

Arrowhead Tools

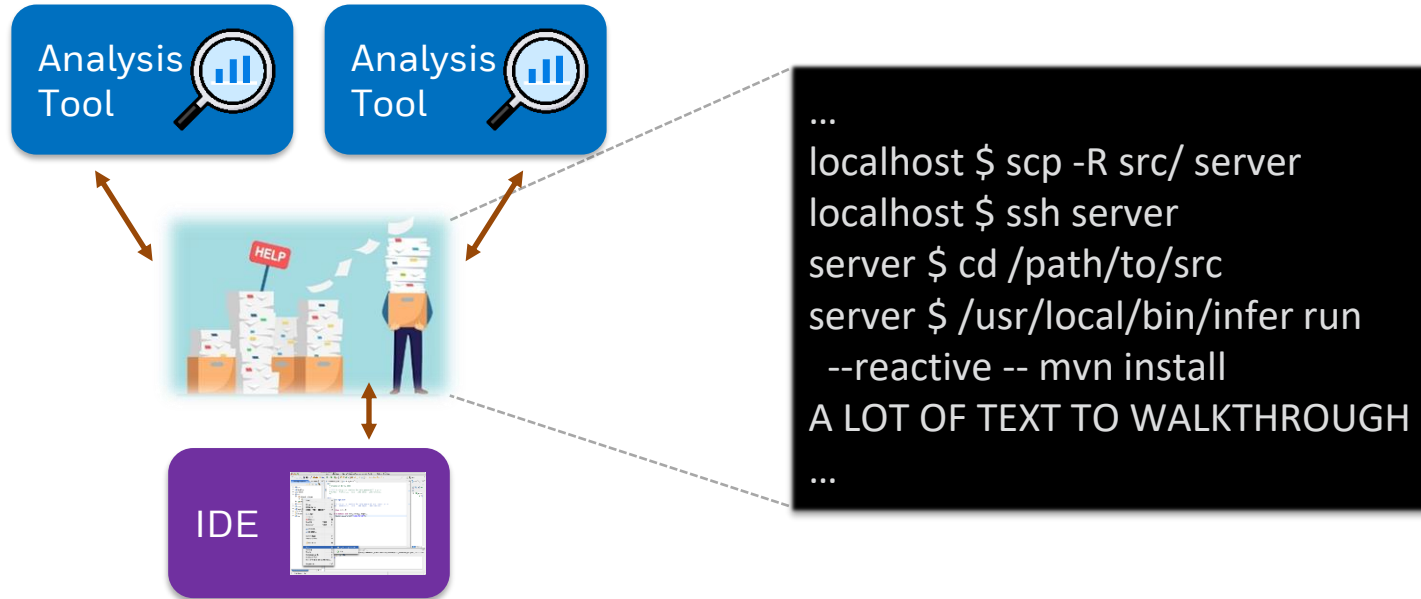
WP4 / WP5 / WP8 UC1

Automated Formal Verification:
Integrating tools with Eclipse
using Unite and UniC

Jan Fiedor
VeriFIT research group

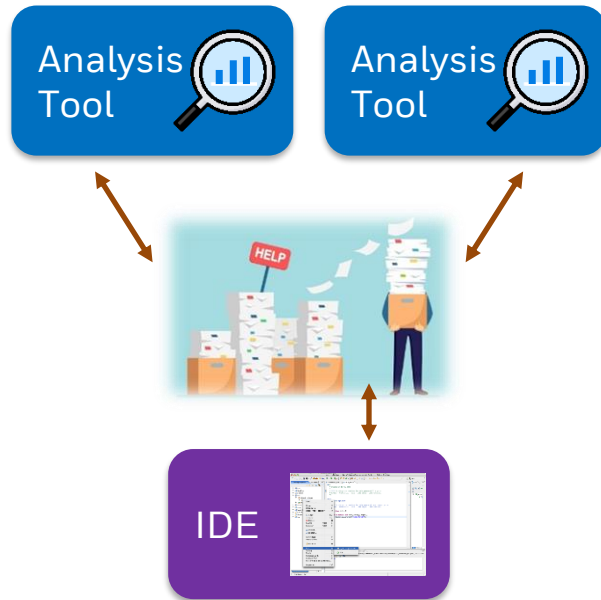
Motivation

Before AHT

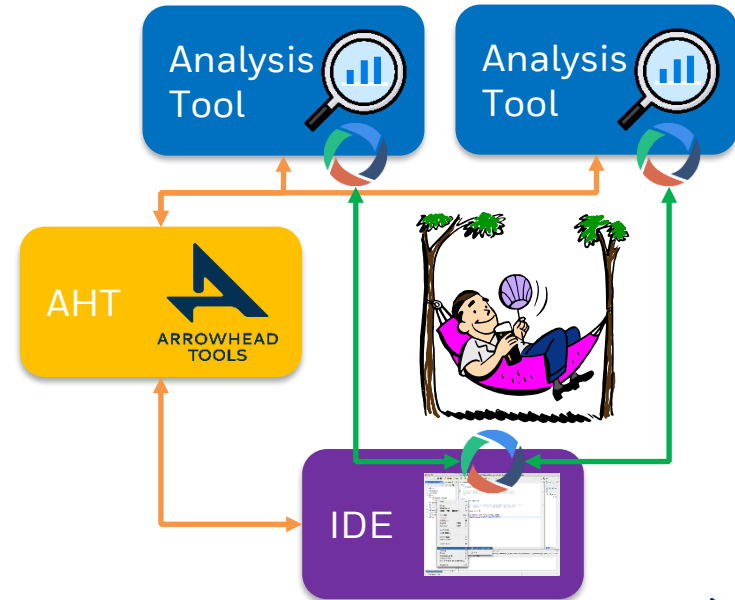


Motivation

Before AHT

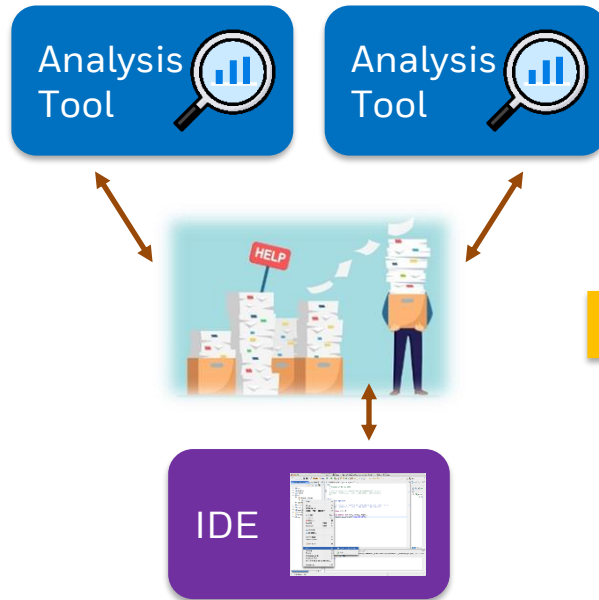


After AHT



Motivation

Before AHT



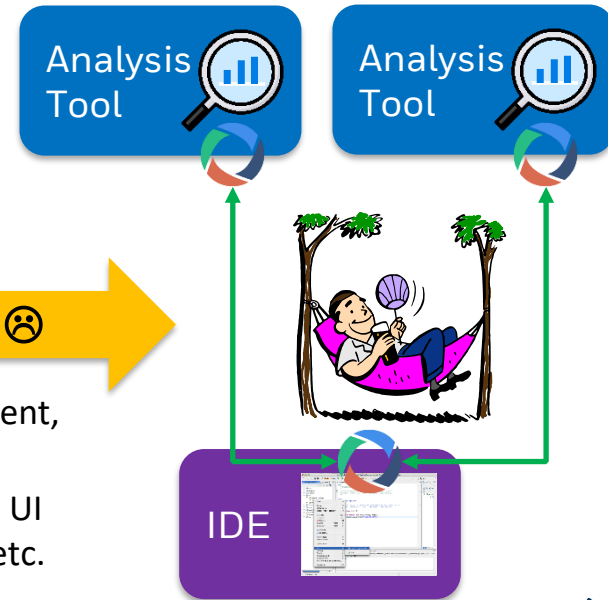
A lot of work 😞

OSLC: adapters, client, domain, etc.

Eclipse: execution, UI integration, data, etc.

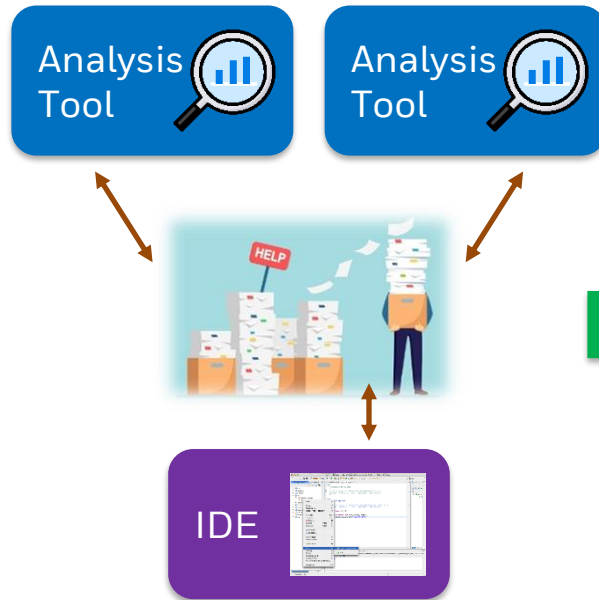


After AHT



Motivation

Before AHT



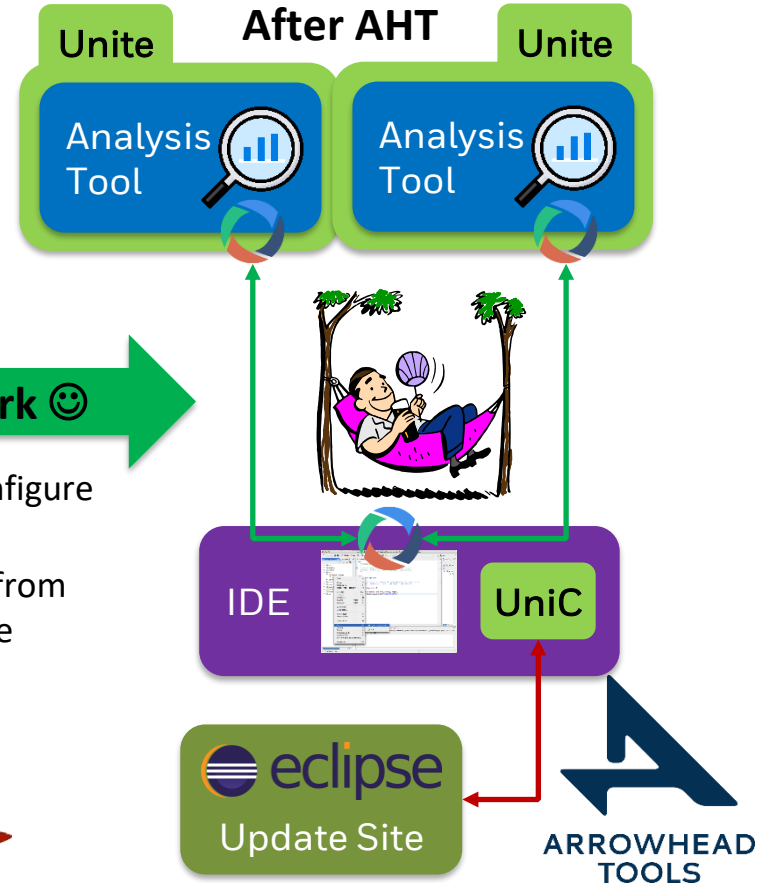
Very little work 😊

Tool provider: configure Unite and UniC

User: install UniC from Eclipse Update Site

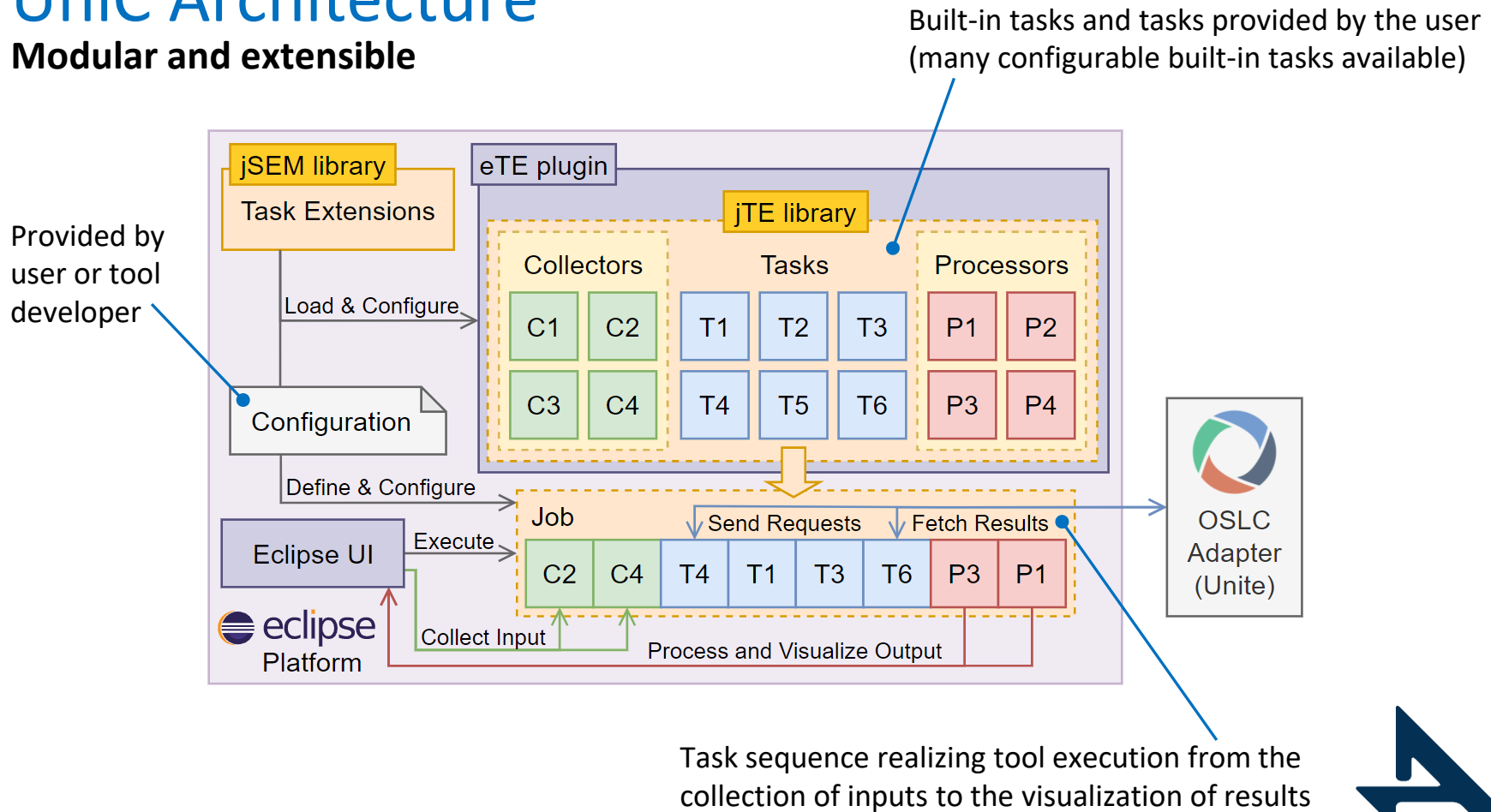


After AHT



UniC Architecture

Modular and extensible



Static Analysis With Infer in a Couple of Minutes

1. Open the Eclipse IDE and add the VeriFIT Eclipse update site
Name: VeriFIT
Location: <https://verifit1.fit.vutbr.cz/eclipse/>
2. Install UniC with Infer configuration
 1. Run **Help -> Install New Software...**
 2. Select the **VeriFIT** update site
 3. Choose **Infer on BUT server (eTE Task)** and start the installation
 - UniC and all of its dependencies will be installed automatically
3. Start analysis of the selected project
 1. Set **build command** in project properties -> **Unite -> SUT**
 2. Right-click on Eclipse project (or any of its files)
 3. Run **Tasks -> Analyse with Facebook Infer (BUT server)**

UniC Q&A

- Can I access (and execute) tasks from my Eclipse plugin? **Yes**

```
IEclipseTaskExecutorService eteSvc = PlatformUI.getWorkbench().getService(  
    IEclipseTaskExecutorService.class);
```

- Can I add a task (sequence) to UniC from my Eclipse plugin? **Yes**

1. Implement the TaskProvider interface
2. Register it using cz.vutbr.fit.group.verifit.ete.task.provider extension point

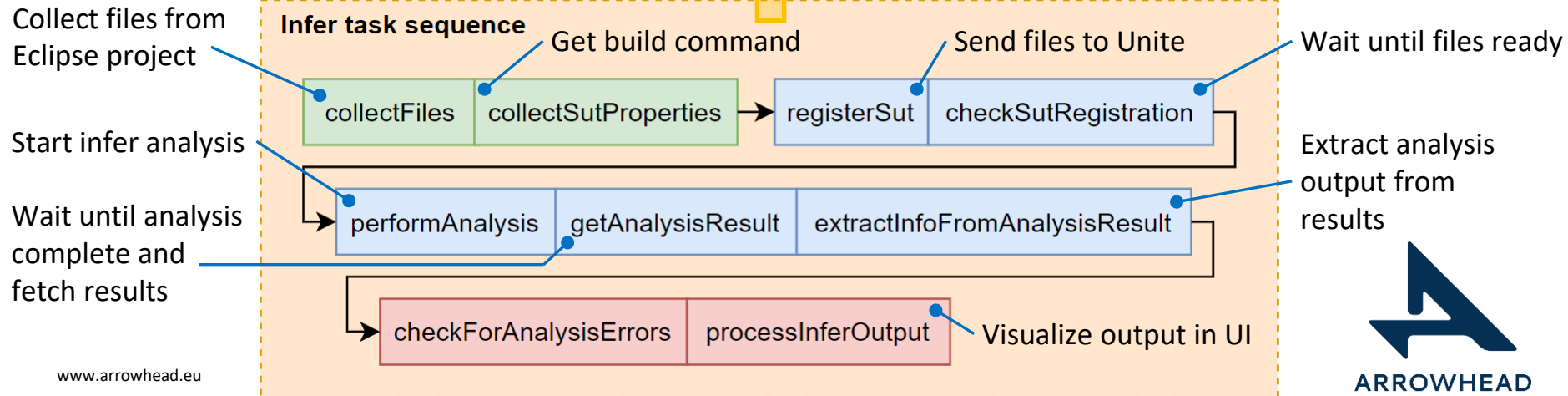
- Can I add a task (sequence) to UniC from a local directory? **Yes**

1. Set the directory in Window -> Preferences -> Task Executor
2. Restart Eclipse IDE to load tasks from all files present in the directory

- Can I add a task (sequence) to UniC from Eclipse Arrowhead? **Not yet**
- Can I reload tasks without restarting Eclipse IDE? **Not yet**
- Can I use UniC without Eclipse? **Yes, it is a set of Java libraries**

Facebook Infer UniC Task Sequence Example

```
infer.extension=cz.vutbr.fit.group.verifit.jte.ConfigurableJobExtension
infer.info.provides=cz.vutbr.fit.group.verifit.jte.Pipeline
infer.info.name=verifit.infer
infer.info.ete.label=Analyse with Facebook Infer (BUT server)
infer.info.ete.category=user-task
infer.config.tasks[]=infer.collectFiles,infer.collectSutProperties,\
infer.registerSut,infer.checkSutRegistration,infer.performAnalysis,\
infer.getAnalysisResult,infer.extractInfoFromAnalysisResult,\
infer.checkForAnalysisErrors,infer.processInferOutput
```



Selecting Tools, Adding Immutable Parameters

```
unite.request.params[ ]=sub-command,options,SUTbuildCommand
unite.request.param.sub-command=run
unite.request.param.options=--reactive
unite.request.param.SUTbuildCommand=true
unite.analysisAutomationPlanName=infer
```

- jTE's **PerformAnalysis** task can invoke any tool wrapped by Unite with given input parameters
 - **unite.analysisAutomationPlanName** parameter defines which tool to invoke on a remote server running Unite
 - **unite.request.params** lists the names of input parameters to add to a Unite request
 - The entries under the **unite.request.param** section define the values of these parameters

Allowing User-Specified Parameters

```
unite.request.params[]=unpackZip,buildCommand,compile
unite.request.param.unpackZip=true
unite.request.param.buildCommand=${unite.sut.buildCommand}
unite.request.param.compile=false
```

- Some parameters, such as jTE's **RegisterSut** task's **buildCommand** parameter, require their values to be supplied by the user
- There are various ways how to do that, however, the recommended approach is the following:
 1. Assign the value to a variable stored in the task's data (the collectSutProperties task stores the **build command** from project properties -> **Unite** -> **SUT** into **unite.sut.buildCommand**)
 2. Use the value of the variable when configuring the parameters (Use **\${unite.sut.buildCommand}** as the value)

How to Define a Task Sequence for Your Tool

- For tools wrapped by Unite, one can modify the Facebook Infer task sequence available here: <https://pajda.fit.vutbr.cz/verifit/ete-verifit-tasks/-/blob/master/plugins/ete-verifit-infer/ete/tasks/infer.tasks>
- The `tasks[]` entry of the task sequence lists the **names** of tasks to use (values of the `info.name` entries each task has), names must be unique, and each named task (instance) has its own configuration!
- In the `performAnalysis` task configuration, set the name of the automation plan of your tool and change the parameters (see sl. 10)
- In the `registerSut` task configuration, change the `compile` parameter to `true` if your tool need to compile the SUT before using it (see sl. 11)
- Replace the `processInferOutput` task with a task which can process the output of your tool, if you only need to print the standard output, you can use eTE's `TaskOutputPrinter` task to print the raw output