Yandex Meetup · October 10, 2018

The Development Manager Career Path

From Junior Developer to CTO: steps and forks



Speaker

Alexander Polomodov

Development Manager, large fintech

What this talk is about

The map of steps from Junior to CTO, and the two forks where the profession changes, not the grade.

Path

From Junior to Development Manager

Today

Several teams and teamleads

What we’ll cover

Agenda Outline

- Steps: from Junior to CTO
- How Senior, Teamlead, CTO differ
- Skills and focus at each step

From specialist to manager



The profession changes, not just the grade

Step 0

00

Education

Lyceum → MIPT → choice between Science and Programming

Chose industry, not academia

Feedback comes back faster

- Physics-and-math school — olympiad training
- MIPT, FUPM — algorithms and CS
- Fork: research or industry
- Real problems instead of papers

Step 1

01

Individual Contributor

Junior, Analyst, Middle and Senior Developer

Depth and ownership grow together

IC track Timeline

Engineering skills and autonomy

Junior

Learns engineering practices

Middle

Works independently

Senior

Solves complex technical problems

Step 2

02

Teamlead

Team: code, people and process

Teamlead owns the team

A new coordinate system

- No longer the most productive
- 1:1s, reviews, hiring, motivation
- Owning the whole team’s code
- Hardest part: delegating and trusting

Responsibility Teamlead

Step 3

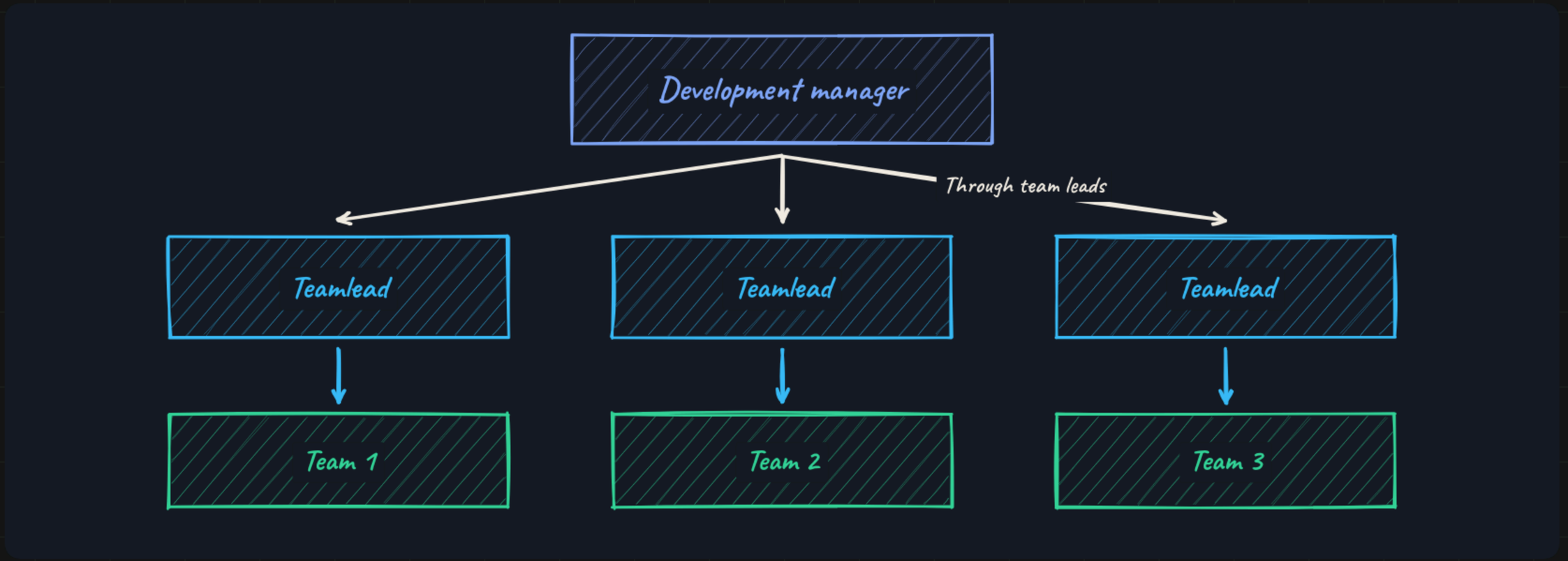
03

Development Manager

Three teams and three teamleads

Manage through teamleads

Manager of managers Dev Manager



Senior is not Teamlead

Fork `Senior ≠ Teamlead`

Technical skills vs people & process

S Senior

- Deep technical expertise
- Architecture and domain
- Complex technical problems

T Teamlead

- People management
- Process and expectations
- Team development

A great Senior ≠ a great Teamlead

Teamlead is not CTO

Fork Teamlead ≠ CTO

Team context vs strategy and resources

T Teamlead

- Context of one team
- Tactical execution
- Tasks here and now

C CTO

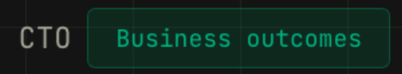
- Strategy and priorities
- Resource allocation
- Strategic outcomes

Different scopes of responsibility

A CTO is measured by business results

Business impact, not code

- Start from business goals
- Own the architecture choice
- Build teams and their processes
- Let people show what they can



Plan the route in advance

Takeaway: Context matters

- Startup/corporation — different speed
- Team size — different skills
- Interests: code / people / strategy

- Careers are path dependent — history matters

polomodov.tech

Thank you!

All slides and links are in the Telegram channel

Alexander Polomodov

Development Manager,
large fintech



@book_cube