

▼ 1 Properties



This is the reference guide to most properties regarding the WorkflowSim, and their respective default values. Please refer to the user guide for a discussion when and which properties to use to configure various components. Please note that the values rely on proper capitalization, unless explicitly noted otherwise.

Some properties rely with their default on the value of other properties. As a notation, the curly braces refer to the value of the named property. For instance, `${workflowsim.home}` means that the value is the root path to where you store WorkflowSim.

Commandline properties have the highest priority. These override any property loaded from a property file. Commandline properties are useful for debugging purposes.

The following example provides a minimum set of properties to be set by the user property file:

```
vmfile.system = LOCAL
num = 20
bandwidth = 1e6
dax.path=${workflowsim.home}/config/balanced/dax/cybershake.xml
```



▼ 1.1 dax.path

Category:	System
Type:	string
Default:	NULL
Required:	YES



The property `dax.path` must be set in the property file. This property is the physical path to the DAX file.

▼ 1.2 vm.num

Category:	System
Type:	int
Default:	0
Required:	YES



The property `vm.num` must be set in the property file. This property is the number of vms during the simulation. However, it should not exceed the maximum allowed in the datacenter. In the WorkflowSimExample1, a CondorVM requires 1000MIPS and 512MB RAM. A datacenter has 28 hosts and each host has 2048MB RAM and 2000 MIPS. Therefore, in this example, the maximum of `vm.num` is 56. If you wish to set `vm.num` be greater than 56, increase the resource (MIPS, RAM, Size, etc.) a datacenter has or decrease the resource a vm requests.



▼ 1.3 bandwidth



Category:	System
Type:	long
Deault:	
Require:	YES

The property bandwidth specifies the bandwidth between the sites/hosts in a data center. This function is not fully completed and thus if your research is related to data transfer time prediction. Please take care.



▼ 1.4 file.system



Category:	System
Type:	enum
Value[0]:	local
Value[1]:	shared
Default:	
Required:	Yes

The property file.system selects what file system to use. The shared filesystem has a central storage and the data transfer etween hosts is ignored. In the local filesystem, every host has a local storage and the data transfer time between hosts is not ignored.



▼ 1.5 Task Clustering Related Parameters (Optional)

▼ 1.5.1 clusters.method




Category:	Clustering
Type:	enum
Value[0]:	HORIZONTAL
Value[1]:	VERTICAL
Value[2]:	BLOCK
Value[3]:	NONE
BALANCED	
Default:	NONE
Required:	NO

The property clusters.method specifies the clustering method.

▼ 1.5.2 clusters.num




Category:	Clustering
Type:	int
Default:	
Required:	NO

The property clusters.num is the number of clustered jobs per level in horizontal task clustering.

▼ 1.5.3 clusters.size



Category:	Clustering
Type:	int
Default:	
Required:	NO

The property clusters.num is the number of tasks per jobs in horizontal task clustering.



▼ 1.6 Overhead Related Parameters (Optional)



Overheads are specified per level with a character indicating the type of overheads and a number indicating the level.

Workflow Engine Delay	d
Queue Delay:	q
Postscript Delay	p
Clustering Delay	c



For example, to specify the workflow engine delay for the first level jobs (root jobs): (the unit is second)

```
d1=45.67
```

To specify the postscript delay for the fifth level jobs:

```
p5=45.67
```



To specify all other levels that are not specified particularly, use 0-level

```
d0=45.67
```

If no other levels have been specified, all levels have the same workflow engine level.

▼ 1.6.1 interval



Category:	Overhead
Type:	int
Default:	infinite

Required:	NO
-----------	----

For example, below we set the interval to 5.

```
interval=5
```

The property interval is the maximum number of jobs workflow engine will release at a period. The larger it is, the workflow engine will release more jobs at a time. If it is not specified (by default), the workflow engine will release all free jobs together.

▼ 1.7 Failure Related Parameters (Optional)

Similar to how we specify overhead, failure rates are also set per level.

```

a
b

```

Task Failure Rate	a
Job Failure Rate (not supported now)	b

For example, below we set the task failure rate of the second level to be 2%, which means 2% of the tasks at this level will be selected randomly to fail.

```
a2=0.02
```

```

a
b

```

▼ 1.7.1 ftc.method

```

a
b

```

Category:	Failure
Type:	enum
Value[0]:	FTCLUSTERING_DC
Value[1]:	FTCLUSTERING_SR
Value[2]:	FTCLUSTERING_DR
Value[3]:	FTCLUSTERING_NOOP
Value[4]:	FTCLUSTERING_BLOCK
Value[5]:	FTCLUSTERING_BINARY
Default:	FTCLUSTERING_NOOP
Required:	NO

The property ftc.method specifies the fault tolerant clustering method to use. Please refer to our fault tolerant clustering paper for details of their implementation.

```

a
b

```

▼ 1.7.2 ftc.monitor

```

a
b

```

Category:	Failure
Type:	enum
Value[0]:	MONITOR_NONE
Value[1]:	MONITOR_ALL

Value[2]:	MONITOR_VM
Value[3]:	MONITOR_JOB
Default:	MONITOR_NONE
Required:	NO



The property `ftc.monitor` specifies the fault tolerant clustering monitor to use. Please refer to our fault tolerant clustering paper for details of their implementation.

▼ 1.7.3 `ftc.failure`



Category:	Failure
Type:	enum
Value[0]:	FAILURE_NONE
Value[1]:	FAILURE_ALL
Value[2]:	FAILURE_VM
Value[3]:	FAILURE_JOB
Default:	FAILURE_NONE
Required:	NO



The property `ftc.monitor` specifies the failure generation mode to use. Please refer to our fault tolerant clustering paper for details of their implementation.

▼ 1.8 Scheduling (Optional)



▼ 1.8.1 `scheduler.method`



Category:	Scheduling
Type:	enum
Value[0]:	MAXMIN_SCH
Value[1]:	MINMIN_SCH
Value[2]:	ROUNDRO_SCH
Value[3]:	HEFT_SCH
Value[4]:	MCT_SCH
Default:	
Required:	NO

The property `scheduler.method` selects the scheduling method for workflow scheduler.

