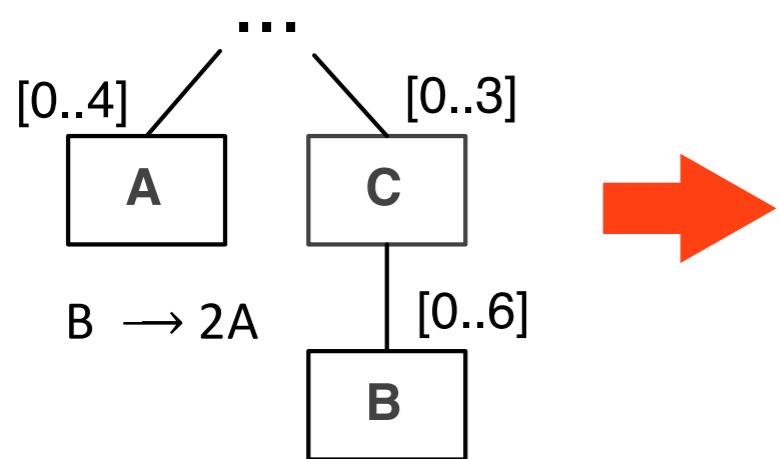
[Bessiere *et al.*, CP'13]



Detecting a local range inconsistency is equivalent to detecting that a CSP is not Global Inverse Consistent

Summary

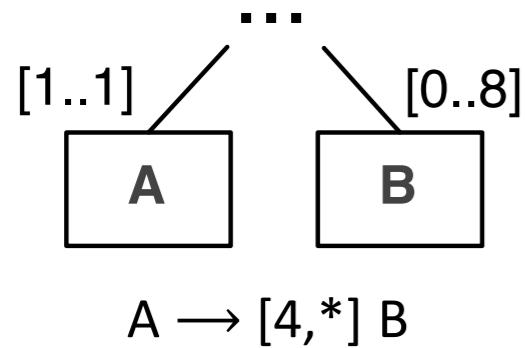
Property: checking the global range consistency of a feature model \mathcal{M} is equivalent to checking the local range consistency of $\mathcal{T}(\mathcal{M})$.



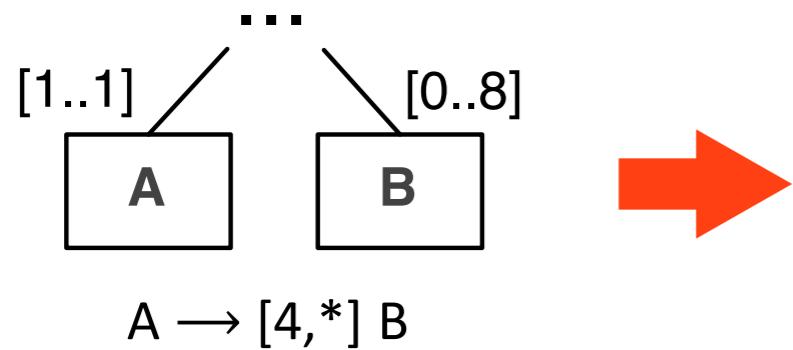
Detecting a local range inconsistency is equivalent to detecting that a CSP is not Global Inverse Consistent

Automated Support

Automated Support



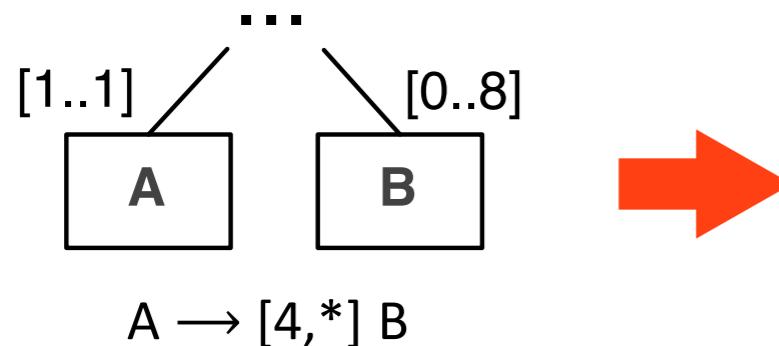
Automated Support



→

```
#(1,1,[A.1]);  
#(1,1,[B.0, B.1, B.2, B.3, B.4, B.5, B.6, B.7, B.8]);  
(A.1 => B.4 | B.5 | B.6 | B.7 | B.8);
```

Automated Support



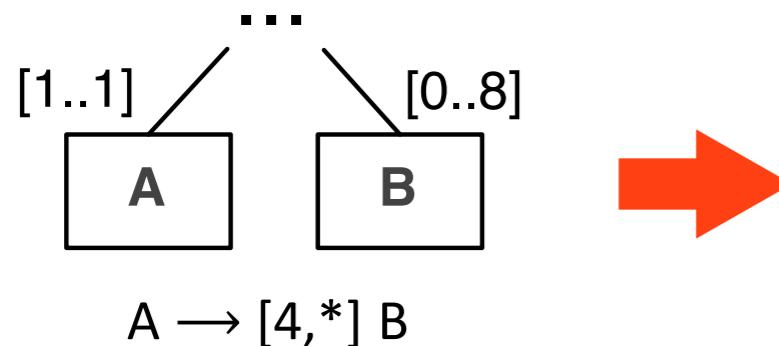
Resulting constraint code:

```
#(1,1,[A.1]);
#(1,1,[B.0, B.1, B.2, B.3, B.4, B.5, B.6, B.7, B.8]);
(A.1 => B.4 | B.5 | B.6 | B.7 | B.8);
```



```
Macintosh:cardFMs clement$ java -jar sat4j.jar fmEvo.txt
computing problem backbone...
rootPropagated: A=1
rootReduced: B=0 B=1 B=2 B=3
done in 0,012s with 5 SAT calls.
```

Automated Support



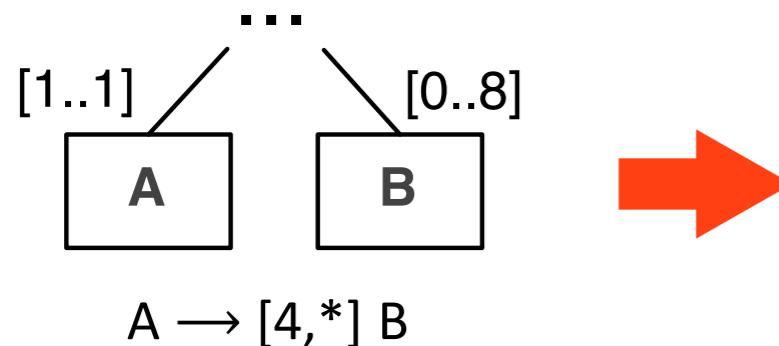
Resulting generated code:

```
#(1,1,[A.1]);
#(1,1,[B.0, B.1, B.2, B.3, B.4, B.5, B.6, B.7, B.8]);
(A.1 => B.4 | B.5 | B.6 | B.7 | B.8);
```



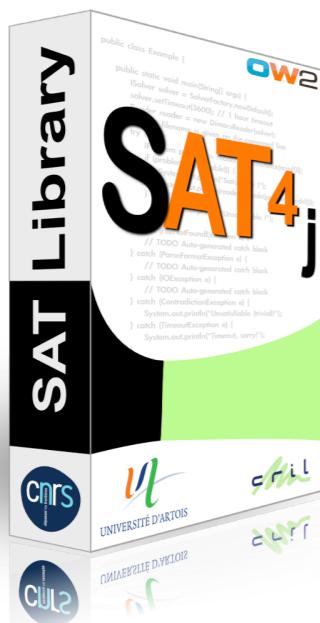
```
Macintosh:cardFMs clement$ java -jar sat4j.jar fmEvo.txt
computing problem backbone...
rootPropagated: A=1
rootReduced: B=0 B=1 B=2 B=3
done in 0,012s with 5 SAT calls.
```

Automated Support



Generated code:

```
#(1,1,[A.1]);
#(1,1,[B.0, B.1, B.2, B.3, B.4, B.5, B.6, B.7, B.8]);
(A.1 => B.4 | B.5 | B.6 | B.7 | B.8);
```



```
Macintosh:cardFMs clement$ java -jar sat4j.jar fmEvo.txt
computing problem backbone...
rootPropagated: A=1
rootReduced: B=0 B=1 B=2 B=3
done in 0,012s with 5 SAT calls.

$> #explain -B=2
A.1=>B.4|B.5|B.6|B.7|B.8
$> |
```

Evaluation: Scalability

Evaluation: Scalability

- Random consistent feature model generation

Evaluation: Scalability

- Random consistent feature model generation
- Edit generation

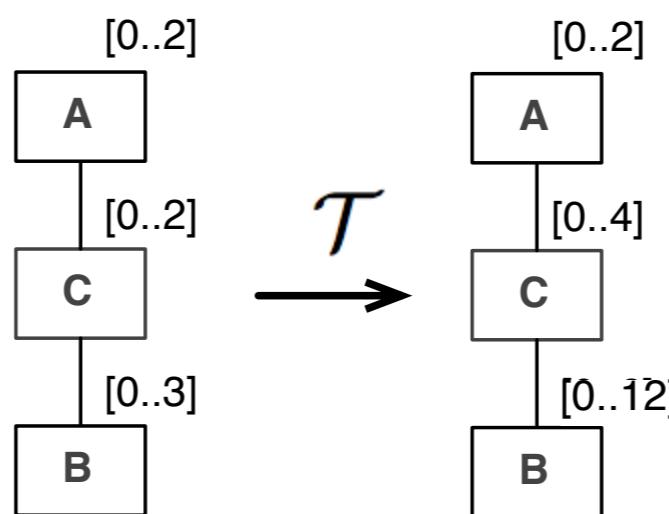
	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	NA

Evaluation: Scalability

- Random consistent feature model generation
- Edit generation

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	NA

- Optimization: only one parent feature considered

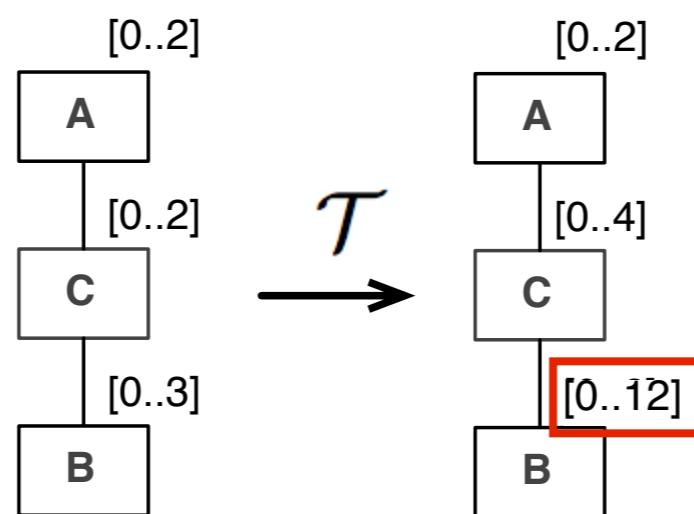


Evaluation: Scalability

- Random consistent feature model generation
- Edit generation

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	NA

- Optimization: only one parent feature considered

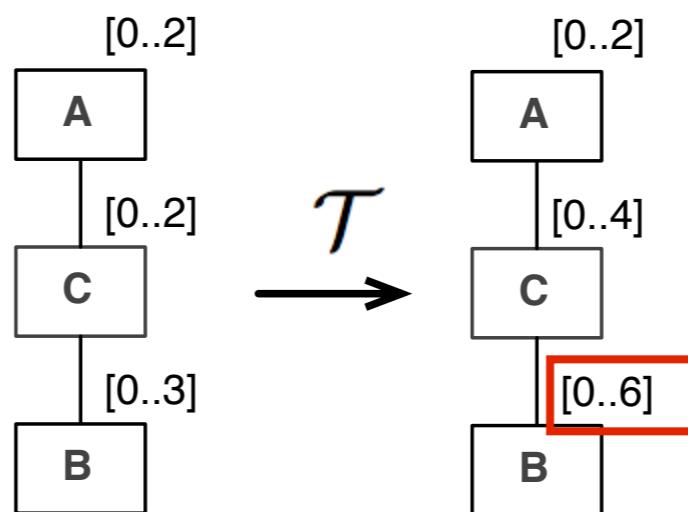


Evaluation: Scalability

- Random consistent feature model generation
- Edit generation

	Add	Remove	Update	Move
Feature	✓	✓	✓	!
Feature Cardinality	NA	NA	!	NA
Group Cardinality	NA	NA	!	NA
Constraint	!	✓	!	NA

- Optimization: only one parent feature considered



Experiment

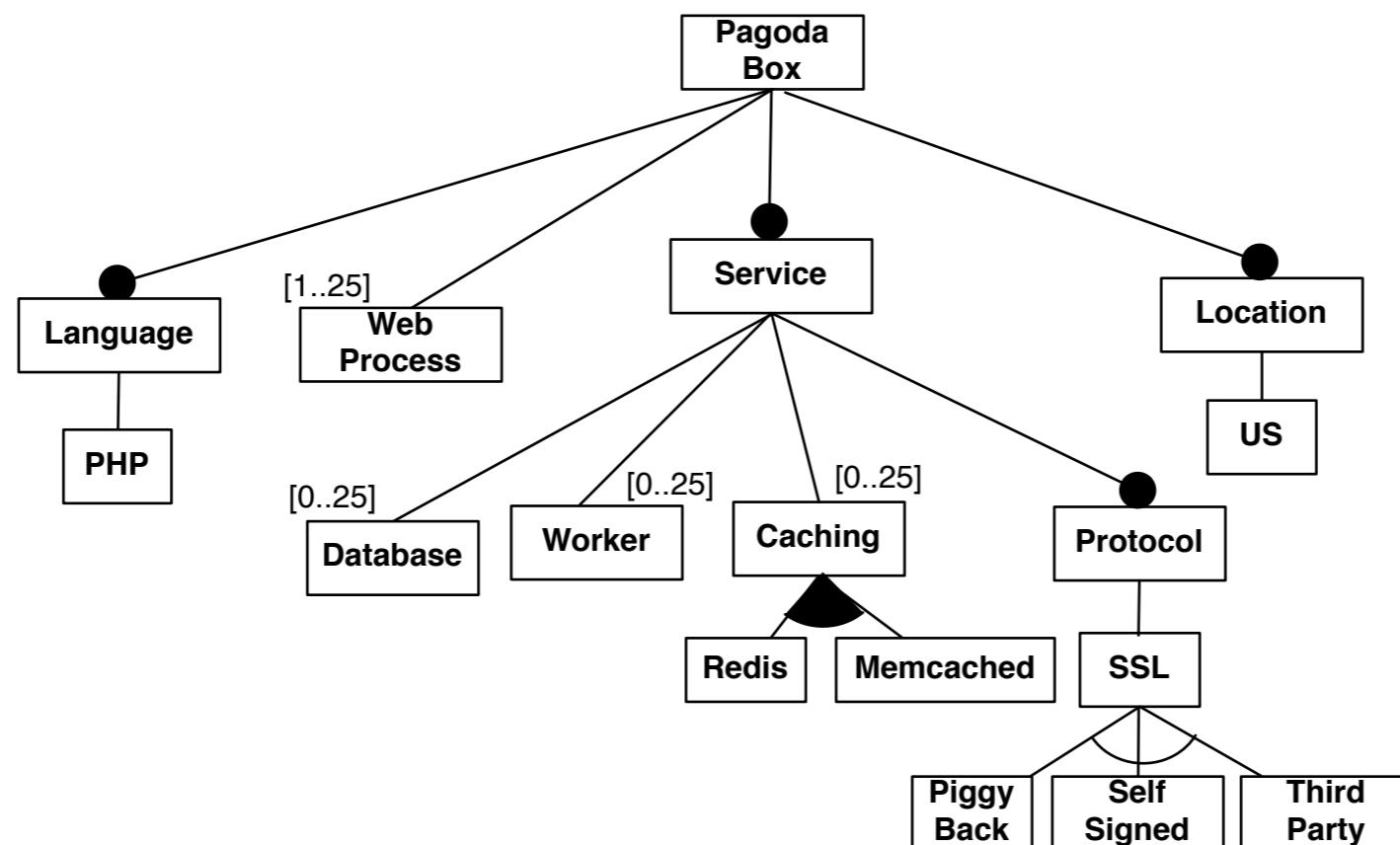
- # edits $\in [1..nbFeatures/20]$

Experiment

- # edits $\in [1..nbFeatures/20]$
- Cardinality upper bound $\in [1..30]$ for 1/10 feature

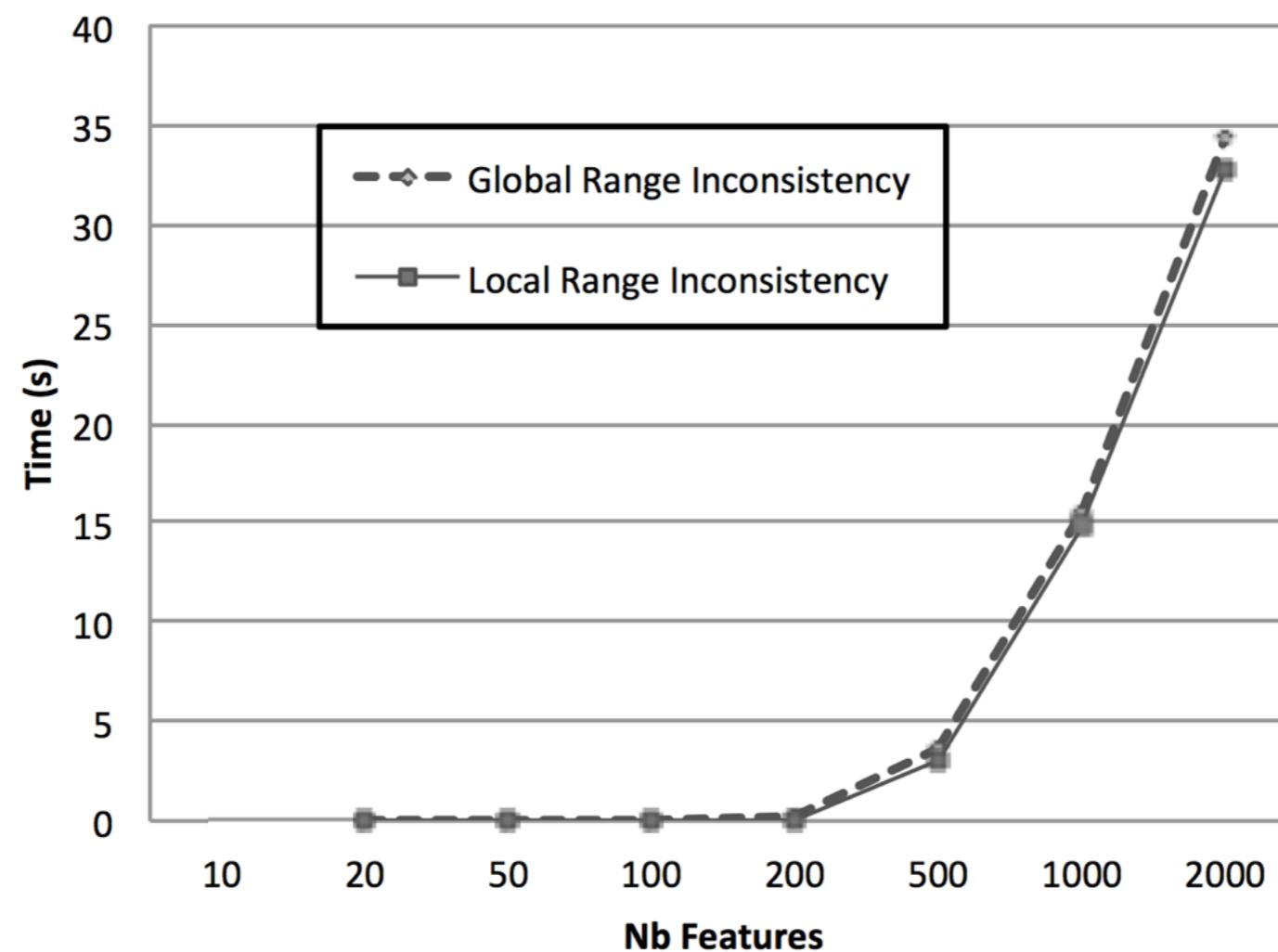
Experiment

- # edits $\in [1..nbFeatures/20]$
- Cardinality upper bound $\in [1..30]$ for 1/10 feature



Experiment

- # edits $\in [1..nbFeatures/20]$
- Cardinality upper bound $\in [1..30]$ for 1/10 feature



Research Goals

- Manage Cloud variability 
- Guarantee environment independance 
- Provide a flexible solution 
- Deliver an automated support 
- Maintain consistency

Research Goals

- Manage Cloud variability 
- Guarantee environment independance 
- Provide a flexible solution 
- Deliver an automated support 
- Maintain consistency 

Agenda

I. Introduction

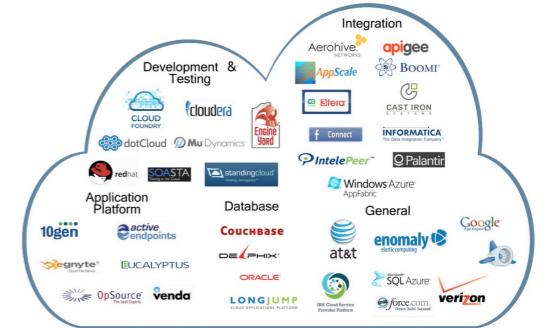
II. Contributions

- Cloud environments variability modeling
- SALOON
- Consistency checking for evolving cloud models

III. Conclusion and Perspectives

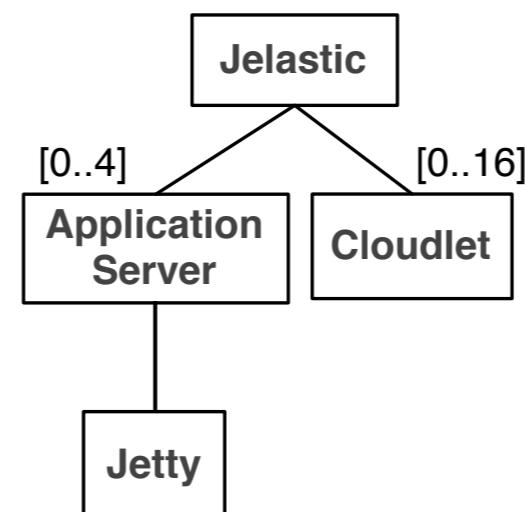
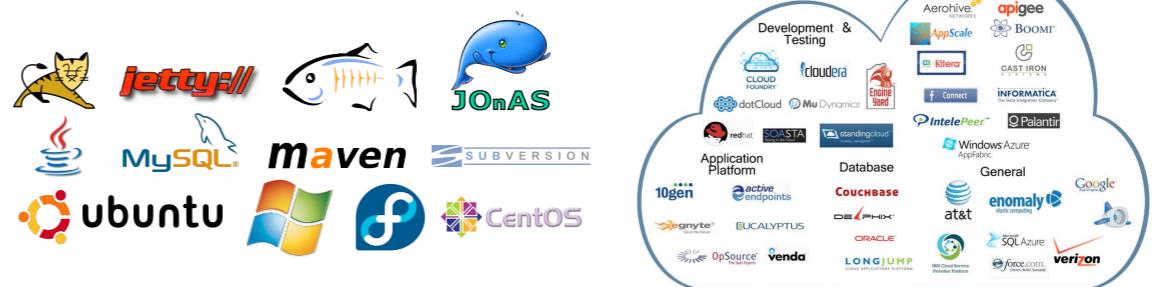
Summary

- Need to select and configure Cloud environments
- Deal with Cloud variability



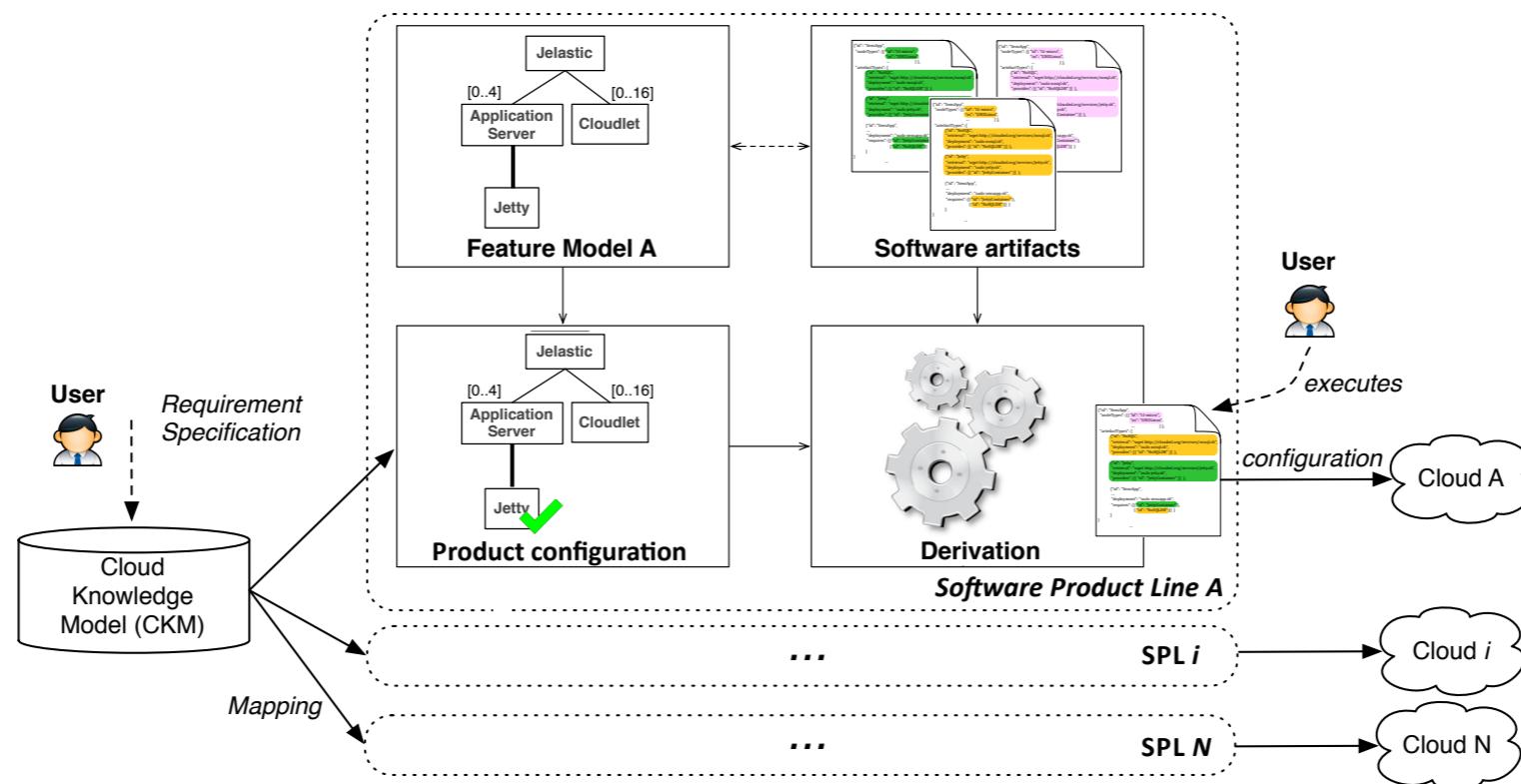
Summary

- Need to select and configure Cloud environments
- Deal with Cloud variability
- Contributions in modeling



Summary

- Need to select and configure Cloud environments
- Deal with Cloud variability
- Contributions in modeling, automated support



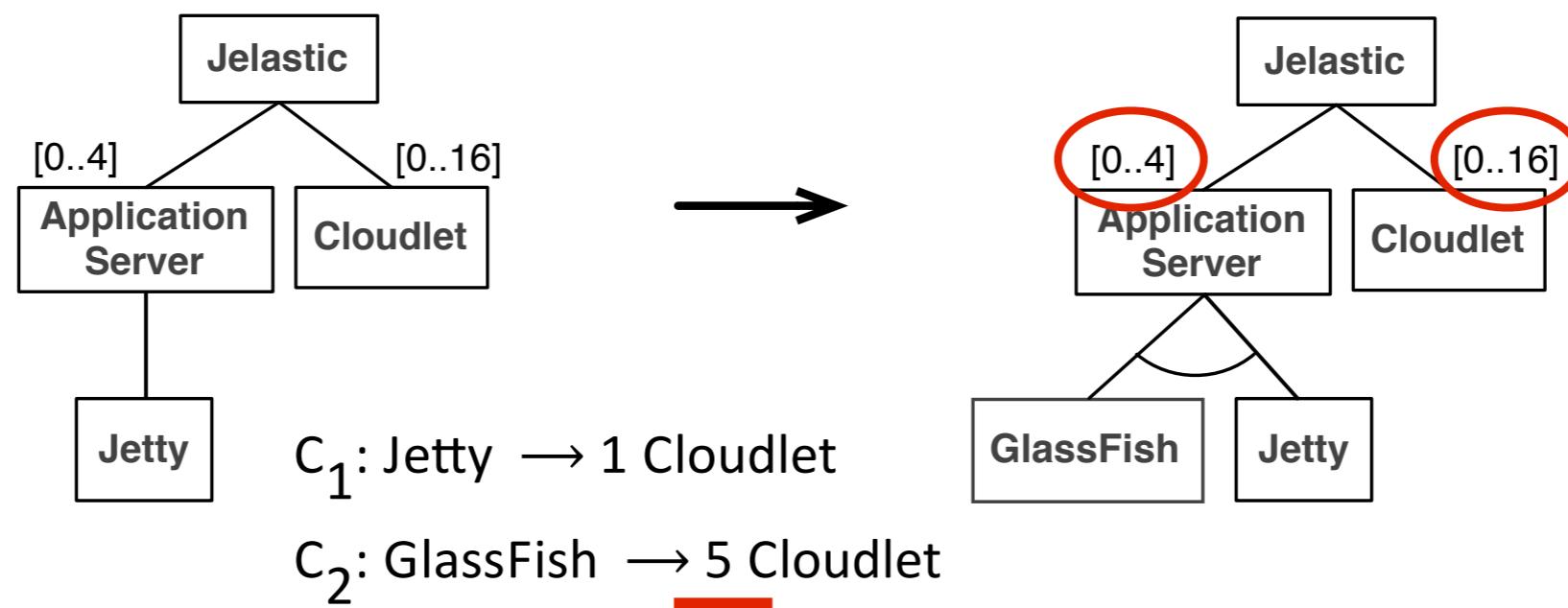
Summary

- Need to select and configure Cloud environments

- Deal with Cloud variability



- Contributions in modeling, automated support and evolution management



Short Term Perspectives

Short Term Perspectives

- Improve SALOON practicality
 - Dedicated GUI or DSL

Short Term Perspectives

- Improve SALOON practicality
 - Dedicated GUI or DSL
- Improve SALOON interoperability
 - Input and output

Short Term Perspectives

- Improve SALOON practicality
 - Dedicated GUI or DSL
- Improve SALOON interoperability
 - Input and output
- Extend cardinality support
 - Other domains/constraints

Long Term Perspectives

Long Term Perspectives

- Consider multi-cloud configurations
 - Model slicing

Long Term Perspectives

- Consider multi-cloud configurations
 - Model slicing
- SALOON @ runtime
 - Monitoring, DSPL, traceability

Long Term Perspectives

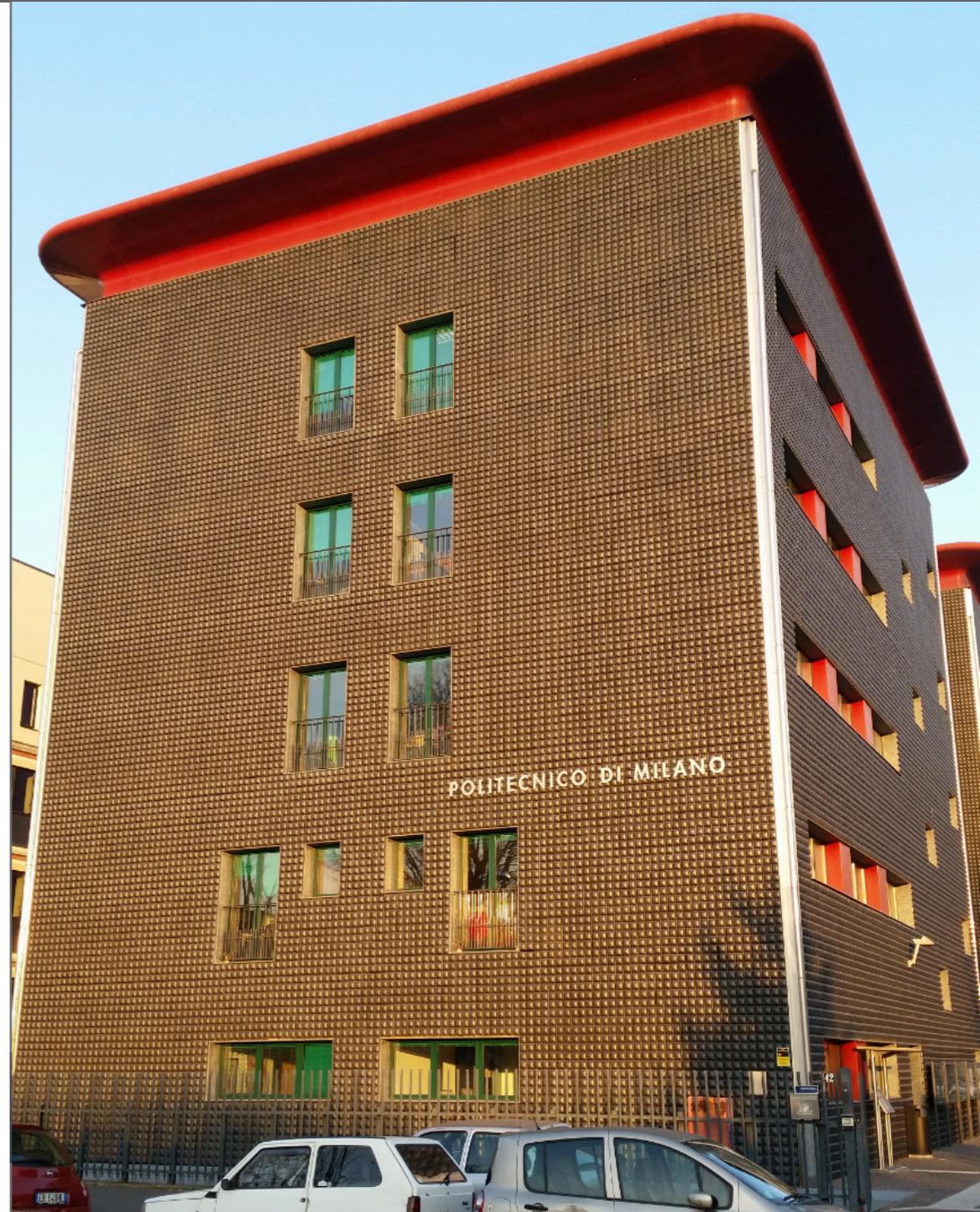
- Consider multi-cloud configurations
 - Model slicing
- SALOON @ runtime
 - Monitoring, DSPL, traceability
- SALOON-as-a-Service

Currently



<http://home.deib.polimi.it/quinton/>

- Dynamic Software Product Lines
 - Cyber-Physical Systems
- Evolution @ runtime
 - Consistency of the SPL/derived product



Cloud Environment Selection and Configuration: A Software Product Lines-Based Approach

Journal

SALOON: A Platform for Selecting and Configuring Cloud Environments.

Clément Quinton, Daniel Romero and Laurence Duchien. *In Software: Practice and Experience journal (SPE). Accepted with minor revisions, September 2014.*

Workshops

SPLEMMMA: A Generic Framework for Controlled Evolution of Software Product Lines. Daniel Romero, Simon Urli, **Clément Quinton**, Mireille Blay-Fornarino, Philippe Collet, Laurence Duchien and Sébastien Mosser. *In Proceedings of the 17th International Software Product Line Conference volume 2, MAPLE'13. Tokyo, Japan, August 2013.*

Towards Multi-Cloud Configurations Using Feature Models and Ontologies. **Clément Quinton**, Nicolas Haderer, Romain Rouvoy and Laurence Duchien. *In Proceedings of the 1st International Workshop on Multi-Cloud Applications and Federated Clouds, Multi-Cloud'13. Prague, Czech Republic, April 2013.*

Using Feature Modelling and Automations to Select among Cloud Solutions. **Clément Quinton**, Patrick Heymans and Laurence Duchien. *In Proceedings of the 3rd International Workshop on Product Line Approaches in Software Engineering, PLEASE'12. Zurich, Switzerland, June 2012.*

Leveraging Feature Models to Configure Virtual Appliances. **Clément Quinton**, Romain Rouvoy and Laurence Duchien. *In Proceedings of the 2nd International Workshop on Cloud Computing Platforms, CloudCP'12. Bern, Switzerland, April 2012.*

Conferences

Consistency Checking for the Evolution of Cardinality-based Feature Models. **Clément Quinton**, Andreas Pleuss, Daniel Le Berre, Laurence Duchien and Goetz Botterweck. *In Proceedings of the 18th International Software Product Line Conference, SPLC'14. Florence, Italy, Septembre 2014.*

Automated Selection and Configuration of Cloud Environments Using Software Product Lines Principles. **Clément Quinton**, Daniel Romero and Laurence Duchien. *In Proceedings of the 7th IEEE International Conference on Cloud Computing, CLOUD'14. Anchorage, Alaska, July 2014.*

Cardinality-Based Feature Models With Constraints: A Pragmatic Approach. **Clément Quinton**, Daniel Romero and Laurence Duchien. *In Proceedings of the 17th International Software Product Line Conference, SPLC'13. Tokyo, Japan, August 2013.*

