Using Python-BRM

# Using BRM in fast growing enterprise architecture

In our days the life of big IT organization can’t be without central repository of rules managed by stable fast rule engine. The [Python-BRM](http://github/zeroprg/python-brm) is one of the big spectrum of BRM engines which is available on the market as open source. This typical standard network topology which all of them (BRM engines) ideally must follow:



The idea is to satisfy current and potential rules consumers of organization. Rules inventory REST server is repository of rules which has set of rules stored as Excel Spread sheets (can be stored as JSON file) for different occasions. It has UI which can be accessed from any browser to upload a new rules set and fire rule set against array of parameters. This UI used as testing point for recently created and uploaded rules set.

# Using Excell SPREADSHEET as key factor for SUCCESs

Based on long experience of working with final users (customers). Introducing any sophisticated tool more then spreadsheet could be a pain. In most successful BRM implementations spreadsheet as rule collector is winner. Here is description how [Python-BRM](http://github/zeroprg/python-brm) looks like

Parameters which used by Rules in this column. (This parameters must be specified in parameters input JSON,CSV,EXCELL file)



Example of Rule two hazardous cars can’t be together

Train based rule :

“Sum\_of”

Hardcoded rule

Rules grouped by some criteria

# BRM inventorY server diagram

In spade of powerful functionality [Python-BRM](http://github/zeroprg/python-brm) has only 3 python classes which is shown bellow. UML sequence diagram shows two type of actors: UI user who may play role of administrator and regular user (actor below first actor) who send only parameters and reuse already existed set of rules.

