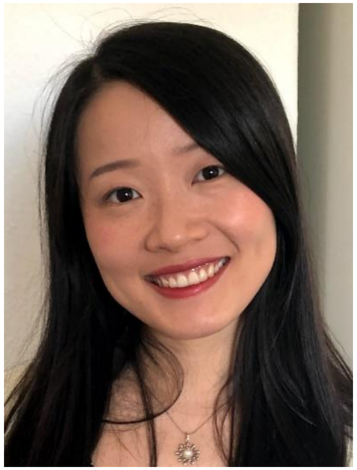
MagpieBridge: A General Approach to Integrating Static Analyses into IDEs and Editors

ECOOP 2019, London

Linghui Luo, Julian Dolby

@LinghuiLuo

@julian_dolby



Linghui Luo

PADERBORN UNIVERSITY

Julian Dolby

IBM Research

Eric Bodden

PADERBORN UNIVERSITY

Fraunhofer IEM

Program Analysis Tools in Academia

SPLLIFT
SpotBugs
Amandroid
CiD
Cheetah
SPLlift
PMD
Lotrack
Andromeda
JOANA
HybriDroid
CheckStyle
ThreadSafe
ParallelChecker
CogniCrypt
TAJ
FlowDroid
DroidSafe
Lint
FindBugs
Cppcheck
Legato

How to achieve **broad** and **lasting** adoption of these tools?

Where Should Analysis Results Be Shown?

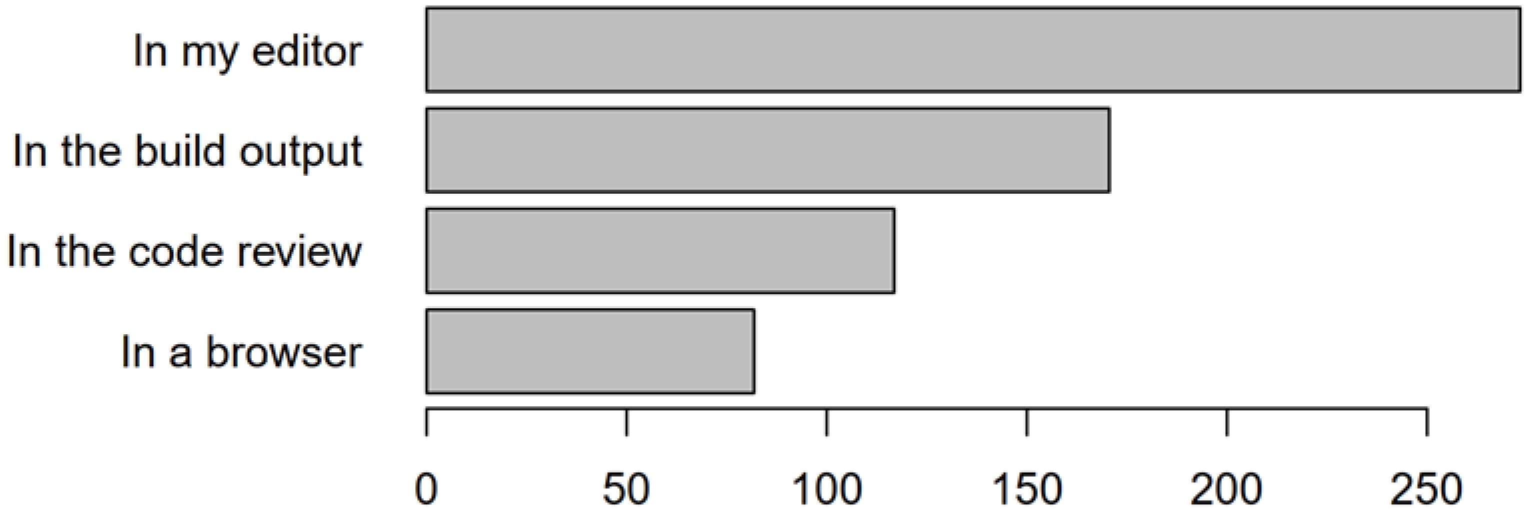
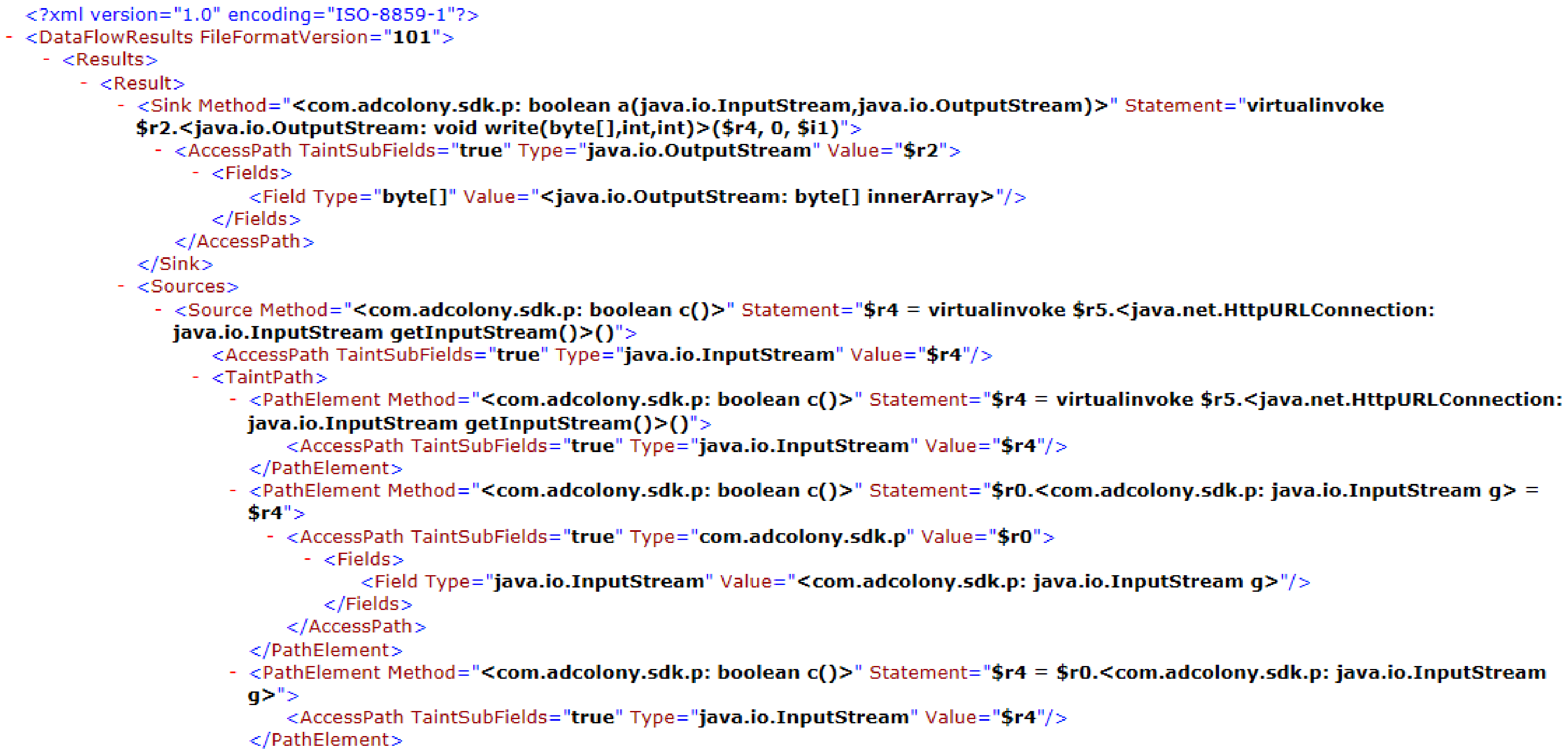


Figure: Where developers would like to have the output of program analyzers [1].

[1] M. Christakis and C. Bird. What developers want and need from program analysis: An empirical study. ASE'*16*, Singapore, 2016, 332-343.

Analysis Result is Often Hard to Understand

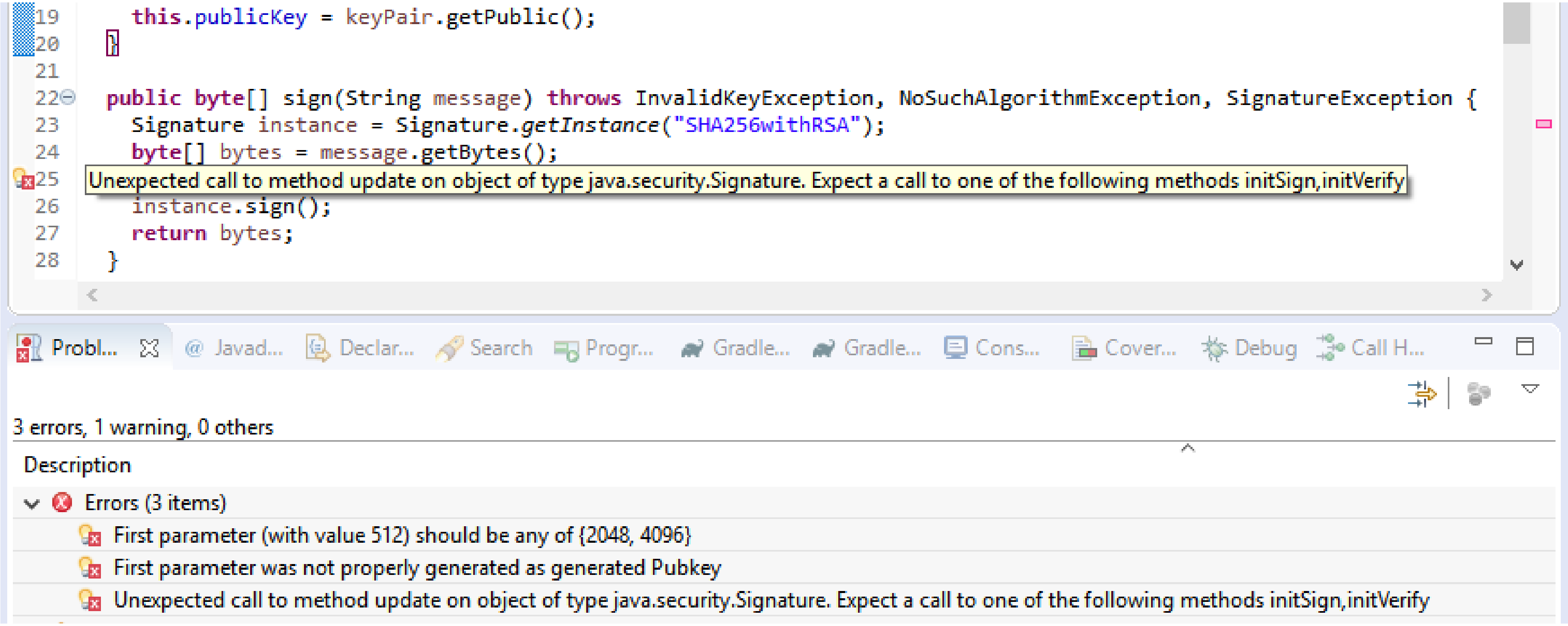


```
  <?xml version="1.0" encoding="ISO-8859-1"?>
- <DataFlowResults FileFormatVersion="101">
  - <Results>
    - <Result>
      - <Sink Method="<com.adcolony.sdk.p: boolean a(java.io.InputStream,java.io.OutputStream)>" Statement="virtualinvoke
        $r2.<java.io.OutputStream: void write(byte[],int,int)>($r4, 0, $i1)">
        - <AccessPath TaintSubFields="true" Type="java.io.OutputStream" Value="$r2">
          - <Fields>
              <Field Type="byte[]" Value="<java.io.OutputStream: byte[] innerArray>"/>
            </Fields>
          </AccessPath>
        </Sink>
      - <Sources>
        - <Source Method="<com.adcolony.sdk.p: boolean c()>" Statement="$r4 = virtualinvoke $r5.<java.net.HttpURLConnection:
          java.io.InputStream getInputStream()>()">
            <AccessPath TaintSubFields="true" Type="java.io.InputStream" Value="$r4"/>
          - <TaintPath>
            - <PathElement Method="<com.adcolony.sdk.p: boolean c()>" Statement="$r4 = virtualinvoke $r5.<java.net.HttpURLConnection:
              java.io.InputStream getInputStream()>()">
                <AccessPath TaintSubFields="true" Type="java.io.InputStream" Value="$r4"/>
              </PathElement>
            - <PathElement Method="<com.adcolony.sdk.p: boolean c()>" Statement="$r0.<com.adcolony.sdk.p: java.io.InputStream g> =
              $r4">
              - <AccessPath TaintSubFields="true" Type="com.adcolony.sdk.p" Value="$r0">
                - <Fields>
                    <Field Type="java.io.InputStream" Value="<com.adcolony.sdk.p: java.io.InputStream g>"/>
                  </Fields>
                </AccessPath>
              </PathElement>
            - <PathElement Method="<com.adcolony.sdk.p: boolean c()>" Statement="$r4 = $r0.<com.adcolony.sdk.p: java.io.InputStream
              g>">
                <AccessPath TaintSubFields="true" Type="java.io.InputStream" Value="$r4"/>
              </PathElement>
```

Figure: XML Output of FlowDroid [2]

[2] S. Arzt, S. Rasthofer, C. Fritz, E. Bodden, A. Bartel, J. Klein, Y. L. Traon, D. Octeau, and P. McDaniel.
FlowDroid: precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for Android apps. PLDI '14, New York, NY, USA, 259-269.

Better Approach - Plugins



```
    this.publicKey = keyPair.getPublic();
  }

  public byte[] sign(String message) throws InvalidKeyException, NoSuchAlgorithmException, SignatureException {
    Signature instance = Signature.getInstance("SHA256withRSA");
    byte[] bytes = message.getBytes();

    instance.sign();
    return bytes;
  }
```

Unexpected call to method update on object of type java.security.Signature. Expect a call to one of the following methods initSign,initVerify

3 errors, 1 warning, 0 others

Description

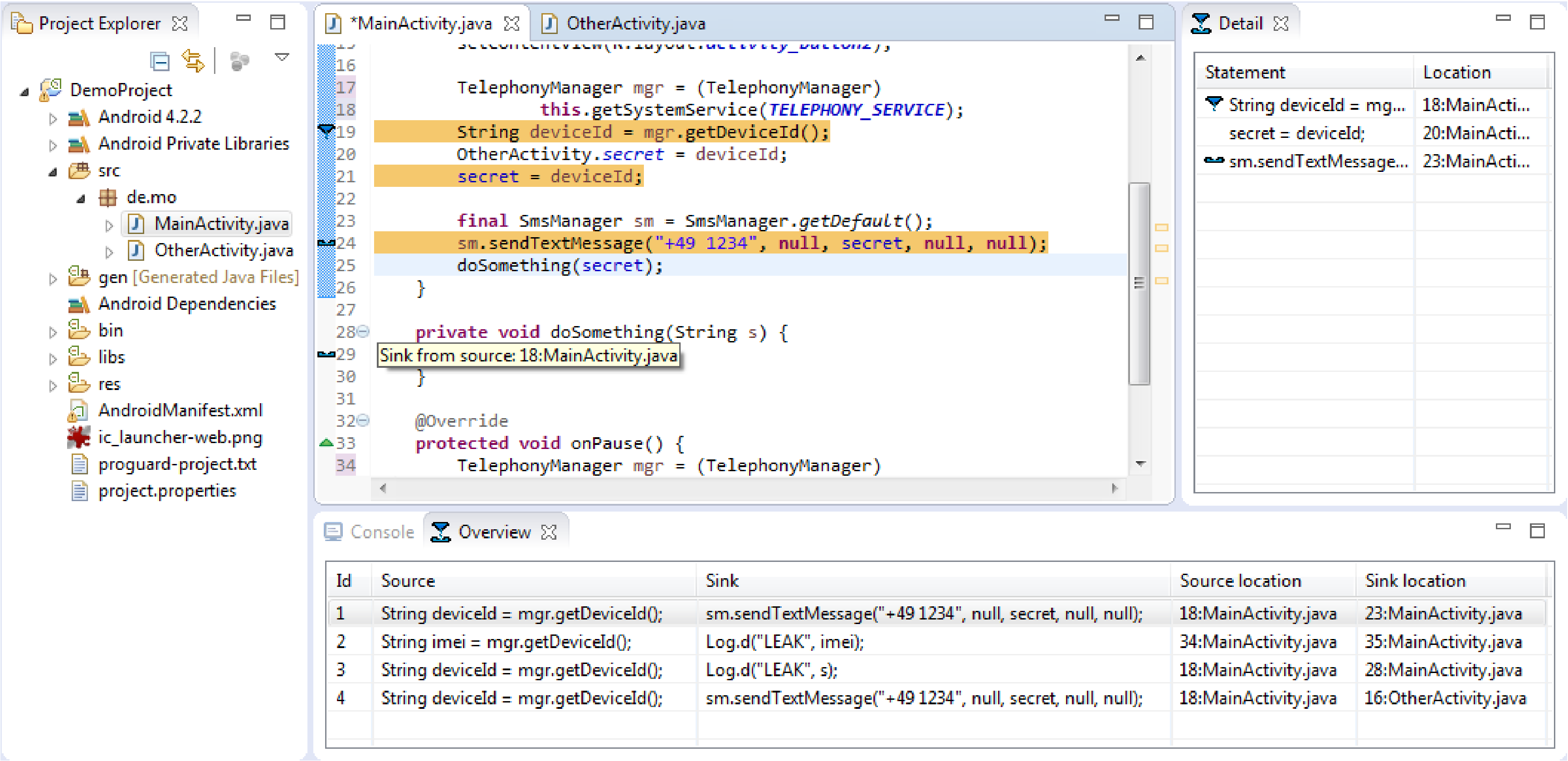
Errors (3 items)

- First parameter (with value 512) should be any of {2048, 4096}
- First parameter was not properly generated as generated Pubkey
- Unexpected call to method update on object of type java.security.Signature. Expect a call to one of the following methods initSign,initVerify

Figure: The CogniCrypt Eclipse Plugin [3]

[3] S. Krüger, S. Nadi, M. Reif, K. Ali, M. Mezini, E. Bodden, F. Göpfert, F. Günther, C. Weinert, D. Demmler, and R. Kamath. CogniCrypt : Supporting Developers in using Cryptography. ASE‘17, NJ, USA, 931-936.

Better Approach - Plugins



```
        setContentView(R.layout.activity_button2);

        TelephonyManager mgr = (TelephonyManager)
                this.getSystemService(TELEPHONY_SERVICE);
        String deviceId = mgr.getDeviceId();
        OtherActivity.secret = deviceId;
        secret = deviceId;

        final SmsManager sm = SmsManager.getDefault();
        sm.sendTextMessage("+49 1234", null, secret, null, null);
        doSomething(secret);
    }

    private void doSomething(String s) {
Sink from source: 18:MainActivity.java
    }

    @Override
    protected void onPause() {
        TelephonyManager mgr = (TelephonyManager)
```

Statement	Location
String deviceId = mg...	18:MainActi...
secret = deviceId;	20:MainActi...
sm.sendTextMessage...	23:MainActi...

Id	Source	Sink	Source location	Sink location
1	String deviceId = mgr.getDeviceId();	sm.sendTextMessage("+49 1234", null, secret, null, null);	18:MainActivity.java	23:MainActivity.java
2	String imei = mgr.getDeviceId();	Log.d("LEAK", imei);	34:MainActivity.java	35:MainActivity.java
3	String deviceId = mgr.getDeviceId();	Log.d("LEAK", s);	18:MainActivity.java	28:MainActivity.java
4	String deviceId = mgr.getDeviceId();	sm.sendTextMessage("+49 1234", null, secret, null, null);	18:MainActivity.java	16:OtherActivity.java

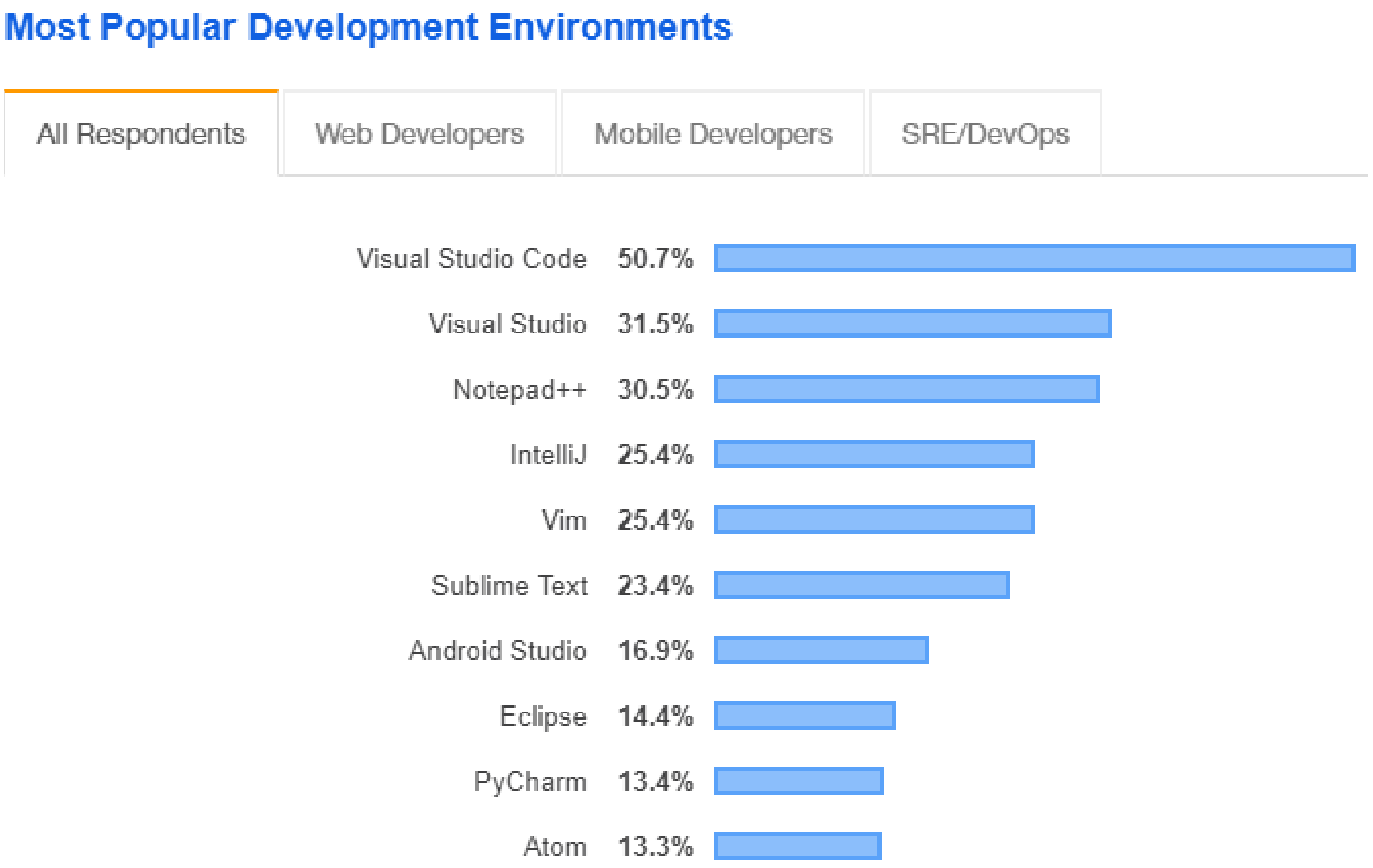
Figure: The Cheetah Eclipse Plugin [4]

[4] L. Nguyen Quang Do, K. Ali, B. Livshits, E. Bodden, J. Smith, and E. Murphy-Hill. Cheetah: just-in-time taint analysis for Android apps. ICSE-C '17. NJ, USA, 39-42.

Tool Integration

	Eclipse	IntelliJ IDEA	Visual Studio	NetBeans	Android Studio	Visual Studio Code
PMD	✓		✓	✓		
FindBugs	✓	✓		✓		
Cheetah	✓					
CogniCrypt	✓					
SonarLint	✓	✓	✓			✓
FixDroid					✓	
SpotBugs	✓					

One Is Not Enough



Source: Stack Overflow Developer Survey 2019 https://insights.stackoverflow.com/survey/2019#technology

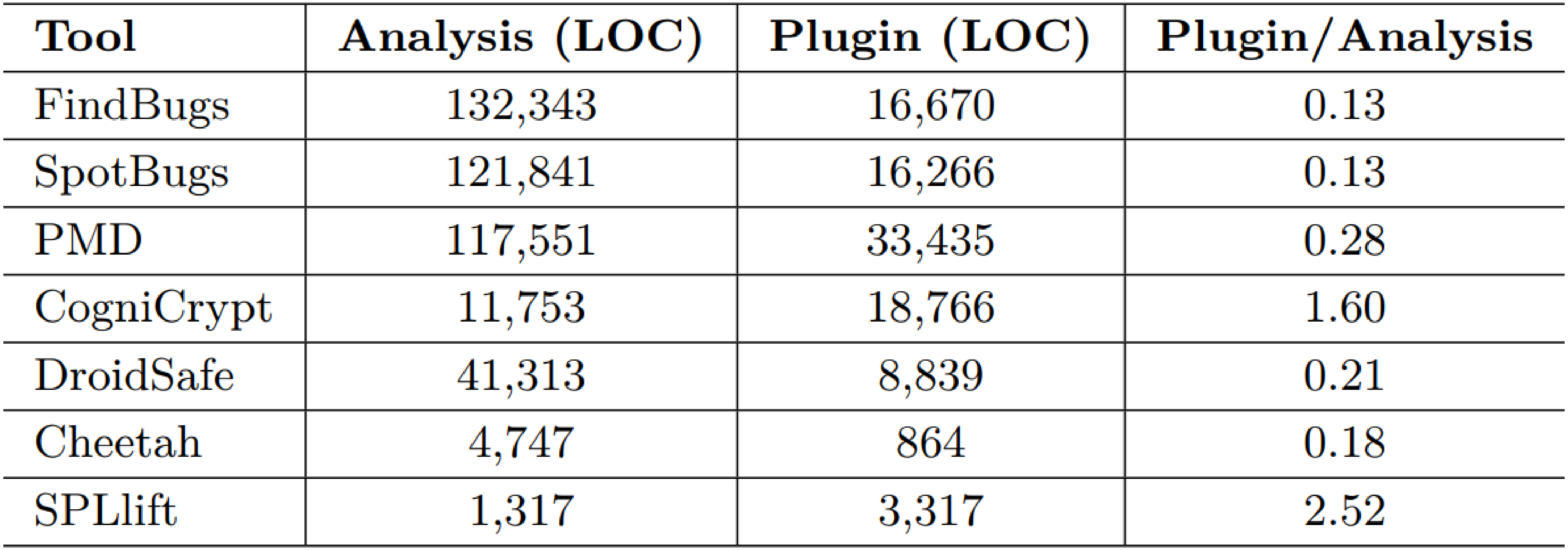
MXN Complexity Problem

N

M

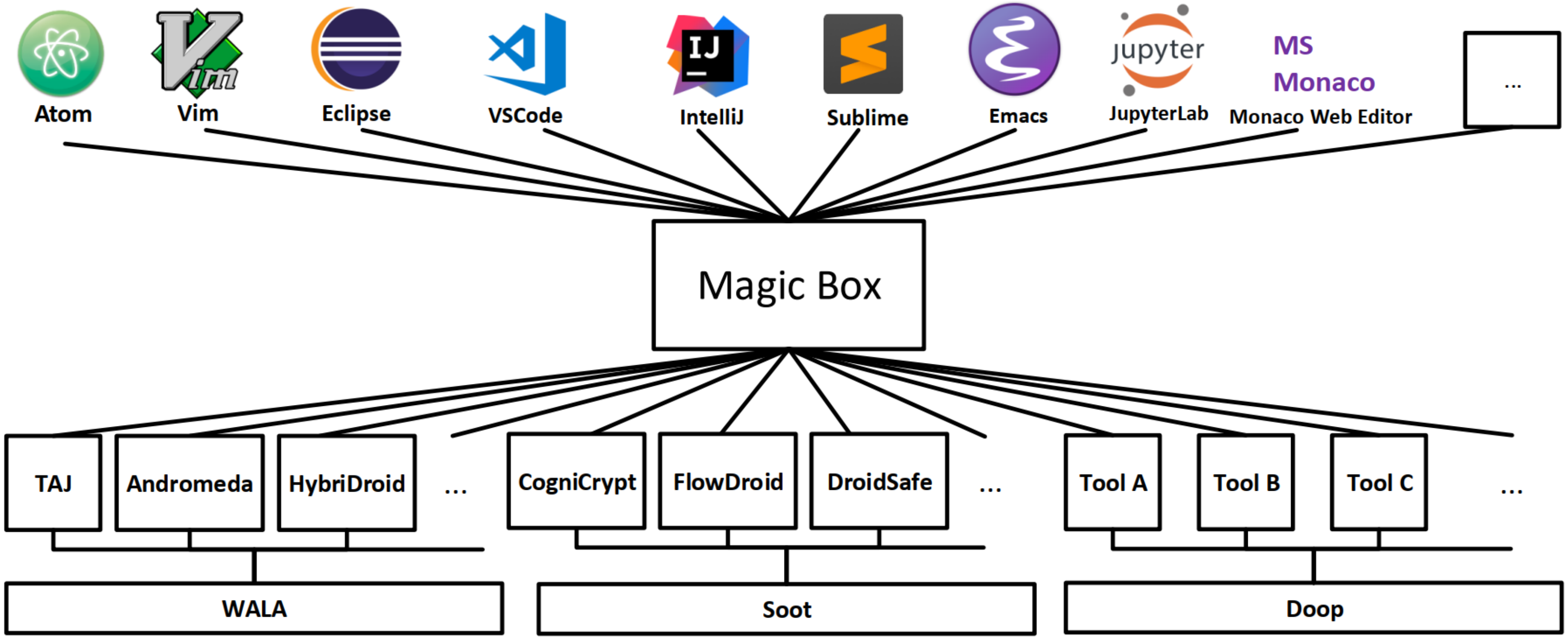
	IDE 1	IDE 2	IDE 3	IDE 4	IDE 5	…
Analysis 1						
Analysis 2						
Analysis 3						
Analysis 4						
Analysis 5						
…						

Relative Costs of the Eclipse Plugins



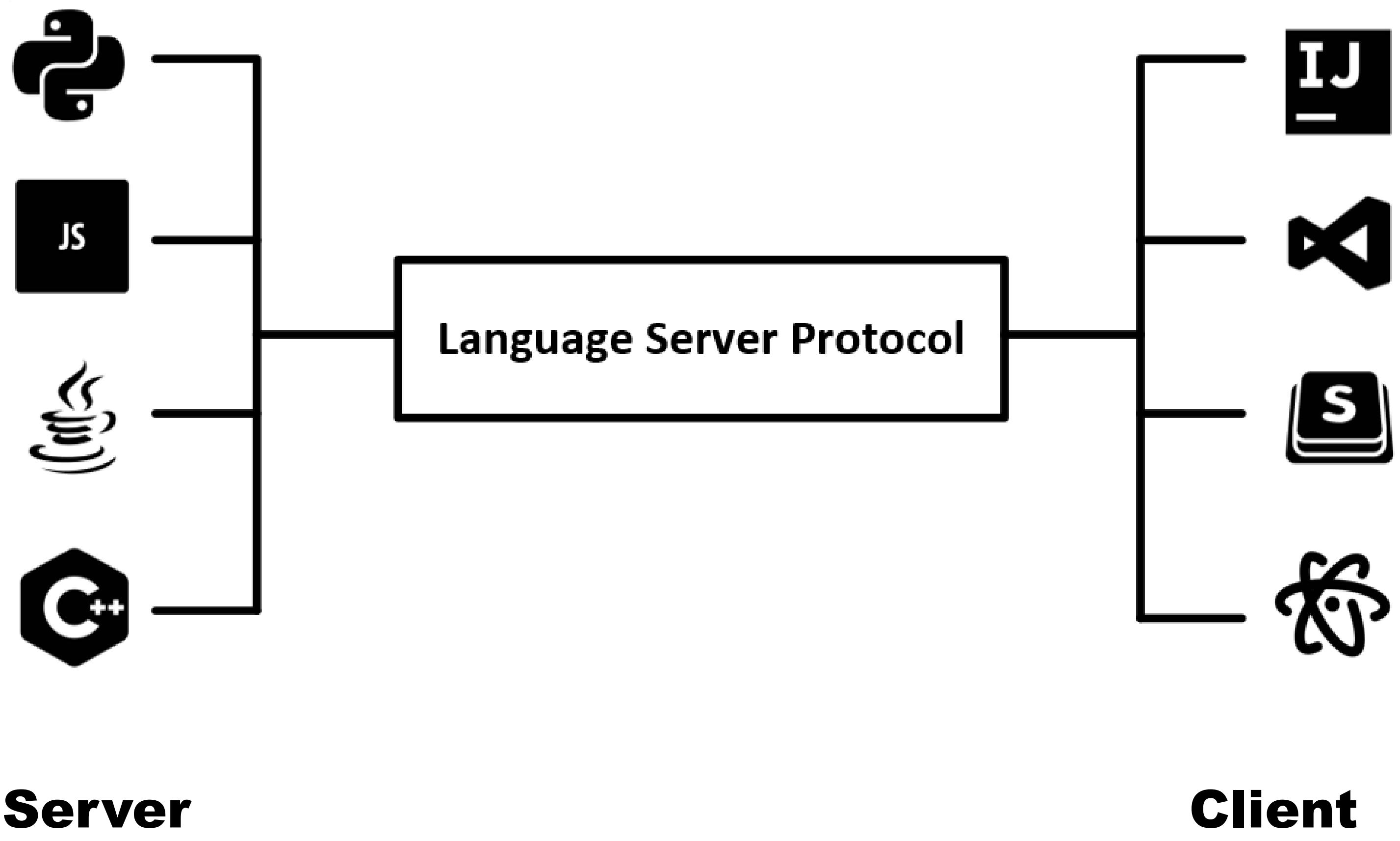
Tool	Analysis (LOC)	Plugin (LOC)	Plugin/Analysis
FindBugs	132,343	16,670	0.13
SpotBugs	121,841	16,266	0.13
PMD	117,551	33,435	0.28
CogniCrypt	11,753	18,766	1.60
DroidSafe	41,313	8,839	0.21
Cheetah	4,747	864	0.18
SPLlift	1,317	3,317	2.52

Desired Solution



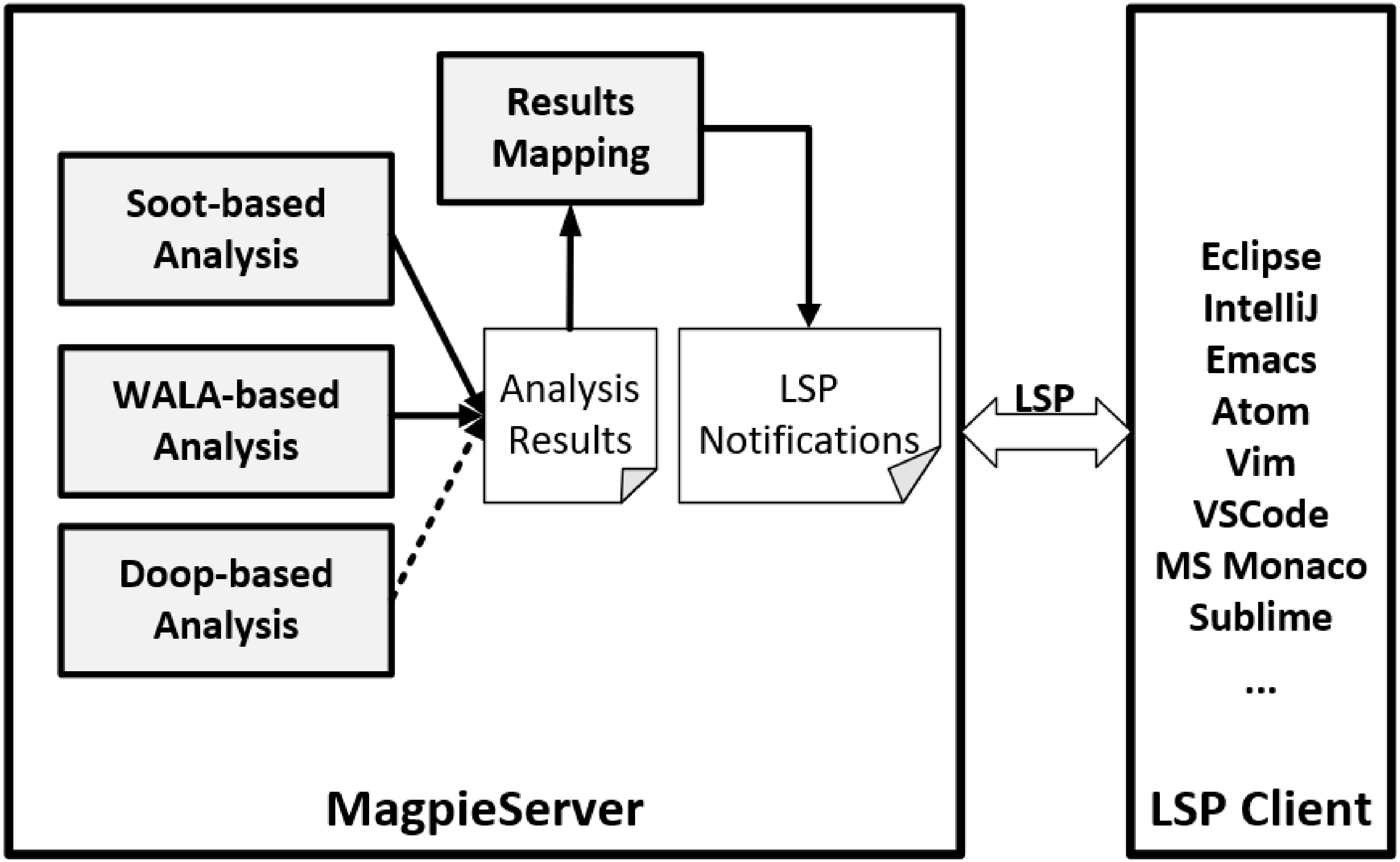
- Provides a common communication protocol between analyses and editors
- Handles required work for good editor support
- Reduces MXN complexity to M+N complexity

Language Server Protocol (LSP)



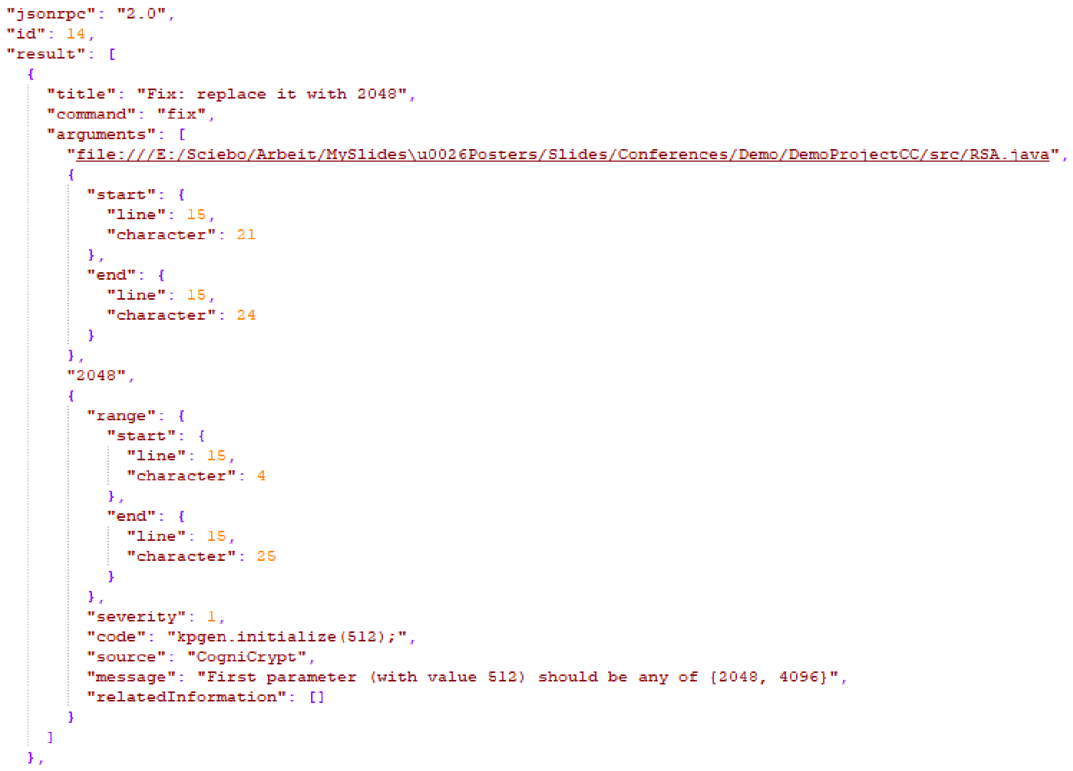
- Syntax Highlighting
- Autocomplete
- Go to Definition
- Find References
- Diagnostics
- Quick Fixes
- Code Actions
- Tooltips
- …

Leverage LSP



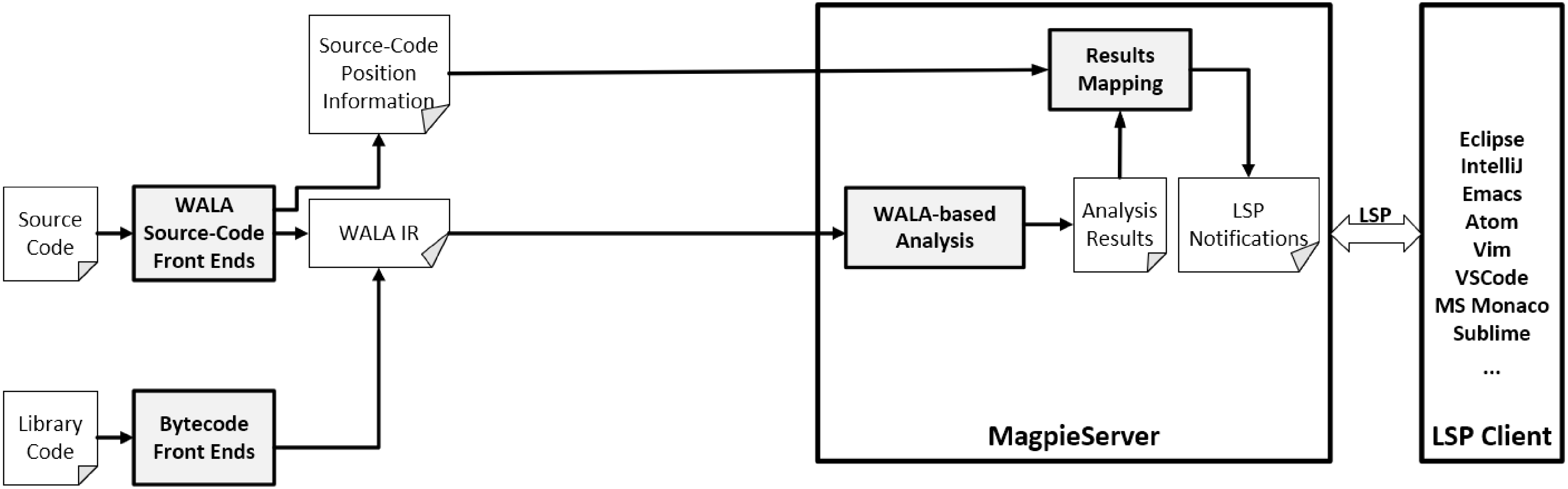
Challenges

- Precise source code info is vital for LSP
  - Code range
  - Line and character number
  - Source code
- Analysis on intermediate representation (IR)
- IRs need precise source code information
  - WALA
  - Soot
  - Doop

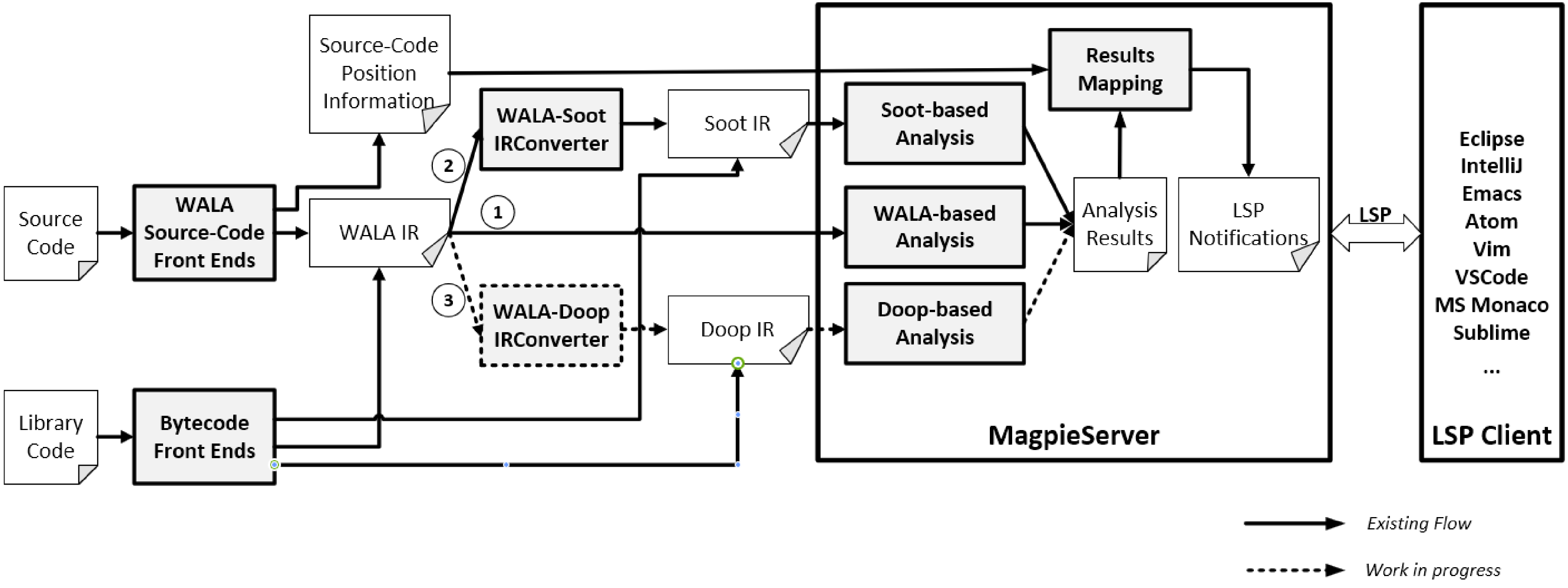


```
"jsonrpc": "2.0",
"id": 14,
"result": [
  {
    "title": "Fix: replace it with 2048",
    "command": "fix",
    "arguments": [
      "file:///E:/Sciebo/Arbeit/MySlides\u0026Posters/Slides/Conferences/Demo/DemoProjectCC/src/RSA.java",
      {
        "start": {
          "line": 15,
          "character": 21
        },
        "end": {
          "line": 15,
          "character": 24
        }
      },
      "2048",
      {
        "range": {
          "start": {
            "line": 15,
            "character": 4
          },
          "end": {
            "line": 15,
            "character": 25
          }
        },
        "severity": 1,
        "code": "kpgen.initialize(512);",
        "source": "CogniCrypt",
        "message": "First parameter (with value 512) should be any of {2048, 4096}",
        "relatedInformation": []
      }
    ]
  },
```

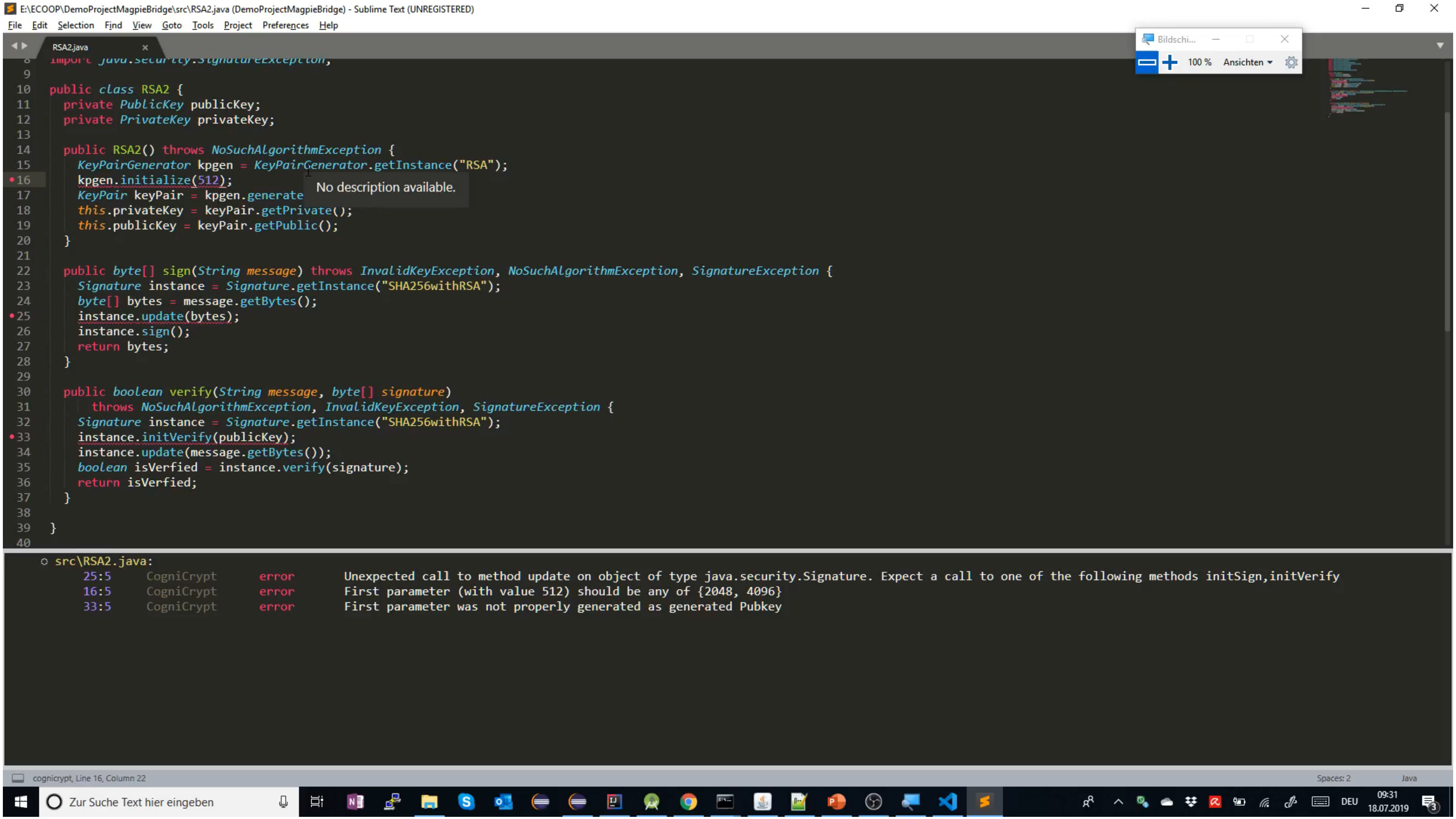
The MagpieBridge System



The MagpieBridge System



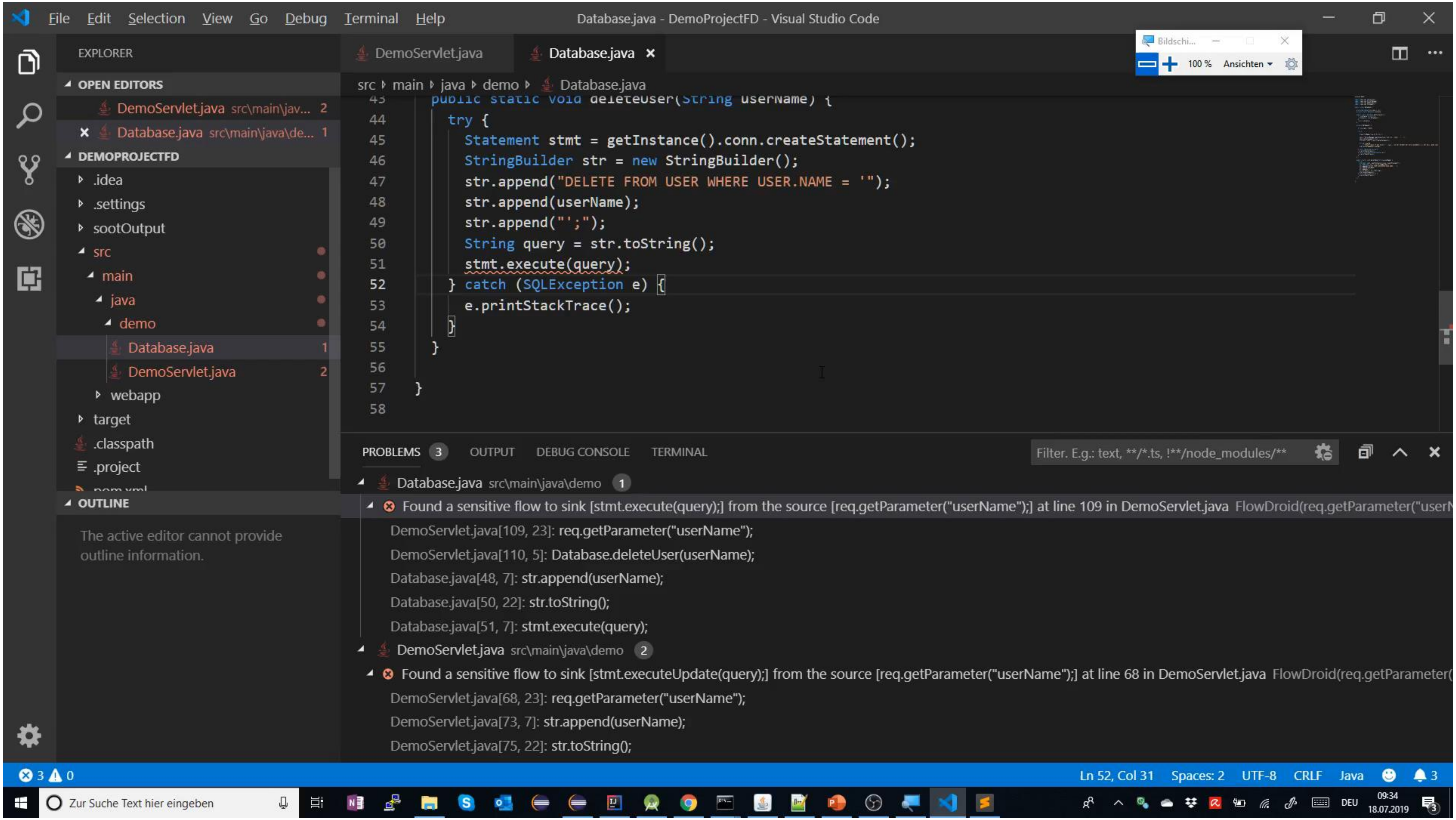
Demo: Providing Quick Fix in Sublime Text (CogniCrypt)


E:\ECOOP\DemoProjectMagpieBridge\src\RSA2.java (DemoProjectMagpieBridge) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
RSA2.java
Bildschi...
100 %
Ansichten
No description available.
cognicrypt, Line 16, Column 22
Spaces: 2
Java
Zur Suche Text hier eingeben
DEU
09:31
18.07.2019

```
 8  import java.security.SignatureException;
 9
10  public class RSA2 {
11    private PublicKey publicKey;
12    private PrivateKey privateKey;
13
14    public RSA2() throws NoSuchAlgorithmException {
15      KeyPairGenerator kpgen = KeyPairGenerator.getInstance("RSA");
16      kpgen.initialize(512);
17      KeyPair keyPair = kpgen.generate
18      this.privateKey = keyPair.getPrivate();
19      this.publicKey = keyPair.getPublic();
20    }
21
22    public byte[] sign(String message) throws InvalidKeyException, NoSuchAlgorithmException, SignatureException {
23      Signature instance = Signature.getInstance("SHA256withRSA");
24      byte[] bytes = message.getBytes();
25      instance.update(bytes);
26      instance.sign();
27      return bytes;
28    }
29
30    public boolean verify(String message, byte[] signature)
31        throws NoSuchAlgorithmException, InvalidKeyException, SignatureException {
32      Signature instance = Signature.getInstance("SHA256withRSA");
33      instance.initVerify(publicKey);
34      instance.update(message.getBytes());
35      boolean isVerfied = instance.verify(signature);
36      return isVerfied;
37    }
38
39  }
40
```

```
○ src\RSA2.java:
    25:5     CogniCrypt    error    Unexpected call to method update on object of type java.security.Signature. Expect a call to one of the following methods initSign,initVerify
    16:5     CogniCrypt    error    First parameter (with value 512) should be any of {2048, 4096}
    33:5     CogniCrypt    error    First parameter was not properly generated as generated Pubkey
```

Demo: Displaying Data-flow Path in Visual Studio Code (FlowDroid)



```
43      public static void deleteUser(String userName) {
44        try {
45          Statement stmt = getInstance().conn.createStatement();
46          StringBuilder str = new StringBuilder();
47          str.append("DELETE FROM USER WHERE USER.NAME = '");
48          str.append(userName);
49          str.append("';");
50          String query = str.toString();
51          stmt.execute(query);
52        } catch (SQLException e) {
53          e.printStackTrace();
54        }
55      }
56
57    }
58
```

Database.java src\main\java\demo 1

Found a sensitive flow to sink [stmt.execute(query);] from the source [req.getParameter("userName");] at line 109 in DemoServlet.java FlowDroid(req.getParameter("userN

DemoServlet.java[109, 23]: req.getParameter("userName");

DemoServlet.java[110, 5]: Database.deleteUser(userName);

Database.java[48, 7]: str.append(userName);

Database.java[50, 22]: str.toString();

Database.java[51, 7]: stmt.execute(query);

DemoServlet.java src\main\java\demo 2

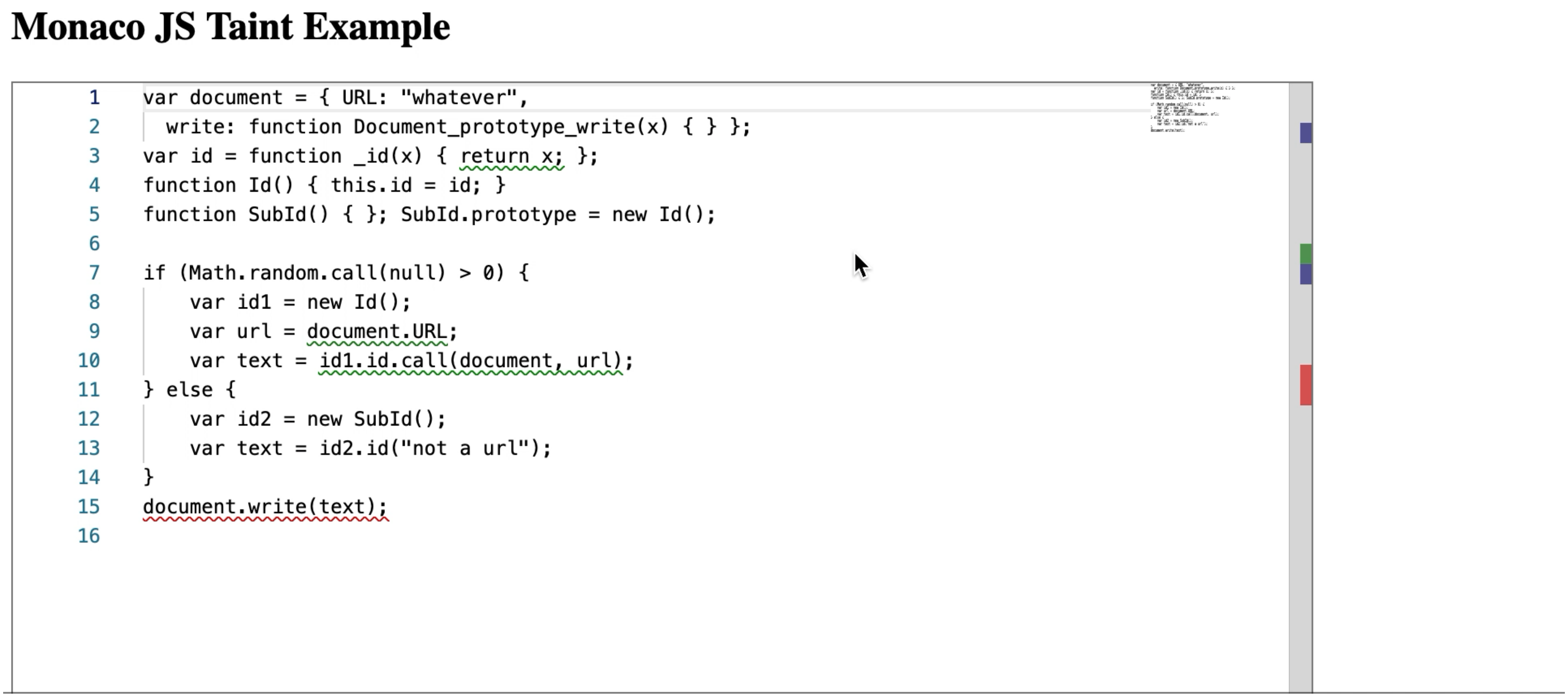
Found a sensitive flow to sink [stmt.executeUpdate(query);] from the source [req.getParameter("userName");] at line 68 in DemoServlet.java FlowDroid(req.getParameter(

DemoServlet.java[68, 23]: req.getParameter("userName");

DemoServlet.java[73, 7]: str.append(userName);

DemoServlet.java[75, 22]: str.toString();

Demo: Analyzing JavaScript Code in Monaco Web Editor



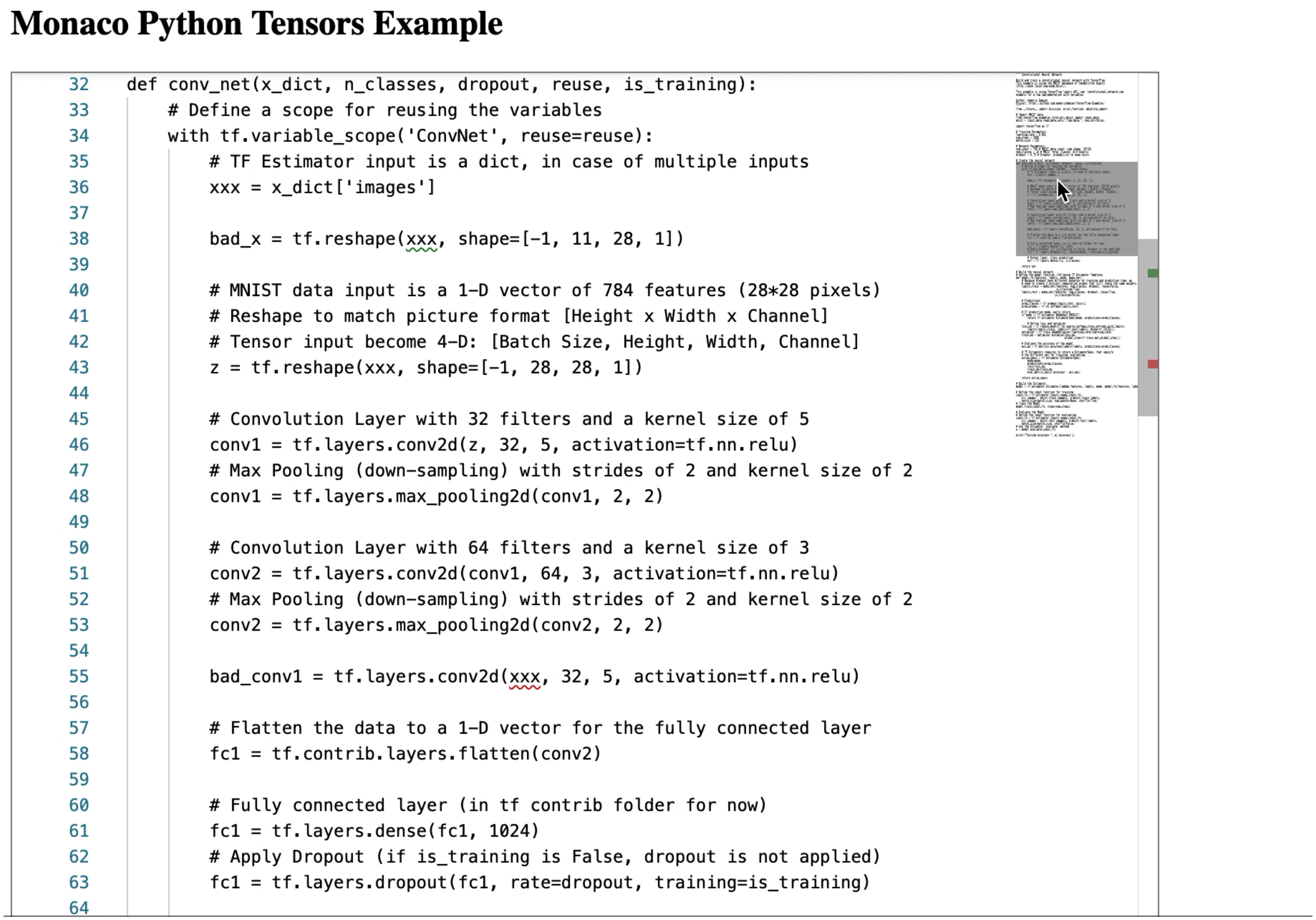
Monaco JS Taint Example

```
var document = { URL: "whatever",
  write: function Document_prototype_write(x) { } };
var id = function _id(x) { return x; };
function Id() { this.id = id; }
function SubId() { }; SubId.prototype = new Id();

if (Math.random.call(null) > 0) {
    var id1 = new Id();
    var url = document.URL;
    var text = id1.id.call(document, url);
} else {
    var id2 = new SubId();
    var text = id2.id("not a url");
}
document.write(text);

```

Demo: Analyzing Python Code in Monaco Web Editor (Ariadne)



Monaco Python Tensors Example

```python
def conv_net(x_dict, n_classes, dropout, reuse, is_training):
    # Define a scope for reusing the variables
    with tf.variable_scope('ConvNet', reuse=reuse):
        # TF Estimator input is a dict, in case of multiple inputs
        xxx = x_dict['images']

        bad_x = tf.reshape(xxx, shape=[-1, 11, 28, 1])

        # MNIST data input is a 1-D vector of 784 features (28*28 pixels)
        # Reshape to match picture format [Height x Width x Channel]
        # Tensor input become 4-D: [Batch Size, Height, Width, Channel]
        z = tf.reshape(xxx, shape=[-1, 28, 28, 1])

        # Convolution Layer with 32 filters and a kernel size of 5
        conv1 = tf.layers.conv2d(z, 32, 5, activation=tf.nn.relu)
        # Max Pooling (down-sampling) with strides of 2 and kernel size of 2
        conv1 = tf.layers.max_pooling2d(conv1, 2, 2)

        # Convolution Layer with 64 filters and a kernel size of 3
        conv2 = tf.layers.conv2d(conv1, 64, 3, activation=tf.nn.relu)
        # Max Pooling (down-sampling) with strides of 2 and kernel size of 2
        conv2 = tf.layers.max_pooling2d(conv2, 2, 2)

        bad_conv1 = tf.layers.conv2d(xxx, 32, 5, activation=tf.nn.relu)

        # Flatten the data to a 1-D vector for the fully connected layer
        fc1 = tf.contrib.layers.flatten(conv2)

        # Fully connected layer (in tf contrib folder for now)
        fc1 = tf.layers.dense(fc1, 1024)
        # Apply Dropout (if is_training is False, dropout is not applied)
        fc1 = tf.layers.dropout(fc1, rate=dropout, training=is_training)

```

Analysis Which Doesn’t Use the Frameworks

- MagpieBridge provides:
  - Different ways of LSP communication: standard I/O, sockets, Websockets
  - A set of LSP features
  - Resolution of project scope like source code path and library code path
  - Useful logs of the interactions with users
- To use MagpieBridge you need only provide source code positions
  - Add analysis to `MagpieServer` by implementing the `ServerAnalysis` interface
  - Adapt analysis results by implementing the `AnalysisResult` interface

https://github.com/MagpieBridge