SEP	12
Title	Spider name
Author	Ismael Carnales, Pablo Hoffman
Created	2009-12-01
Updated	2010-03-23
Status	Final

SEP-012: Spider name

The spiders are currently referenced by its `domain_name` attribute. This SEP proposes adding a `name` attribute to spiders and using it as their identifier.

Current limitations and flaws

1. You can't create two spiders that scrape the same domain (without using workarounds like assigning an arbitrary `domain_name` and putting the real domains in the `extra_domain_names` attributes)
2. For spiders with multiple domains, you have to specify them in two different places: `domain_name` and `extra_domain_names`.

Proposed changes

1. Add a `name` attribute to spiders and use it as their unique identifier.
2. Merge `domain_name` and `extra_domain_names` attributes in a single list `allowed_domains`.

Implications of the changes

General

In general, all references to `spider.domain_name` will be replaced by `spider.name`

OffsiteMiddleware

`OffsiteMiddleware` will use `spider.allowed_domains` for determining the domain names of a spider

scrapy-ctl.py

crawl

The new syntax for crawl command will be:

```
crawl [options] <spider|url> ...
```

If you provide an url, it will try to find the spider the processes it. If no spider is found or more than one spider is found, it will raise an error. So, to crawl in those cases you must set the spider to use using the `--spider` option

genspider

The new signature for genspider will be:

```
genspider [options] <name> <domain>
```

example:

```
$ scrapy-ctl genspider google google.com

$ ls project/spiders/
project/spiders/google.py

$ cat project/spiders/google.py

class GooglecomSpider(BaseSpider):
   name = 'google'
   allowed_domains = ['google.com']
```

> **Note**
>
> `spider_allowed_domains` becomes optional as only `OffsiteMiddleware` uses it.