AsTeRICS Packaging Environment (APE)

Manage AsTeRICS AT solution as dedicated SW project

`ant APE-copy`

- Project structure template
- Set of tools and **build infrastructure** (ant)
- Creates a downstripped version of the ARE
  - (including plugins, configuration files and data files to execute the models.)
- Enables customization of ARE (Replace images, config files,…)

Name
- custom
- package
- Tests
- APE.properties
- build.xml
- imported.xml
- README.md

```
APE-copy -DAPE.models=../ARE/models/ImageDemo.acs
```

What is AsTeRICS Packaging Environment?

- Enables creation of **native installer** and **native launcher**
  - **Windows (.msi, .exe)**
  - **Linux incl. Raspberry Pi (.deb)**
  - **Mac OS X (.dmg)**
  - Desktop integration
  - With / without Java Runtime embedded
- Helps in license management
  - **GPLv3 / GPLv3 with CLASSPATH exception** or **MIT**?
  - Provides set of involved license files
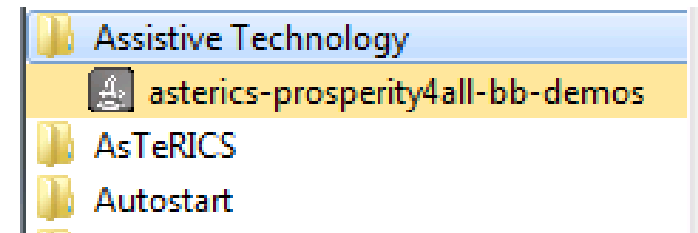
Implementers can **create**, **host**, **customize**, **release/deploy** AT SW based on the AsTeRICS framework **themselves**.

Intended workflow using APE
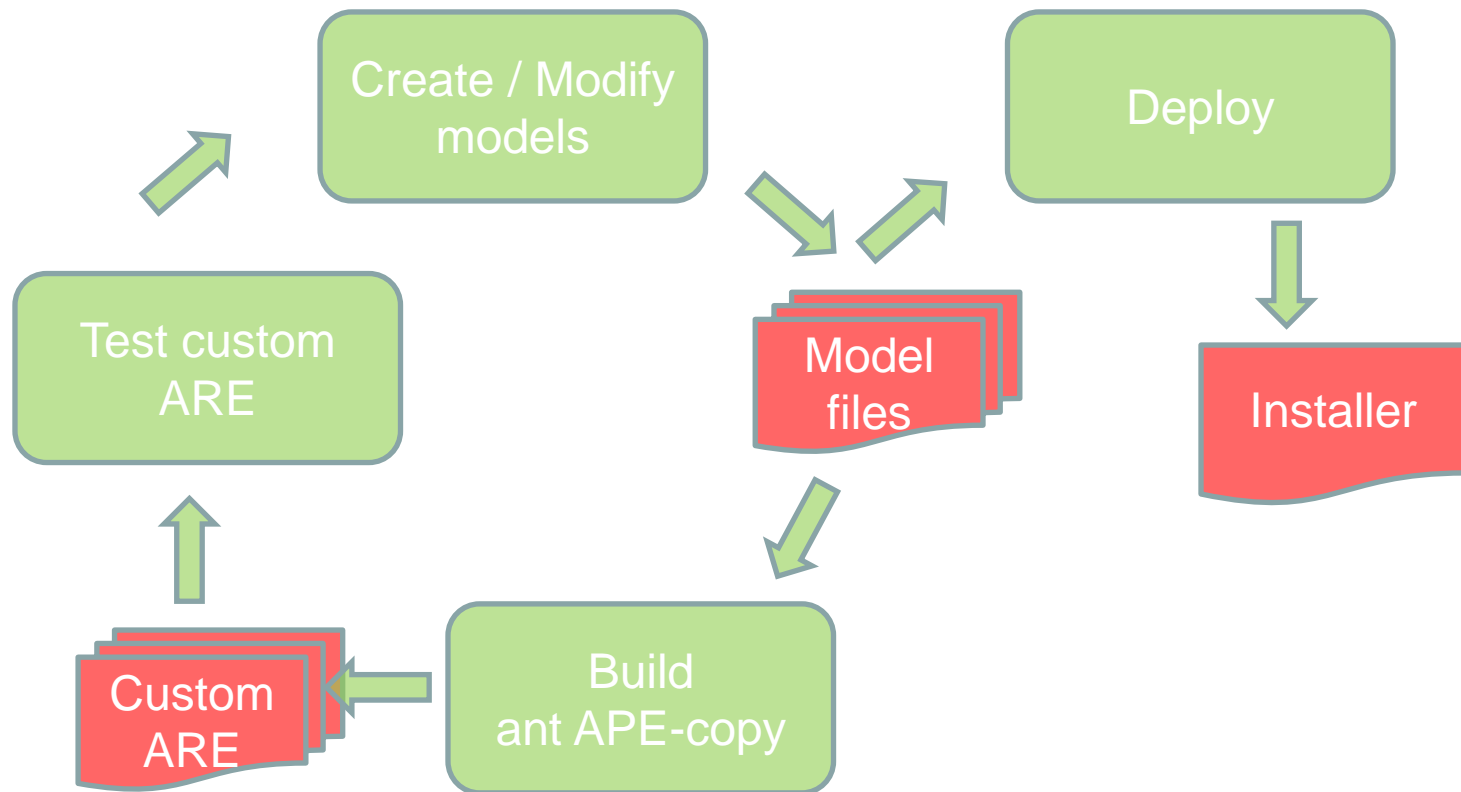
APE folder location

- Full AsTeRICS installer
  - C:\Program Files (x86)\AsTeRICS\APE
- Platform independent .zip
  - APE
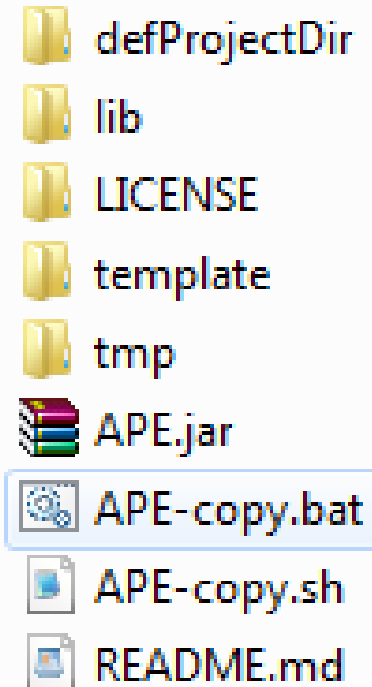
A command line tool to create a down-stripped version of the ARE based on provided model files.

One model file

Create a downstripped ARE package of the model file `ImageDemo.acs` located in the `ARE/models` folder. By default, the result is written to the folder `<APE.buildDir>/merged`.

```
cd <APE.baseURI>
APE-copy -DAPE.models=../ARE/models/ImageDemo.acs
```

Build infrastructure

Template project folder

ant build files

Project properties (APE.properties)

One model file, ant APE-copy

Copy the model file `<ARE.baseURI>/models/ImageDemo.acs` to the location `<APE.projectDir>/custom/bin/ARE/models` or edit the `APE.models` property in the file `<APE.projectDir>/APE.properties`. Then execute the following commands:

```
cd <APE.projectDir>
ant APE-copy
```

```
#Set the base URI of the ARE to use
#Default: Relative to the location of APE.jar
#ARE.baseURI=../../ARE/

#Set the model src file(s) / src folder(s), either relative to APE.projectDir
or as absolute directory or file URIs
#Seperate several URIs by semicolon (;)
#Default: bin/ARE/models (this value will always be appended to the property
to include custom models in the custom/bin/ARE/models subfolder)
#APE.models=../../ARE/models/CameraMouse.acs;../../ARE/models/ImageDemo.acs

#The path to the build folder, can be relative or absolute
#Default: build
#APE.buildDir=build/

#currently: exe|msi|deb|image|all|none
#image: Only the final folder structure is created which can be used to create
a .zip file.
#all: all supported installer types for the platform are created
fx.deploy.nativeBundles=all

#Name and version of the application
fx.application.name=CamMouse-Webinar
fx.application.version=0.1

#Provide more detailed information here
fx.info.title=Assistive Technology Application
fx.info.vendor=Asterics consortium
fx.info.description=Assistive Technology Application based on AsTeRICS
fx.info.license=MIT | GPLv3 with Exception
fx.info.category=Assistive Technology
```

Demo: Create custom Camera Mouse

Requirements
[Java Development Kit 8 (32-bit)](#)
[Apache ant >= 1.9.1](#)

1. Create simple camera mouse model of tutorial before

2. Save model to folder
   `C:\Program Files (x86)\AsTeRICS\APE\defProjectDir\custom\bin\ARE\models`

3. Change to
   `cd C:\Program Files (x86)\AsTeRICS\APE\defProjectDir\`

4. Edit `APE.properties` (e.g. change `fx.application.name`)

5. Open command window (Shift+Right click, Open command window here)

6. Execute
   `ant APE-copy`

7. Start `ARE.exe` in
   `C:\Program Files (x86)\AsTeRICS\APE\defProjectDir\build\merged\bin\ARE`

Shrink of complete ARE of AsTeRICS framework to custom ARE of camera mouse

269 MB → 62.5 MB

Repositories of AsTeRICS demos (building blocks)

- The demos are hosted as **APE-based projects** at the **P4AllBuildingBlocks repository**

- The repositories are **organized in sub-folders**

- Can be built and deployed **stand-alone** (as single building block) or **in a bundle**

- Can be reused as template projects (forked)

- Check screencast about camera mouse packaging

Create a standalone
Camera Mouse
SW package in minutes

Link Summary

- AsTeRICS homepage: http://www.asterics.eu/

- AsTeRICS github: https://github.com/asterics/AsTeRICS

- Demo page: http://asterics.github.io/AsTeRICS/demos.html

- P4AllBuildingBlocks: https://github.com/asterics/P4AllBuildingBlocks

- APE: https://github.com/asterics/AsTeRICS/wiki/AsTeRICS-Packaging-Environment-(APE)