

Advanced Typing for the Kieker Instrumentation Languages

SSP 2016, Kiel

Reiner Jung Christian Wulf

8th November 2016

iObserve

Kieker



Motivation

Instrumentation of software systems

- Multiple technologies and languages
- Vast set of event types (currently 50)

Kieker instrumentation languages

[Jung et al. 2013]

- Provide language and technology independent instrumentation
- Support
 - **type declaration**
instrumentation record language (IRL)
 - **probe specification & aspect weaving**
instrumentation aspect language (IAL)

Motivation

Instrumentation of software systems

- Multiple technologies and languages
- Vast set of event types (currently 50)

Kieker instrumentation languages

[Jung et al. 2013]

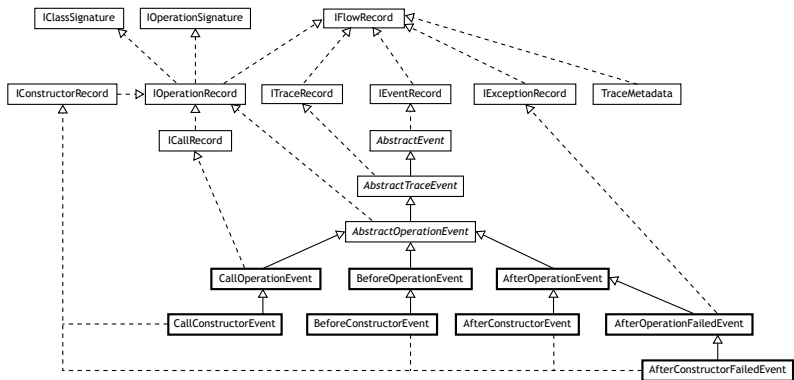
- Provide language and technology independent instrumentation
- Support
 - **type declaration**
instrumentation record language (IRL)
 - **probe specification & aspect weaving**
instrumentation aspect language (IAL)

Two issues

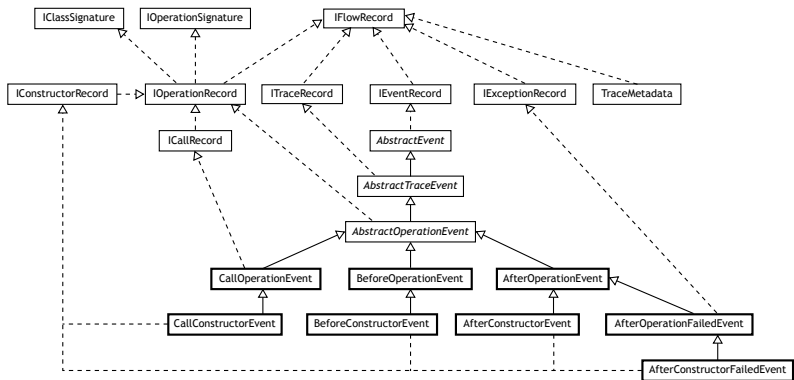
- Extending multiple event types is **cumbersome**
- Instrumentation advice construction is **error prone**

Extending event types

Example - Extending trace records

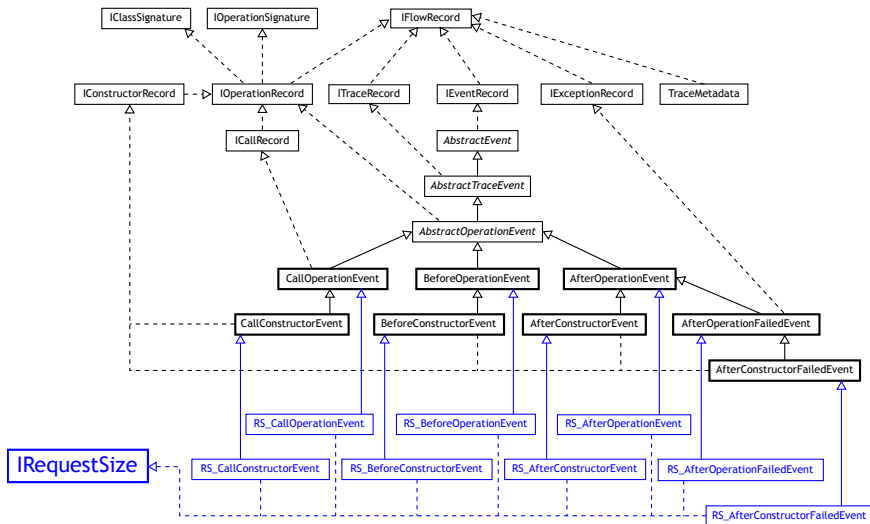


Example - Extending trace records



IRequestSize

Example - Extending trace records

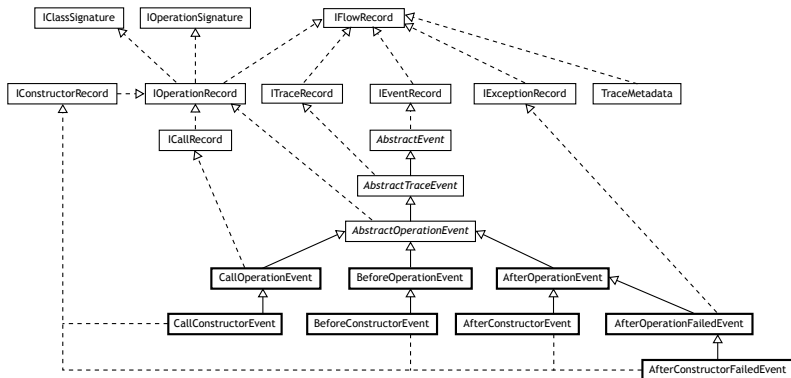


Extending trace records - using the IRL

```
template IRequestSize {  
    long size  
}
```

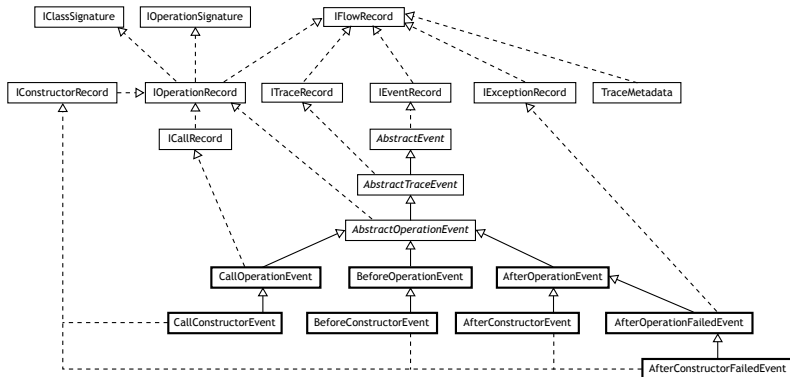
```
entity RS_CallOperationEvent extends CallOperationEvent : IRequestSize  
entity RS_BeforeOperationEvent extends BeforeOperationEvent : IRequestSize  
entity RS_AfterOperationEvent extends AfterOperationEvent : IRequestSize  
entity RS_AfterOperationFailedEvent extends AfterOperationFailedEvent : IRequestSize  
entity RS_CallConstructorEvent extends CallConstructorEvent : IRequestSize  
entity RS_BeforeConstructorEvent extends BeforeConstructorEvent : IRequestSize  
entity RS_AfterConstructorEvent extends AfterConstructorEvent : IRequestSize  
entity RS_AfterConstructorFailedEvent extends AfterConstructorFailedEvent : IRequestSize
```


Extending trace records with model types



based on model types introduced by Steel et al. 2007

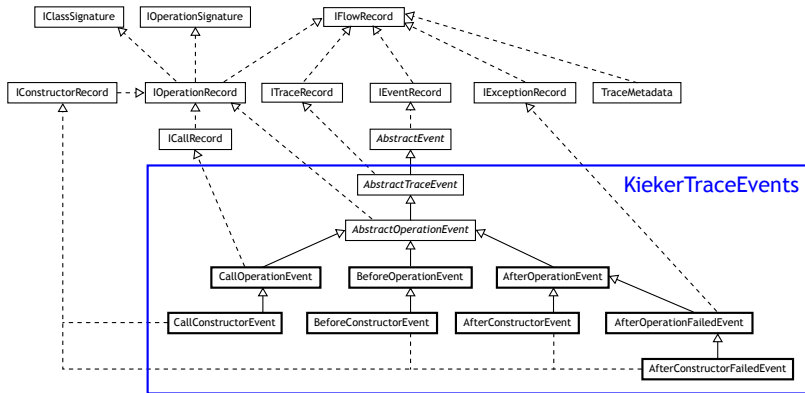
Extending trace records with model types



IRequestSize

based on model types introduced by Steel et al. 2007

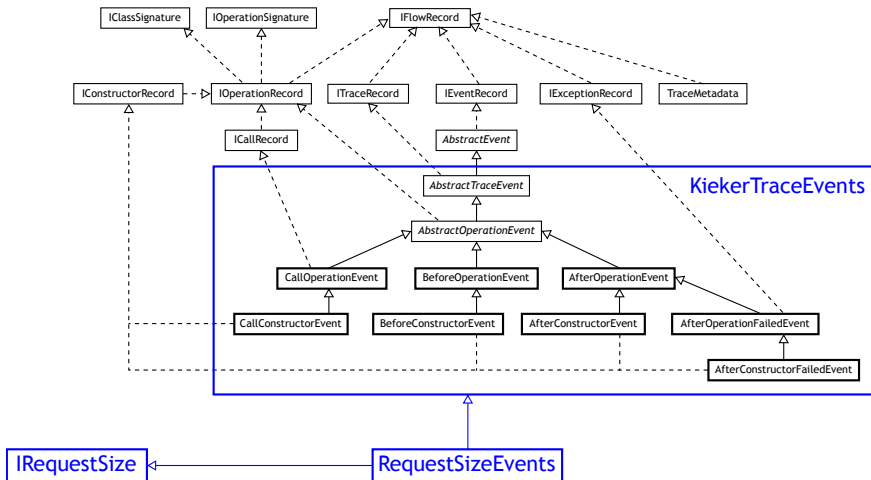
Extending trace records with model types



IRequestSize

based on model types introduced by Steel et al. 2007

Extending trace records with model types



based on model types introduced by Steel et al. 2007

Extending trace records with model types

Template based

model KiekerTraceEvents AbstractTraceEvent

```
template IRequestSize {  
    long size  
}
```

sub RS KiekerTraceEvents IRequestSize

Extending trace records with model types

Template based

model KiekerTraceEvents AbstractTraceEvent

```
template IRequestSize {  
    long size  
}
```

sub RS KiekerTraceEvents IRequestSize

Attribute based

model KiekerTraceEvents AbstractTraceEvent

```
sub RS KiekerTraceEvents {  
    long size  
}
```

Instrumentation advice construction

Old advice declaration

Event type declaration

```
entity BeforeOperationEvent {  
    long timestamp  
    string classSignature  
    string operationSignature  
    long traceId  
    int orderIndex  
}
```

Advice declaration

```
advice LogOperationInvocation() {  
    before BeforeOperationEvent(time, $clsSignature, $opsSignature, traceId, orderIndex)  
    after AfterOperationEvent(time, $opsSignature, $clsSignature, traceId, orderIndex)  
}
```


Old advice declaration

Event type declaration

```
entity BeforeOperationEvent {  
    long timestamp  
    string classSignature  
    string operationSignature  
    long traceId  
    int orderIndex  
}
```

Advice declaration

```
advice LogOperationInvocation() {  
    before BeforeOperationEvent(time, $clsSignature, $opsignature, traceId, orderIndex)  
    after AfterOperationEvent(time, $opsignature, $clsSignature, traceId, orderIndex)  
}
```

Utilizing semantic annotations

Event type declaration

```
entity BeforeOperationEvent {  
    long timestamp      : time  
    string classSignature : class-signature  
    string operationSignature : operation-signature  
    long traceId        : trace-id  
    int  orderIndex     : order-index  
}
```

Advice declaration

```
advice LogOperationInvocation() {  
    before BeforeOperationEvent(time, $clsSignature, $opsignature, traceId, orderIndex)  
    after  AfterOperationEvent(time, $opsignature, $clsSignature, traceId, orderIndex)  
}
```

Utilizing semantic annotations

Event type declaration

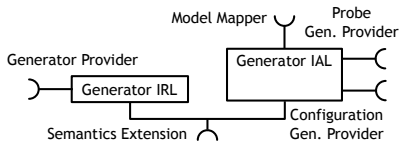
```
entity BeforeOperationEvent {  
    long timestamp      : time  
    string classSignature : class-signature  
    string operationSignature : operation-signature  
    long traceId        : trace-id  
    int  orderIndex     : order-index  
}
```

Advice declaration

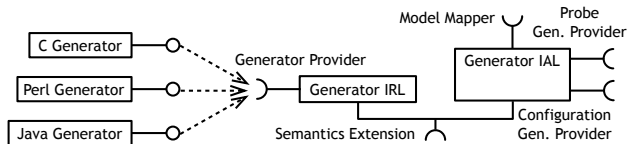
```
advice LogOperationInvocation() {  
    before BeforeOperationEvent  
    after  AfterOperationEvent  
}
```

Realization

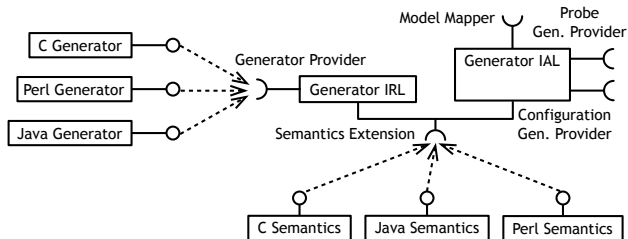
Architecture



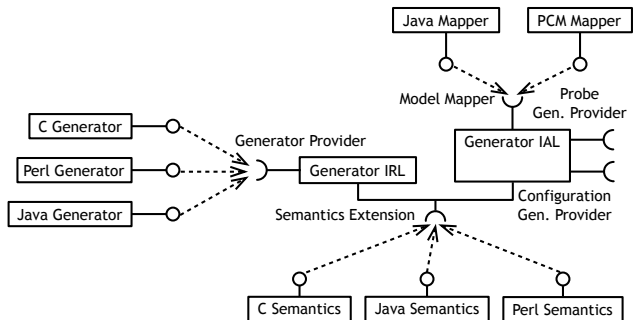
Architecture



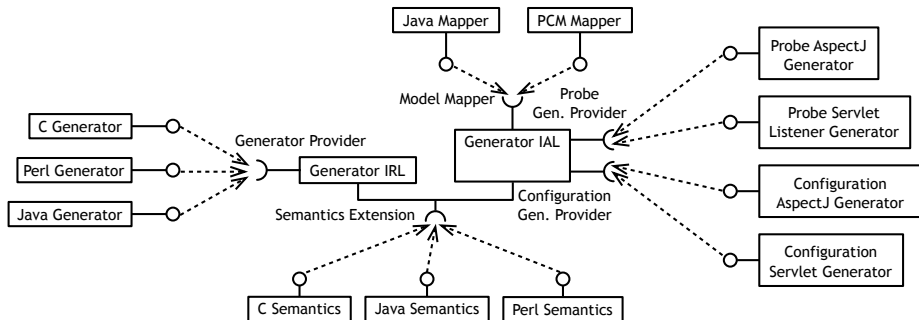
Architecture



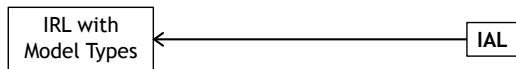
Architecture



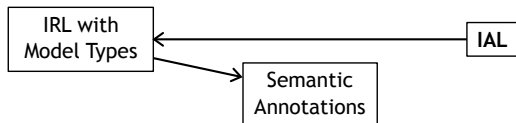
Architecture



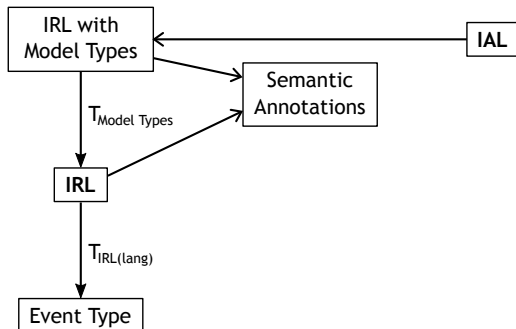
Model transformation



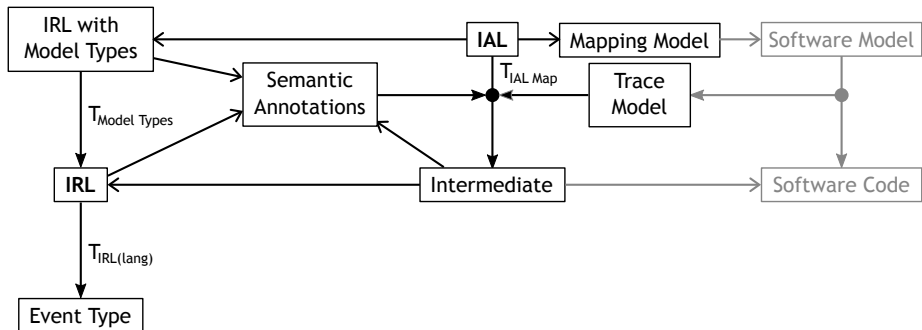
Model transformation



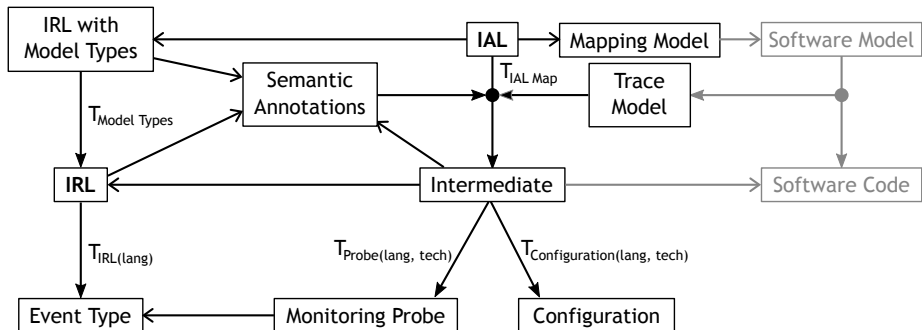
Model transformation



Model transformation



Model transformation



Conclusion

Conclusion

Summary

- Entity type extensions mechanism based on model types
- Semantic properties for save probe construction

Future Work

- Complete PCM and Java IAL support
- Adaptation of IAL generators to new API

Github

<https://github.com/kieker-monitoring/instrumentation-languages>

Eclipse update site - snapshot

<https://build.se.informatik.uni-kiel.de/eus/mdm/snapshot/>

References

Bibliography I

- Jung, Reiner et al. (2013). “Model-driven Instrumentation with Kieker and Palladio to forecast Dynamic Applications.” In: *Proceedings Symposium on Software Performance: Joint Kieker/Palladio Days 2013*. Vol. 1083. CEUR, pp. 99-108.
- Steel, Jim et al. (2007). “On model typing.” In: *Software & Systems Modeling 6.4*, pp. 401-413. ISSN: 1619-1366.