SArray *)arrayForKey:(NSString *)defaultName

Invokes objectForKey: with key defaultName. Returns the corresponding value if it's an NSArray object (according to the isKindOfClass: test) and nil otherwise.

OOL)boolForKey:(NSString *)defaultName

Invokes stringForKey: with key defaultName. Returns YES if the corresponding value is an NSString containing uppercase or lowercase ªYESº or responds to the intValue message by returning a non-zero value. Otherwise, returns NO.

SData *)dataForKey:(NSString *)defaultName

Invokes objectForKey: with key defaultName. Returns the corresponding value if it's an NSData object (according to the isKindOfClass: test) and nil otherwise.

SDictionary *)dictionaryForKey:(NSString *)defaultName

Invokes objectForKey: with key defaultName. Returns the corresponding value if it's an NSDictionary object (according to the isKindOfClass: test) and nil otherwise.

id)removeObjectForKey:(NSString *)defaultName

Removes the value for the given default in the standard ap
Removing a default has no effect on the value returned
method if the same key exists in a domain that precede
application domain in the search list.

id)setBool:(BOOL)value
forKey:(NSString *)defaultName

Sets the value of the specified default to a string
representation of YES or NO, depending on value. Inv
as part of its implementation.

id)setFloat:(float)value
forKey:(NSString *)defaultName

Sets the value of the specified default to a string
representation of value. Invokes setObject:forKey: as p
implementation.

id)setInteger:(int)value
forKey:(NSString *)defaultName

Sets the value of the specified default to a string
representation of value. Invokes setObject:forKey: as p
implementation.

id)setObject:(id)value
forKey:(NSString *)defaultName

Sets the value of the specified default in the standard
application domain. Setting a default has no effect on t
the objectForKey: method if the same key exists in a d
application domain in the search list.

SArray *)stringArrayForKey:(NSString *)defaultName

Invokes objectForKey: with key defaultName. Returns the
it's an NSArray object containing NSStrings, and nil o
each object is determined using the isKindOfClass: tes

SString *)stringForKey:(NSString *)defaultName

Invokes objectForKey: with key defaultName. Returns the
it's an NSString object (according to the isKindOfClas
otherwise.

)init

Initializes defaults for the current user (who's identified b
environment). This method doesn't put anything in the
only if you've allocated your own NSUserDefaults obj
shared one. Returns self.

)initWithUser:(NSString *)userName

Like init, but initializes defaults for the specified user.

SMutableArray *)searchList

Returns a mutable array of domain names, signifying the d
objectForKey: will search. You can customize the sear
array that's returned. Non-existent domain names in th

SDictionary *)persistentDomainForName:(NSString *)domainName

Returns a dictionary corresponding to the specified persist
in the dictionary are names of defaults, and the value c
key is a property list data object.

SArray *)persistentDomainNames

Returns an array containing the names of the persistent do
can then be retrieved by invoking persistentDomainFo

id)removePersistentDomainForName:(NSString *)domainName

domains that were not modified to what is on disk. Ret
save data to disk. Since the synchronize method is auto
periodic intervals, use this method only if you cannot w
synchronization (for example if your application is abo
want to update user defaults to what is on disk even tho
made any changes.

id)removeVolatileDomainForName:(NSString *)domainName

Removes the named volatile domain from the user's defau

id)setVolatileDomain:(NSDictionary *)domain
forName:(NSString *)domainName

Sets the dictionary to domain for the volatile domain name
method raises an NSInvalidArgumentException if a pe
domainName already exists.

SDictionary *)volatileDomainForName:(NSString *)domainName

Returns a dictionary corresponding to the specified volatil
the dictionary are names of defaults, and the value corr
is a property list data object.

SArray *)volatileDomainNames

Returns an array containing the names of the volatile dom
then be retrieved by calling volatileDomainForName:.

SDictionary *)dictionaryRepresentation

Returns a dictionary that contains a union of all key-value
the search list. As with objectForKey:, key-value pairs
earlier in the search list take precedence. The combine
preserve information about which domain each entry c

id)registerDefaults:(NSDictionary *)dictionary

Adds the contents of dictionary to the registration domain.
registration domain yet, it's created using dictionary, a
NSRegistrationDomain is added to the end of the searc