

Making Apps More Resilient By Applying Darwinism

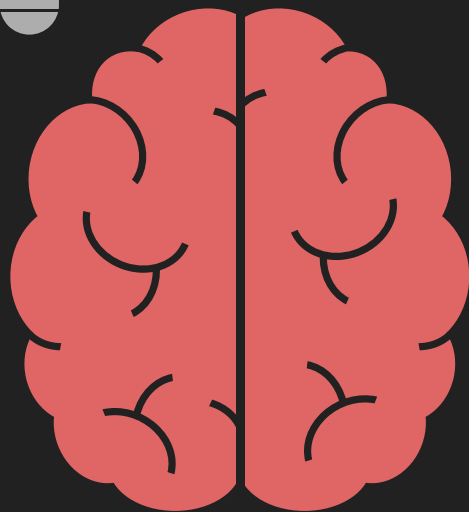


Masking Technology
hello@masking.tech
[linkedin.com/company/maskingtechnology](https://www.linkedin.com/company/maskingtechnology)



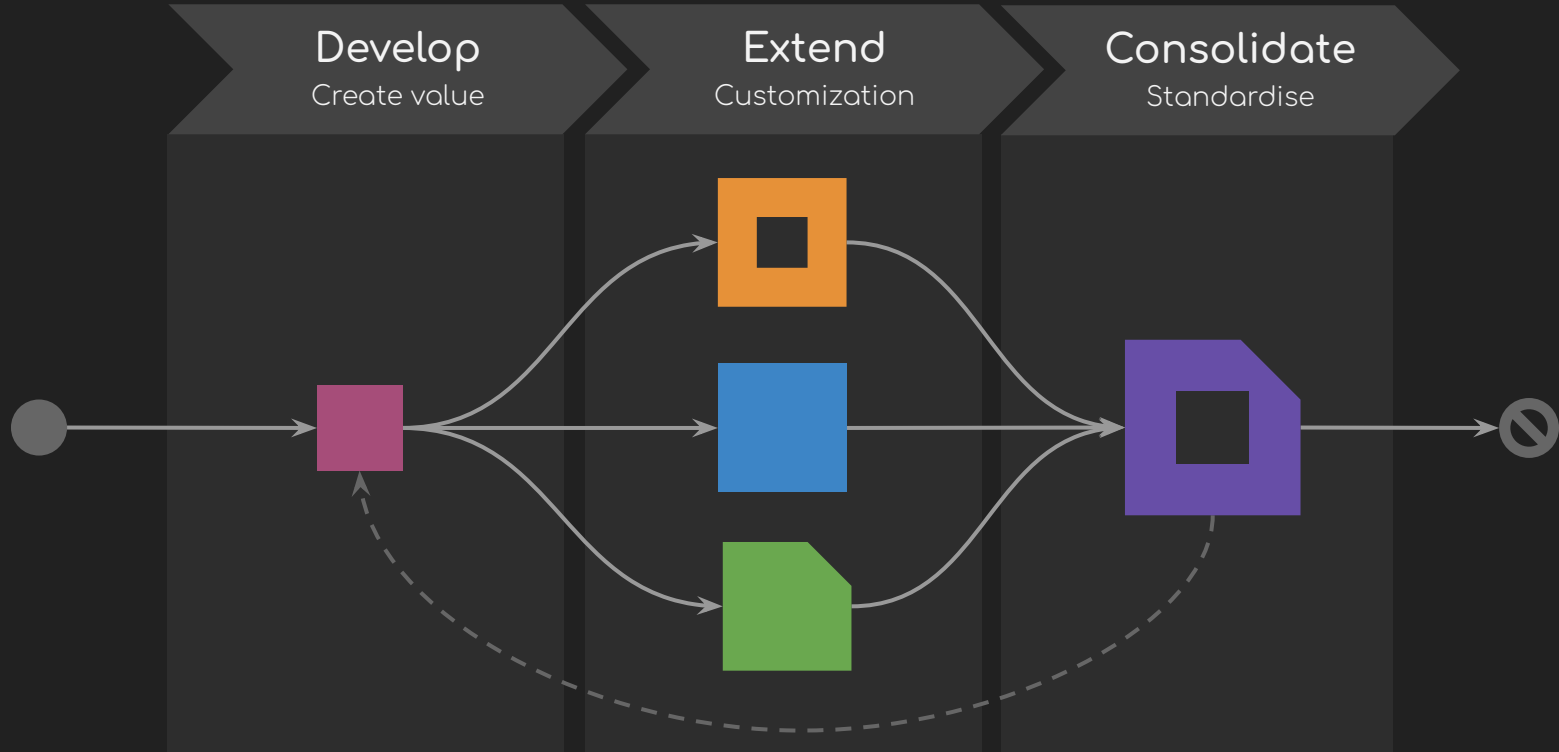
Evolution in software development



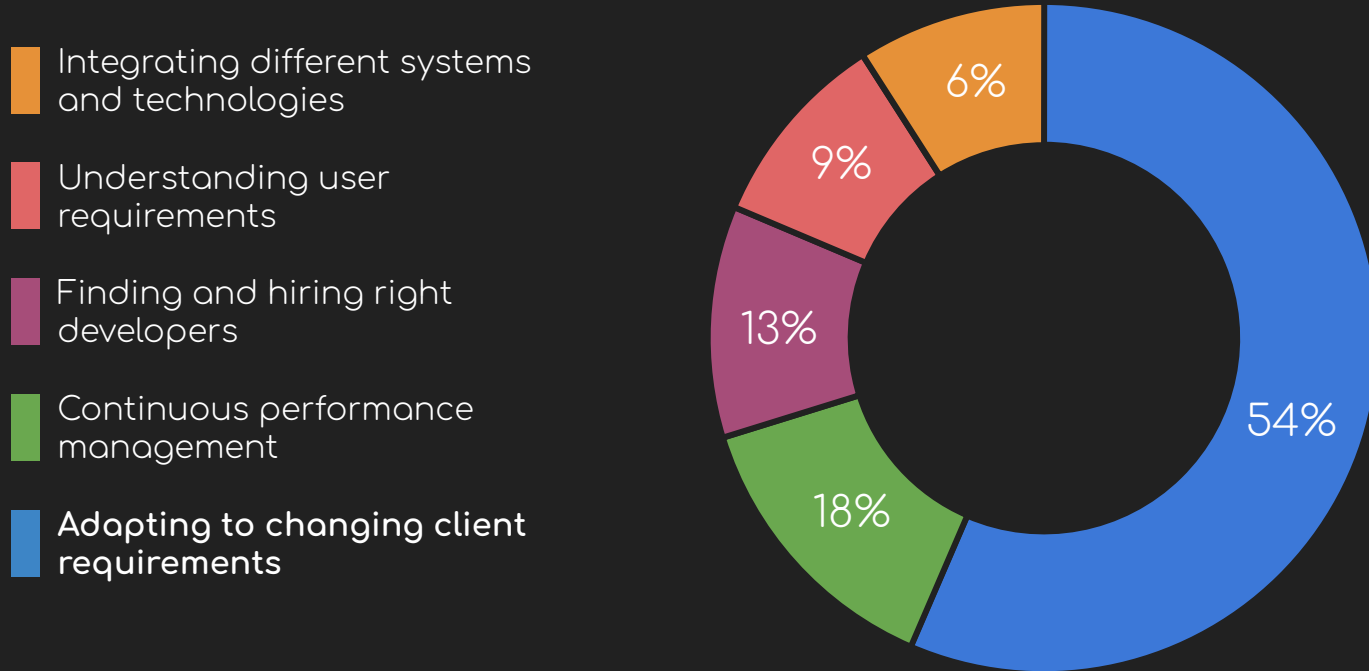


Sharing
ideas

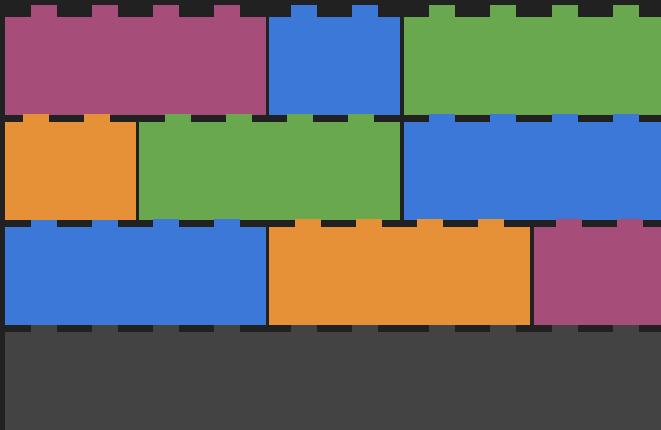
Common application life cycle



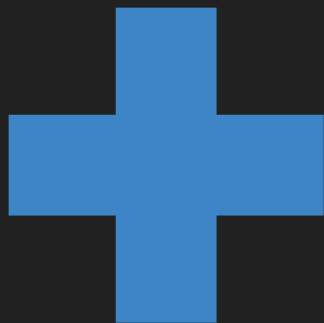
Biggest challenges of software development



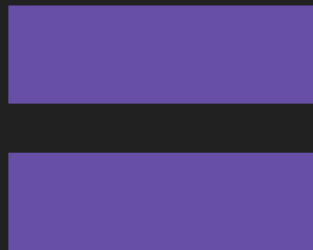
(GoodFirms. (2024, November 8). Remarkably Useful Stats and Trends on Software Development)



Common
application
structure



Extendable



Modifiable

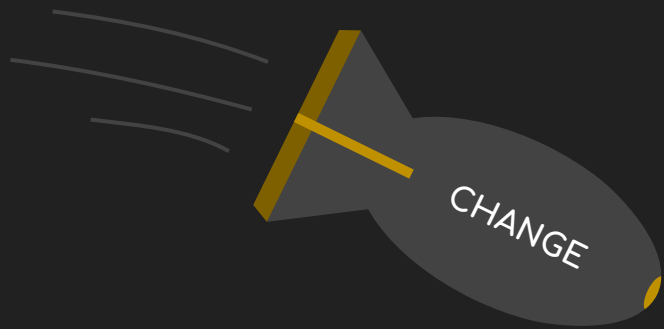


Reducible

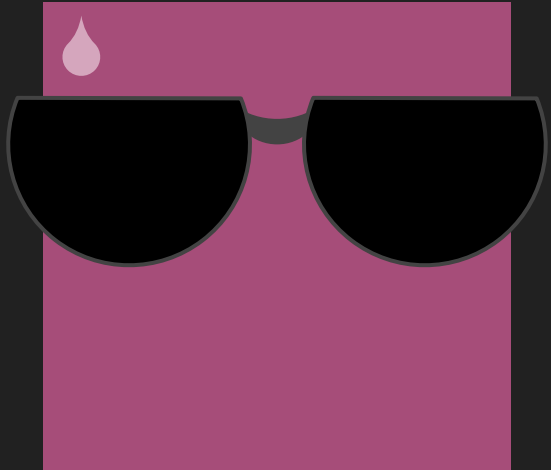
Resilient applications are
adaptable by nature.

With minimal refactoring, recover from
disruptions, and avoid catastrophic failure.

- Masking Technology -



Can I get a hint on how to do that?



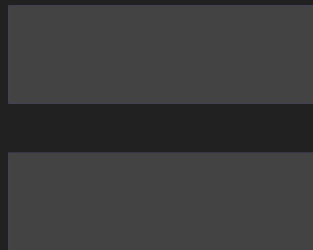
A penny on its side is
stable unless disturbed.

This is the same for a software system.

- Robert C. Martin (CEO Clean Coder) -



Extendable



Modifiable

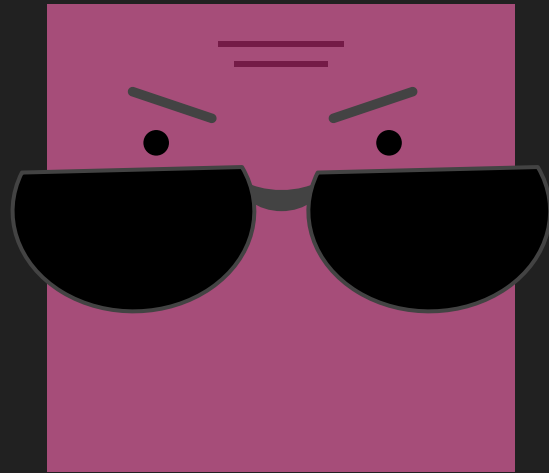


Reducible

Disturbances

The word "Disturbances" is written in white text above the center bars. Two red curved arrows originate from the word, one pointing left towards the center bars and one pointing right towards the single bar on the right.

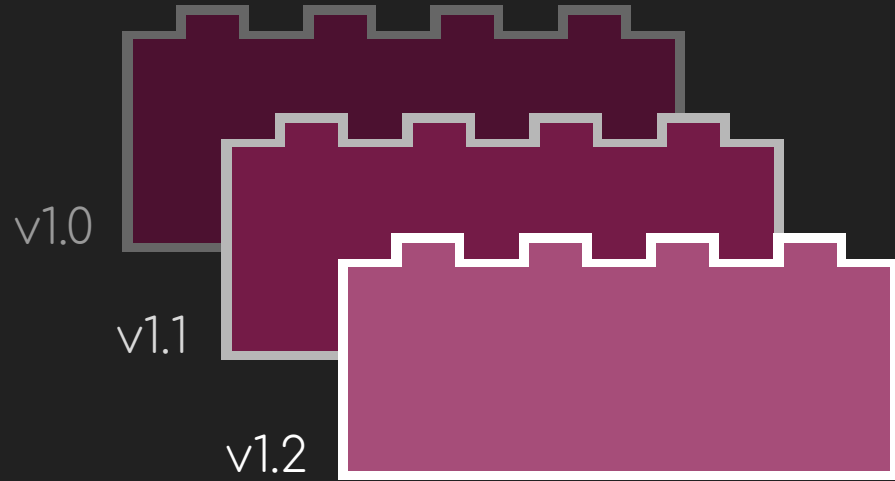
I'm listening...



Make
components
immutable

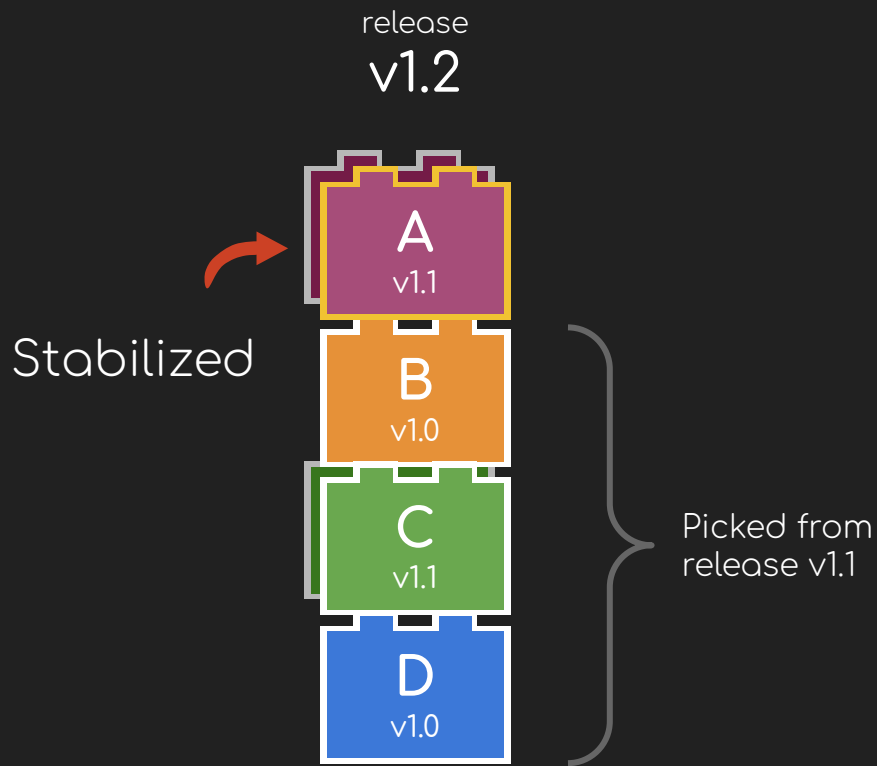


Create a
new version
per change



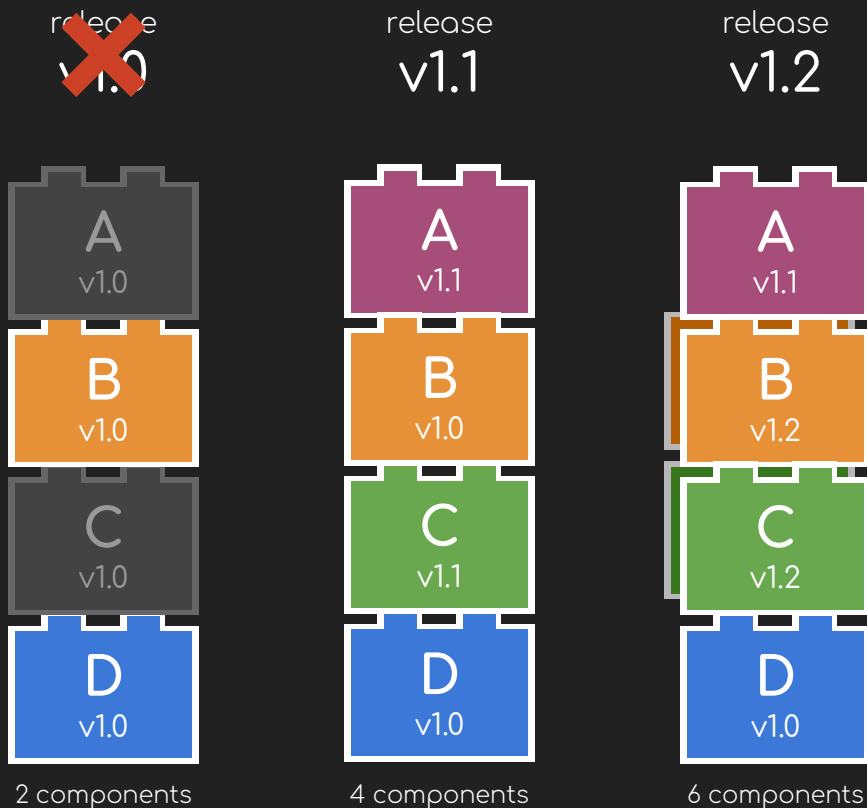
Compose by release





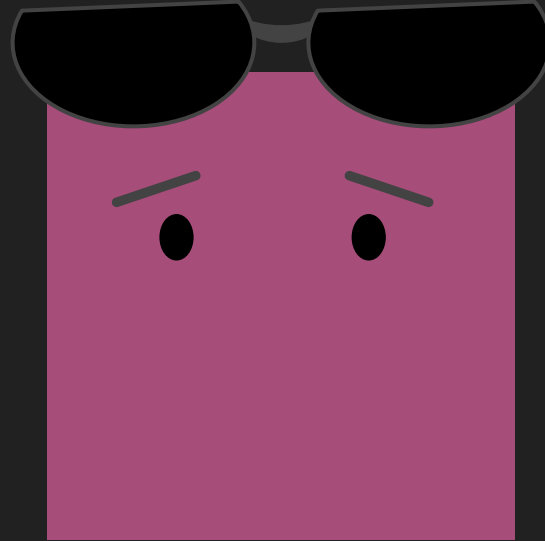
Stabilize by feature

Cleanup unused components



Looks great!
But, euhm...

Any ideas on the
implementation?

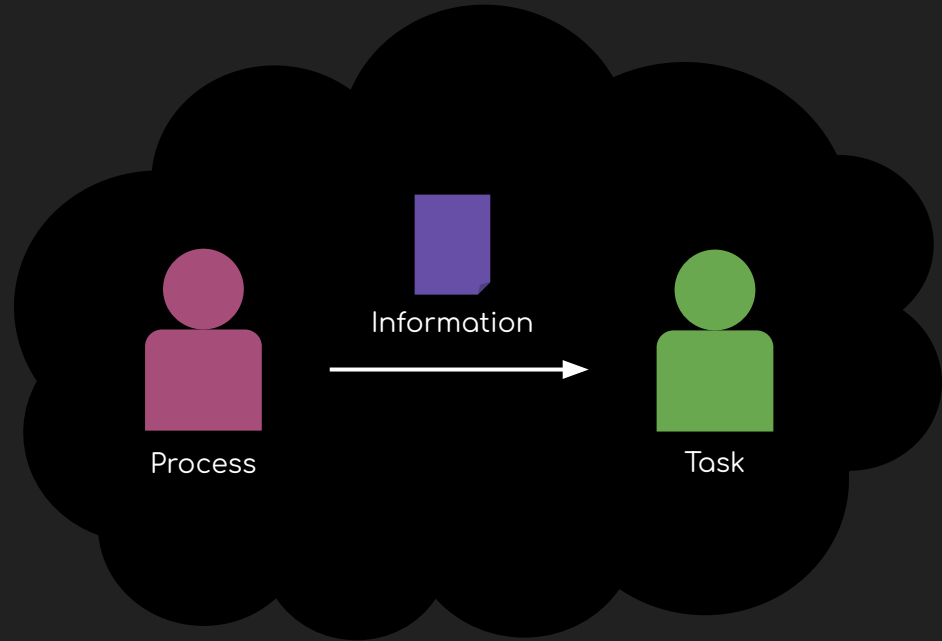


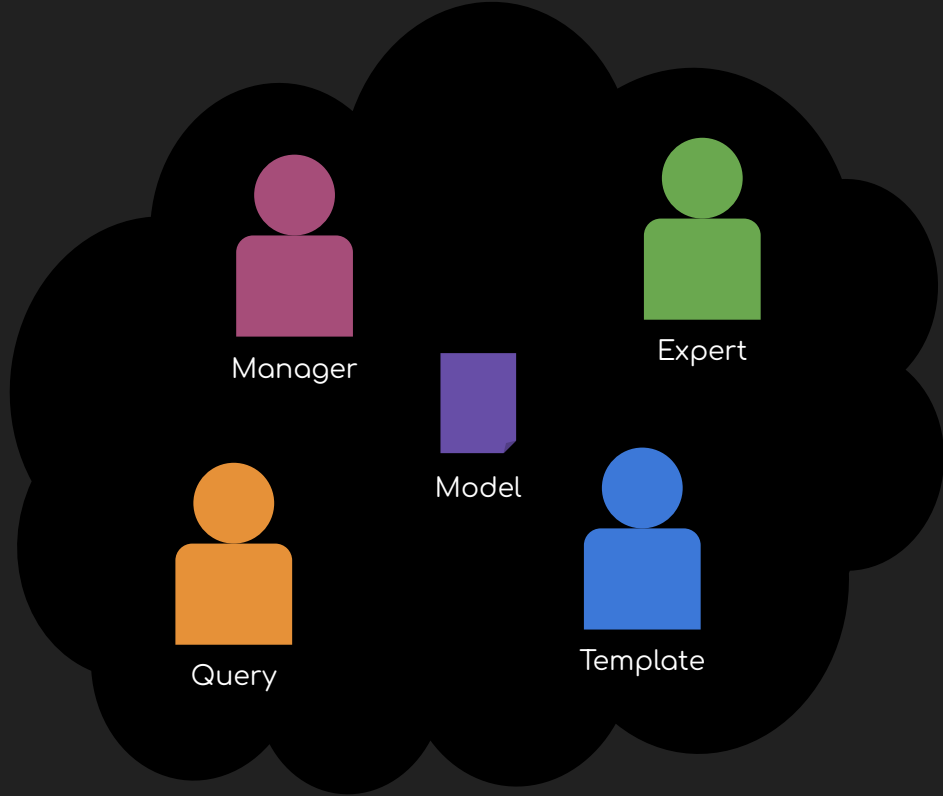
CHANGES

- Create version 1.1
- Update block A
- Create block C
- Create version 1.2
- Remove block E
- Update block C

Keep track
of changes

Separate concerns





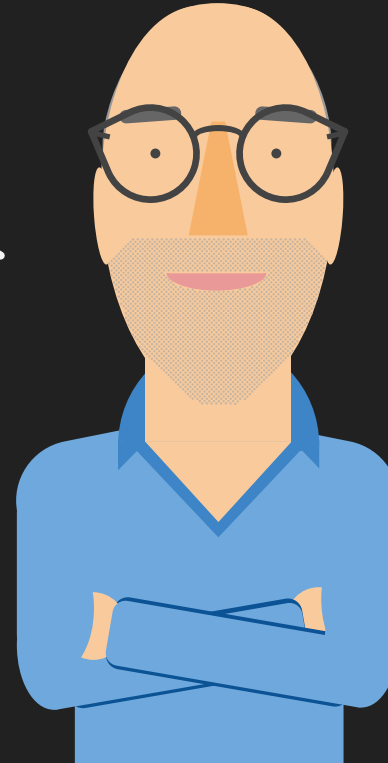
Predefine
building
blocks



ReciPie

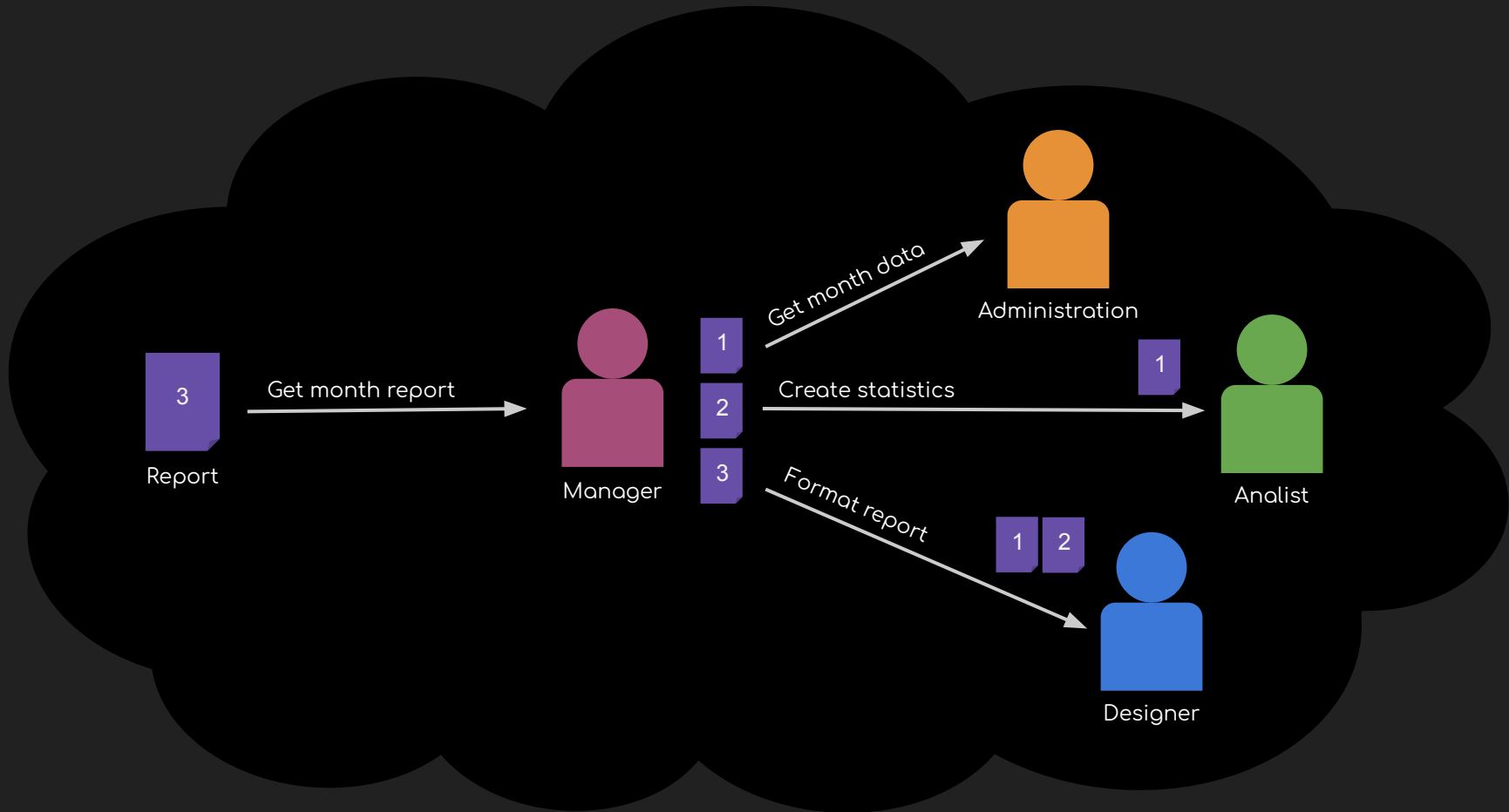
Homemade pie for everyone

As sales manager
I want a monthly
report of the
revenu per day.



Sales
manager





Get month report

manager

Process for creating the sales report for the given month.

v1.0.0 | Peter van Vliet

Stabilized



```
1 RUN 'Get month data' WITH year, month GET data;  
2 RUN 'Create statistics' WITH data GET statistics;  
3 RUN 'Format report' FROM dtp WITH data, statistics GET report;
```

Parameters

```
! year : string  
! month : string
```

Result

```
! report : string
```

Month data

model

Data structure for the month data.

v1.0.0 | Bas Meeuwissen

```
1 FIELD string AS year;  
2 FIELD month AS year;  
3 FIELD real AS numbers MULTIPLE;
```

Parameters

(none)

Get month data

query

Queries the database for sales data per month.

v1.0.0 | Bas Meeuwissen

```
1 SELECT 'Month data'  
2 FROM sales  
3 WHERE year = year AND month = month  
4 GET year, month, numbers
```

Parameters

```
! year : string  
! month : string
```

Create statistics

expert

Sums the numbers in the given data.

v1.0.0 | Bas Meeuwissen

```
1 let total = 0;  
2  
3 for (const number of data.numbers)  
4 {  
5     total += number;  
6 }  
7  
8 return total;
```

Parameters

! data : object

Result

! : real

Format report

template

Creates the graphical report in html.

v1.0.0 | Peter van Vliet

```
1 <h1>Sales {{ data.month }} {{ data.year }}</h1>
2 <table>
3   {{ REPEAT data.numbers AS number }}
4   <tr>
5     <td>{{ #number | inc }}</td>
6     <td>{{ number | usd }}</td>
7   </tr>
8   {{ DONE }}
9   <tr>
10    <td>Total</td>
11    <td>{{ statistics | usd }}</td>
12  </tr>
13 </table>
```

Parameters

! data : object



ReciPie

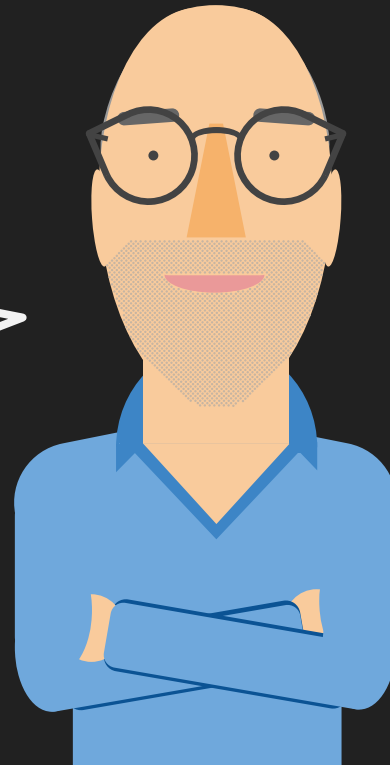
v1.00

Sales january 2021

Day	Revenue
1	\$ 7.50
2	\$ 22.50
3	\$ 30.00
4	\$ 15.00
5	\$ 22.50
Total	\$ 97.50

Lovely!

Where can I see
the number of
orders per day?



Month data

model

Data structure for the month data.

v1.1.0 | Bas Meeuwissen

```
1  FIELD string AS year;  
2  FIELD month AS year;  
3  ~FIELD real AS numbers MULTIPLE~  
4  OBJECT numbers MULTIPLE;  
5      FIELD string AS date;  
6      FIELD int AS orders;  
7      FIELD real AS revenue;  
8  DONE;
```

Parameters

(none)

Create statistics

expert

Sums the numbers in the given data.

v1.1.0 | Bas Meeuwissen

```
1 let total = 0;
2
3 for (const number of data.numbers)
4 {
5     total += number.revenue;
6 }
7
8 return total;
```

Parameters

! data : object

Result

! : real

Format report

template

Creates the graphical report in html.

v1.1.0 | Peter van Vliet

```
1 <h1>Sales {{ data.month }} {{ data.year }}</h1>
2 <table>
3   {{ REPEAT data.numbers AS number }}
4   <tr>
5     <td>{{ number.date }}</td>
6     <td>{{ number.orders }}</td>
7     <td>{{ number.revenue | usd }}</td>
8   </tr>
9   {{ DONE }}
10  <tr>
11    <td colspan="2">Total</td>
12    <td>{{ statistics | usd }}</td>
13  </tr>
14 </table>
```

Parameters

! data : object

Result

! : string



ReciPie

v1.1.0

Sales february 2021

Date	Orders	Revenue
Mon 1 february	5	\$ 37.50
Tue 2 february	3	\$ 22.50
Wed 3 february	6	\$ 45.00
Thu 4 february	4	\$ 30.00
Fri 5 february	8	\$ 60.00
Total		\$ 195.00



PERFECT!

Does this
also work for
January?

Get month report

manager

Process for creating the sales report for the given month.

v1.1.0 | Peter van Vliet

Stabilized



```
1 RUN 'Get month data' WITH year, month GET data COMPLY;  
2 RUN 'Create statistics' WITH data GET statistics;  
3 RUN 'Format report' FROM dtp WITH data, statistics GET report;
```

Parameters

```
! year : string  
! month : string
```



ReciPie

v1.1.0

Sales january 2021

Day	Revenue
1	\$ 7.50
2	\$ 22.50
3	\$ 30.00
4	\$ 15.00
5	\$ 22.50
Total	\$ 97.50

Sales february 2021

Date	Orders	Revenue
Mon 1 february	5	\$ 37.50
Tue 2 february	3	\$ 22.50
Wed 3 february	6	\$ 45.00
Thu 4 february	4	\$ 30.00
Fri 5 february	8	\$ 60.00
Total		\$ 195.00



Demo time!

Thanks!



<https://masking.tech>