

SmartFrog Quick Reference

Integer: 345

Long: 65325L or 65325l
Float: 34.76F or 34.76f or 34.76E-10F or 34.76e+10f or 34.76E10f
Double: 1534.45 or 1534.45D or 1534.45d or 1534.45E10 or 1534.45E-10D
String: "this is a string"
Multi-line String: ## This is a string
Over many lines #
Boolean: true or false
SFNull: NULL // alternatively, leave the value empty

Operators: ! + - / == != >= <= < * ++ <> && ||

IF (val1 > val2) THEN (val1 - val2) ELSE (val2 - val1) FI;

Simple Vectors:

```
userList [| "fred", "harry" |];  
empty [| |];  
listOfLists [| [| 1,2,3 |], [| 4,5,6 |] |];
```

Full vectors can contain *references*:

```
v1 [|1,2,3|];  
v2 [|9,8,7|];  
v3 [v1, v2]; // same as [[1,2,3],[9,8,7]]
```

Components

Component ::= **extends** [LAZY] Type Body
Type ::= [NULL | BaseReference]
Body ::= (; | { AttributeList })

```
/* example */  
sfConfig extends Prim {  
    portNum 4047;  
    sfProcessHost "ahost.smartfrog.org";  
    administrators ["patrick",ATTRIB ROOT:sysadmin];  
}
```

The sfConfig attribute identifies the component in a description to deploy.

Include files

Include files are resources on the classpath of the deploying JVM; they are expanded before deployment.

```
system extends {  
    -- extends foo; //anonymous component  
    #include "/absolute/path/foo.sf"  
}
```

References

```
Reference ::= [LAZY] BaseReference  
BaseReference ::= ReferencePart ( : ReferencePart)*  
ReferencePart ::= ROOT // root of the descriptor '/'  
| PARENT // parent '..'  
| (ATTRIB WORD) //closet attribute 'WORD'  
| (HERE WORD) //local attribute 'WORD' only  
| THIS //local context '.'  
| WORD //ATTRIB if first/only part, else HERE  
| (PROPERTY WORD) //JVM string property (see LAZY)  
| (IPROPERTY WORD) //JVM integer property (see LAZY)  
| PROCESS //current process  
| (HOST (WORD | STRING)) //host by reference/address
```

As non-lazy attributes are evaluated at parse time, PROPERTY and PROCESS references will be evaluated in the JVM parsing the file, not the process hosting the components, *unless* LAZY is used to declare late binding of information.

Functions

```
#include "org/smartfrog/functions.sf";  
myString extends formatString {  
    format "the meaning of $2 is $1";  
    s1 42;  
    s2 "life";  
}
```

When a function is evaluated, its declaration is replaced by the result: myString *becomes* the string. LAZY references can not be used in functions.

Shell commands:

```
smartfrog -a SFACT -f SFREF -e -?  
  
sfRun -a SFREF -e (= smartfrog -a :DEPLOY:SFREF:: -e)  
  
sfStart hostname appname SFREF (= smartfrog -a appname:DEPLOY:SFREF::hostname: -e)  
  
sfTerminate hostname appname (=smartfrog -a appname:TERMINATE:::hostname: -e)  
  
sfStopDaemon hostname (= smartfrog -a rootProcess:TERMINATE:::hostname: -e)  
  
sfDetachAndTerminate hostname appname (=smartfrog -a appname:DETaTERM:::hostname: -e)  
  
sfParse [-v] [-q] [-r] [-R] {[-f] filename | SFREF}
```

SmartFrog Action Descriptor (SFACT): NAME : ACTION : SFREF : SUBREF : HOST : PROCESS

SmartFrog Resource Reference (SFREF): URL to the file; as a relative or absolute path name to a file; as a path to a resource in a jar file on the classpath or code base.

Well-Known Attributes

Deployment attributes:

sfConfig	the root of a SmartFrog description
----------	-------------------------------------

Component:

sfClass	class that implements a component
sfCodeBase	Java codebase for a component
sfDeployerClass	The class name for deployer
sfExport	enable/disable remote method calls
sfSchemaDescription	the definition of a schema
sfSyncTerminate	asynchronous or synchronous termination of compound

Location:

sfProcessComponentName	name a component in root process
sfProcessHost	host where to deploy component
sfProcessName	subprocess where to deploy component

Liveness:

sfLivenessDelay	how often to send liveness in seconds.
sfLivenessFactor	how many multiples of the liveness delay to wait till a liveness failure of the parent is declared

SubProcesses:

sfProcessAllow	enable/disable subprocesses
sfProcessTimeout	Garbage collection time out for subprocesses
sfRootLocatorPort	Registry port used by the <code>rootProcess</code> daemon. Default: 3800
sfSubprocessGCTimeout	garbage collection time out for subprocesses

Runtime attributes:

sfHost	The host address where a component runs
sfProcess	the process/subprocess name where a component runs.

Useful Components

Prim	
Compound	
ExecuteProgram, RunShellScripts, Java	Run external programs, scripts, Java applications.
Downloader	Download a file from a URL
File, Mkdir, TextFile, TempFile	File manipulation components.
LivenessPage	Check that an HTTP URL is valid, as a liveness check.
SetProperty	Set a JVM Property
Assert	Simple assertions that are checked at deployment and during liveness.
sfPersist	Persist a component to a description file
Trace	Tracing for life cycle hooks. org/smartfrog/services/trace/components.sf

Management Console

`sfManagementConsole -h <HOSTNAME> -p <PORT_NUMBER> -r: -?`

Default configuration files for shell scripts

- Ini file: dist/bin/default.ini
- Default deployment description: dist/bin/default.sf
- Environment variables used:
 - Smartfrog distribution directory: SFHOME
 - User libs: SFUSERHOME,SFUSERHOME1, ...,SFUSERHOME4