

# Chronometer



## Description

Chronometer is a bundle of two apps:

- **Start Chrono**  
Starts chronometer(s) for the input job.
- **Stop Chrono**  
Evaluates and stores the elapsed time for each chronometer from the input job.

## Compatibility

Switch 2020 spring (Node.js).

## Connections

Both elements require at least one incoming connection.

Start Chrono requires at least one outgoing connection.

Stop Chrono requires a TrafficLight connection.

## Use cases

It can be used to benchmark velocity between softwares or flow designs.

It is often useful for development / pre-production.

## Start Chrono properties

- **Chronometer(s)**

List of all the chronometers you wish to start.  
One per line.

Example:

```
firstChrono
secondChrono
...
```



## Stop Chrono properties

- **Chronometer(s)**

List of all the chronometers you wish to get value from.  
One per line.

If at least one of the specified chronometers has an invalid value, the input job will be sent to error connection(s).

Example:

```
firstChrono
secondChrono
...
```



- **Unit**

Choose the time unit in which you want to format the result.

Possible values:

- Millisecond
- Second
- Minute
- Hour

## Result

The result will be stored in private data keys as:

<b>&lt;ChronometerName&gt;</b>	Formatted elapsed time
<b>&lt;ChronometerName&gt;.unit</b>	Formatted time unit
<b>&lt;ChronometerName&gt;.startingDate</b>	Time elapsed since 01/01/1970 in millisecond <i>(used for calculation)</i>
<b>&lt;ChronometerName&gt;.finishDate</b>	

## Example flow



**Properties**

Property	Value
Element type	Start Chrono
Name	Start Chrono
Description	
Chronometer(s)	IdChrono MyNthChrono >

**Edit**

Enter a description...

MyFirstChrono  
MySecondChrono  
MyNthChrono

**Properties**

Property	Value
Element type	Stop Chrono
Name	Stop Chrono
Description	
Chronometer(s)	IdChrono MyNthChrono >
Unit	Second

**Edit**

Enter a description...

MyFirstChrono  
MySecondChrono  
MyNthChrono