Defining Messages

Michael Van Sickle

@vansimke

```
message Customer {

    uint64 id = 1;

    string username = 2;

    bool is_active = 3;

}
```
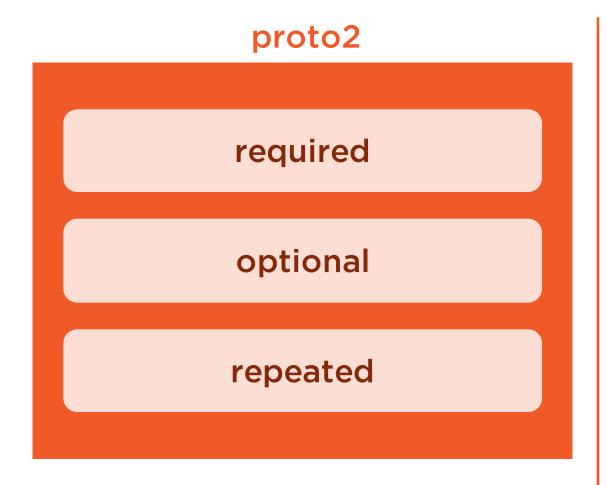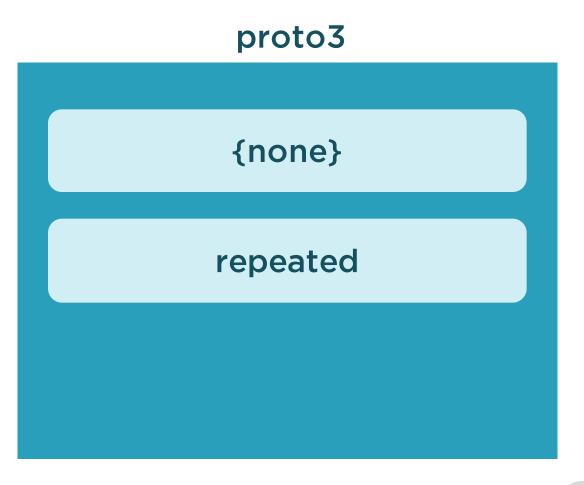
Messages

Fields

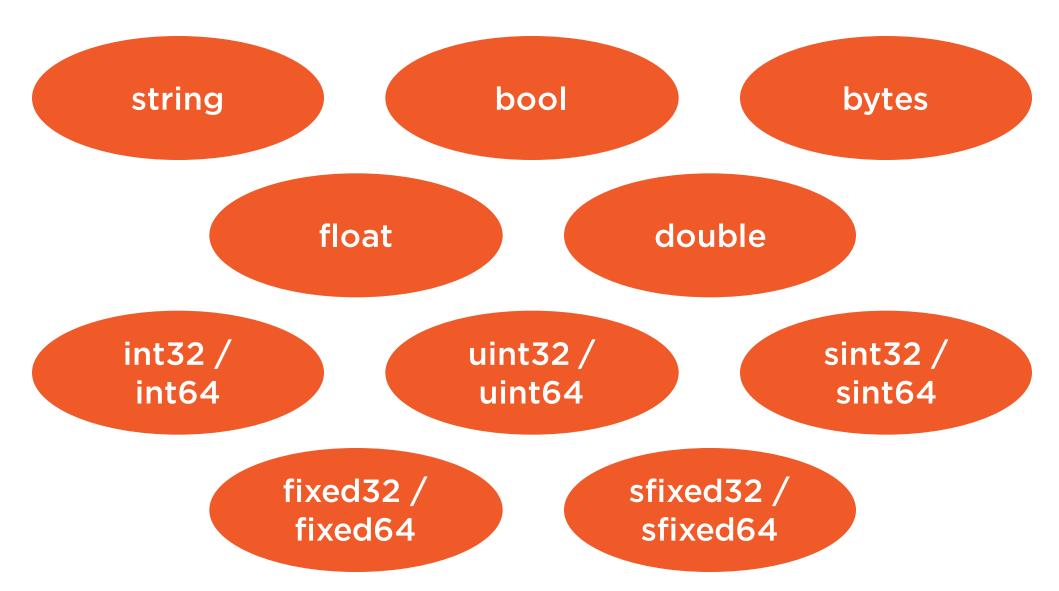**Other message members**

Field Elements

```
repeated uint64 vals = 1;
```

rule	type	name	tag

Field Rule

Field Type

Scalar

Enumeration

Message Type

Scalar Types

string

bool

bytes

float

double

int32 / int64

uint32 / uint64

sint32 / sint64

fixed32 / fixed64

sfixed32 / sfixed64

Enumerations

```
message Customer {

    enum Type {

        REGULAR = 0;              var customer = new Customer();

        MEMBER = 1;

        SPONSOR = 2;              customer.setType(

    }                                 Customer.Type.REGULAR);

    Type type = 1;

}
```

```
message Customer {

    string first_name = 1;

}
```

Field Name

- lower case

- use underscore (_) as separator

- converted to correct style for each language

Field Tags

unique

integer
[1, 2^29 – 1]

reserved
[19000, 19999]

smaller values are more efficient

Default Values

proto2	proto3
Normally the zero value	**Always the zero value**

```
optional int32 results_per_page = 1 [default = 25];
```

Nested Message Types

```
message Customer {

    message Address {

        . . .

    }

    repeated Address addresses = 1;

}
```

```
var address = new Customer.Address()
```

Other Message Members

oneof

map

the 'Any' type

```
message Customer {

    oneof access_type {

        string email = 1;

        string username = 2;

    }

}
```

oneof

- **Only one field can hold value**

- **Setting one field clears others**

- **More efficient memory usage (when possible)**

maps

```
message Customer {

    map<string, string> email_addresses = 1;

}
```

keys must be strings or integer types

The 'Any' Type

```
import "google/protobuf/any.proto";


message Customer {

    int32 id = 1;

    string username = 2;

    repeated google.protobuf.Any details = 3;
}
```

Summary

Fields
- Rules
- Types
- Name
- Tag

Enumerations

Nested Messages

Oneof

Map

The 'Any' Type