

## V3 - Alternative schemas

### Alternative schemas

Sometimes there is more than one acceptable schema. For example if you were sending money to the UK, there are two types of account details you could use, a sort code and account number, or an IBAN. These would require two different JSON schemas. JSON schema has a mechanism to cover exactly this eventuality: 'oneOf'. 'oneOf' lets you specify that any one (and exactly one) of the following schemas is acceptable.

```
{
  oneOf: [{
    type: "object",
    title: "Sort code & account number"
    properties: {
      sortCode: {
        type: "string",
        format: "number",
        title: "UK sort code"
      },
      accountNumber: {
        type: "string",
        format: "number",
        title: "Account number"
      }
    }
  }, {
    type: "object"
    title: "IBAN"
    properties: {
      iban: {
        type: "string",
        format: "number",
        title: "IBAN"
      }
    }
  }
]
}
```

When we encounter a oneOf we need to render a selector that allows the user to choose which type of object they wish to create. This could be a select box, a set of tabs, radio buttons, or an extra step before we show the form.

➤ model that we create does not require any additional nesting. Either of the following models would satisfy the above specification:

```
{
  sortCode: "123456",
  accountNumber: "12345678"
}
```

```
{  
  iban: "123456789012345678"  
}
```

Client interface might render these alternatives as tabs, or as a two step process where you select the type first, and then complete the form.

Sort code & account number

IBAN

UK Sort code

Account number