
system-intelligence Documentation

Release 2.0.2

Lukas Heumos

Jan 24, 2021

CONTENTS:

1	system-intelligence	3
1.1	Features	3
1.2	Credits	3
2	Installation	5
2.1	Stable release	5
2.2	From sources	5
3	Usage	7
3.1	command-line interface	7
3.2	System-intelligence on MacOS	8
3.3	System-intelligence on Windows	8
3.4	Generate HTML report	8
3.5	Module	8
4	Credits	11
4.1	Development Lead	11
4.2	Contributors	11
5	Changelog	13
5.1	2.0.2 (2021-01-24)	13
5.2	2.0.1 (2021-01-03)	13
5.3	2.0.0 (2021-01-03)	13
5.4	1.2.4 (2020-08-20)	14
5.5	1.2.3 (2020-08-18)	14
5.6	1.2.2 (2020-07-06)	14
5.7	1.2.1 (2020-06-25)	15
5.8	1.2.0 (2020-06-24)	15
5.9	1.1.0 (2020-06-23)	15
5.10	1.0.0 (2020-06-23)	15
6	Contributor Covenant Code of Conduct	17
6.1	Our Pledge	17
6.2	Our Standards	17
6.3	Our Responsibilities	17
6.4	Scope	18
6.5	Enforcement	18
6.6	Attribution	18
7	Indices and tables	19



SYSTEM-INTELLIGENCE

Query your system for all hardware and software related information.

- Free software: Apache2.0
- Documentation: <https://system-intelligence.readthedocs.io>.

1.1 Features

- Query your system for hardware and software related information
- Supports queries for Hostname, OS, CPU, GPUs, RAM, HDDs, Network and Software
- Supports Rich stdout or saving to a raw, json or yml file

1.2 Credits

This package was created with [cookiетemplate](#) based on a modified [audreyr/cookiecutter-pypackage](#) project template using [Cookiecutter](#).

INSTALLATION

System-intelligence supports all three major platforms: Linux, MacOS and Windows

2.1 Stable release

To install system-intelligence, run this command in your terminal:

```
$ pip install system-intelligence
```

This is the preferred method to install system-intelligence, as it will always install the most recent stable release.

If you don't have [pip](#) installed, this [Python installation guide](#) can guide you through the process.

2.2 From sources

The sources for system-intelligence can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/mlf-core/system-intelligence
```

Once you have a copy of the source and are in the top level directory, you can install it with:

```
$ pip install .
```


3.1 command-line interface

To get an overview of system-intelligence run

```
$ system-intelligence --help
```

system-intelligence queries your system for hardware and software related information. Available scopes are 'all', 'cpu', 'gpus', 'ram', 'host', 'os', 'hdd', 'swap', 'network', 'software'. To query for your desired scope run

```
$ system-intelligence <scope> <optionally more scopes> ... <optionally more scopes>
```

If system-intelligence is run with scope 'all', all other scopes will be queried for. Please note, that for GPU support you need to have [pycuda](#) installed. The installation instructions can be found [here](#).

To save all results into a file run

```
$ system-intelligence <scope> --output_format <raw/json/yml> --output <name_of_file.  
↪output_format>
```

As an example you may run

```
$ system-intelligence all --output_format json --output info.json
```

As of version 2.0.0, you can also run queries by querying all scopes except for some of them. An example, where one queries every scope except RAM and Software, would look like the following:

```
$ system-intelligence -e ram software
```

To suppress the standard output you may add the option

```
$ system_intelligence <scope> --silent
```

3.2 System-intelligence on MacOS

As with version 2.0.0, system-intelligence can also query under MacOS. However, there are a few peculiarities:

- 1.) Querying host will also display the model marketing name of the users device (if available)
- 2.) As RAM info on MacOS is limited compared to Linux or Windows, the RAM query will only display Slot, Type, Size, Speed and Serial Number of each RAM slot as well as the total memory size
- 3.) There is no need to run it with sudo, since it's not required like in Linux for example
- 4.) All memory size units (so every Byte unit) is displayed using base10, as this is 'Darwin's' way to compute such units
- 5.) The L1 Cache in the CPU query summarizes the L1i cache and the L1d cache together

3.3 System-intelligence on Windows

As with version 2.0.0, system-intelligence can also query under Windows. There are only a few minor things different to whats displayed under linux:

- 1.) L1 CPU cache size is not available (as for now) and thus will be not displayed
- 2.) Size units (bytes basically) are displayed using base2 computation as this is how Windows computes those numbers

3.4 Generate HTML report

System-intelligence can not only print the results to stdout, but is also capable of creating nice HTML reports that can be included in your research or appliaction. To generate such report in a file called result.html above the current working directory simply run the following:

```
$ system_intelligence all -f json -g -o result.html
```

Note, that under Windows, this must be:

```
$ system_intelligence all -f json -g -o result.html
```

3.5 Module

To use system-intelligence in a project:

```
import system_intelligence
```

To run queries you have to import

```
from system_intelligence.query import query_and_export
```

which will allow you to call the `query_and_export` function. Please note that it requires a list of a tuple as input. This is pretty much a relict of the command line library Click.

```
query_and_export(query_scope=list(('all',)), verbose=True, export_format='json',  
↳ generate_html_table=True, output='system_intelligence.json')::
```


CREDITS

4.1 Development Lead

- Lukas Heumos <lukas.heumos@posteo.net>
- Philipp Ehmele <philipp_ehm@protonmail.com>

4.2 Contributors

None yet. Why not be the first?

CHANGELOG

This project adheres to [Semantic Versioning](#).

5.1 2.0.2 (2021-01-24)

Added

Fixed

Dependencies

- less strict pin of rich and click

Deprecated

5.2 2.0.1 (2021-01-03)

Added

- Updated documentation
- Updated PyPi classifiers

Fixed

Dependencies

Deprecated

5.3 2.0.0 (2021-01-03)

Added

- Complete support for MacOS
- Complete support for Windows
- Modelname query for MacOS
- Python 3.9 WFs for all GitHub Actions
- Run system-intelligence WF after build
- Multiple comments and docstrings

- Redesigned code structure to more OOP concept
- RAM memory attributes will now be displayed in Bytes (was Hertz)

Fixed

- Accelerated software query (about 5 times faster now)
- Fixed a bug that causes SI to crash and not generate a html report when path given via cli option was different to the cwd

Dependencies

- plistlib (from std lib)

Deprecated

- Unit-util file (most functions are now part of the base info class)

5.4 1.2.4 (2020-08-20)

Added

- Richified all prints

Fixed

Dependencies

Deprecated

5.5 1.2.3 (2020-08-18)

Added

- mkl to supported software to query for

Fixed

- Python dependency version handling

Dependencies

Deprecated

5.6 1.2.2 (2020-07-06)

Added

Fixed

- cpu cache raw_level is now always a string (fixes py-cpuinfo 7.0.0 returning integers)
- Replaced gif with full size gif

Dependencies

Deprecated

5.7 1.2.1 (2020-06-25)

Added

Fixed

- Import error if pycuda is not available

Dependencies

Deprecated

5.8 1.2.0 (2020-06-24)

Added

- html table output via `--generate_html_table`

Fixed

Dependencies

Deprecated

5.9 1.1.0 (2020-06-23)

Added

- Python packages to stdout

Fixed

- nvcc version is now correctly reported

Dependencies

Deprecated

5.10 1.0.0 (2020-06-23)

Added

- Scopes all, Hostname, OS, CPU, GPUs, RAM, HDDs, Network and Software
- saving to raw, json, yml
- rich stdout

Fixed

Dependencies

Deprecated

CONTRIBUTOR COVENANT CODE OF CONDUCT

6.1 Our Pledge

In the interest of fostering an open and welcoming environment, we as contributors and maintainers pledge to making participation in our project and our community a harassment-free experience for everyone, regardless of age, body size, disability, ethnicity, gender identity and expression, level of experience, nationality, personal appearance, race, religion, or sexual identity and orientation.

6.2 Our Standards

Examples of behavior that contributes to creating a positive environment include:

- Using welcoming and inclusive language
- Being respectful of differing viewpoints and experiences
- Gracefully accepting constructive criticism
- Focusing on what is best for the community
- Showing empathy towards other community members

Examples of unacceptable behavior by participants include:

- The use of sexualized language or imagery and unwelcome sexual attention or advances
- Trolling, insulting/derogatory comments, and personal or political attacks
- Public or private harassment
- Publishing others' private information, such as a physical or electronic address, without explicit permission
- Other conduct which could reasonably be considered inappropriate in a professional setting

6.3 Our Responsibilities

Project maintainers are responsible for clarifying the standards of acceptable behavior and are expected to take appropriate and fair corrective action in response to any instances of unacceptable behavior.

Project maintainers have the right and responsibility to remove, edit, or reject comments, commits, code, wiki edits, issues, and other contributions that are not aligned to this Code of Conduct, or to ban temporarily or permanently any contributor for other behaviors that they deem inappropriate, threatening, offensive, or harmful.

6.4 Scope

This Code of Conduct applies both within project spaces and in public spaces when an individual is representing the project or its community. Examples of representing a project or community include using an official project e-mail address, posting via an official social media account, or acting as an appointed representative at an online or offline event. Representation of a project may be further defined and clarified by project maintainers.

6.5 Enforcement

Instances of abusive, harassing, or otherwise unacceptable behavior may be reported by opening an issue. The project team will review and investigate all complaints, and will respond in a way that it deems appropriate to the circumstances. The project team is obligated to maintain confidentiality with regard to the reporter of an incident. Further details of specific enforcement policies may be posted separately.

Project maintainers who do not follow or enforce the Code of Conduct in good faith may face temporary or permanent repercussions as determined by other members of the project's leadership.

6.6 Attribution

This Code of Conduct is adapted from the Contributor Covenant, version 1.4, available at <https://www.contributor-covenant.org/version/1/4/code-of-conduct.html>

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`