

# APP4RTA

For Analyzing Response Time & End-to-End Event-Chain Latency

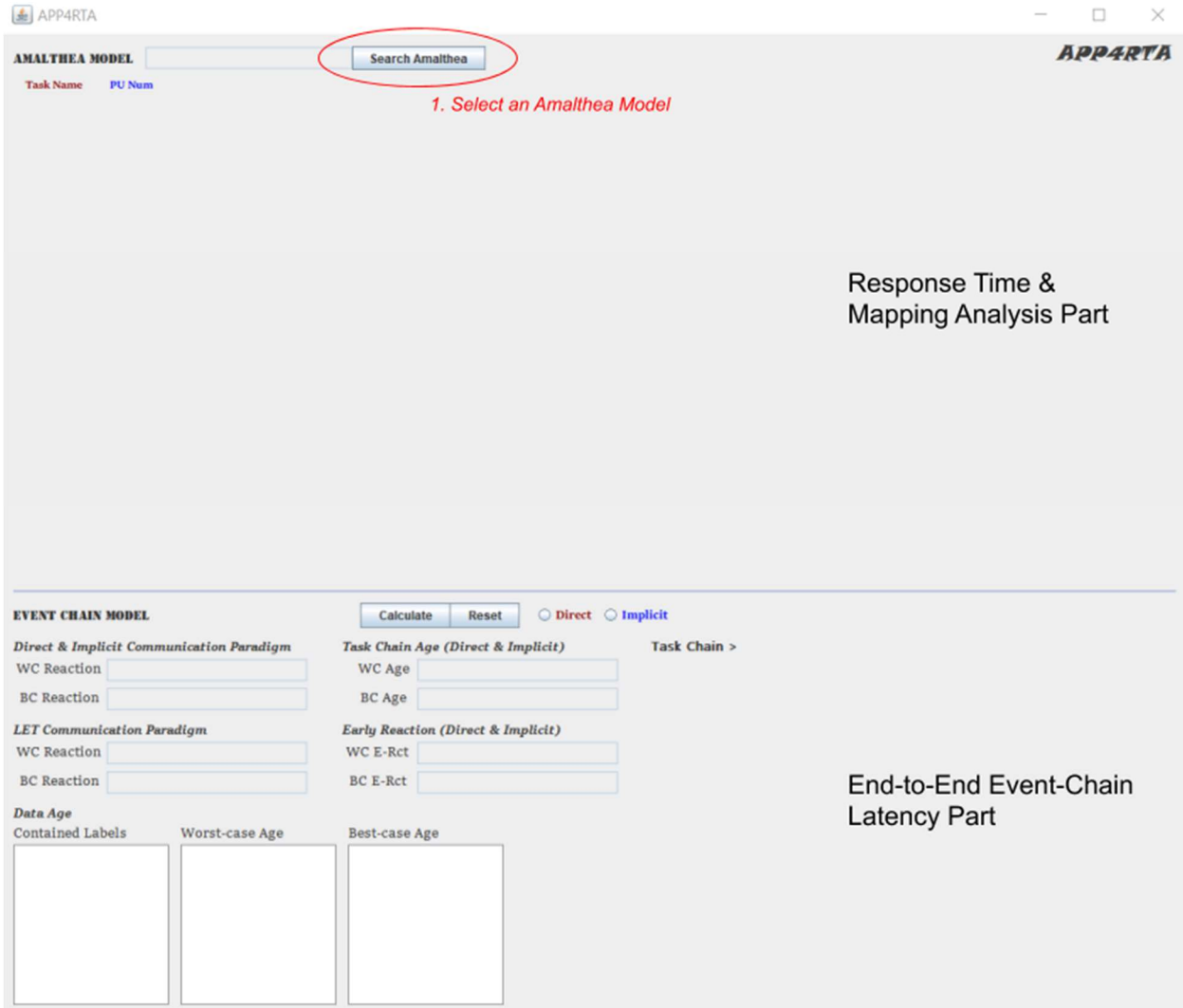
The screenshot displays the APP4RTA software interface, which is used for analyzing response time and end-to-end event-chain latency. The interface is divided into several sections:

- AMALTHEA MODEL:** This section allows users to configure tasks and their parameters. The task name is "ChallengeModel\_TCs.amxml". Tasks listed include OS\_Overhead, Lidar\_Grabber, DASM, CANbus\_polling, EKF, Planner, PRE\_SFM\_gpu..., PRE\_Localizati..., PRE\_Lane\_det..., PRE\_Detection..., SFM, Localization, Lane\_detection, and Detection. Each task has a PU Num (Priority/Utilization) and a Response Time. The interface also includes buttons for "Default IA", "Enter IA", "Calculate", and "Reset".
- Summary Metrics:** This section provides a summary of the model's performance, including:
  - Cumulated Memory-Access Cost: 5361668000 ps
  - Cumulated Contention: 24795710000 ps
  - Computation: 6350750505000 ps
  - Response Time Sum: 665232428500 ps
- EVENT CHAIN MODEL:** This section shows the event chain configuration, including:
  - Model Name: CA-EK-P-DA
  - Buttons: Calculate, Reset
  - Paradigm:  Direct  Implicit
  - Direct & Implicit Communication Paradigm: WC Reaction (81302942000 ps), BC Reaction (16086298500 ps)
  - LET Communication Paradigm: WC Reaction (80000000000 ps), BC Reaction (45000000000 ps)
  - Data Age: Contained Labels (Occupancy\_grid\_host, Vehicle\_status\_host, x\_car\_host, y\_car\_host, yaw\_car\_host, vel\_car, yaw\_rate, steer\_objective, ...), Worst-case Age, and Best-case Age.
  - Task Chain: 1: Core3 (A57) [CANbus\_polling], 2: Core4 (A57) [EKF], 3: Core0 (Denver) [Planner], 4: Core1 (Denver) [DASM]

Before executing the code, please install the Java GUI software.

➤ To install Java GUI softwares:

1. Eclipse > `Help`
2. `Install New Software` > Work with: Eclipse Repository (<http://download.eclipse.org/releases/oxygen>)
3. `General Purpose Tools` > all click from `Swing Designer` to `WindowBuilder XML Core (requires Eclipse WTP/WST)`
4. `Next` > `Next` > `accept` > `Finish`

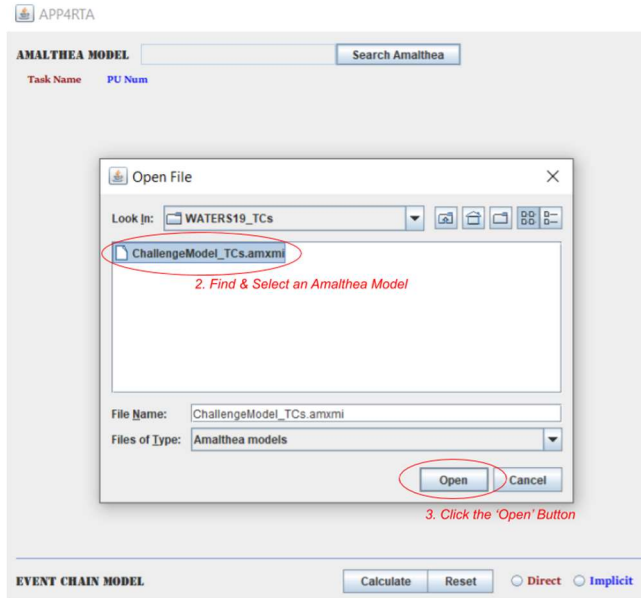


Run `APP4RTA.java` in `org.eclipse.app4mc.gsoc\_rta.ui` package, then this window will show up.

Based on the horizontal line on the middle, the upper part is for response time & mapping analysis, and the lower part is for end-to-end event-chain latency analysis.

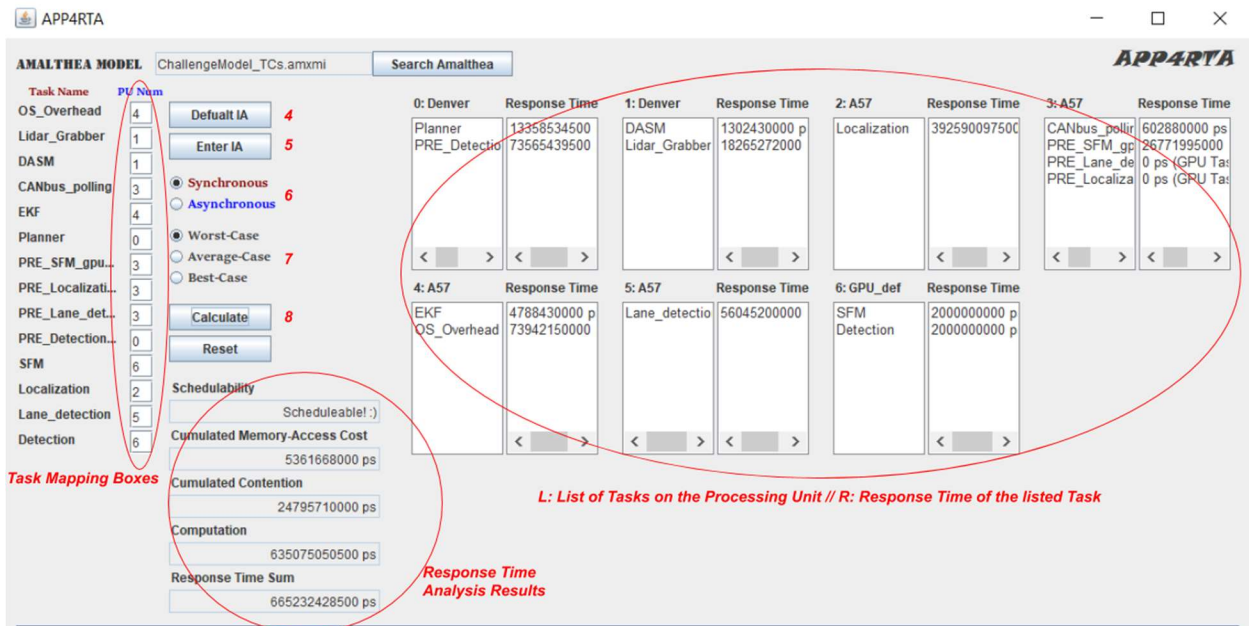
The first thing to do is deciding a target Amalthea model.

1. The window browser for searching Amalthea models shows up when the `Search Amalthea` button clicked.



2. When the search browser shows up, direct to the path where the target Amalthea model file is located and select the model file.

3. Click the `Open` button.



Then the empty space will be filled with the the tasks and processing units of the selected model.

On the left-hand side, tasks' names with empty boxes can be found.

On the right-hand side, seven pairs of lists are seen (It means the selected model has seven processing units).

The list on the left side of each pair is for listing names of the tasks which are mapped to the corresponding processing unit while one on the right side is for listing response times of the corresponding tasks.

Basically, we can map the tasks with these boxes by entering the number of each processing unit which is stated on the top of the lists on the left-side.

4. The user can either manually type numbers for every box or simply click the `Default IA` button which would automatically fill up every box with the pre-defined integer array values.
5. Once every `PU Num` box is filled, click `Enter IA` button to assign tasks to processing units according to each integer value. Once this is done, the mapped tasks would appear on the left-side lists.
6. Choose the offloading mode between `Synchronous` case and `Asynchronous` case.
7. Choose the execution case between `Worst` case and `Average` case and `Best` case.
8. By clicking the `Calculate` button, all calculation results will be printed out on the text-fields (`Schedulability`, `Cumulated Memory-Access Cost`, `Cumulated Contention`, `Computation`).

The screenshot displays the APP4RTA software interface. At the top, the window title is 'APP4RTA'. Below it, the 'AMALTHEA MODEL' section is active, showing a file named 'ChallengeModel\_TCs.amxmi'. The interface is divided into several sections:

- Task Mapping:** A table with columns for 'Task Name', 'PU Num', and 'Response Time'. Tasks listed include OS\_Overhead, Lidar\_Grabber, DASM, CANbus\_polling, EKF, Planner, PRE\_SFM\_gpu..., PRE\_Localizati..., PRE\_Lane\_deL..., PRE\_Detection..., SFM, Localization, Lane\_detection, and Detection. Each task has a corresponding 'PU Num' value and a 'Response Time' value.
- Configuration:** Radio buttons for 'Synchronous' (selected) and 'Asynchronous', and radio buttons for 'Worst-Case' (selected), 'Average-Case', and 'Best-Case'. There are also 'Calculate' and 'Reset' buttons.
- Results:** A section showing calculated values: 'Schedulability' (Scheduleable!), 'Cumulated Memory-Access Cost' (5361668000 ps), 'Cumulated Contention' (24795710000 ps), 'Computation' (635075050500 ps), and 'Response Time Sum' (665232488500 ps).
- Event Chain Model:** A section with a dropdown menu for 'EVENT CHAIN MODEL' (circled in red) and buttons for 'Calculate' and 'Reset'. It includes fields for 'Task Chain Age (Direct & Implicit)', 'WC Age', 'BC Age', and 'BC E-Rct'.
- Data Age:** A section with three columns: 'Contained Labels', 'Worst-case Age', and 'Best-case Age', each with a corresponding empty text box.

The event-chain combo-box becomes visible once the user clicks `Enter IA` to assign tasks to processing units according to each integer value in the boxes.

9. To analyze end-to-end event-chain latency, an event-chain in the combo-box should be selected first.

**EVENT CHAIN MODEL** CA-EK-P-DA    Direct  Implicit **10. Select a Communication Paradigm**

**11. Click the 'Calculate' Button.**

**Direct & Implicit Communication Paradigm**

WC Reaction  BC Reaction

**LET Communication Paradigm**

WC Reaction  BC Reaction

**Task Chain Age (Direct & Implicit)**

WC Age  BC Age

**Early Reaction (Direct & Implicit)**

WC E-Rct  BC E-Rct

**Data Age**

Contained Labels	Worst-case Age	Best-case Age
Occupancy_grid_host	17900000000 ps	12100000000 ps
Vehicle_status_host	10200000000 ps	9800000000 ps
x_car_host	15780000000 ps	12100000000 ps
y_car_host	15780000000 ps	12100000000 ps
yaw_car_host	15780000000 ps	12100000000 ps
vel_car	15780000000 ps	12100000000 ps
yaw_rate	15780000000 ps	12100000000 ps
steer_objective	5250000000 ps	4750000000 ps
steer_objective	5250000000 ps	4750000000 ps
steer_objective	17900000000 ps	12100000000 ps

**Task Chain >**

- 1: Core3 (A57)  
CANbus\_polling
- 2: Core4 (A57)  
EKF
- 3: Core0 (Denver)  
Planner
- 4: Core1 (Denver)  
DASM

10. Select the communication paradigm between direct Communication and implicit communication.

11. Finally, click the `Calculate` button.

Then all calculation results regarding reaction, age of data, task-chain in the worst and best cases will be printed out to the corresponding text fields or lists.