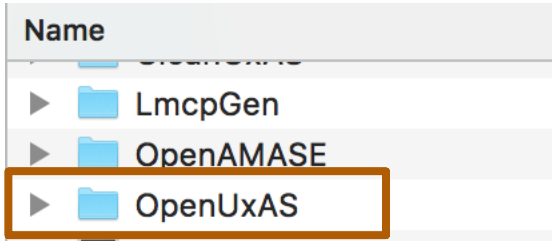
Unmanned Systems Autonomy Services: Build & File System

Dr. Derek Kingston
Control Science Center of Excellence
Aerospace Systems Directorate
Air Force Research Laboratory

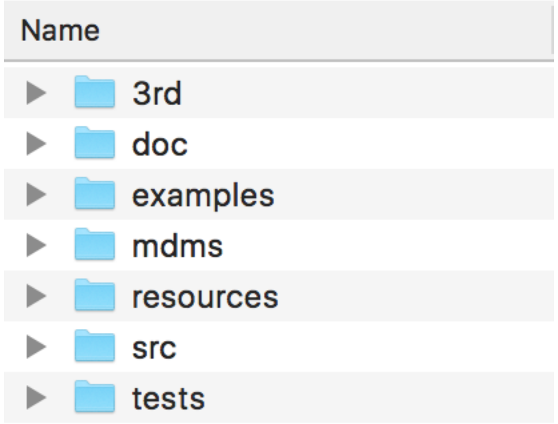
Integrity ★ Service ★ Excellence

File Structure



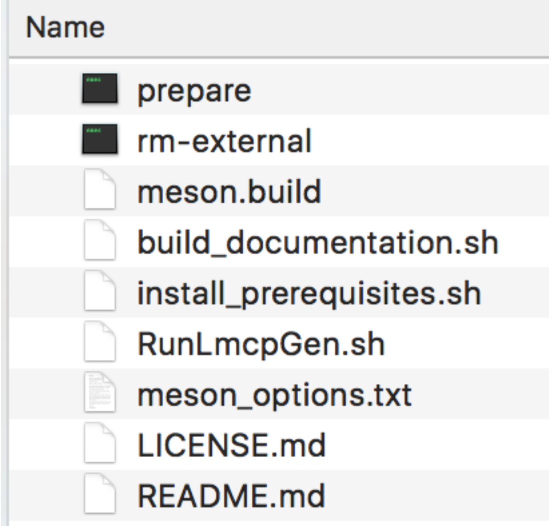
Name

- LmcpGen
- OpenAMASE
- OpenUxAS



Name

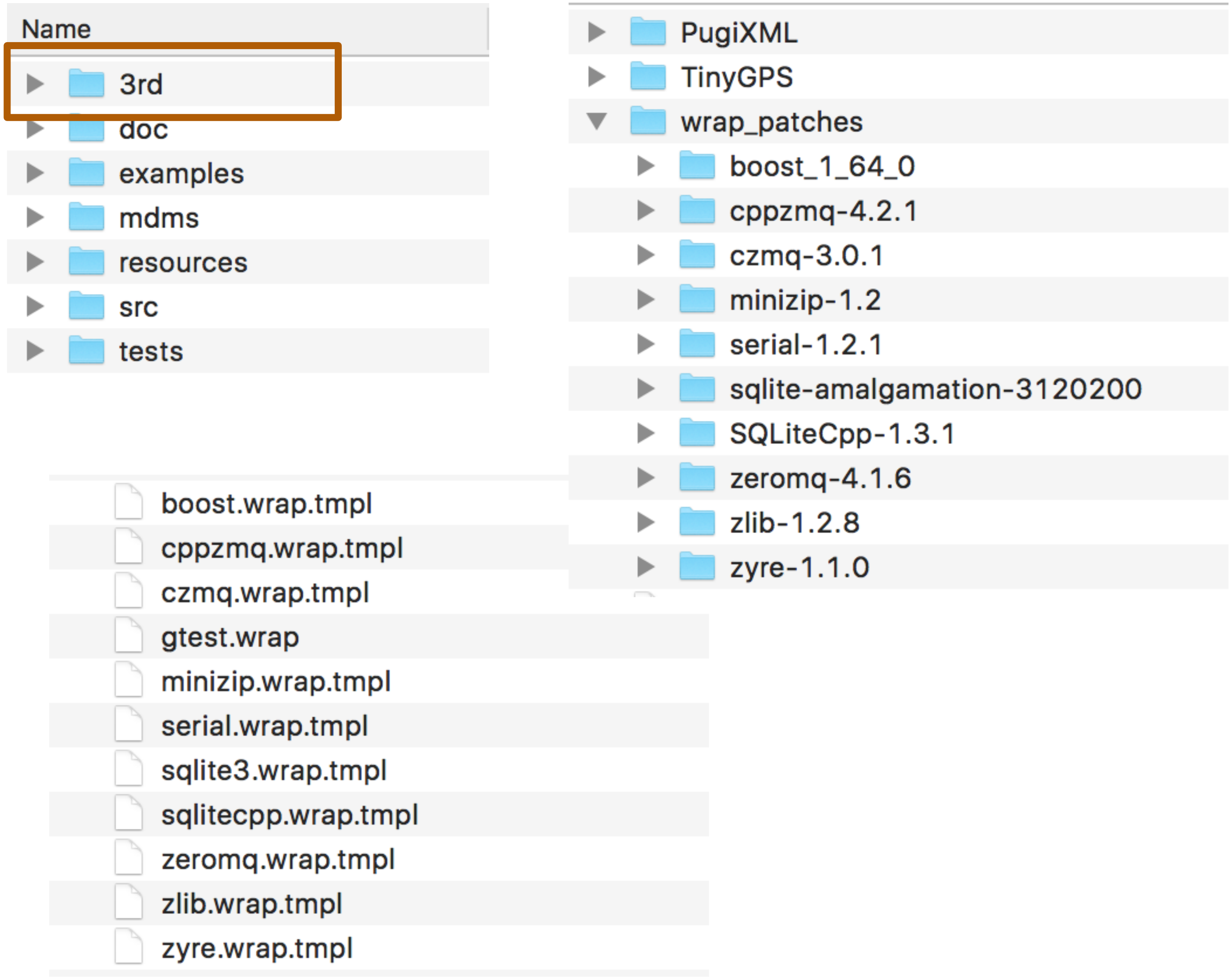
- 3rd
- doc
- examples
- mdms
- resources
- src
- tests



Name

- prepare
- rm-external
- meson.build
- build_documentation.sh
- install_prerequisites.sh
- RunLmcpGen.sh
- meson_options.txt
- LICENSE.md
- README.md

3rd Party



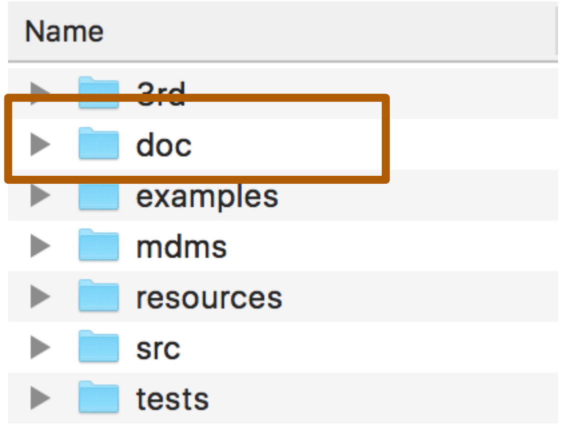
Name

- 3rd
- doc
- examples
- mdms
- resources
- src
- tests

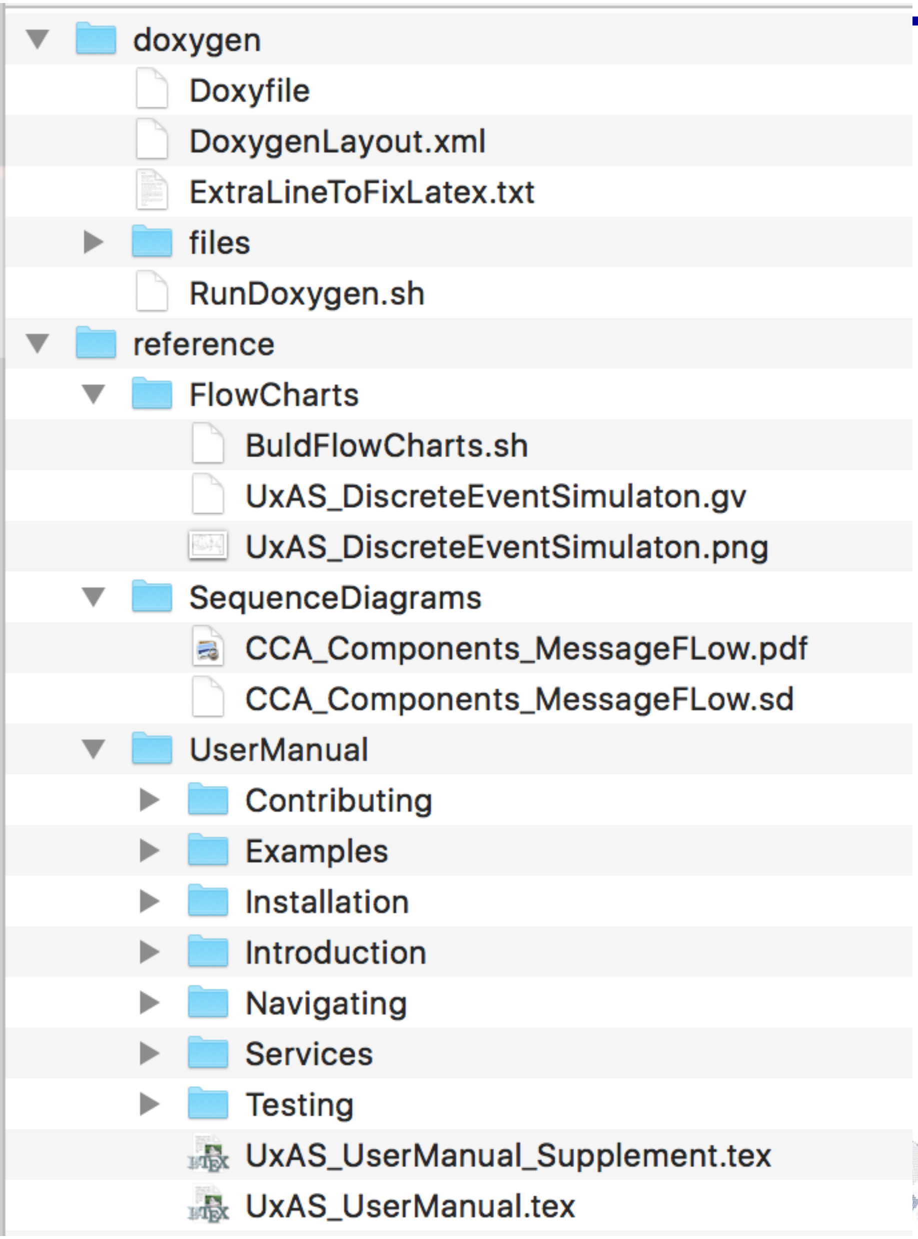
- PugiXML
- TinyGPS
- wrap_patches
  - boost_1_64_0
  - cppzmq-4.2.1
  - czmq-3.0.1
  - minizip-1.2
  - serial-1.2.1
  - sqlite-amalgamation-3120200
  - SQLiteCpp-1.3.1
  - zeromq-4.1.6
  - zlib-1.2.8
  - zyre-1.1.0

- boost.wrap.tmpl
- cppzmq.wrap.tmpl
- czmq.wrap.tmpl
- gtest.wrap
- minizip.wrap.tmpl
- serial.wrap.tmpl
- sqlite3.wrap.tmpl
- sqlitecpp.wrap.tmpl
- zeromq.wrap.tmpl
- zlib.wrap.tmpl
- zyre.wrap.tmpl

Doc Directory

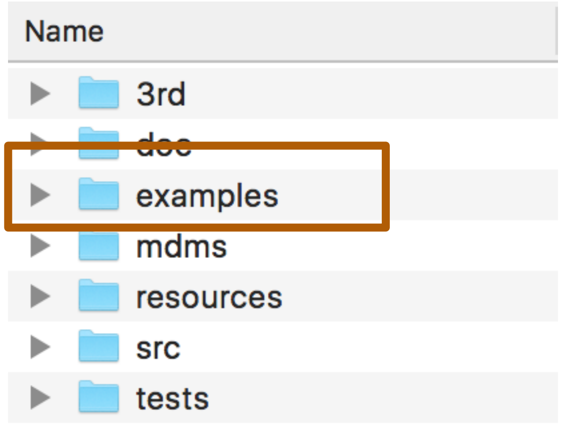


```
Name
▶ 3rd
▶ doc
▶ examples
▶ mdms
▶ resources
▶ src
▶ tests
```

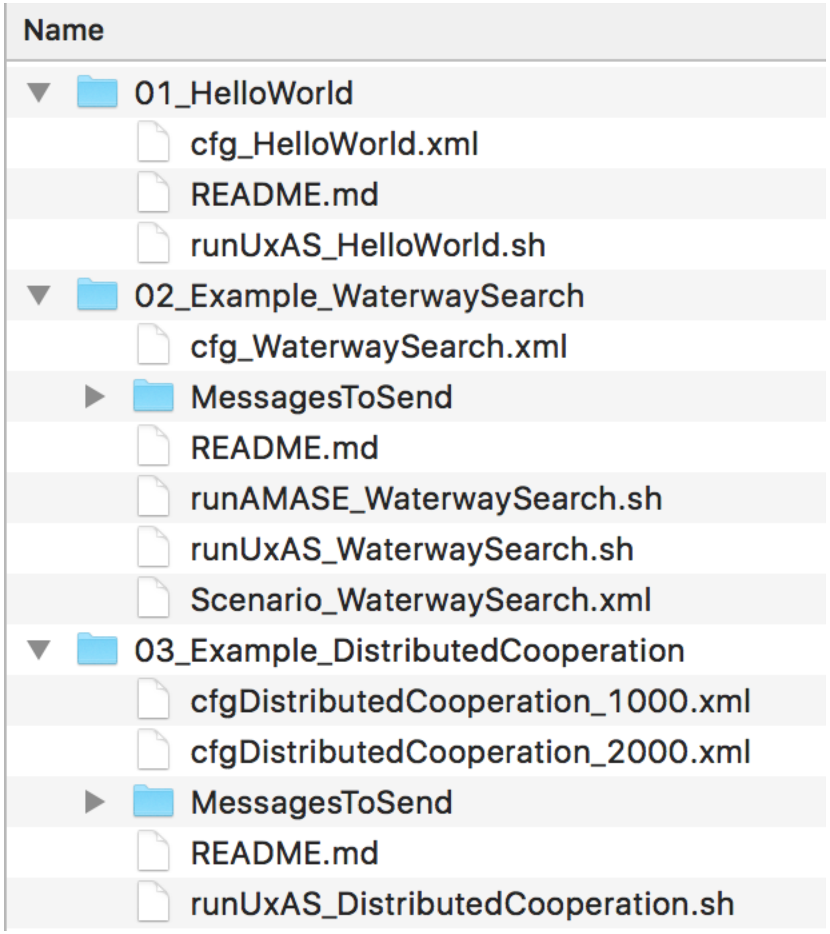


```
▼ doxygen
      Doxyfile
      DoxygenLayout.xml
      ExtraLineToFixLatex.txt
  ▶ files
      RunDoxygen.sh
▼ reference
  ▼ FlowCharts
        BuldFlowCharts.sh
        UxAS_DiscreteEventSimulaton.gv
        UxAS_DiscreteEventSimulaton.png
  ▼ SequenceDiagrams
        CCA_Components_MessageFLow.pdf
        CCA_Components_MessageFLow.sd
  ▼ UserManual
    ▶ Contributing
    ▶ Examples
    ▶ Installation
    ▶ Introduction
    ▶ Navigating
    ▶ Services
    ▶ Testing
        UxAS_UserManual_Supplement.tex
        UxAS_UserManual.tex
```

Examples Directory

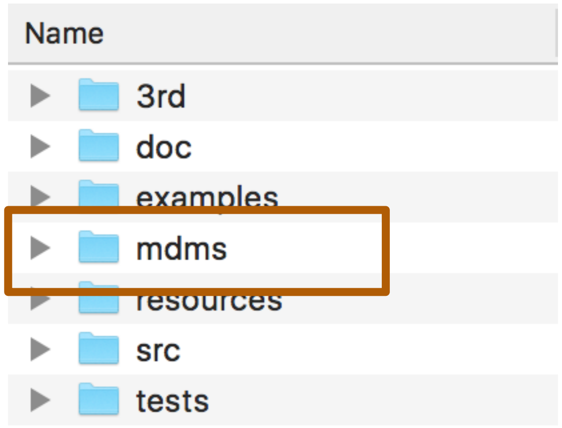


```
Name
▶ 3rd
▶ doc
▶ examples
▶ mdms
▶ resources
▶ src
▶ tests
```



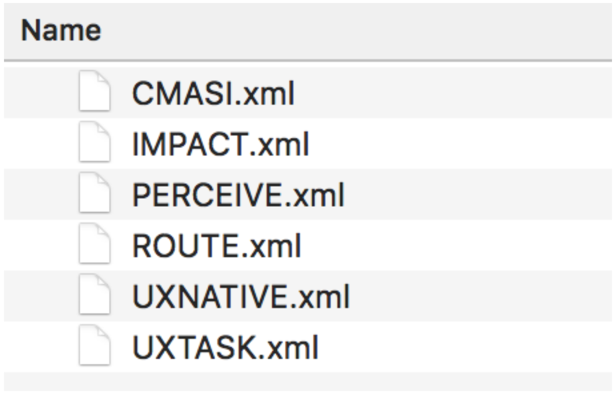
```
Name
▼ 01_HelloWorld
      cfg_HelloWorld.xml
      README.md
      runUxAS_HelloWorld.sh
▼ 02_Example_WaterwaySearch
      cfg_WaterwaySearch.xml
    ▶ MessagesToSend
      README.md
      runAMASE_WaterwaySearch.sh
      runUxAS_WaterwaySearch.sh
      Scenario_WaterwaySearch.xml
▼ 03_Example_DistributedCooperation
      cfgDistributedCooperation_1000.xml
      cfgDistributedCooperation_2000.xml
    ▶ MessagesToSend
      README.md
      runUxAS_DistributedCooperation.sh
```

MDMs Directory



Name

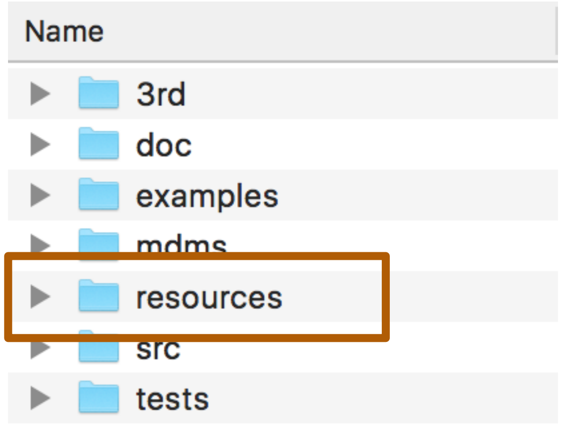
- 3rd
- doc
- examples
- mdms
- resources
- src
- tests



Name

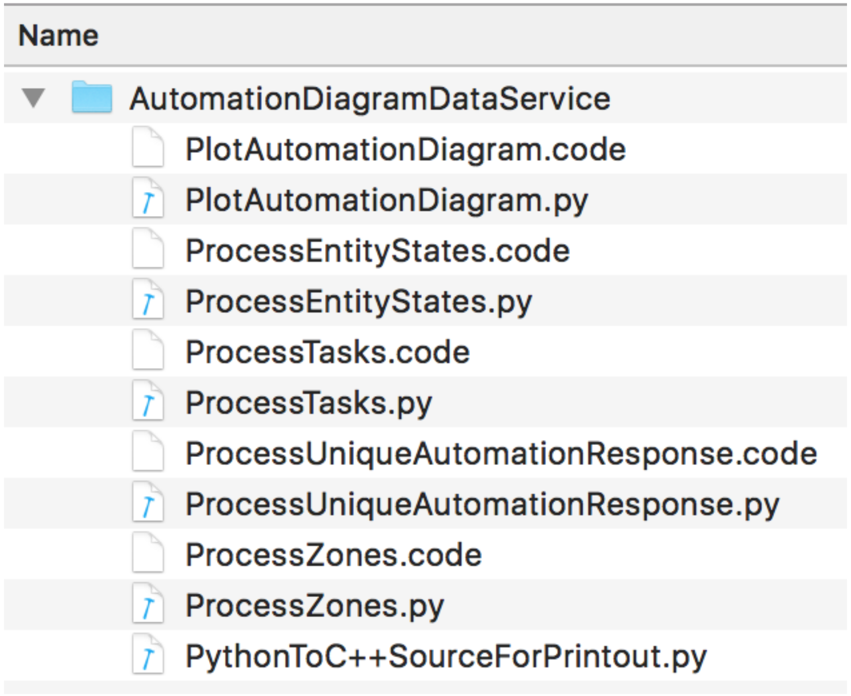
- CMASI.xml
- IMPACT.xml
- PERCEIVE.xml
- ROUTE.xml
- UXNATIVE.xml
- UXTASK.xml

Resources Directory



Name

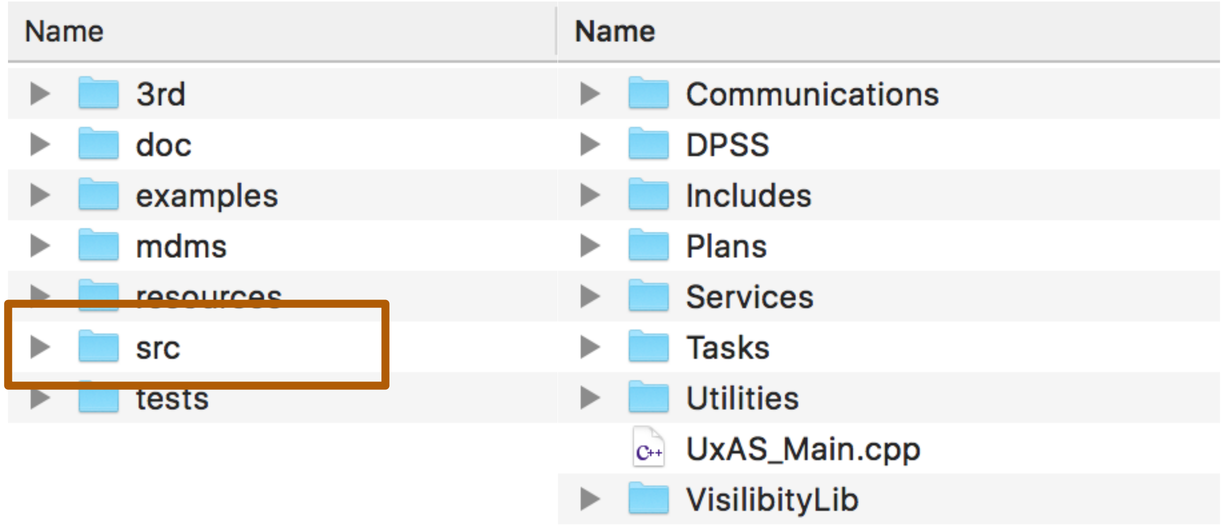
- 3rd
- doc
- examples
- mdms
- resources
- src
- tests



Name

- AutomationDiagramDataService
  - PlotAutomationDiagram.code
  - PlotAutomationDiagram.py
  - ProcessEntityStates.code
  - ProcessEntityStates.py
  - ProcessTasks.code
  - ProcessTasks.py
  - ProcessUniqueAutomationResponse.code
  - ProcessUniqueAutomationResponse.py
  - ProcessZones.code
  - ProcessZones.py
  - PythonToC++SourceForPrintout.py

Source Directory



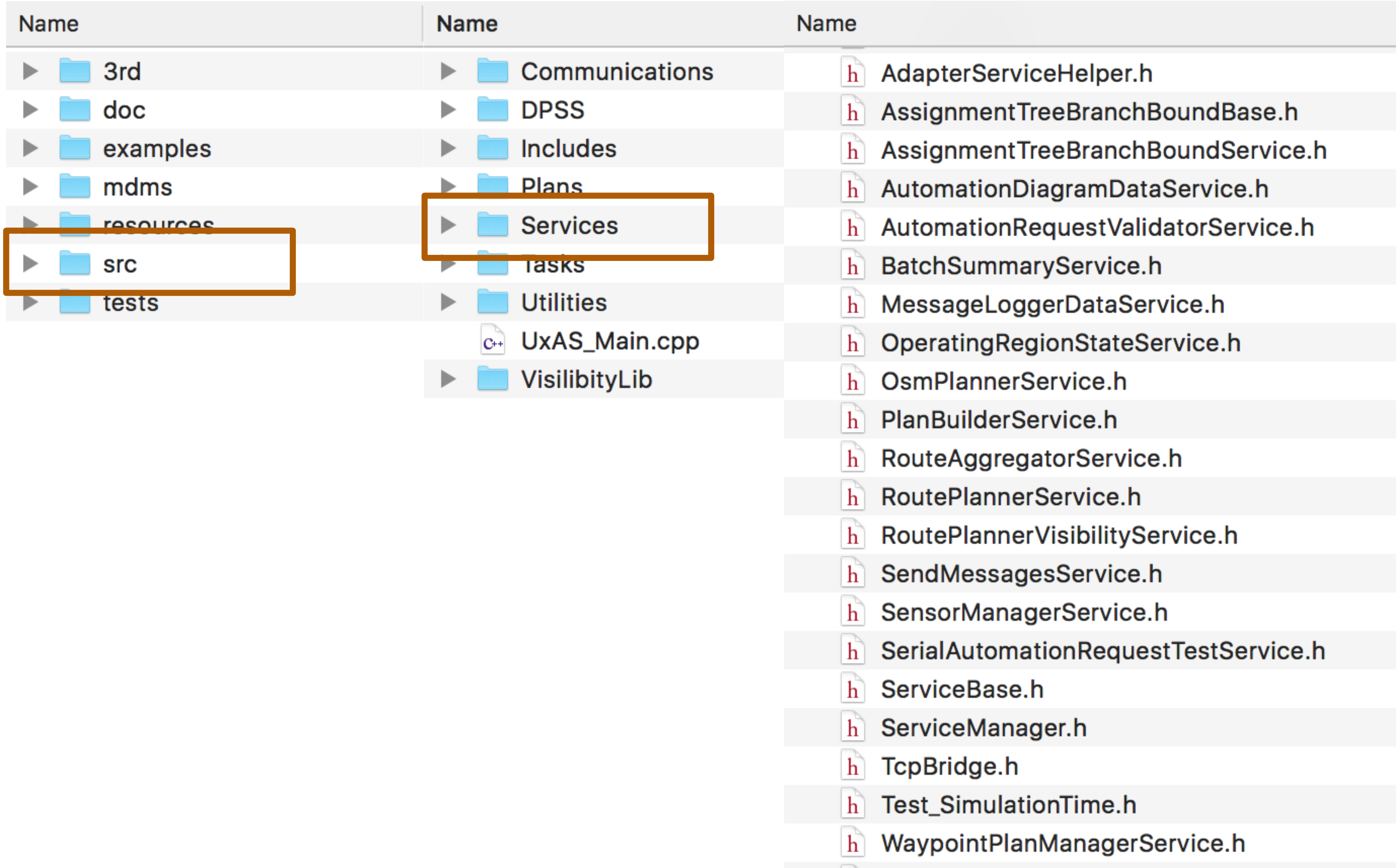
Name

- 3rd
- doc
- examples
- mdms
- resources
- src
- tests

Name

- Communications
- DPSS
- Includes
- Plans
- Services
- Tasks
- Utilities
- UxAS_Main.cpp
- VisilibityLib

Services Directory



Name

- 3rd
- doc
- examples
- mdms
- resources
- src
- tests

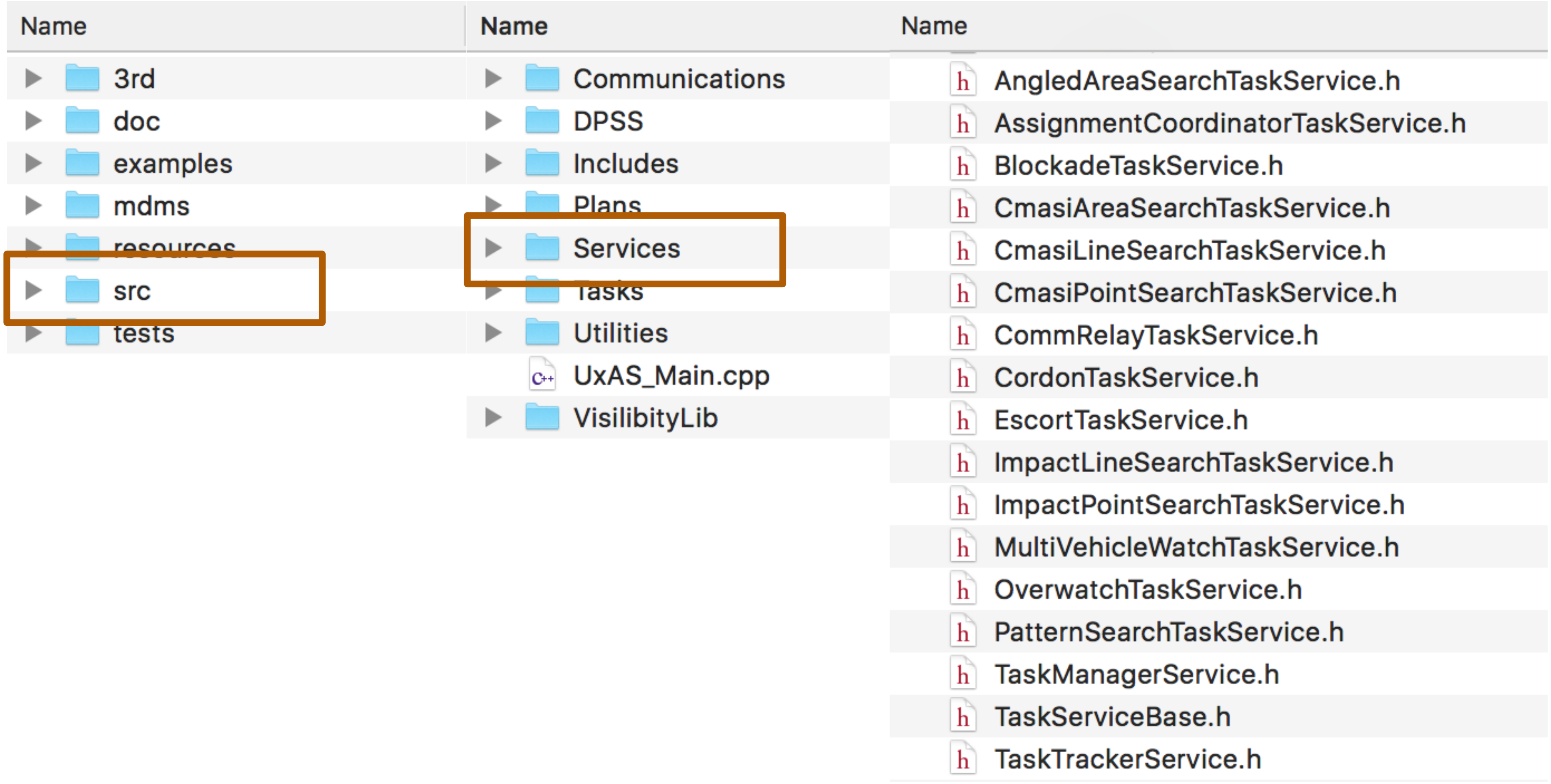
Name

- Communications
- DPSS
- Includes
- Plans
- Services
- Tasks
- Utilities
- UxAS_Main.cpp
- VisilibityLib

Name

- AdapterServiceHelper.h
- AssignmentTreeBranchBoundBase.h
- AssignmentTreeBranchBoundService.h
- AutomationDiagramDataService.h
- AutomationRequestValidatorService.h
- BatchSummaryService.h
- MessageLoggerDataService.h
- OperatingRegionStateService.h
- OsmPlannerService.h
- PlanBuilderService.h
- RouteAggregatorService.h
- RoutePlannerService.h
- RoutePlannerVisibilityService.h
- SendMessagesService.h
- SensorManagerService.h
- SerialAutomationRequestTestService.h
- ServiceBase.h
- ServiceManager.h
- TcpBridge.h
- Test_SimulationTime.h
- WaypointPlanManagerService.h

Tasks Directory



Name

- 3rd
- doc
- examples
- mdms
- resources
- src
- tests

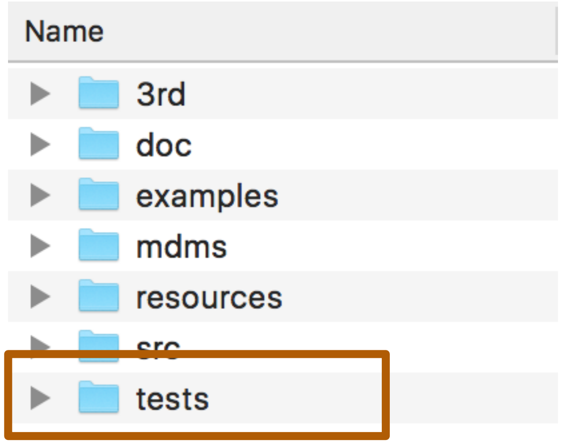
Name

- Communications
- DPSS
- Includes
- Plans
- Services
- Tasks
- Utilities
- UxAS_Main.cpp
- VisilibityLib

Name

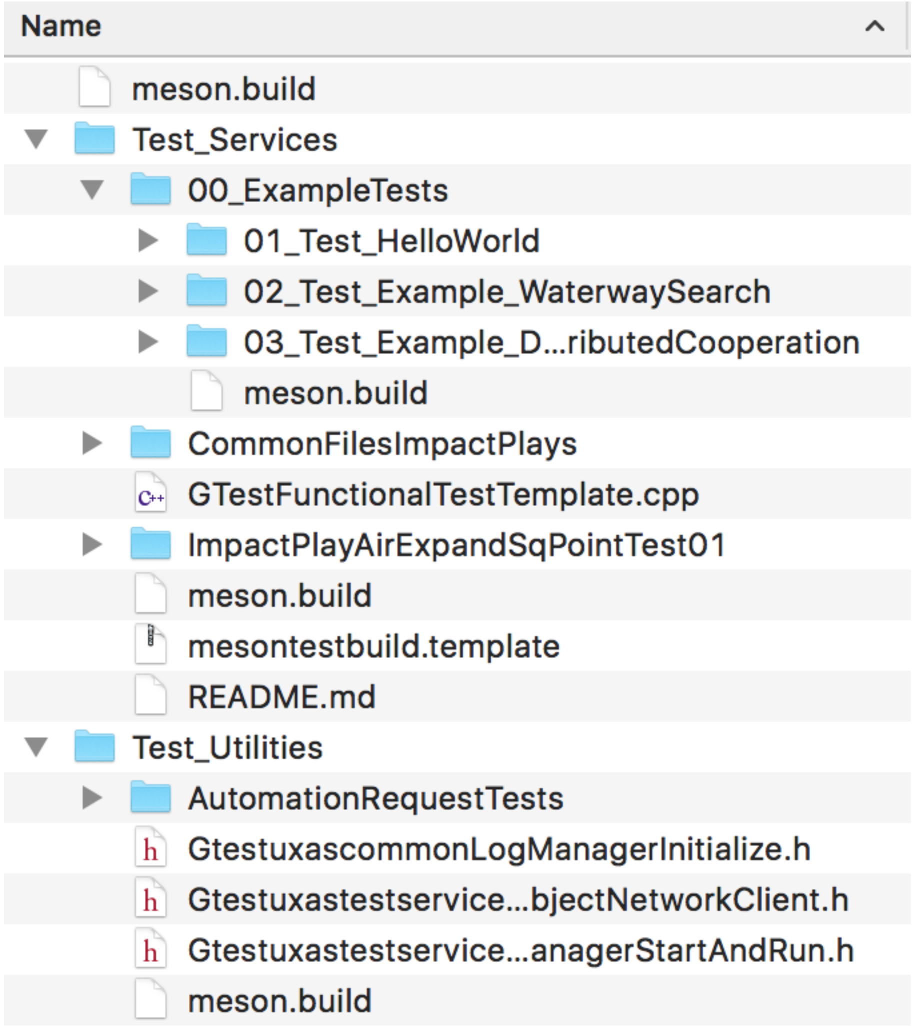
- AngledAreaSearchTaskService.h
- AssignmentCoordinatorTaskService.h
- BlockadeTaskService.h
- CmasiAreaSearchTaskService.h
- CmasiLineSearchTaskService.h
- CmasiPointSearchTaskService.h
- CommRelayTaskService.h
- CordonTaskService.h
- EscortTaskService.h
- ImpactLineSearchTaskService.h
- ImpactPointSearchTaskService.h
- MultiVehicleWatchTaskService.h
- OverwatchTaskService.h
- PatternSearchTaskService.h
- TaskManagerService.h
- TaskServiceBase.h
- TaskTrackerService.h

Tests Directory



Name

- 3rd
- doc
- examples
- mdms
- resources
- src
- tests



Name

- meson.build
- Test_Services
  - 00_ExampleTests
    - 01_Test_HelloWorld
    - 02_Test_Example_WaterwaySearch
    - 03_Test_Example_D...ributedCooperation
    - meson.build
  - CommonFilesImpactPlays
  - GTestFunctionalTestTemplate.cpp
  - ImpactPlayAirExpandSqPointTest01
  - meson.build
  - mesontestbuild.template
  - README.md
- Test_Utilities
  - AutomationRequestTests
  - GtestuxascommonLogManagerInitialize.h
  - Gtestuxastestservice...bjectNetworkClient.h
  - Gtestuxastestservice...anagerStartAndRun.h
  - meson.build

UxAS Repositories

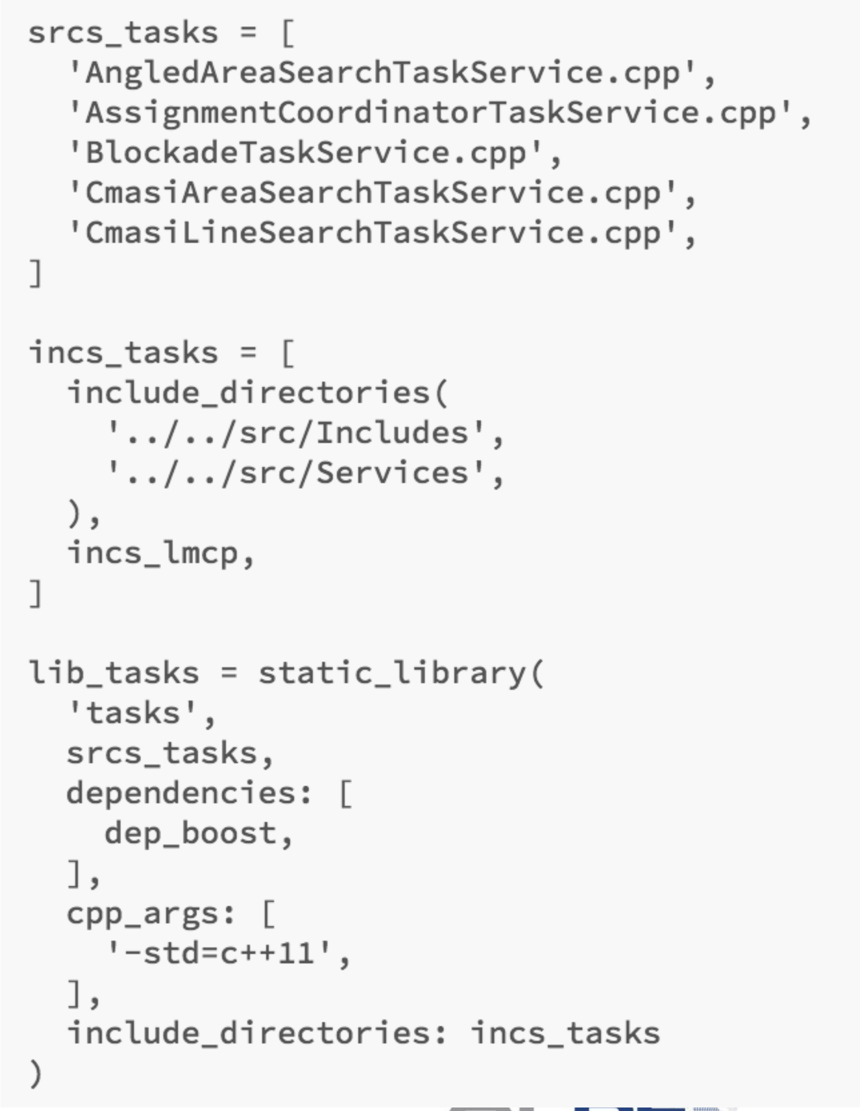
- UxAS uses Git for source code management

- OpenUxAS: github.com/afrl-rq/OpenUxAS.git
- LmcpGen: github.com/afrl-rq/LmcpGen.git
- OpenAMASE: github.com/afrl-rq/OpenAMASE.git
- Closed source additions to UxAS: repos.vdl.afrl.af.mil/gitlab/tcas-uxas/UxAS-afrl_internal.git

- Use ‘git clone <repo>’ to checkout source code

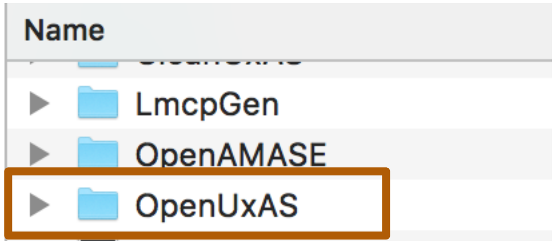
Building UxAS

- Uses the Meson build system
  - Multi-platform build system
  - Meta-compile: builds framework and input files to actual compile system
  - Creates ninja or visual studio build projects
  - ‘meson.build’ files organized hierarchically
  - Primary ‘meson.build’ in root OpenUxAS directory
- Ninja compile is attempt to use g++ in highly parallelizable manner

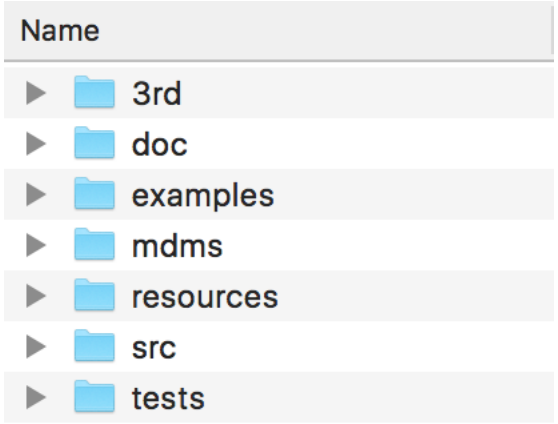


```
srcs_tasks = [
  'AngledAreaSearchTaskService.cpp',
  'AssignmentCoordinatorTaskService.cpp',
  'BlockadeTaskService.cpp',
  'CmasiAreaSearchTaskService.cpp',
  'CmasiLineSearchTaskService.cpp',
]

incs_tasks = [
  include_directories(
    '../../src/Includes',
    '../../src/Services',
  ),
  incs_lmcp,
]

lib_tasks = static_library(
  'tasks',
  srcs_tasks,
  dependencies: [
    dep_boost,
  ],
  cpp_args: [
    '-std=c++11',
  ],
  include_directories: incs_tasks
)
```

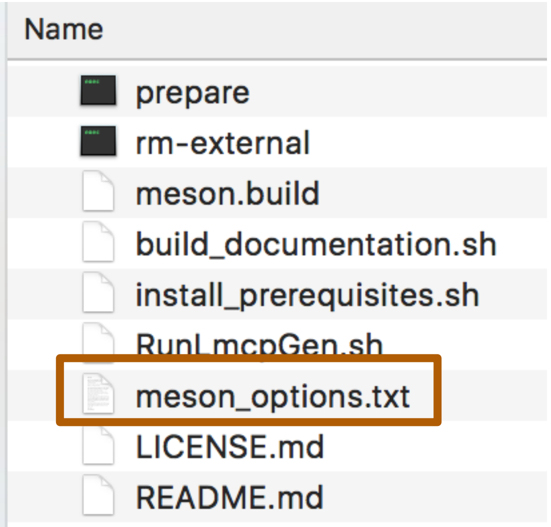
Meson Options


Name

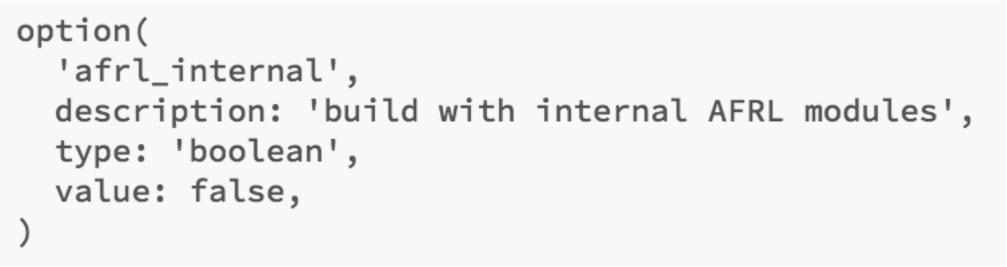
- LmcpGen
- OpenAMASE
- OpenUxAS


Name

- 3rd
- doc
- examples
- mdms
- resources
- src
- tests


Name

- prepare
- rm-external
- meson.build
- build_documentation.sh
- install_prerequisites.sh
- RunLmcpGen.sh
- meson_options.txt
- LICENSE.md
- README.md



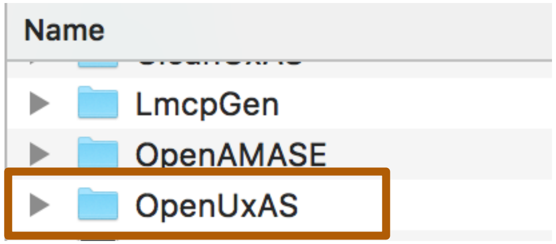
```
option(
  'afrl_internal',
  description: 'build with internal AFRL modules',
  type: 'boolean',
  value: false,
)
```

Building UxAS

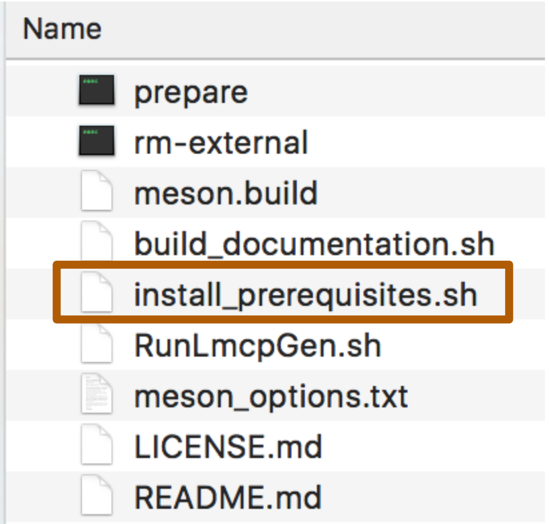
1. Pre-requisites
2. Configure
3. Build

Building UxAS

1. Pre-requisites
   - Libraries, tools for build
   - ‘install_prerequisites.sh’ or follow detailed step-by-step instructions
     - Mostly system libraries needed for meson and some 3rd party libraries (e.g. uuid)
2. Configure
3. Build

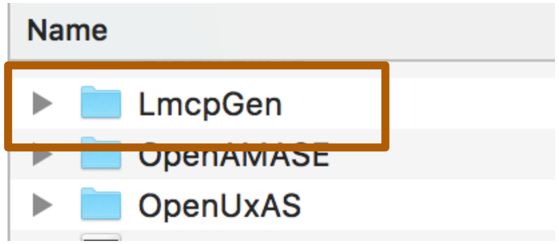


Name
- LmcpGen
- OpenAMASE
- OpenUxAS



Name
- prepare
- rm-external
- meson.build
- build_documentation.sh
- install_prerequisites.sh
- RunLmcpGen.sh
- meson_options.txt
- LICENSE.md
- README.md

Building UxAS



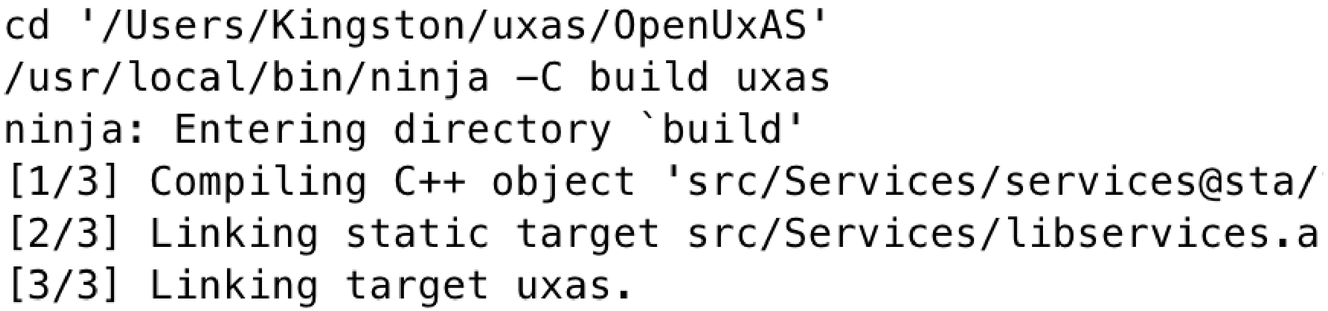
Name

- LmcpGen
- OpenAMASE
- OpenUxAS

1. Pre-requisites
2. Configure
   - Setup 3rd party dependency download './prepare'
   - Auto-generate library 'RunLmcpGen.sh'
     - Requires LmcpGen at same file level as OpenUxAS
   - Setup complete build:
     'meson build --buildtype=release' or
     'meson build –Dafrl_internal=true'
3. Build

Building UxAS

1. **Pre-requisites**
2. **Configure**
3. **Build**
   - **Start build: ‘ninja –C build’**



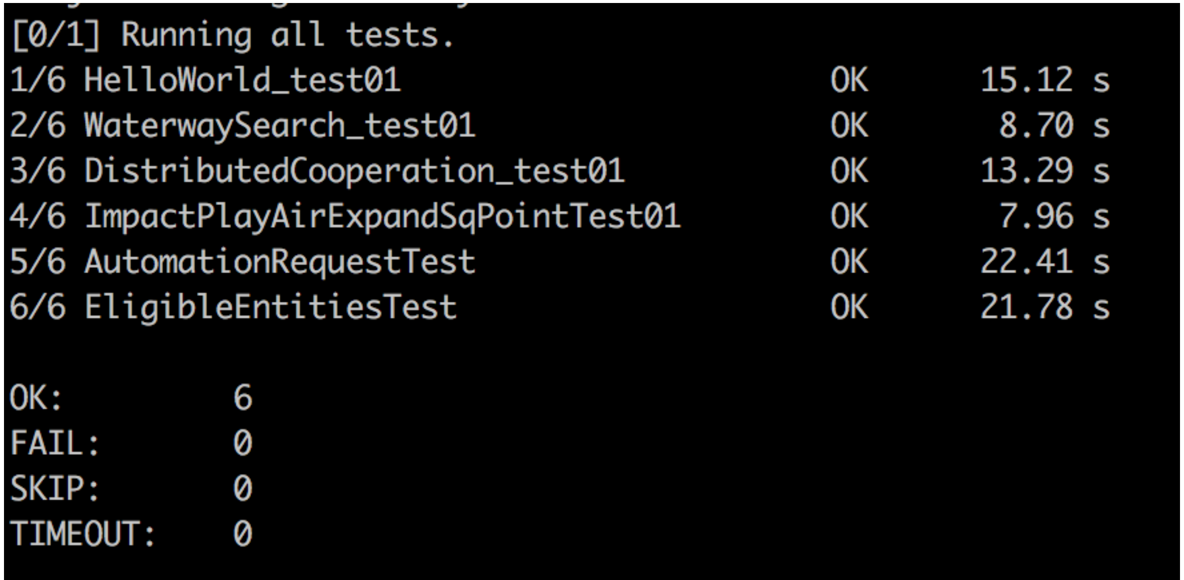
```
cd '/Users/Kingston/uxas/OpenUxAS'
/usr/local/bin/ninja -C build uxas
ninja: Entering directory `build'
[1/3] Compiling C++ object 'src/Services/services@sta/
[2/3] Linking static target src/Services/libservices.a
[3/3] Linking target uxas.
```

Re-Building UxAS

- Changes in MDMs need to re-run ‘RunLmcpGen.sh’
- Changes in 3rd party libraries require reconfiguring with ‘meson’
- Source changes will build directly with ninja without re-configuration

Automated Testing

- Uses Google Test for reporting pass/fail
- The ‘meson.build’ file in the tests directory is the top of the test hierarchy
  - Each sub-folder can contain additional tests
- Tests inject messages and then check the log file to determine proper response message reception
- Run all tests with: ‘ninja –C build test’



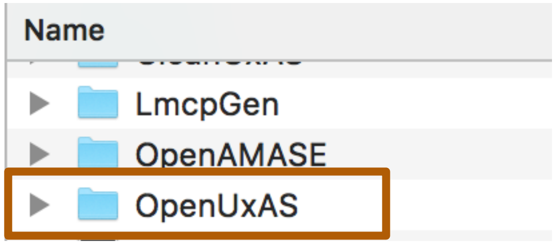
```
[0/1] Running all tests.
1/6 HelloWorld_test01                      OK       15.12 s
2/6 WaterwaySearch_test01                  OK        8.70 s
3/6 DistributedCooperation_test01          OK       13.29 s
4/6 ImpactPlayAirExpandSqPointTest01       OK        7.96 s
5/6 AutomationRequestTest                  OK       22.41 s
6/6 EligibleEntitiesTest                   OK       21.78 s

OK:          6
FAIL:        0
SKIP:        0
TIMEOUT:     0
```

Automated Testing

- **Due to limitations in Google Test, no tests in the same file may collectively run longer than 30 seconds**
- **OpenUxAS is connected to Travis CI for automated tests on source code check in**
- **All tests configured to run via ‘ninja –C build test’ will be run on the server**
- **Complete Travis setup script is found in: OpenUxAS/.travis.yml**

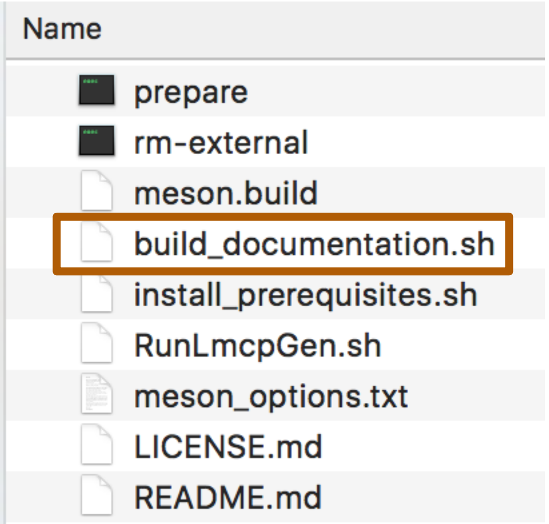
Documentation

- Three major parts to documentation
  1. User Manual
  2. LMCP messages, auto-created at: OpenUxAS/doc/LMCP/index.html
  3. Doxygen class descriptions
- User Manual and Doxygen documentation can be created with: ‘build_documentation.sh’
- Relies on LaTeX
- Continuing work in progress



Name

- LmcpGen
- OpenAMASE
- OpenUxAS



Name

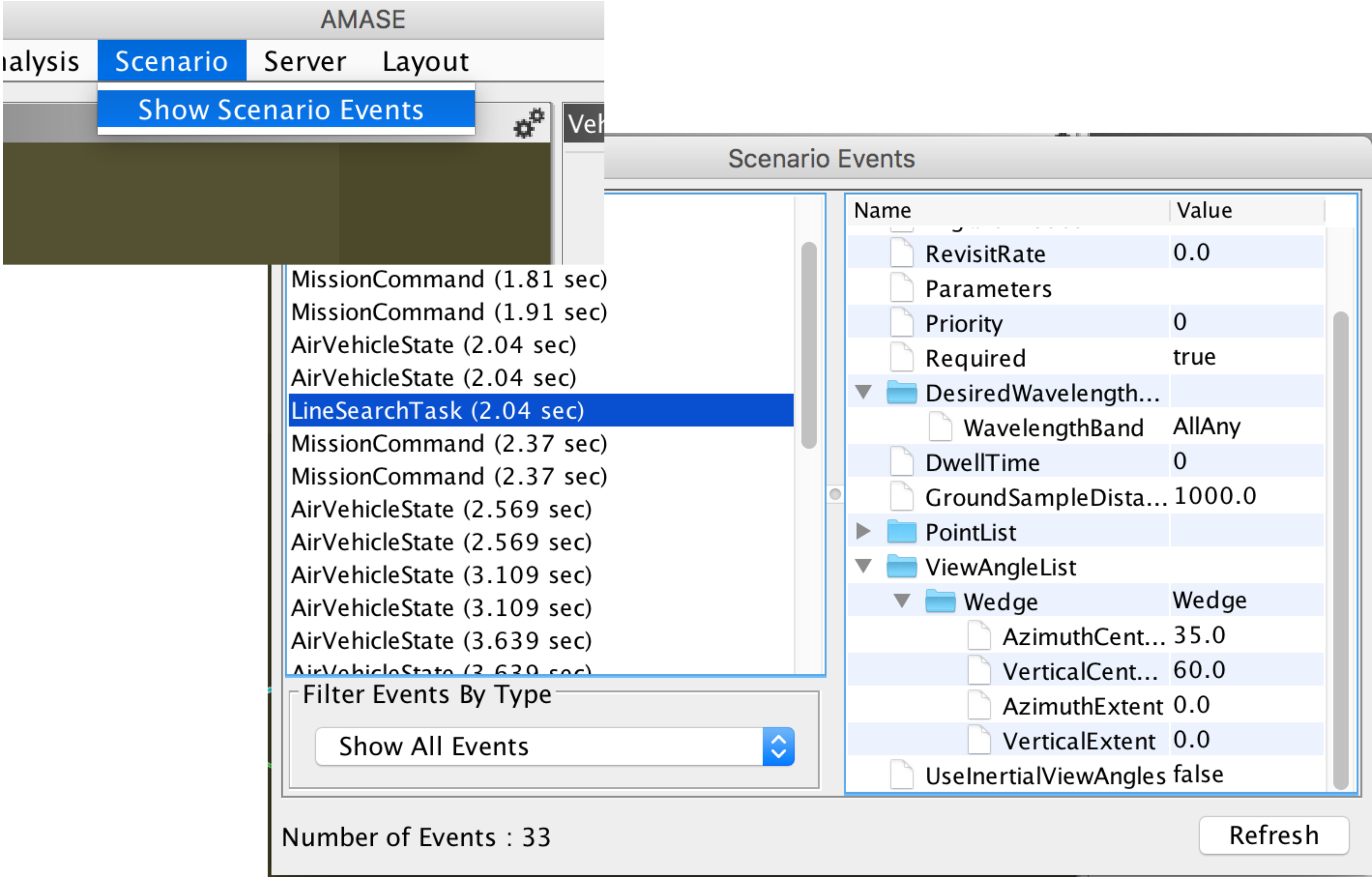
- prepare
- rm-external
- meson.build
- build_documentation.sh
- install_prerequisites.sh
- RunLmcpGen.sh
- meson_options.txt
- LICENSE.md
- README.md

Debugging

- UxAS is a collection of services each running in its own thread, so serial step-through can be challenging
- Gdb (and IDEs that use gdb) can facilitate breakpoints, tracebacks, and step-throughs

- Due to multiple threads as well as multiple instances of UxAS running for collaborative behaviors, traditional debugging (gdb) is of limited value
- Key tools:
  - MessageLogger service places all messages in a database for searching
  - AMASE provides log of messages received

AMASE Message Viewer



MissionCommand (1.81 sec)
MissionCommand (1.91 sec)
AirVehicleState (2.04 sec)
AirVehicleState (2.04 sec)
LineSearchTask (2.04 sec)
MissionCommand (2.37 sec)
MissionCommand (2.37 sec)
AirVehicleState (2.569 sec)
AirVehicleState (2.569 sec)
AirVehicleState (3.109 sec)
AirVehicleState (3.109 sec)
AirVehicleState (3.639 sec)
AirVehicleState (3.639 sec)

Name	Value
RevisitRate	0.0
Parameters	
Priority	0
Required	true
DesiredWavelength...	
WavelengthBand	AllAny
DwellTime	0
GroundSampleDista...	1000.0
PointList	
ViewAngleList	
Wedge	Wedge
AzimuthCent...	35.0
VerticalCent...	60.0
AzimuthExtent	0.0
VerticalExtent	0.0
UseInertialViewAngles	false

Number of Events : 33

Questions?