Everything in Sight: Kieker's WebGUI in Action

— Joint Kieker / Palladio Days 2013 —

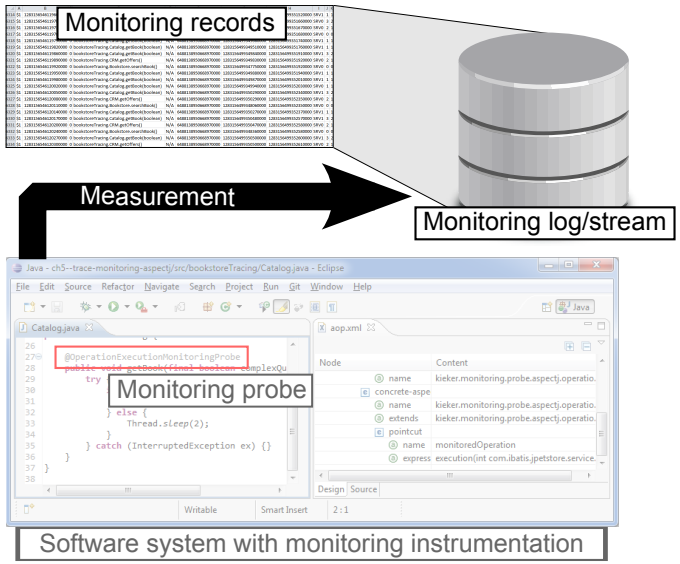
Nils Christian Ehmke

Software Engineering Group
Kiel University, Germany

November 27, 2013 @ Karlsruhe

Motivation

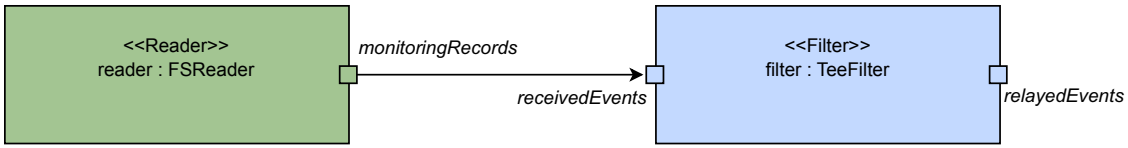
- Kieker can monitor and analyze Java, Cobol, Perl, ... based applications



Motivation

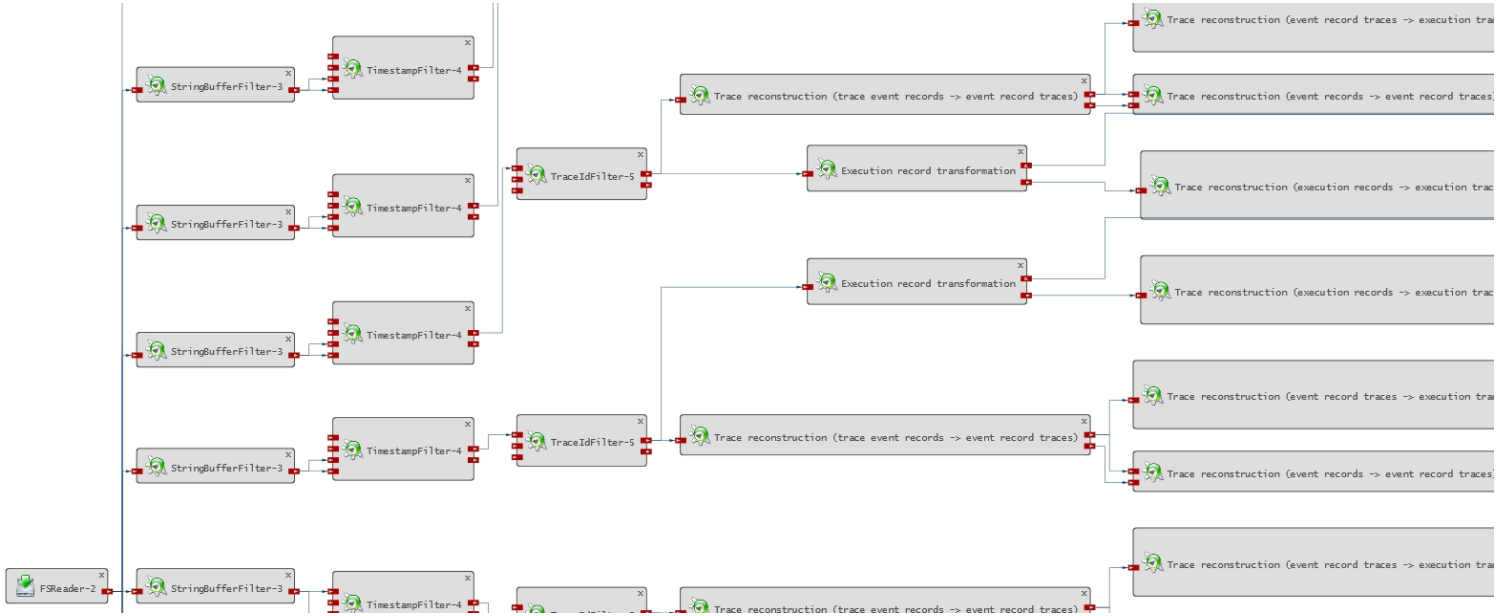
- Kieker can monitor and analyze Java, Cobol, Perl, ... based applications
- An API can be used to create analysis networks

```
public static void main(String[] args) throws IllegalStateException, AnalysisConfigurationException {
    final IAnalysisController ac = new AnalysisController();

    final Configuration readerConfig = new Configuration();
    readerConfig.setProperty(FSReader.CONFIG_PROPERTY_NAME_INPUTDIRS, "~/monitoring-data");

    final FSReader reader = new FSReader(readerConfig, ac);

    final TeeFilter filter = new TeeFilter(new Configuration(), ac);

    ac.connect(reader, FSReader.OUTPUT_PORT_NAME_RECORDS, filter, TeeFilter.INPUT_PORT_NAME_EVENTS);

    ac.run();
}
```



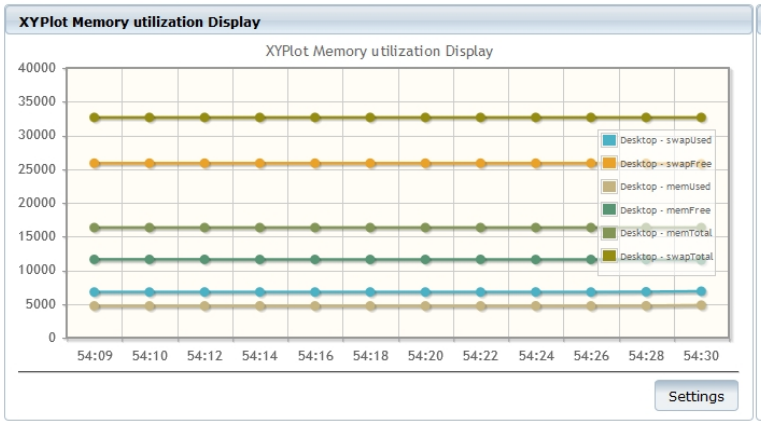
Motivation

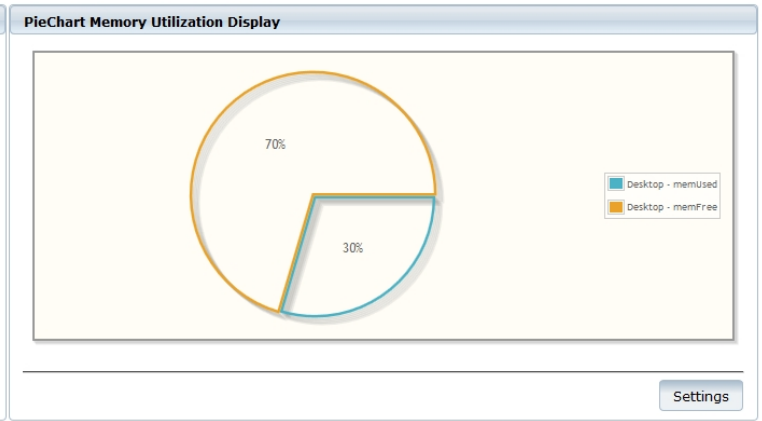
- Kieker can monitor and analyze Java, Cobol, Perl, ... based applications
- An API can be used to create analysis networks
- Larger analysis networks?



Motivation

- Kieker can monitor and analyze Java, Cobol, Perl, ... based applications
- An API can be used to create analysis networks
- Larger analysis networks?
- Interactive visualizations?





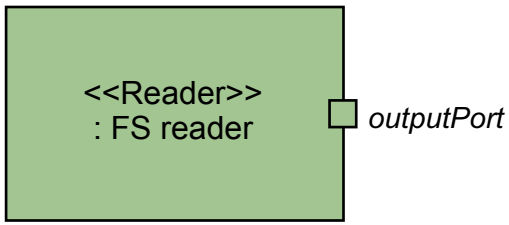
Motivation

- Kieker can monitor and analyze Java, Cobol, Perl, ... based applications
- An API can be used to create analysis networks
- Larger analysis networks?
- Interactive visualizations?

$\rightsquigarrow$ Kieker's WebGUI

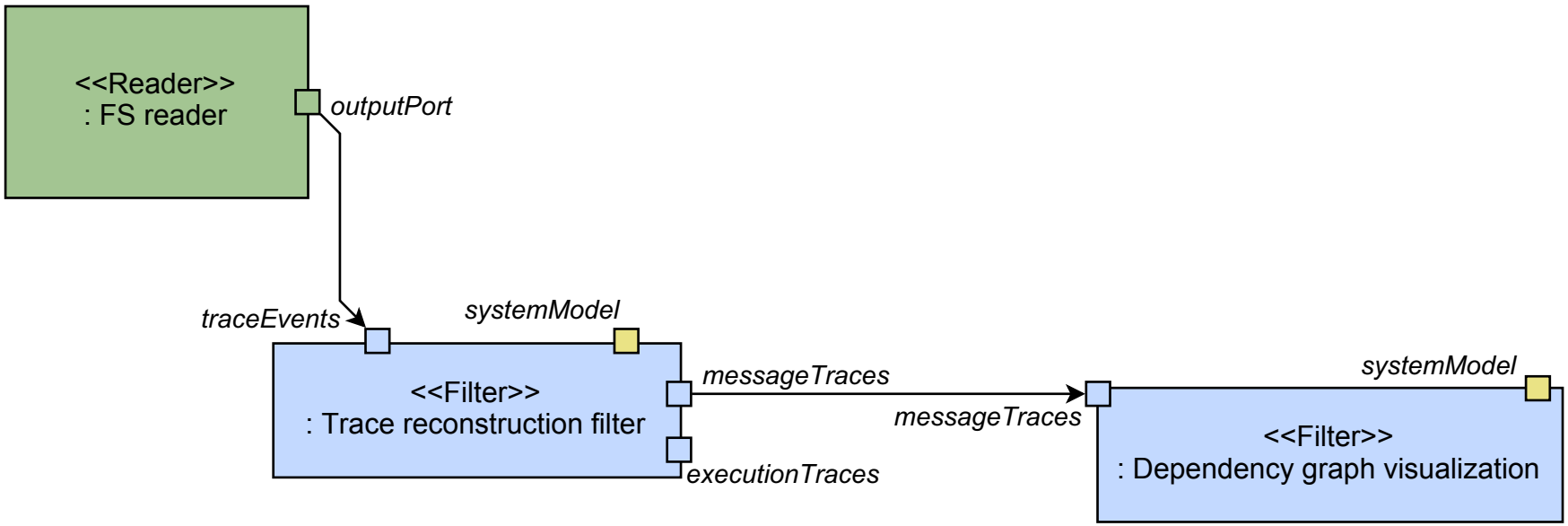
Analysis Networks

- Three types of components:
  - Readers



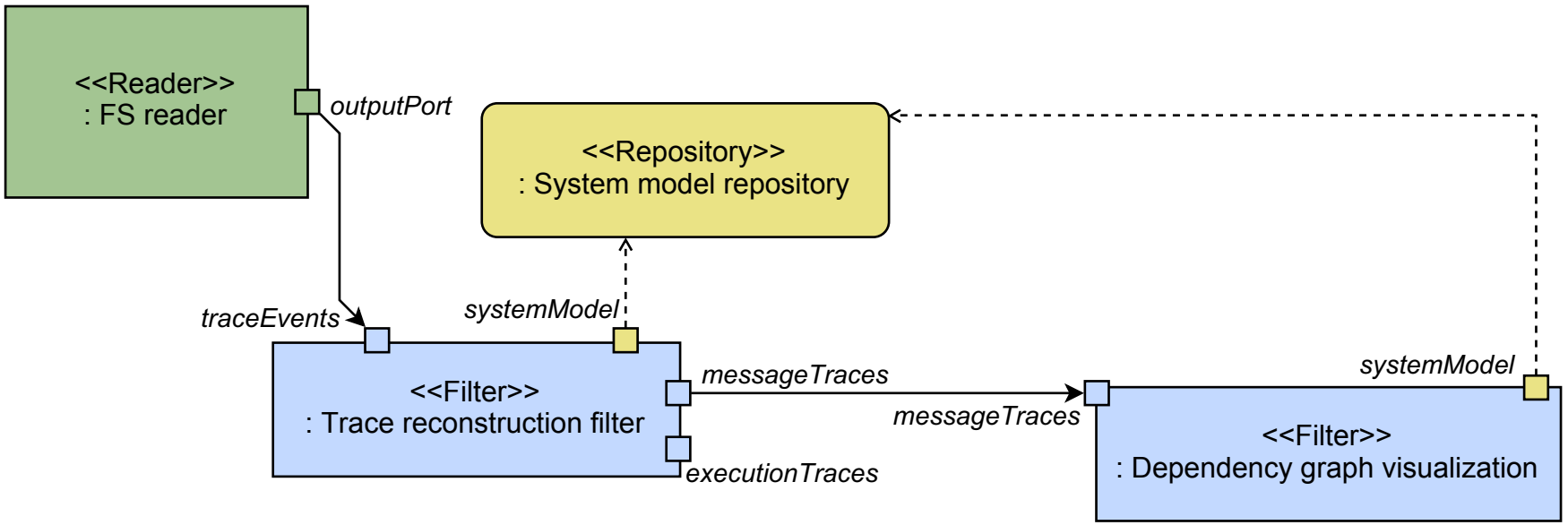
Analysis Networks

- Three types of components:
  - Readers
  - Filters



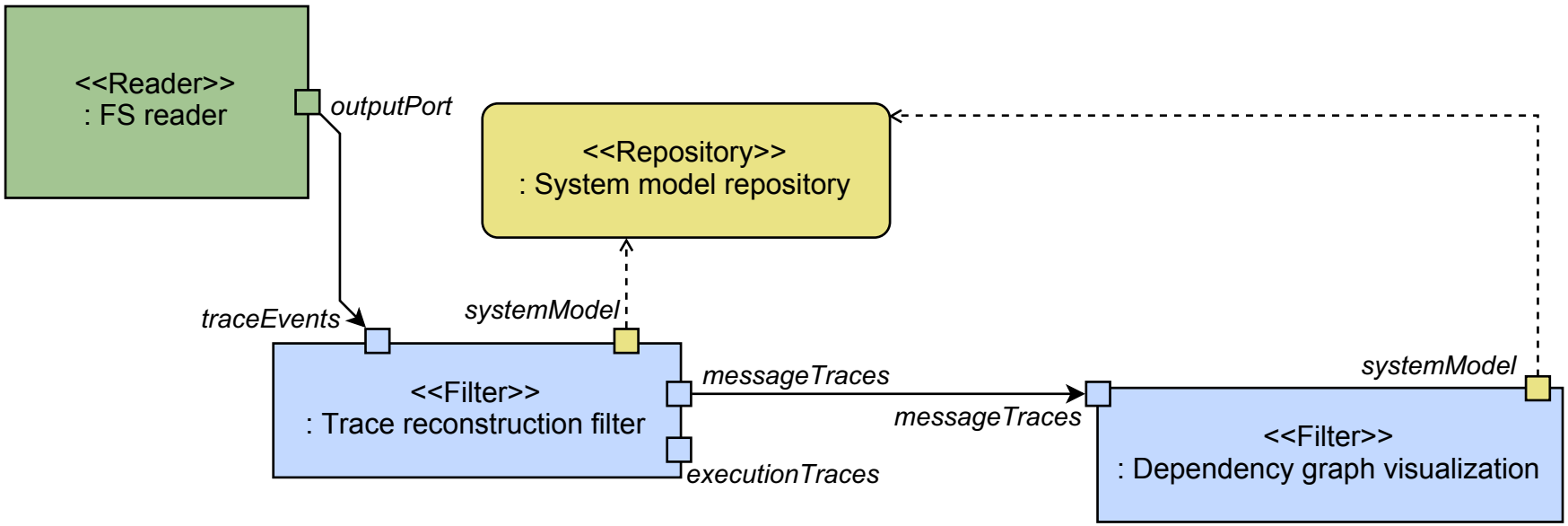
Analysis Networks

- Three types of components:
  - Readers
  - Filters
  - Repositories



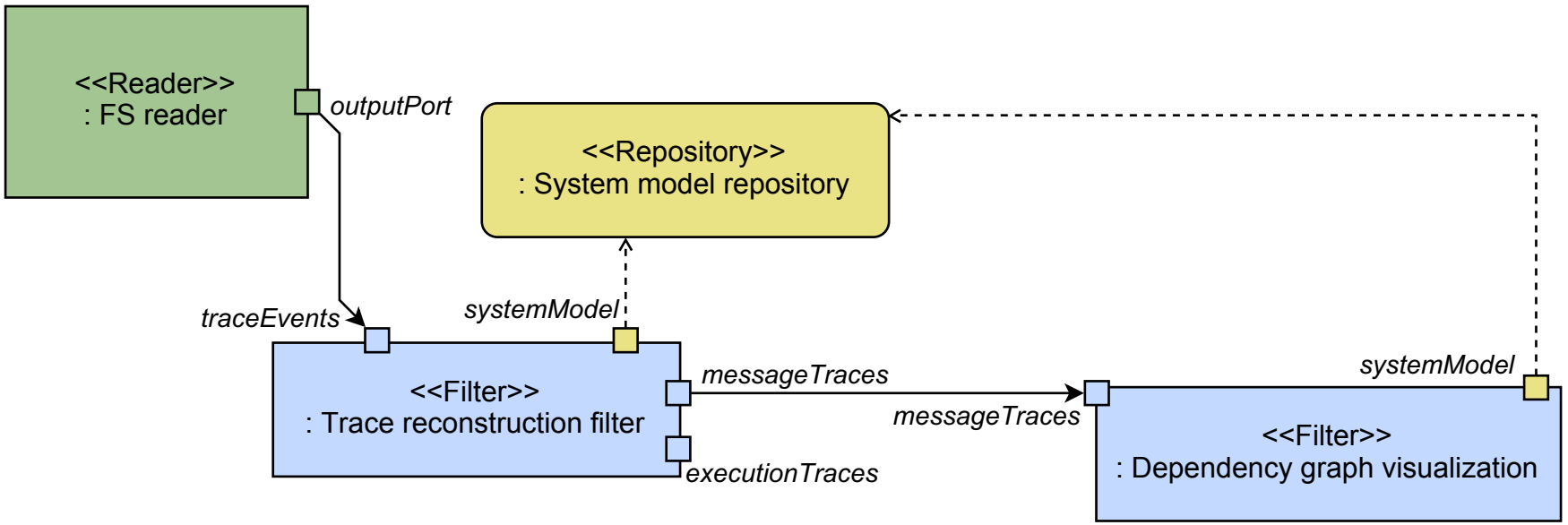
Analysis Networks

- Three types of components:
  - Readers
  - Filters
  - Repositories
- Connection via named ports



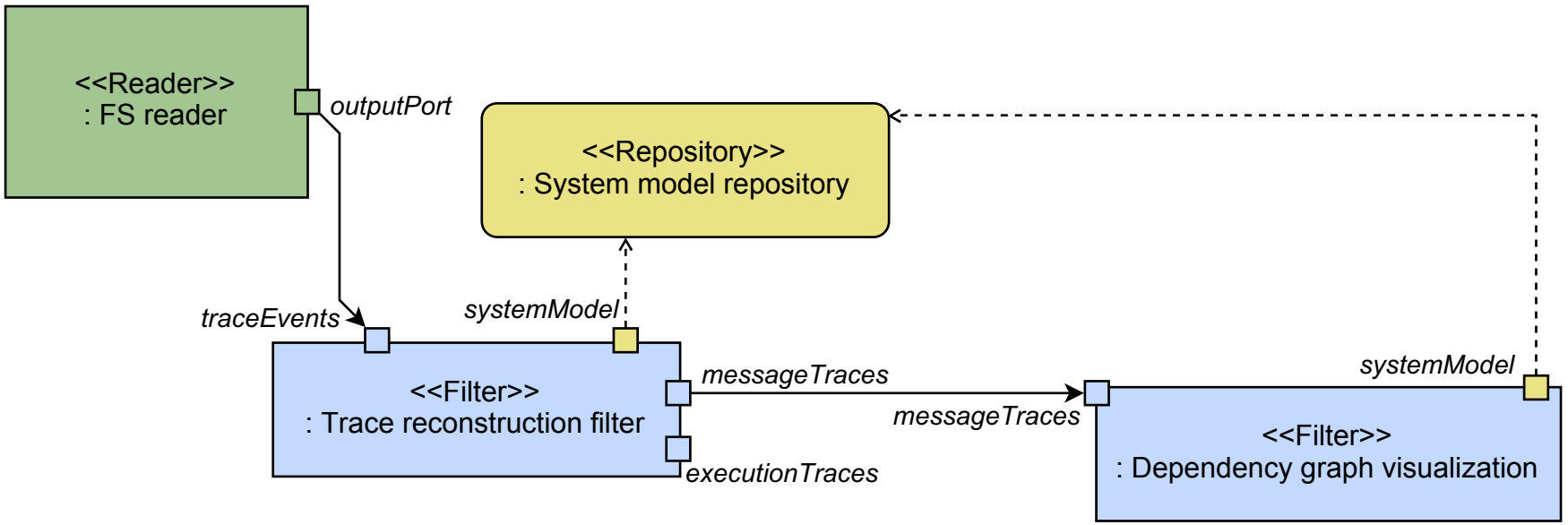
Analysis Networks

- Three types of components:
  - Readers
  - Filters
  - Repositories
- Connection via named ports
- Configuration via named properties



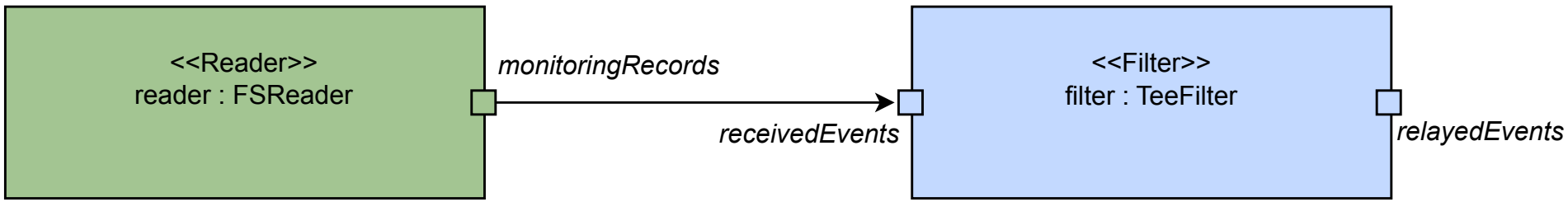
Analysis Networks

- Three types of components:
  - Readers
  - Filters
  - Repositories
- Connection via named ports
- Configuration via named properties
- API can be used to create, save, and load networks



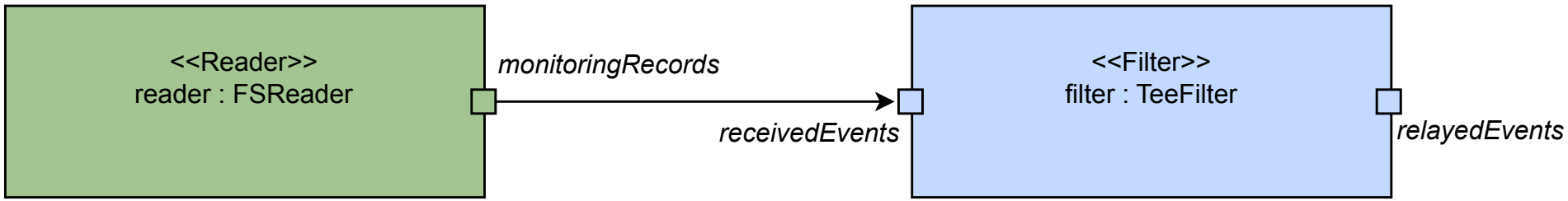
An Exemplaric Network


```
// Prepare the controller for the analysis network
final IAnalysisController controller = new AnalysisController();

// Create and configure the file system reader
final Configuration rConfig = new Configuration();
rConfig.setProperty(FSReader.CONFIG_PROPERTY_NAME_INPUTDIRS,
"home/nie/monitoring-logs/log-2013-10-03-12-00-00");
final FSReader reader = new FSReader(rConfig, controller);

// Create and configure the tee filter for the printing
final Configuration fConfig = new Configuration();
final TeeFilter filter = new TeeFilter(fConfig, controller);

// Connect the reader to the filter
controller.connect(reader, FSReader.OUTPUT_PORT_NAME_RECORDS,
filter, TeeFilter.INPUT_PORT_NAME_EVENTS);

// Execute the analysis
controller.run();
```

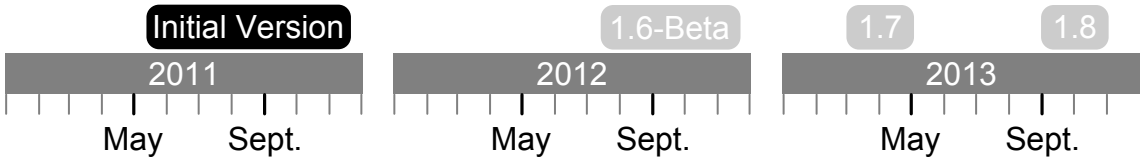


An Exemplaric Network (cont'd)

```
<plugins xsi:type="Reader" name="reader"
          classname="kieker.analysis.plugin.reader.filesystem.FSReader">
        <properties name="inputDirs"
          value="home/nie/monitoring-logs/log-2013-10-03-12-00-00"/>
        <outputPorts name="monitoringRecords" id="2" subscribers="3"/>
  </plugins>
  <plugins xsi:type="Filter" name="filter"
          classname="kieker.analysis.plugin.filter.forward.TeeFilter">
        <outputPorts name="relayedEvents" id="5"/>
        <inputPorts name="receivedEvents" id="3"/>
  </plugins>
```

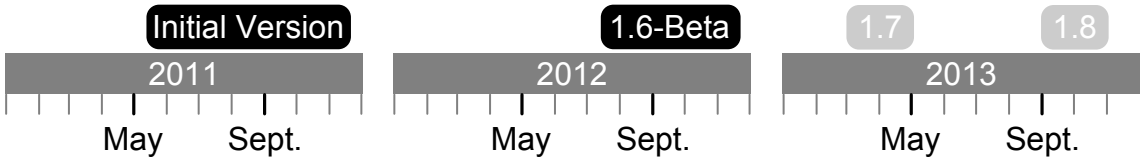


Kieker's WebGUI



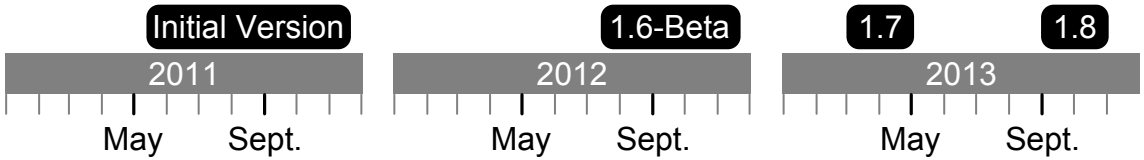
- Developed since December 2011

Kieker's WebGUI



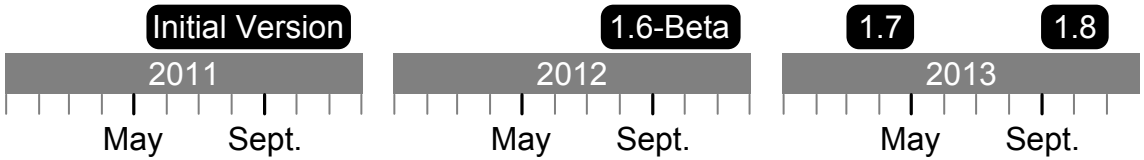
- Developed since December 2011
- First beta release October 2012

Kieker's WebGUI



- Developed since December 2011
- First beta release October 2012
- Synchronized with Kieker release cycle

Kieker's WebGUI



- Developed since December 2011
- First beta release October 2012
- Synchronized with Kieker release cycle

Let's take a look!

Kieker's WebGUI

- A multi-user web application for Kieker analyses
- Cockpits visualize live results from running analyses
- Included in the Kieker releases
- Open-source (Apache License, V. 2.0)

- Future development:
  - Usability
  - Performance
  - Stability
  - More displays

- `http://kieker-monitoring.net`