

EVERYONE IS BUSY. THE WORK IS WAITING.

UTILISATION BY TEAM

95%

booked to work

Payments	17		97%
Onboarding	15		96%
Lending	16		94%
Mobile	14		93%
Platform	18		95%
Data	12		96%

TEN ITEMS, START TO FINISH

Working Waiting

82%

waiting

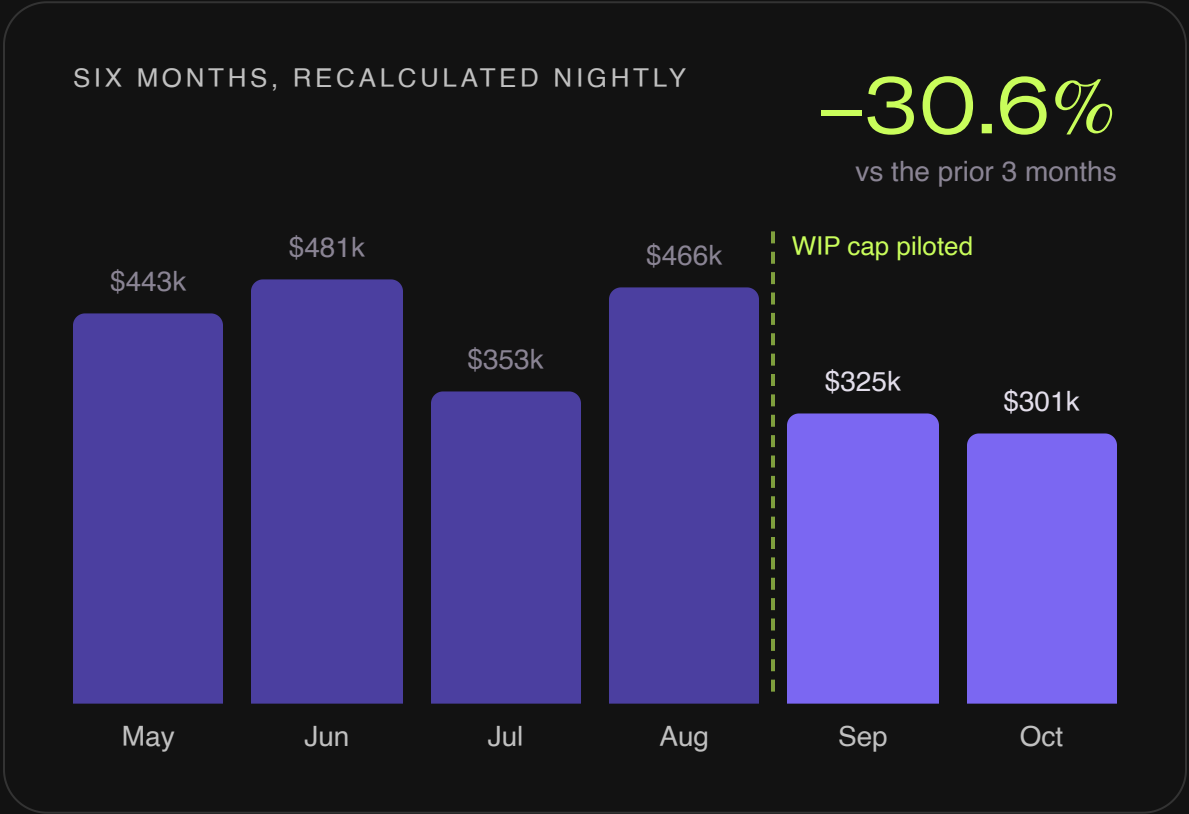
PAY-6599	
ONB-2210	
LEN-4471	
PLT-1893	
MOB-3302	
PAY-6641	
DAT-0917	
ONB-2268	
LEN-4520	
PAY-6702	

WAITING HAS A PRICE.

Delay and rework are tying up an estimated

\$300,978

▼ 7.5% against \$325,351 in September

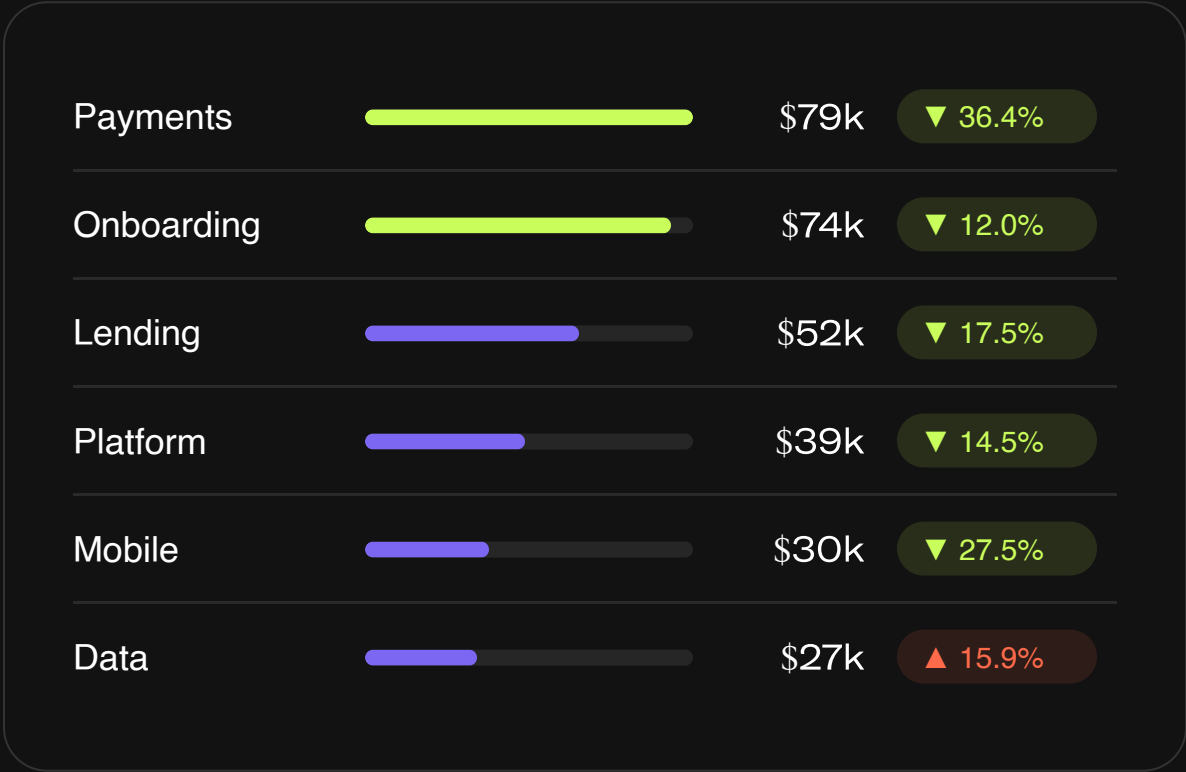


WHERE IT SITS

NOW YOU KNOW **WHERE.**

Two teams carry **51%** of the cost.

Payments runs 3.1 items in progress per engineer.



WHERE IT STARTS

IT STARTS IN THE BACKLOG.

WITH AGREED CRITERIA

7.5 days

305 items · \$129k

WITHOUT

16.0 days

125 items · \$172k

2.1x

longer when work starts poorly defined

WHAT TO DO

NUMBERS YOU CAN ARGUE WITH.

1

Put an acceptance-criteria gate on Payments and Onboarding

Refinement quality · low effort

\$586k
a year

2

Cap work in progress in Payments at two items per engineer

WIP discipline · low effort

\$80k
a year

3

Add contract tests on the direct debit retry path

Defect rework · medium effort

\$413k
a year

Together \$1.08m a year of modeled value at stake, not banked savings. Accept only 50% of the impact and it is still \$539k.

HOW IT WORKS

READS WHAT YOU ALREADY RUN. FIRST NUMBER IN WEEKS.

01

CONNECT

Read-only access to the tools your teams already use.

02

SEE

Your first number in three to four weeks.

03

ACT

Pick the changes that matter, then track whether the number moves.

SEE IT ON YOUR OWN DATA

A short demo, then a focused proof of concept on one area · octy.dev/try/demo

Dmitri Koleno
dmitri@octy.dev