

Types and Constants

Defined Types

Cache

```
typedef struct objc_cache *Cache;
```

Category

```
typedef struct objc_category *Category;
```

Ivar

```
typedef struct objc_ivar *Ivar;
```

marg_list

```
typedef void *marg_list;
```

Method

```
typedef struct objc_method *Method;
```

Module

```
typedef struct objc_module *Module;
```

Symbolic Constants

Type Constants	Meaning	Defined As
_C_ID	id	`@'
_C_CLASS	Class	`#'
_C_SEL	SEL	`:'
_C_VOID	void	`v'
_C_CHR	char	`c'
_C_UCHR	unsigned char	`C'
_C_SHT	short	`s'
_C_USHT	unsigned short	`S'
_C_INT	int	`i'
_C_UINT	unsigned int	`I'
_C_LNG	long	`l'
_C_ULNG	unsigned long	`L'
_C_FLT	float	`f'
_C_DBL	double	`d'

_C_UNDEF	an undefined type	`?'
_C_PTR	a pointer	`^'
_C_CHARPTR	char *	`*'
_C_BFLD	a bitfield	`b'
_C_ARY_B	begin an array	`['
_C_ARY_E	end an array	`]'
_C_UNION_B	begin a union	`('
_C_UNION_E	end a union	`)'
_C_STRUCT_B	begin a structure	`{'
_C_STRUCT_E	end a structure	`}'

Structures

objc_cache

```
struct objc_cache {
    unsigned int mask;
    unsigned int occupied;
    Method buckets[1];
};
```

objc_category

```
struct objc_category {
    char *category_name;
    char *class_name;
    struct objc_method_list *instance_methods;
    struct objc_method_list *class_methods;
    struct objc_protocol_list *protocols;
};
```

objc_class

```
struct objc_class {
    struct objc_class *isa;
    struct objc_class *super_class;
    const char *name;
    long version;
    long info;
    long instance_size;
    struct objc_ivar_list *ivars;
    struct objc_method_list *methods;
    struct objc_cache *cache;
    struct objc_protocol_list *protocols;
};
```

objc_ivar

```
struct objc_ivar {
    char *ivar_name;
    char *ivar_type;
    int ivar_offset;
};
```

objc_ivar_list

```
struct objc_ivar_list {  
    int ivar_count;  
    struct objc_ivar ivar_list[1];  
};
```

objc_method

```
struct objc_method {  
    SEL method_name;  
    char *method_types;  
    IMP method_imp;  
};
```

objc_method_description

```
struct objc_method_description {  
    SEL name;  
    char *types;  
};
```

objc_method_description_list

```
struct objc_method_description_list {  
    int count;  
    struct objc_method_description list[1];  
};
```

objc_method_list

```
struct objc_method_list {  
    struct objc_method_list *method_next;  
    int method_count;  
    struct objc_method method_list[1];  
};
```

objc_module

```
struct objc_module {  
    unsigned long version;  
    unsigned long size;  
    const char *name;  
    Symtab symtab;  
};
```

objc_protocol_list

```
struct objc_protocol_list {  
    struct objc_protocol_list *next  
    int count;  
    Protocol *list[1];  
};
```

objc_super

```
struct objc_super {
    id receiver;
    Class class;
};
```

Global Variables

Function Pointers

```
id (*_alloc)          (Class aClass, unsigned int indexedIvarBytes)
id (*_dealloc)        (Object *anObject)
id (*_realloc)        (Object *anObject, unsigned int numBytes)
id (*_copy)           (Object *anObject, unsigned int indexedIvarBytes)
id (*_zoneAlloc)      (Class aClass, unsigned int indexedIvarBytes, NXZone *zone)
id (*_zoneRealloc)    (Object *anObject, unsigned int numBytes, NXZone *zone)
id (*_zoneCopy)       (Object *anObject, unsigned int indexedIvarBytes, NXZone *zone)
id (*_error)          (Object *anObject, char *format, va_list ap)
```