

AliasRecord structure

```
#include <Aliases.h>
```

		<u>Size</u>	<u>Offset</u>	<u>Description</u>
typedef struct AliasRecord {	<u>OSType</u> userType;	4	0	application's signature
	<u>unsigned short</u> aliasSize;	2	4	size of record when created
		<i>n</i>	6	variable length private data

```
} AliasRecord;
```

```
typedef AliasRecord *AliasPtr;
```

```
typedef AliasRecord **AliasHandle;
```

Notes: Your application can use the userType field to store its own signature or any other data that fits into 4 bytes. When the **Alias Manager** creates an **AliasRecord**, it stores 0 in that field.

The **Alias Manager** stores the size of the record when it was created in the aliasSize field. Knowing the starting size allows you to store and retrieve data of your own at the end of the record (see **Customizing Alias Records** under **Using the Alias Manager**). An **AliasRecord** is typically 200 to 300 bytes long.

The private **Alias Manager** data includes all of the location, verification, and mounting information needed to resolve the **AliasRecord** with the various search strategies described in **Search Strategies for Resolving Alias Records**.