

Safely Managing Data Variety in Big Data Software Development

Thomas Cerqueus (Université de Lyon, INSA Lyon, LIRIS)

Eduardo Cunha de Almeida (Federal University of Paraná)

Stefanie Scherzinger (OTH Regensburg)

1st International Workshop on BIG Data Software Engineering, 23 May 2015

PaaS over NoSQL Databases

- No explicit DB schema
- Entities are defined at the application level
 - Class → Entity
 - Attribute → Field
- Developers are in charge of defining and maintaining the DB

Object Mappers

```
package project;

import com.googlecode.objectify.annotation.Entity;
import com.googlecode.objectify.annotation.Id;

@Entity
public class Player {

    @Id
    String login;

    String name;

    Integer level;

    Float score;

}
```

Object Mappers

```
package project;

import com.googlecode.objectify.annotation.AlsoLoad;
import com.googlecode.objectify.annotation.Entity;
import com.googlecode.objectify.annotation.Id;
import com.googlecode.objectify.annotation.Ignore;

@Entity
public class Player {

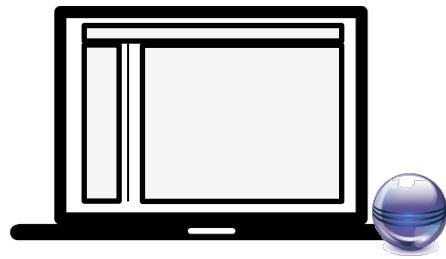
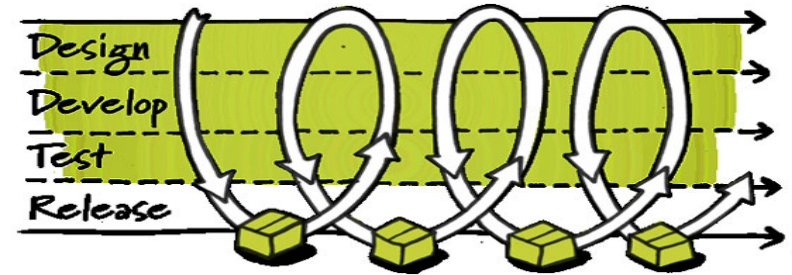
    @Id
    String login;

    String name;

    @AlsoLoad("level")
    Integer rank;

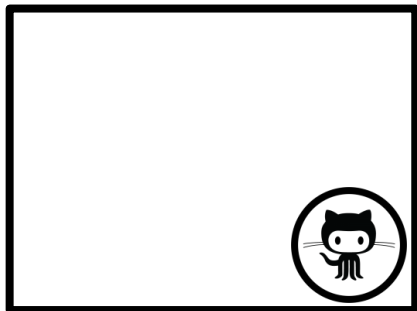
    @Ignore
    Float score;

}
```



development IDE

↕ *commit*



code repository

Development Environment

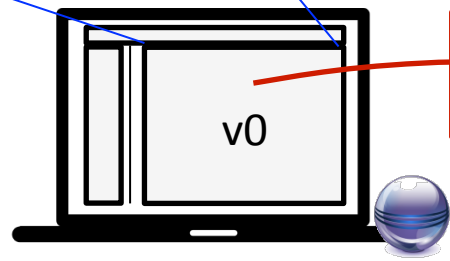
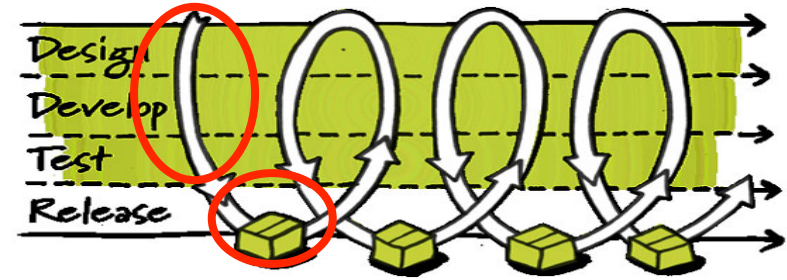


Production Environment

```

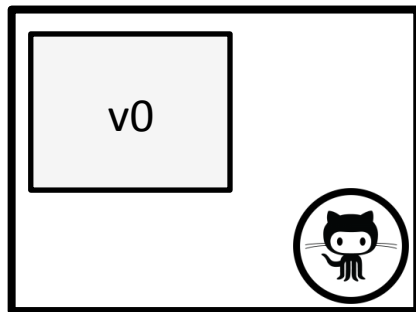
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    String rank;
15 }
16

```



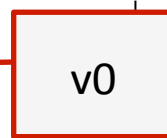
development IDE

↕ *commit*

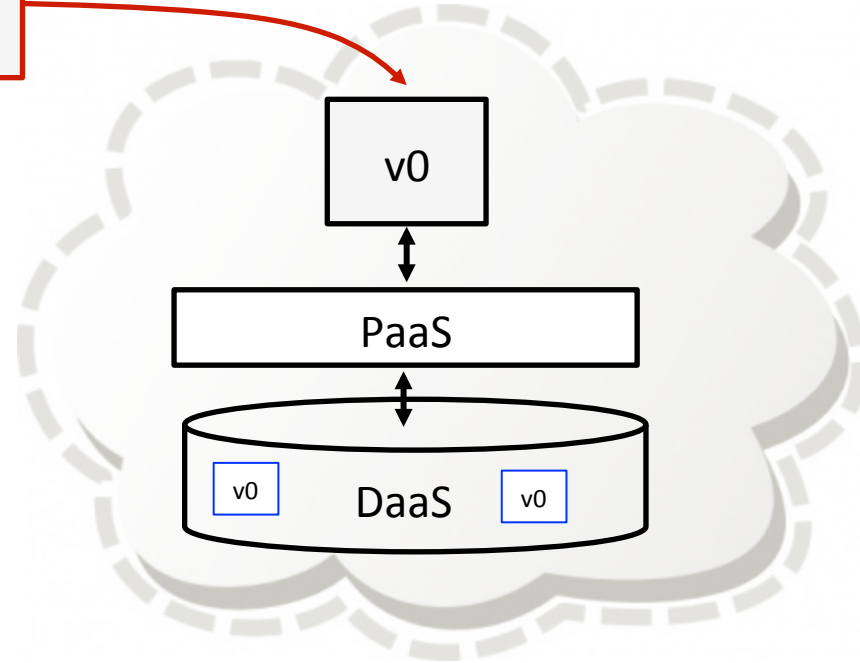


code repository

Development Environment



deploy



Production Environment

```

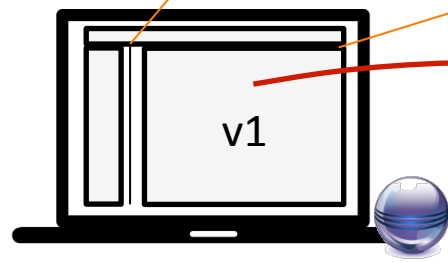
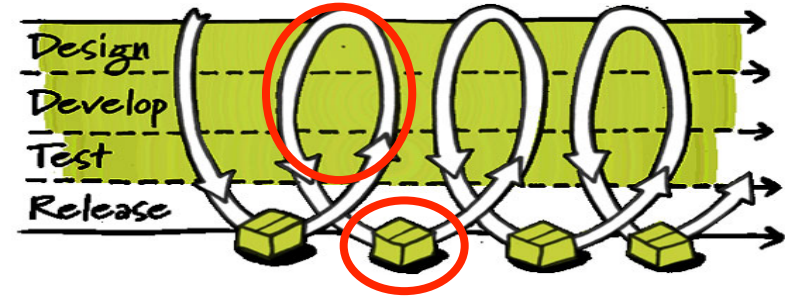
Player.java
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    String rank;
15 }
16

```

```

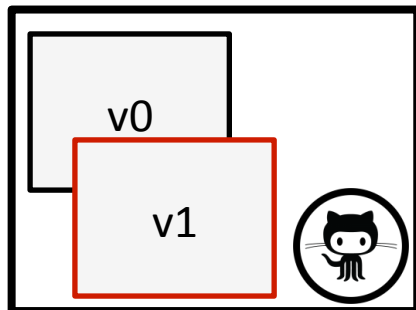
Player.java
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    Integer rank;
15 }
16

```



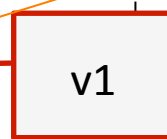
development IDE

↕ *commit*

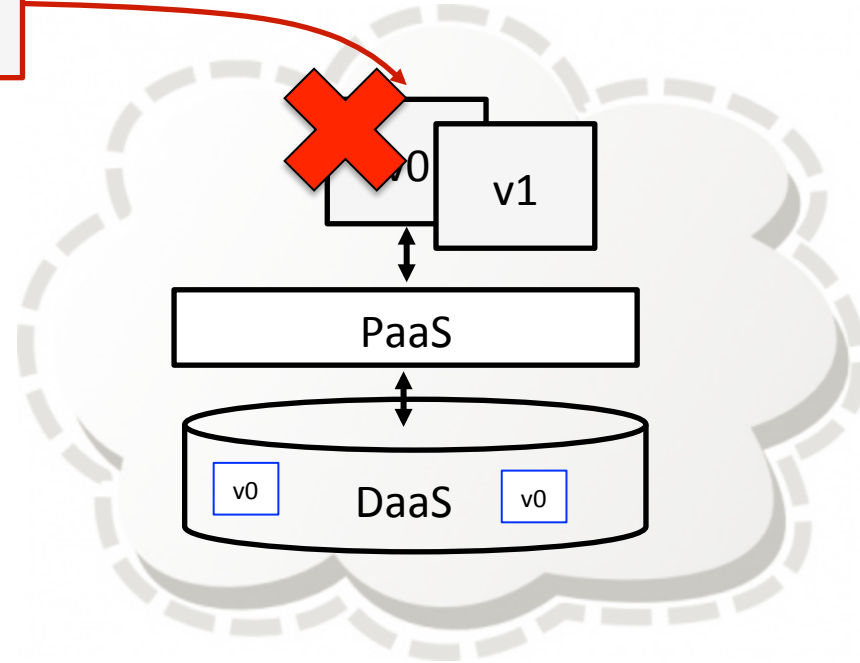


code repository

Development Environment



deploy

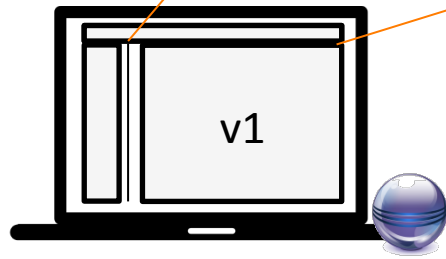
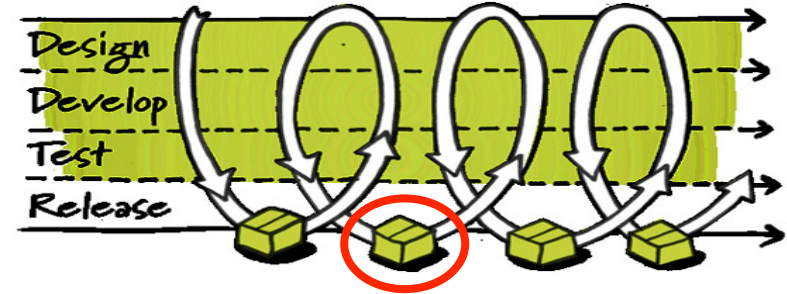


Production Environment

```

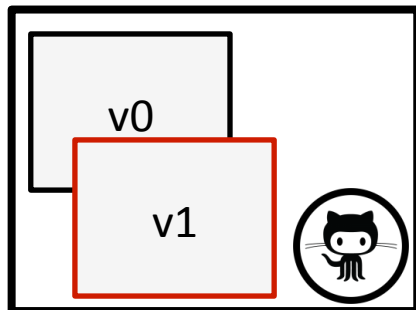
Player.java
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    String rank;
15 }
16
Player.java
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    Integer rank;
15 }
16

```



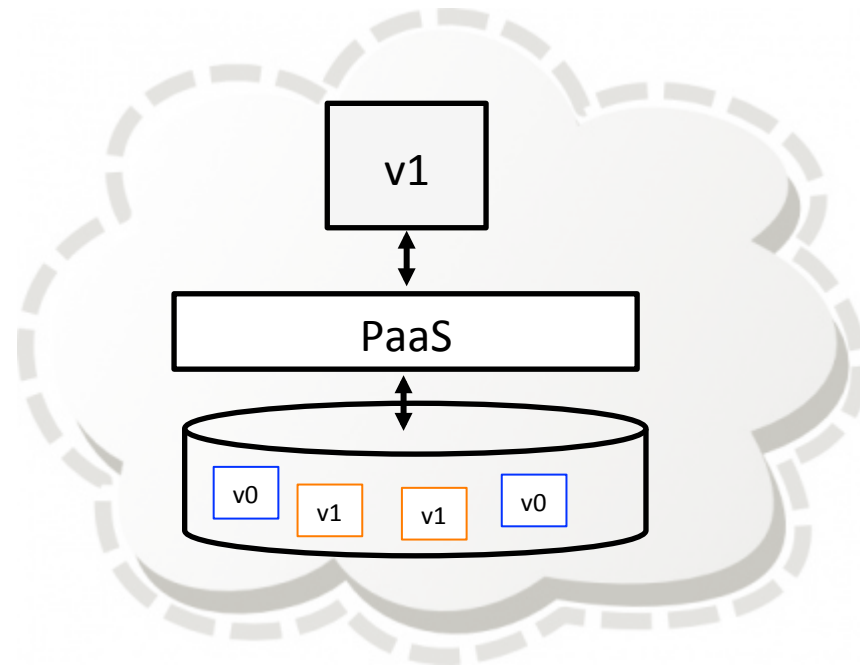
development IDE

↕ commit



code repository

Development Environment

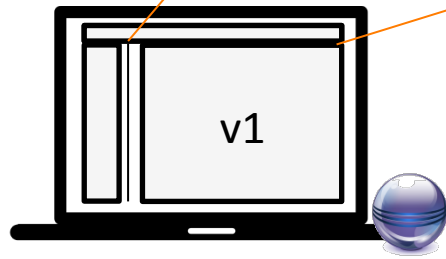
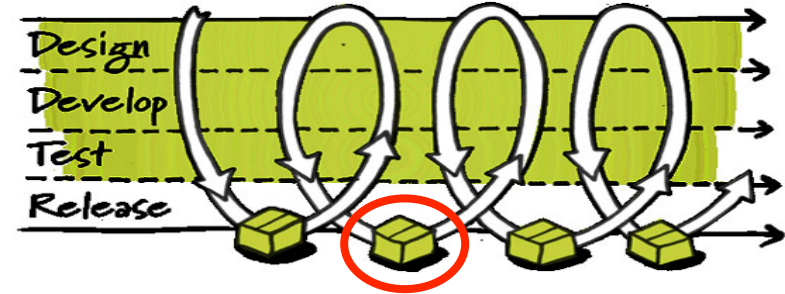


Production Environment


```

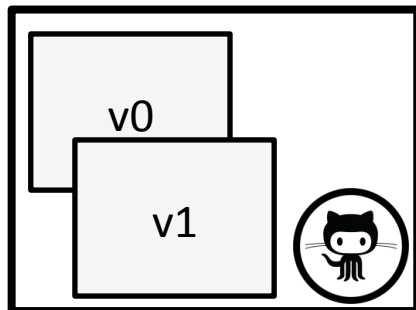
Player.java
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    String rank;
15 }
16
Player.java
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    Integer rank;
15 }
16

```



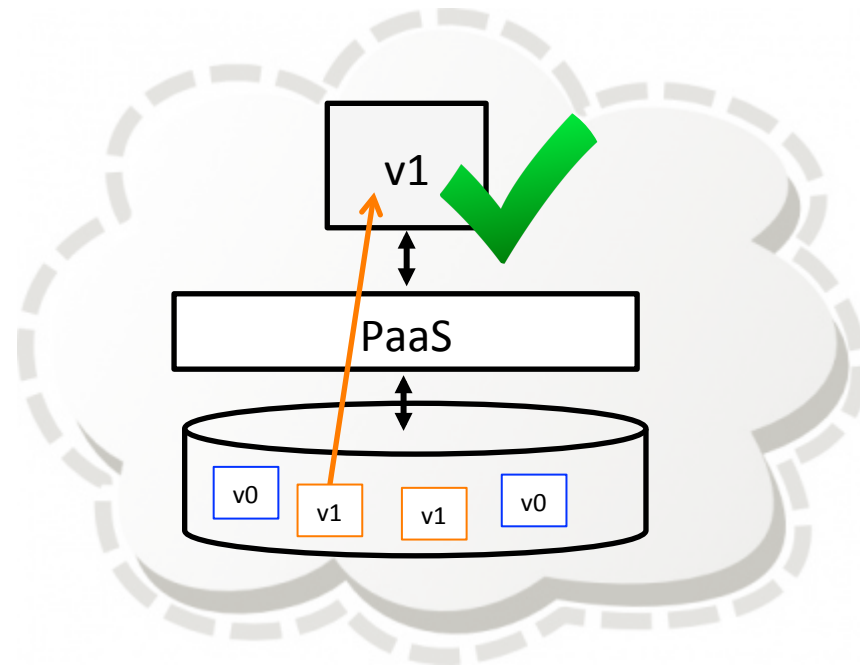
development IDE

↕ commit



code repository

Development Environment



Production Environment

```

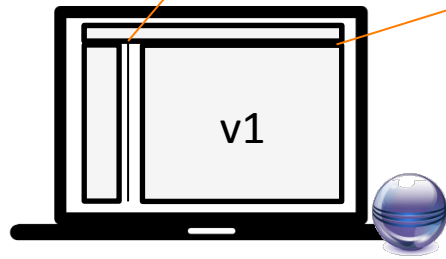
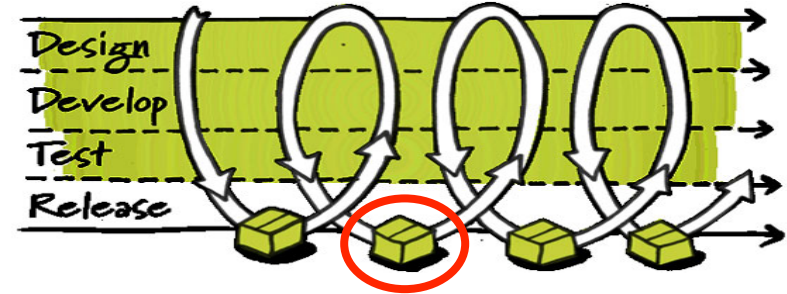
Player.java
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    String rank;
15 }
16

```

```

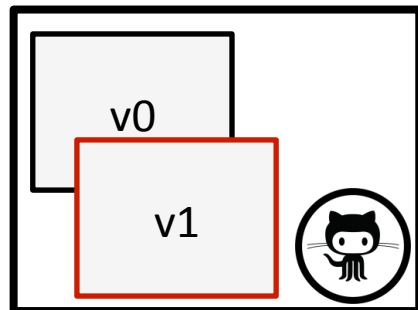
Player.java
1 package com.example;
2
3 import com.googlecode.objectify.annotat
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    Integer rank;
15 }
16

```



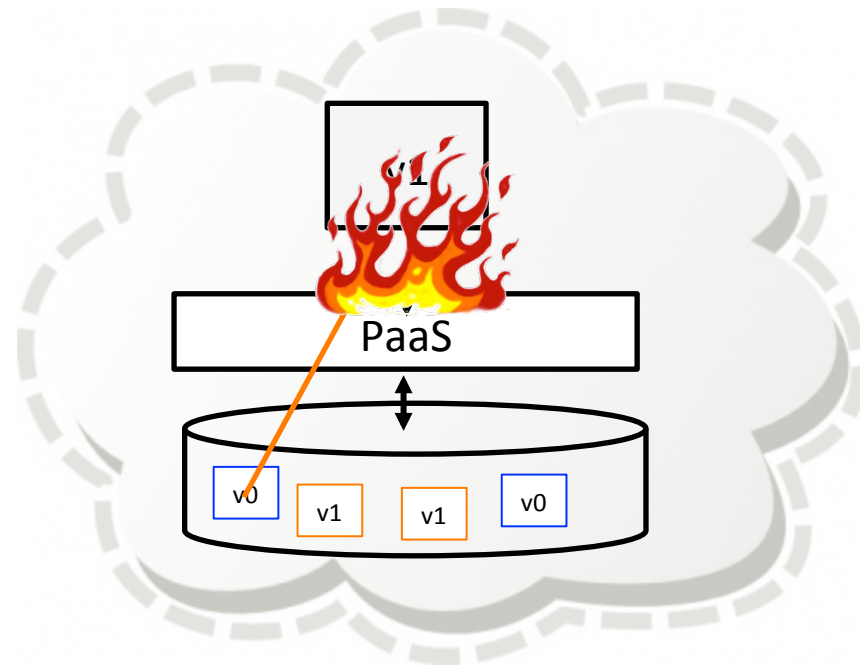
development IDE

↕ commit



code repository

Development Environment



Production Environment

Checking Rules (1/2)

$$\frac{C_W \vdash @Id \text{ type } att;}{C_R \circ C_W \vdash @Id \text{ type } att; \quad ok}$$

$$\frac{C_W \vdash \text{ type } att;}{C_R \circ C_W \vdash \text{ type } att; \quad ok}$$

$$\frac{}{C_R \circ C_W \vdash \text{ type } att; \quad ok} \left(\begin{array}{l} att \text{ not declared} \\ \text{in } C_W \end{array} \right)$$

$$\frac{C_W \vdash @Ignore \text{ type}_1 \text{ att};}{C_R \circ C_W \vdash \text{ type}_2 \text{ att}; \quad ok}$$

$$\frac{C_W \vdash \text{ type } att;}{C_R \circ C_W \vdash @IgnoreLoad \text{ type } att; \quad ok}$$

$$\frac{C_W \vdash \text{ type } att;}{C_R \circ C_W \vdash @IgnoreSave \text{ type } att; \quad ok}$$

$$\frac{C_W \vdash \text{ type } att;}{C_R \circ C_W \vdash @Ignore \text{ type } att; \quad ok}$$

Checking Rules (2/2)

$$\frac{C_W \vdash \text{type } att_1;}{C_R \circ C_W \vdash @AlsoLoad("att_2") \text{ type } att_1; \quad ok} \quad (att_2 \text{ not declared in } C_W)$$

$$\frac{C_W \vdash \text{type } att_2;}{C_R \circ C_W \vdash @AlsoLoad("att_2") \text{ type } att_1; \quad ok} \quad (att_1 \text{ not declared in } C_W)$$

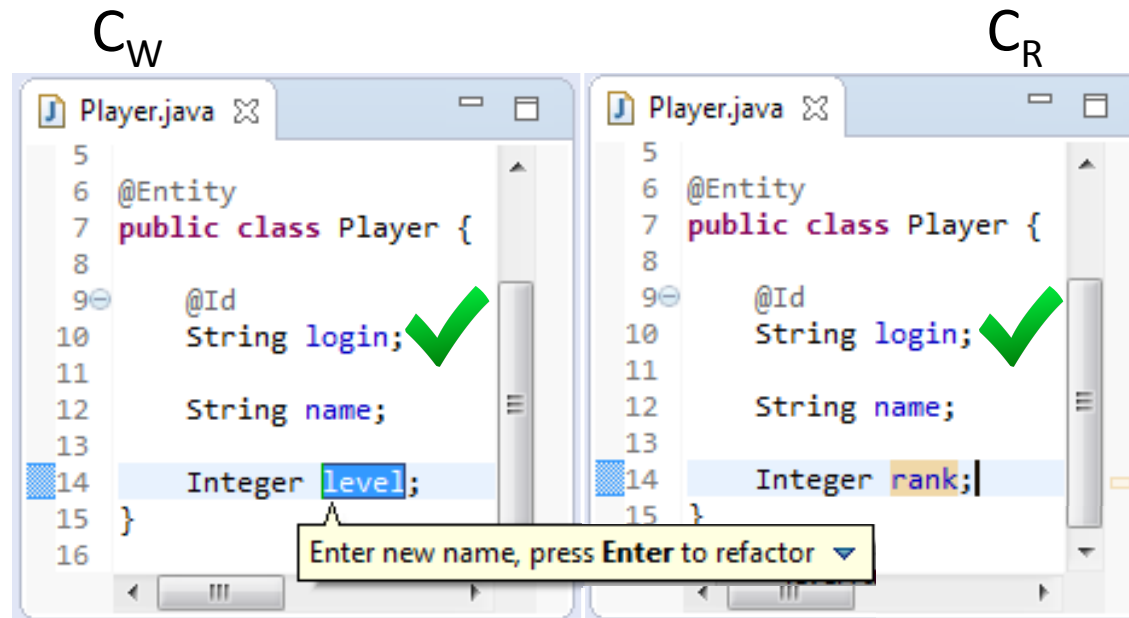
$$\frac{C_W \vdash @annot \text{ type } att_1 \quad C_W \vdash \text{type } att_2;}{C_R \circ C_W \vdash @AlsoLoad("att_2") \text{ type } att_1; \quad ok} \quad (@annot \text{ is } @Ignore \text{ or } @IgnoreSave)$$

$$\frac{C_W \vdash \text{type } att_1 \quad C_W \vdash @annot \text{ type } att_2;}{C_R \circ C_W \vdash @AlsoLoad("att_2") \text{ type } att_1; \quad ok} \quad (@annot \text{ is } @Ignore \text{ or } @IgnoreSave)$$

$$\frac{}{C_R \circ C_W \vdash @AlsoLoad("att_2") \text{ type } att_1; \quad ok} \quad (att_1 \text{ and } att_2 \text{ not declared in } C_W)$$

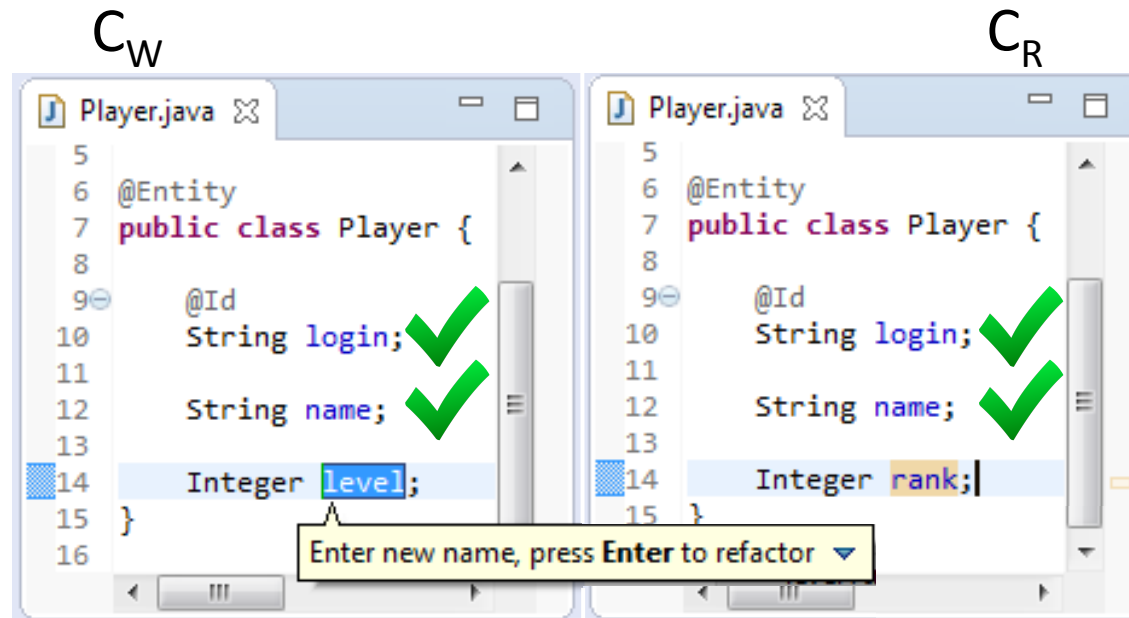
$$\frac{C_W \vdash @annot_1 \text{ type } att_1;}{C_R \circ C_W \vdash @annot_2 \text{ type } att_1; \quad ok} \quad \left(\begin{array}{l} @annot_1 \text{ and } @annot_2 \text{ are any of} \\ @Ignore, @IgnoreSave, @IgnoreLoad \end{array} \right)$$

Example



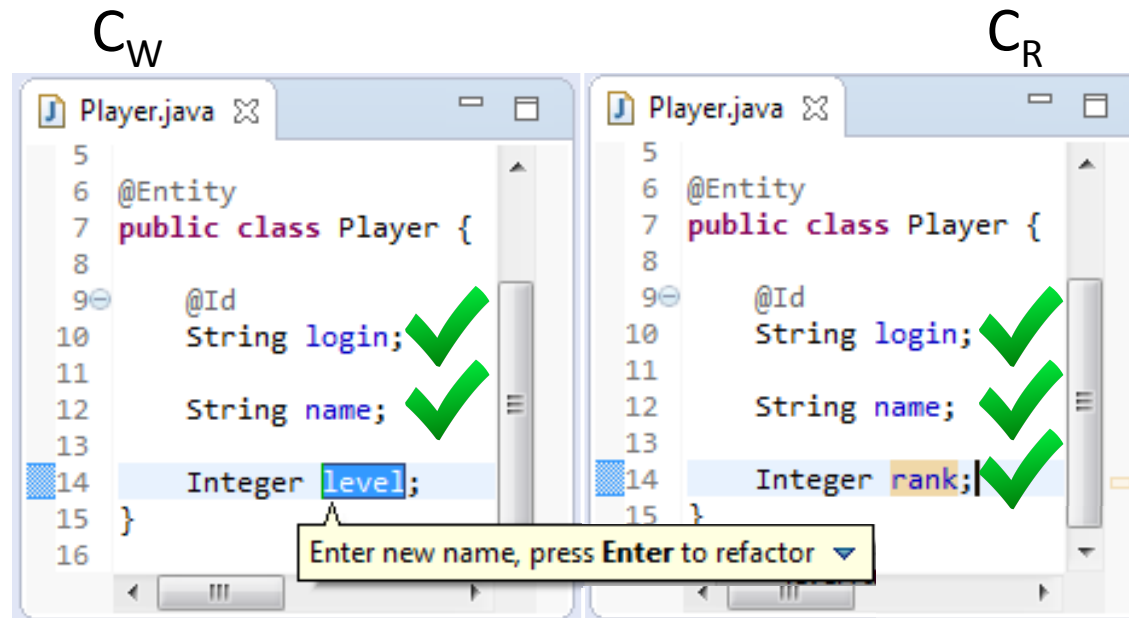
$$\frac{C_W \vdash @Id \text{ type } att;}{C_R \circ C_W \vdash @Id \text{ type } att; \quad ok}$$

Example



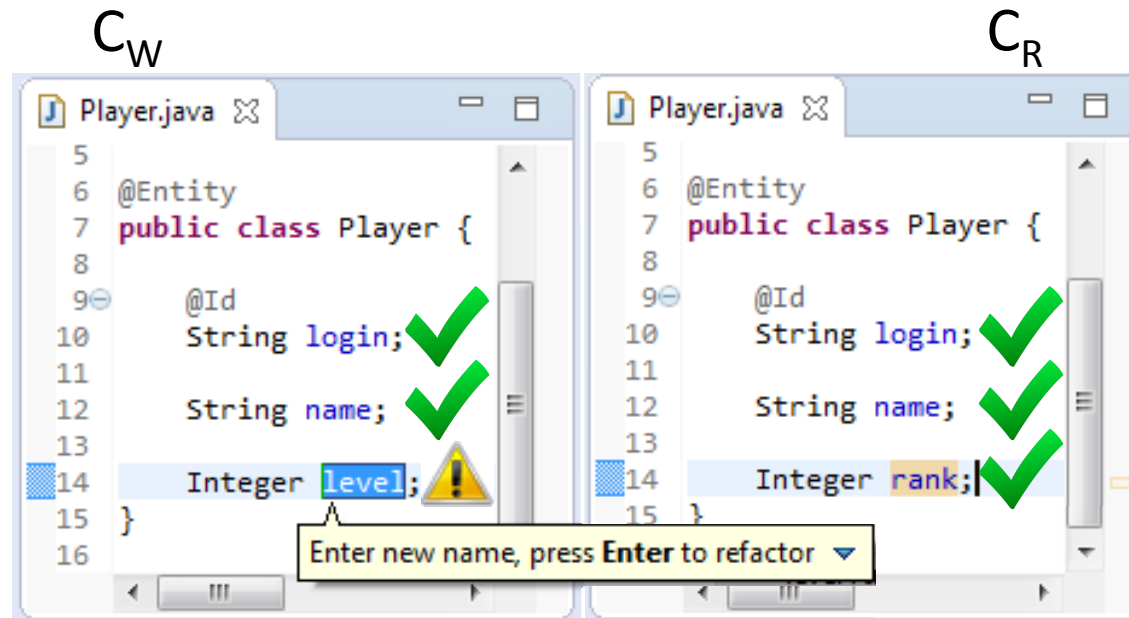
$$\frac{C_W \vdash \text{type att;}}{C_R \circ C_W \vdash \text{type att;}} \quad ok$$

Example

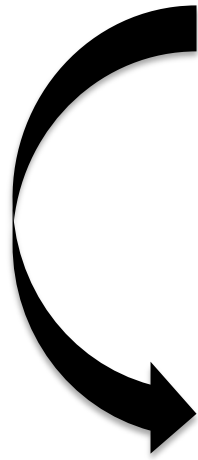


$$\frac{}{C_R \circ C_W \vdash \text{type } att; \quad ok} \quad \left(\begin{array}{l} \text{att not declared} \\ \text{in } C_W \end{array} \right)$$

Example



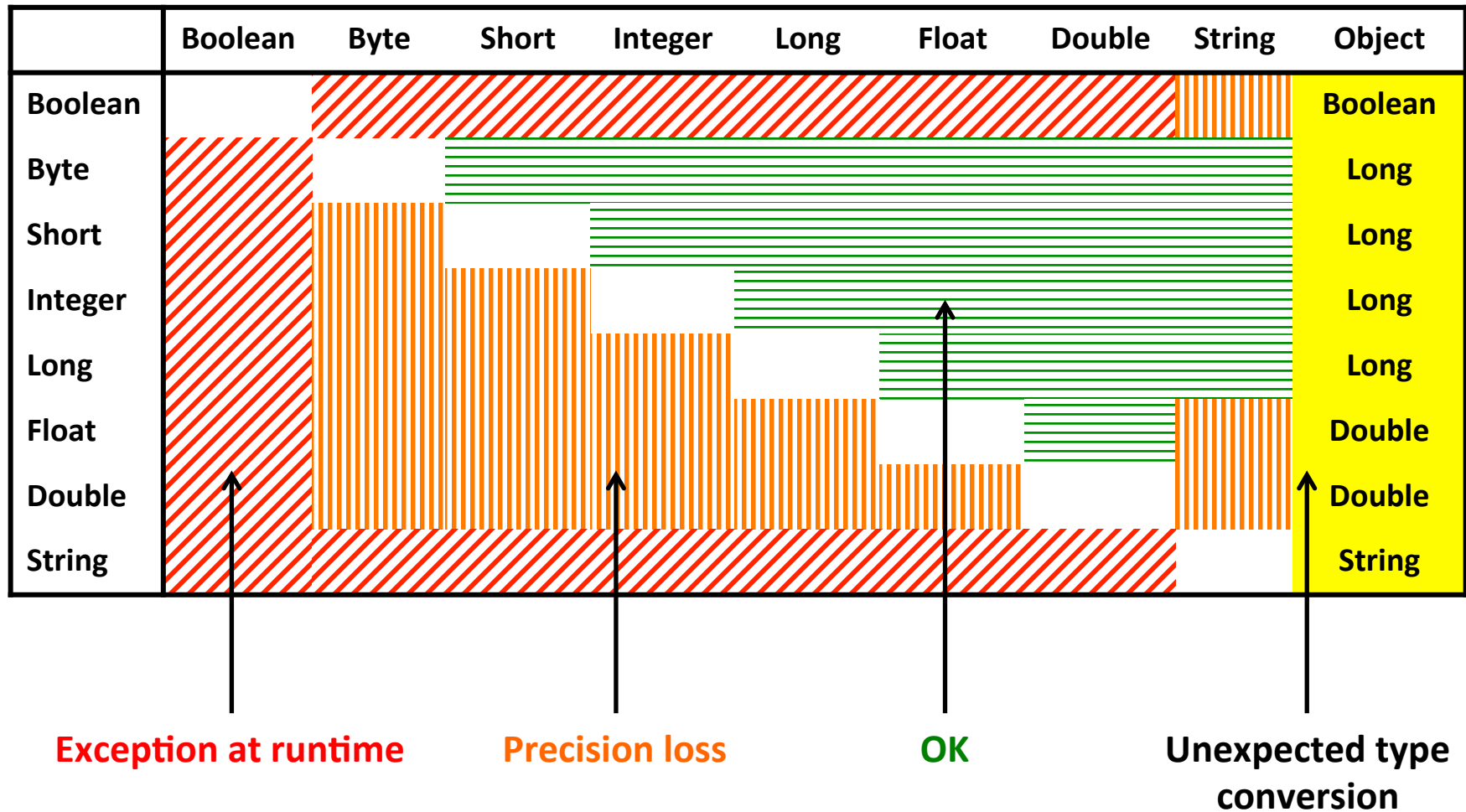
Type Checking Extension



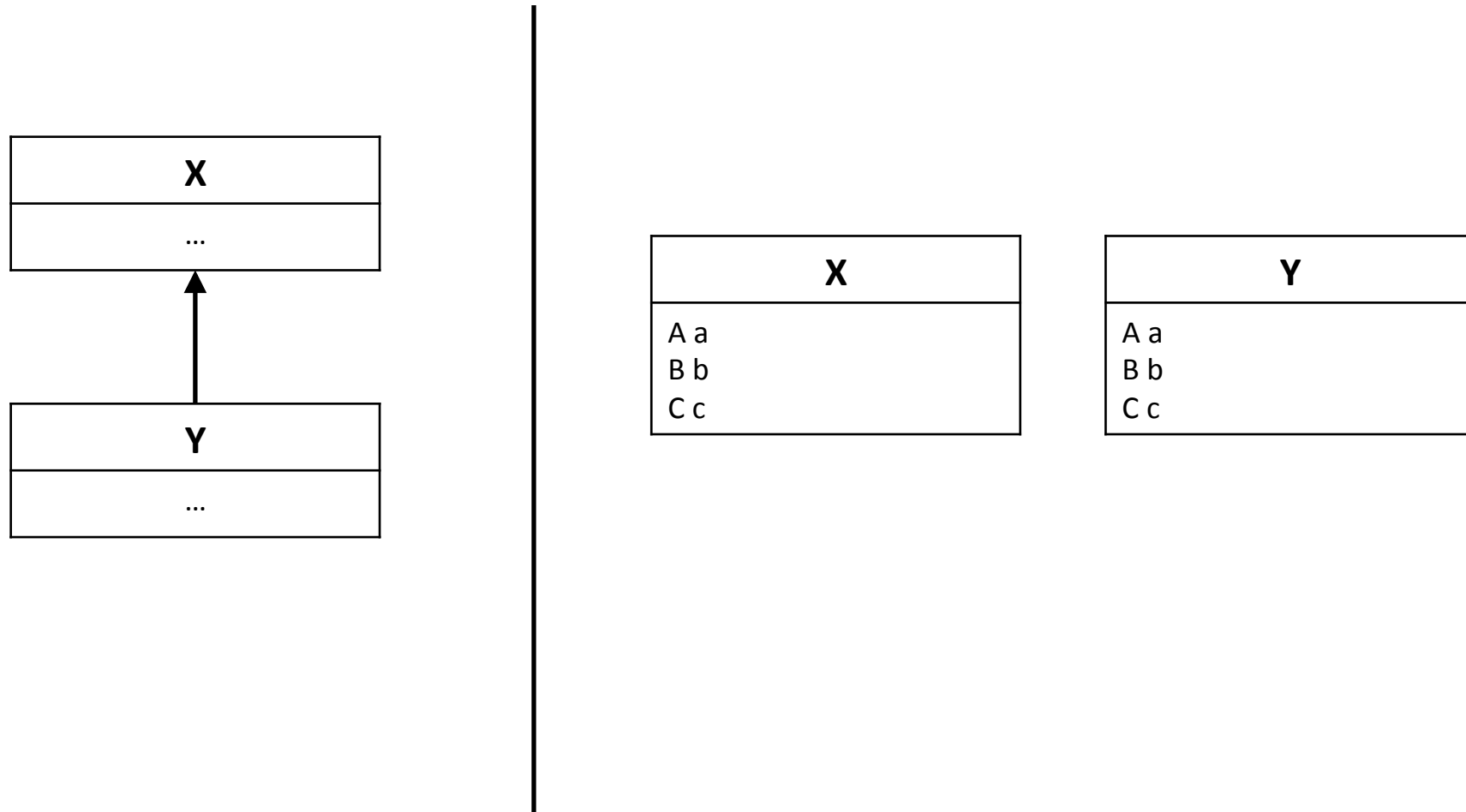
$$\frac{C_W \vdash \textit{type att};}{C_R \circ C_W \vdash \textit{type att}; \quad \textit{ok}}$$

$$\frac{C_W \vdash \textit{type att};}{C_R \circ C_W \vdash \textit{type}' \textit{ att}; \quad \textit{ok}} \quad (\textit{type} \triangleright \textit{type}')$$

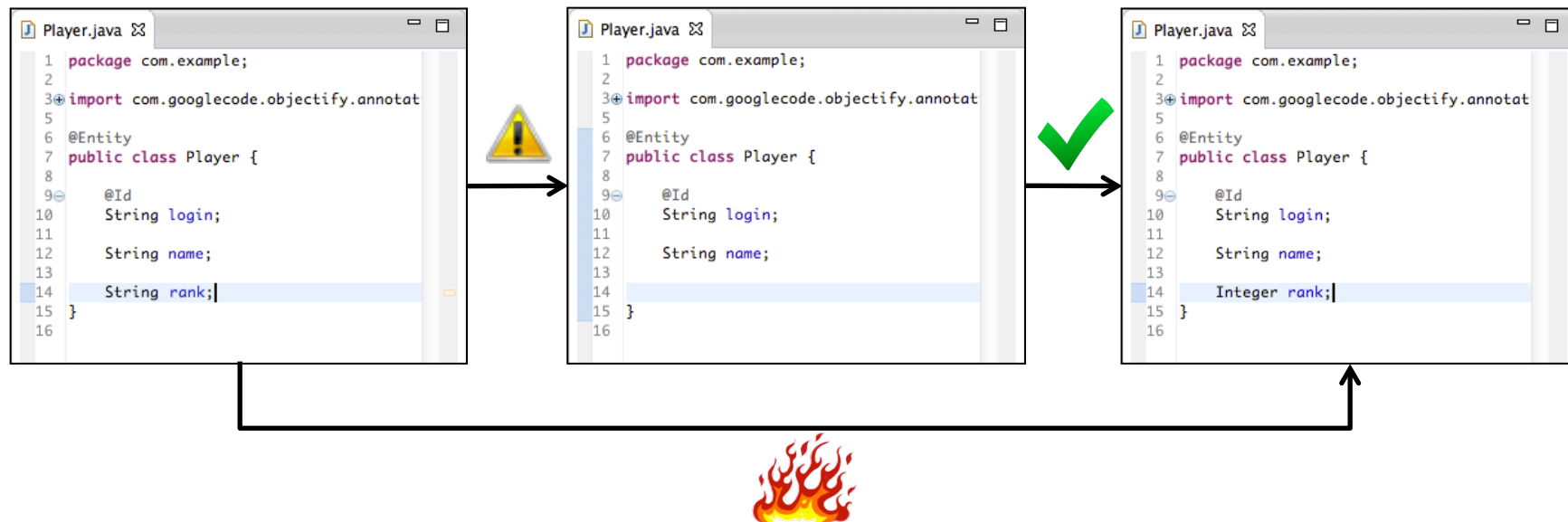
Type Checking Extension – Primitive Types



Type Checking Extension – Complex Types



Chains of Declarations



Conclusion

ControVol: schema evolution checking framework for NoSQL databases

- Takes action prior to application release
- Integrated in the IDE (Eclipse)
- Warns developers against perilous evolutions: addition/deletion/renaming of attributes, re-typing
- Suggests quick fixes

Conclusion

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows a project structure with a package named 'player'. The main editor displays the code for 'Player.java', which includes an import for 'com.googlecode.objectify.annotation.Entity' and a class definition for 'Player' with attributes 'login', 'name', and 'level'. A warning icon is present next to the 'level' attribute. A 'Quick Fix' dialog is open, offering three options to resolve the warning: 'Add annotation @AlsoLoad("rank") to prevent data loss', 'Add annotation @Ignore to explicitly remove attribute 'rank'', and 'Restore attribute 'rank''. The 'Problems' view at the bottom shows a table with one warning item.

```
1 package com.example;
2
3 import com.googlecode.objectify.annotation.Entity;
4
5
6 @Entity
7 public class Player {
8
9     @Id
10    String login;
11
12    String name;
13
14    String level;
15
16 }
17
```

Quick Fix dialog content:

Quick Fix

Select the fix for 'Attribute 'rank' will be lost when loading legacy entities'.

Select a fix:

- Add annotation @AlsoLoad("rank") to prevent data loss
- Add annotation @Ignore to explicitly remove attribute 'rank'
- Restore attribute 'rank'.

Problems:

Resource	Location
<input checked="" type="checkbox"/> Player.java	line 14

Buttons: Select All, Deselect All, Cancel, Finish

Problems view table:

Description	Resource	Path	Location	Type
⚠ Attribute 'rank' will be lost when loading legacy entities	Player.java	/player/src/main/java/com/example	line 14	ControVol Marker

Status bar: Attribute 'rank' will be lost when loading legacy entities

Publications

- T. Cerqueus, E. Cunha de Almeida, S. Scherzinger. **ControVol: Let yesterday's data catch up with today's application code.** In *24th International World Wide Web Conference (WWW 2015)*
- S. Scherzinger, T. Cerqueus, E. Cunha de Almeida. **ControVol: A Framework for Controlled Schema Evolution in NoSQL Application Development.** In *31st International Conference on Data Engineering (ICDE 2015)*
- T. Cerqueus, E. Cunha de Almeida, S. Scherzinger. **Safely Managing Data Variety in Big Data Software Development.** In *1st International Workshop on Big Data Software Engineering (BIGDSE'15)*

Safely Managing Data Variety in Big Data Software Development

Thomas Cerqueus (Université de Lyon, INSA Lyon, LIRIS)

Eduardo Cunha de Almeida (Federal University of Paraná)

Stefanie Scherzinger (OTH Regensburg)

1st International Workshop on BIG Data Software Engineering, 23 May 2015