

# TemPerf

## Temporal Correlation between Performance Metrics and Source Code

Jürgen Cito, Genc Mazlami, Philipp Leitner

 @citostyle @gmazlami @xl

University of Zurich, Switzerland



**Universität  
Zürich**<sup>UZH</sup>



# Feedback-Driven Development

(including TemPerf)

Jürgen Cito, Genc Mazlami, Philipp Leitner

 @citostyle @gmazlami @xLeitix

University of Zurich, Switzerland



**Universität  
Zürich**<sup>UZH</sup>



# Plan for this Demo

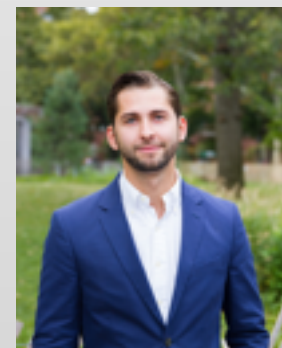
## Part I - The Theory:

Feedback-Driven Development, PerformanceHat, and Permo



## Part II - The (Actual) Demo:

Short look at PerformanceHat and Permo





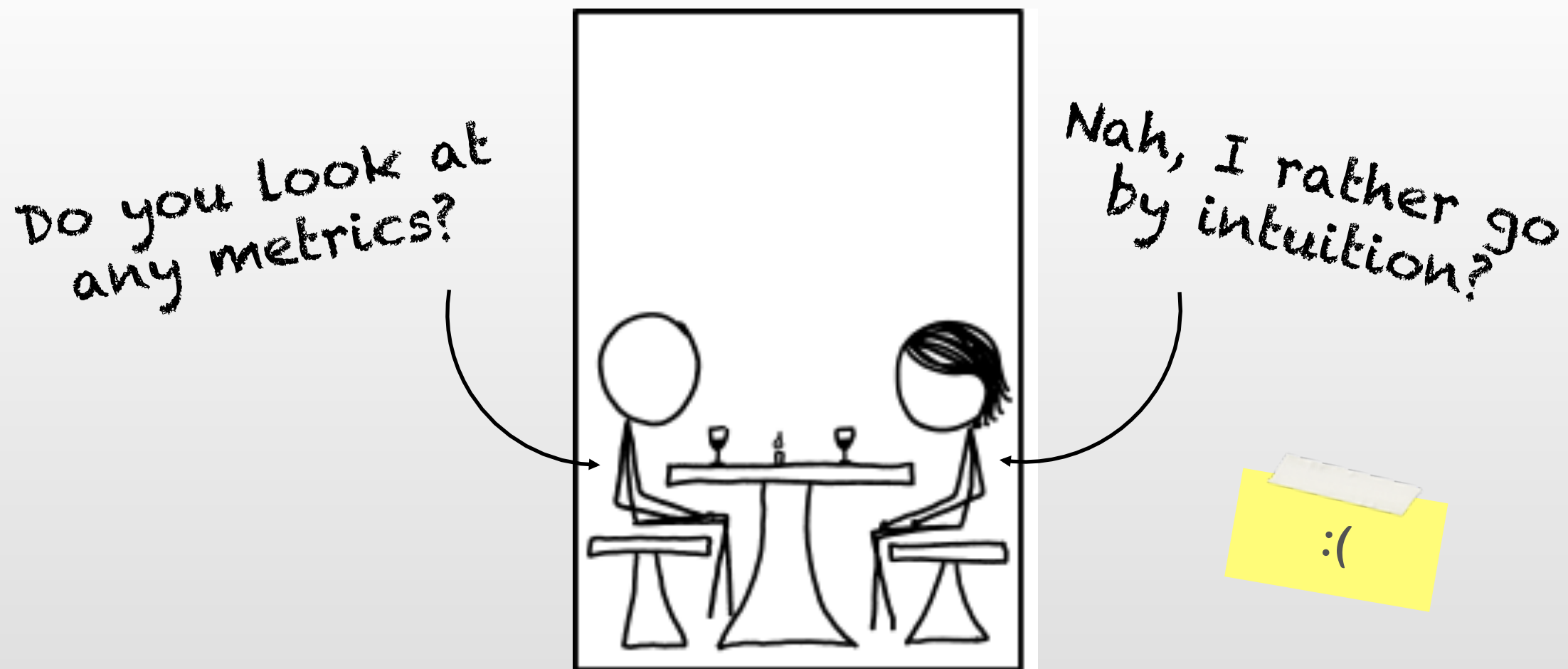
A 3D illustration featuring a dark grey network wall jack on the left and a clear plastic RJ45 connector with a grey Ethernet cable on the right. The background is a blue grid with a glowing blue globe on the left and a starry space-like pattern on the right. The text 'Part I: FDD' is overlaid in the bottom left corner.

# Part I: FDD



# How are performance dashboards used?

Interview Topic: Solving production performance problems



Adapted from <https://xkcd.com/1423/>

# Feedback-Driven Development (FDD)

## Basic Promise:

Making **runtime data actionable** for software  
developers Infrastructure

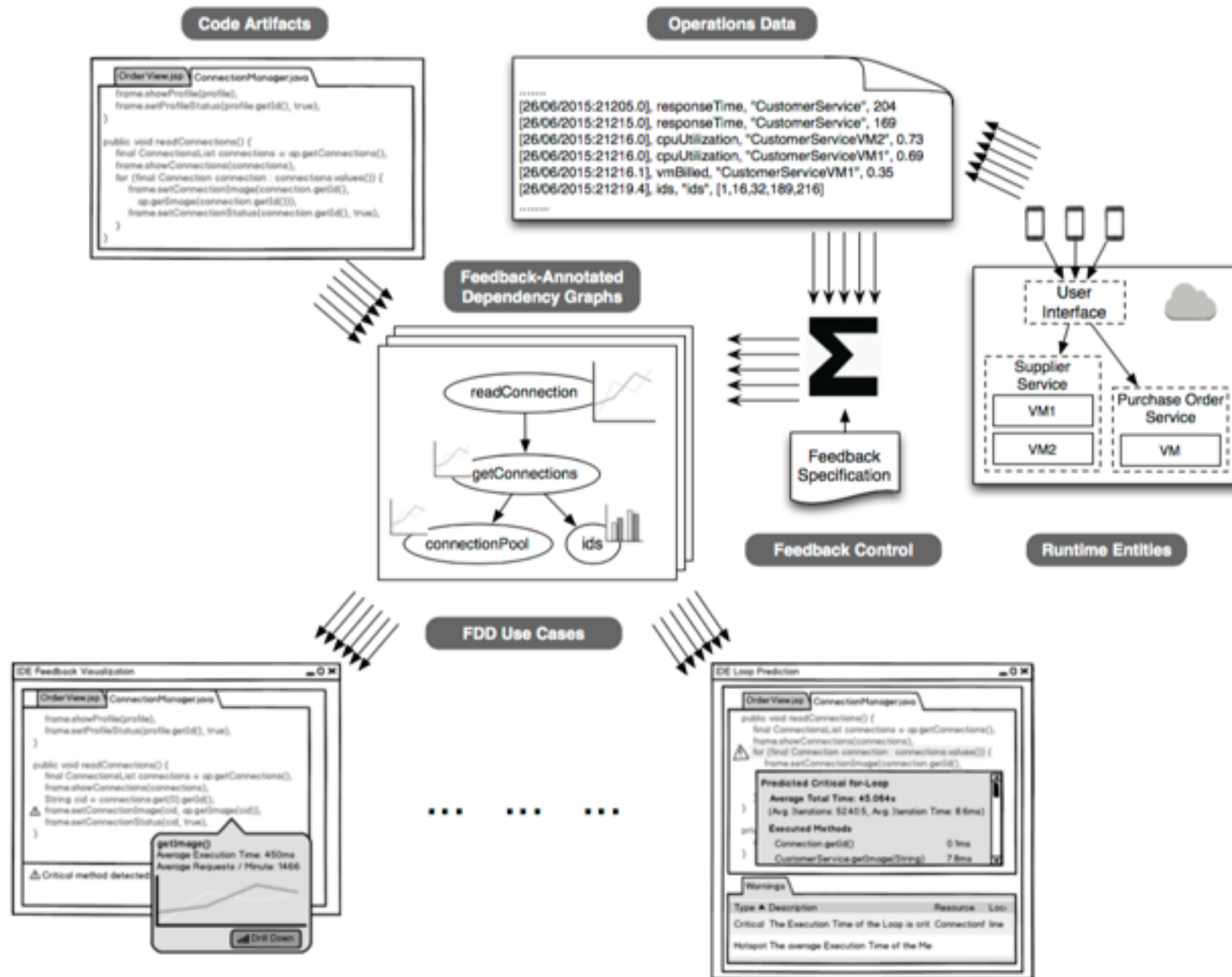
## Approach:

**Associate** runtime data to design-time  
artefacts

Display **directly in the IDE**

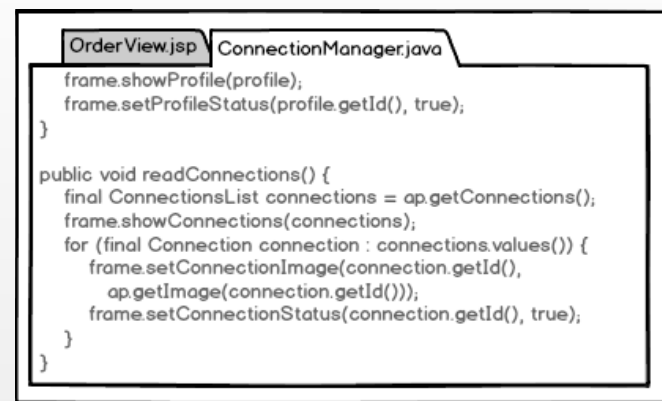
Support **drill-down**

# Main FDD Concepts - Feedback Mapping



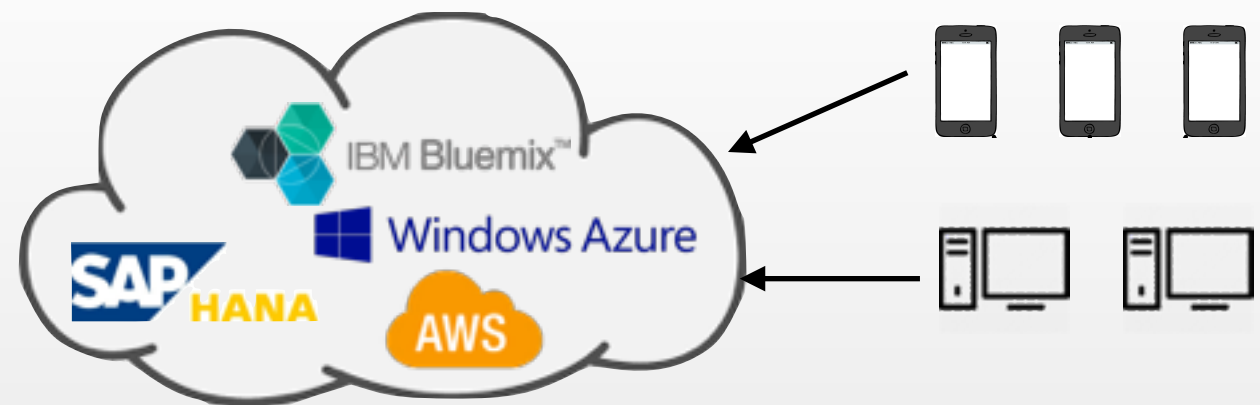
# Dependency Graph

## Code Artifacts



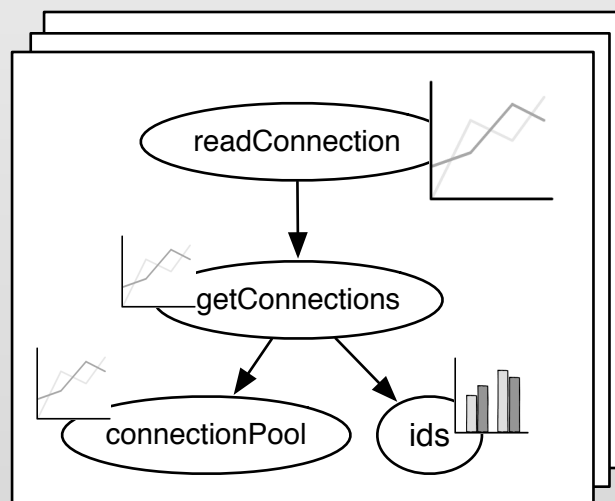
Deployment

## Cloud Infrastructure



observe

## Feedback Annotated Dependency Graph



## Operations Data

```
.....
[26/06/2015:21205.0], responseTime, "CustomerService", 204
[26/06/2015:21215.0], responseTime, "CustomerService", 169
[26/06/2015:21216.0], cpuUtilization, "CustomerServiceVM2", 0.73
[26/06/2015:21216.0], cpuUtilization, "CustomerServiceVM1", 0.69
[26/06/2015:21216.1], vmBilled, "CustomerServiceVM1", 0.35
[26/06/2015:21219.4], ids, "ids", [1,16,32,189,216]
.....
```



# Visualization (SAP HANA)

```
getPOs: function(oEvent, oController) {
    //Check for selected item for all other events
    var oTable = sap.ui.getCore().byId("poTable");
    var data = oTable.getModel();
    var poId = data.getProperty("PURCHASEORDERID",
    if (poId == undefined) {
        sap.ui.commons.MessageBox.show(oBundle.getText(
        "ERROR",
        oBundle.getText("error_action"));
    } else {
        oController.getPOItems(poId);

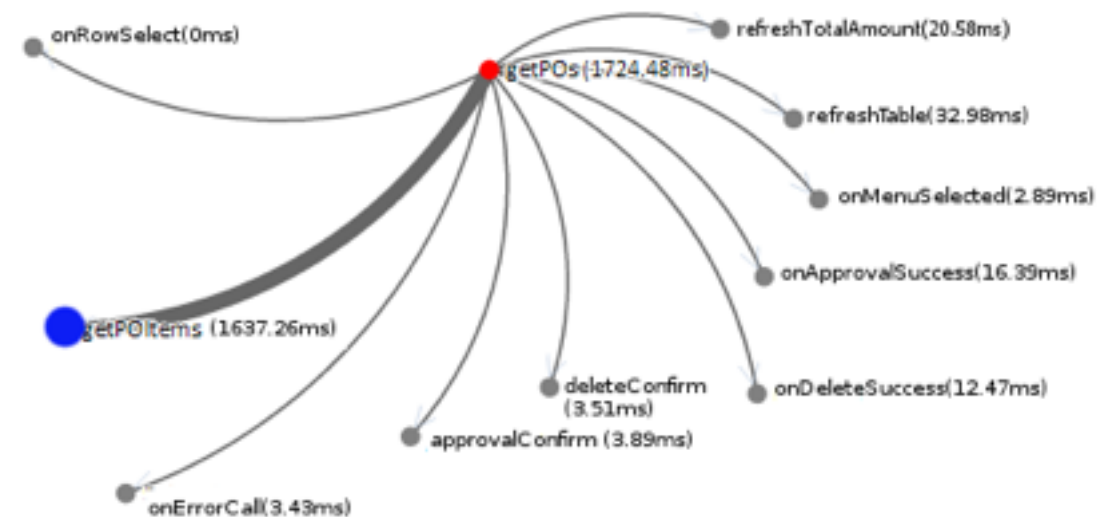
        switch (onMenuSelected(oEvent, oController)
        case "btnDelete":
            sap.ui.commons.MessageBox.confirm(o
            function(bResult) {
                oController.deleteConfirm(b
            },
            oBundle.getText("title_delete")
            break;
        case "btnEdit":
            oController.editConfirm(poId);
            break;
        case "btnApprove":
            oController.approvalConfirm(poId);
            break;
        }
    }
    refreshTable(oTable);
    refreshTotalAmount(oTable);
},

//Toolbar Menu Select Event Handler - Reject & Acce
onMenuSelected: function(oEvent, oController) {

    //Check for selected item for all other events
    var oTable = sap.ui.getCore().byId("poTable");
```

Average Execution Time

- ▲ getPOs(oEvent, oController)
- ▲ anonymous\_0 (bResult)
- ▲ onMenuSelected(oEvent, oController)
- ▲ anonymous\_0 (bResult)
- ▲ onRowSelect(oEvent)
- ▲ deleteConfirm(bResult, oController, poId)
- ▲ success(myTxt)
- ▲ approvalConfirm(bResult, oController, poId, action)
- ▲ success(myTxt)
- ▲ onDeleteSuccess(myTxt, oController)
- ▲ onApprovalSuccess(myTxt, oController, action)
- ▲ onErrorCall(jqXHR, textStatus, errorThrown)
- ▲ refreshTable(oController)
- ▲ success(myTxt)
- ▲ refreshTotalAmount(myTxt)
- ▲ getPOItems()



# Prediction of Regressions

The screenshot displays an IDE interface. On the left is a project explorer showing a Java project with a package structure including `eu.cloudwave.wp5.demo.voip`. The main editor shows a Java file with a `login()` method. A tooltip is overlaid on the code, indicating that a loop has been identified as critical and providing performance metrics.

```
267 */
268 private void login() {
269     call_all.setEnabled(true);
270
271     for(String user : this.getUsers()) {
272         // ...
273         item().toString();
274     }
275 }
276
277 private
278 call
279 stop
280 login_all.setText(getProp( LOGINALL_TXT ));
```

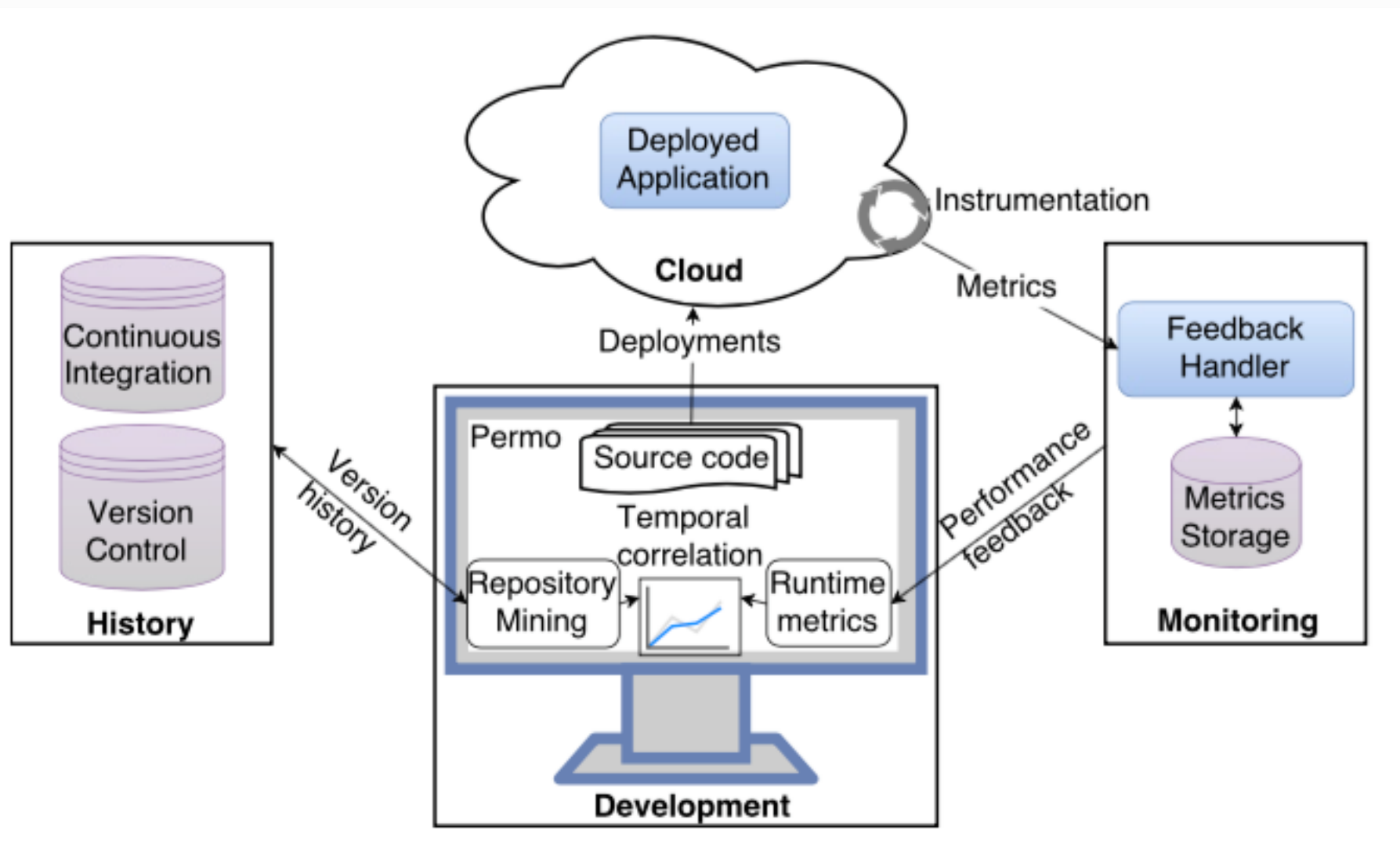
This loop has been identified as critical.

**The Average Total Time is: 19.352s**  
Average Iterations: 14.0; Average Time per Iteration: 1.382s

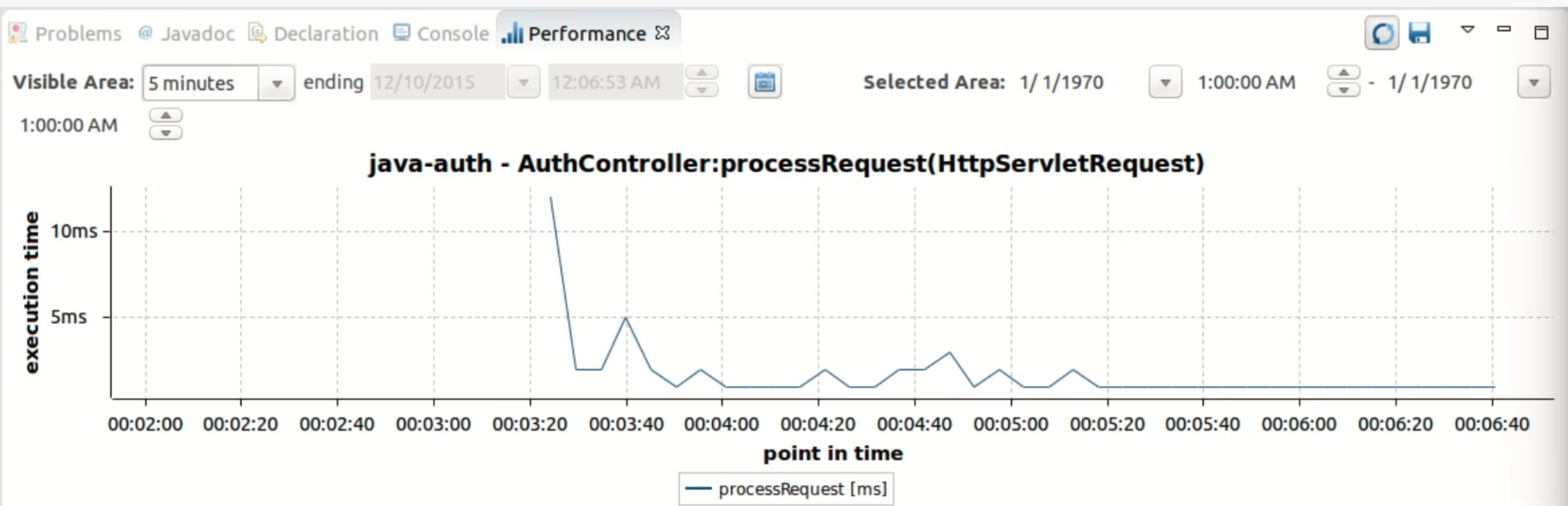
**Executed methods:**

VoIPFrame.getVboxes:	0.0ms
Hashtable.get:	0.0ms
JComboBox.getSelectedItem:	0.0ms
VBox.getPassword:	0.0ms
DataManager.DataManager:	1.964ms
Object.toString:	0.0ms
DataManager.start:	1.38s

# TempPerf



# TempPerf



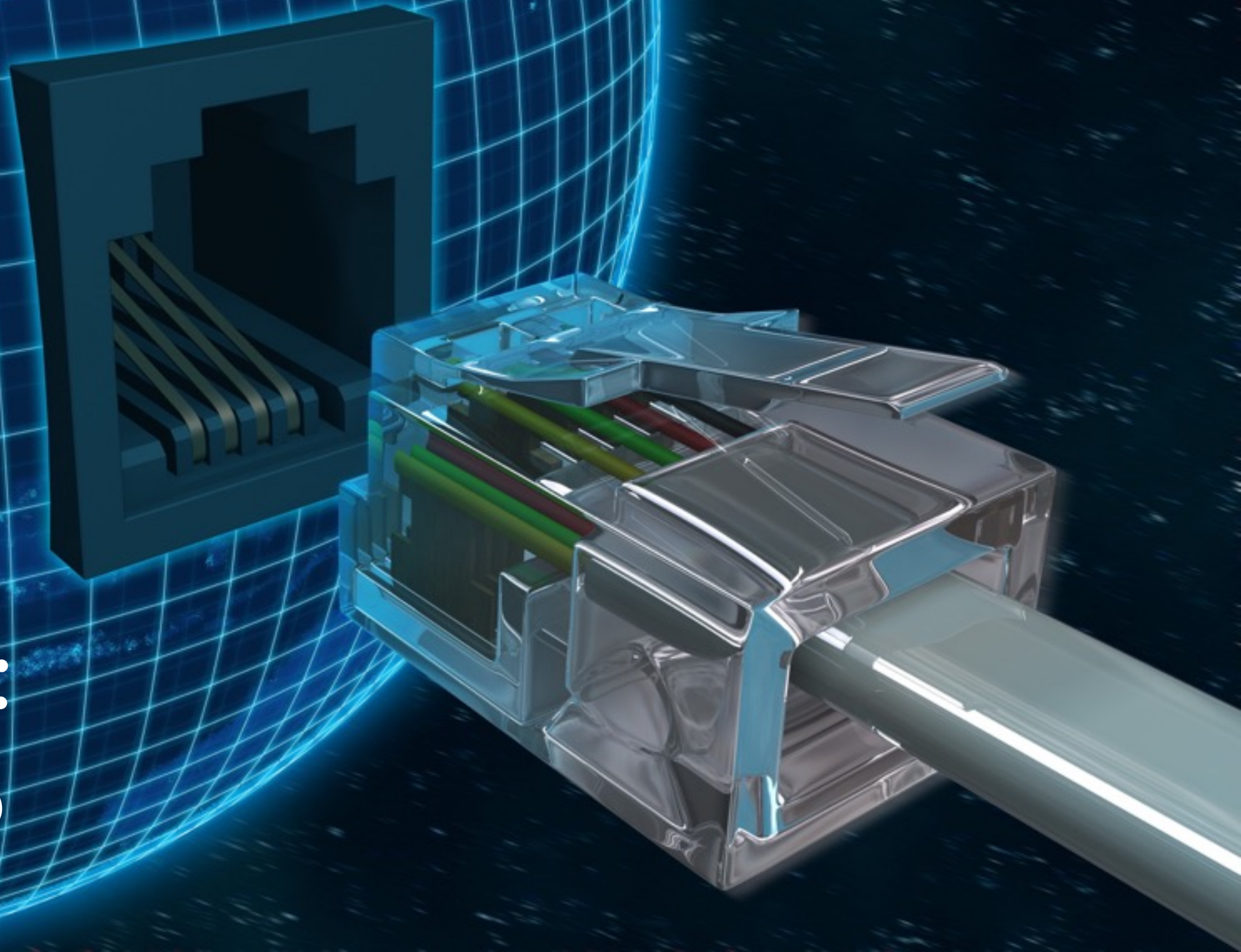


# TempPerf

Outline		Project History	
<b>guava</b>		23.03.15	23.03.15
		17:23:01	20:32:45
23.03.15 19:21	Prevent rare disastrous classloading in first call to LockSupport.p		
23.03.15 19:20	Eliminate boxing in RegularImmutableMultiset by inlining the h		
23.03.15 18:58	Round 2 in the great ImmutableCollection javadoc rewrite. Try t		
23.03.15 18:55	When a new entry exceeds the segment's maxWeight all by itsel		
23.03.15 18:16	Apply some simplifications and improvements. ----- Cr		
23.03.15 18:15	Use Hashing.smearedHash directly in HashBiMap. -----		
23.03.15 18:15	Make implicit narrowing compound assignments explicit.		
23.03.15 18:15	Make it explicit that a memoized Supplier's get() method is invo		
23.03.15 18:15	Improve the implementation of Multisets.ElementSet.remove to		
23.03.15 18:15	Document that callers of catching() should prefer more specific		
23.03.15 18:15	Migrate calls of to-be-deprecated Futures.get to equivalent Futu		



# Part II: Demo





# Prediction of Regressions

```
/**
 *
 * LOGIN Connection to server
 */
private void login() {
    call_all.setEnabled(true);
    VTestBox b = singleUserVBox;

    String userName = b.user.getSelectedItem().toString();
    String pwd = b.getPassword();
    new DataManager(b).start(userName, pwd);
}
```


# Discussion Points

How should we communicate **uncertainty** of our models?

How much feedback is **too much** feedback?



<http://cloudwave-fp7.eu>

 @cloudwavefp7

Jürgen Cito, Genc Mazlami, Philipp leitner

 @citostyle    @gmazlami    @xLeitix