BlockIdentification

Block URI Scheme

All blocks (behaviors and implementations) are identified by a URI. The format of the URI is as follows:

```
http://organization/name/x.y(.z)
```

where

- `http:` is the hypertext transfer protocol. This is used to allow the identifier (which is a URI and should be considered as such) also to be dereferencable (the URI can be treated like a URL)
- `organization` is the unique identifier for the organization that is responsible for the maintenance of that identifier. the ICANN domain name should be used (for example, `apache.org` for the ASF). It is up to the URI creator to make sure that the URI is potentially dereferencable (i.e., if you don't use "www." in front, your DNS should be properly configured to map the domain name as a machine name)
- `name` is the unique name of the identifier. it is suggested that a path delimiter is used to further specialize the name (see belows for examples)
- `x.y.z` is the version identifier, where:
    - x -> major (>= 1)
    - y -> minor (>= 0)
    - z -> bugfix (>= 0) (only for implementations)

NOTE: even if URI are generally case sensitive, block identifiers will be treated as case insensitive.

Example of good identifiers are

- http://apache.org/cocoon/PDF/2.6
- http://apache.org/cocoon/Fop/3.4.34
- http://apache.org/cocoon/iText/1.0.43
- http://mycompany.com/mydepartment/myself/myblock/3.2.23

example of bad identifiers

- http://apache.org/cocoon/PDF/Fop/2.3.43

information of what behavior is implemented by a given block implementation should not be included in the identifier.

Dependency Ranges

When a block implementation depends on another block (either implementation or behavior), it should be able to have an 'elastic' dependency which doesn't connect it to the versioned identifier, but to a range of those versions.

Instead of explicitly indicate the range description language, it is suggested to implicity describe range rules. These implicit range rules are:

- if the dependency doesn't include the version, all versions are matched

ex: both "http://apache.org/blah/1.0" and "http://apache.org/blah/3.43.342" are matched by "http://apache.org/blah"

- if the dependency includes a version, versions are matched with the following rules
    - if major is equal
    - if minor is greater or equal
    - in case of implementations and if minor is equal, if bugfix is greater or equal

ex: depending on "http://apache.org/blah/2.0.34" will match

- http://apache.org/blah/2.0.345
- http://apache.org/blah/2.3.23

but not

- http://apache.org/blah/1.0.0
- http://apache.org/blah/34.323.324534