

Navigating the five stages of a project to product transformation

Carmen DeArdo

Senior Director VSM R&D, Tasktop

Naomi Lurie

VP, Product Marketing, Tasktop



According to Gartner, Inc.

In a 2018 survey, 85% of participating respondents said their organization has adopted or plans to adopt (fully or partially) a product-centric model

Main drivers:

- Faster delivery
- Digital business transformation

In today's talk

- The 5 stages of the shift
- 7 dimensions to self-assess and re-orient yourself
- Early findings: state of the industry
- Practical tips for advancing through the stages

5 stages to this multi-year process



Operating primarily in a project-based model.

Teams are constantly being formed, stormed, etc.

Application integrity and dependencies are hard to manage.

Characterized by large batch sizes and long release cycles.

Typically, digital portfolios are the first to implement new approaches, such as long-lived cross-functional teams and the product manager role. Frequently coupled with the adoption of other lean and agile frameworks.

Expanding the mindset, but the principles are not evenly applied, particularly between internal and external products. Agile, DORA and engineering metrics are the primary measures of success. Funding is still fixed annually, and lead times exceed a quarter.

The product operating model is increasingly widespread across the entire technology portfolio. Most products have an independent path to production and can incorporate customer feedback within a quarter. Metrics evolve to emphasize business outcomes and flow.

Entire portfolio is organized in continuously funded cross-functional build-and-run teams. Product management discipline is strong. Customer feedback is incorporated within weeks. Strong culture of continuous improvement.

7 differences between project and product operating models

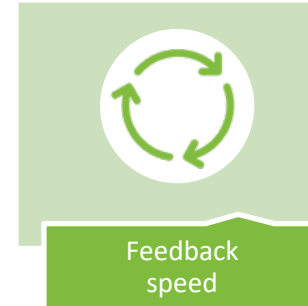
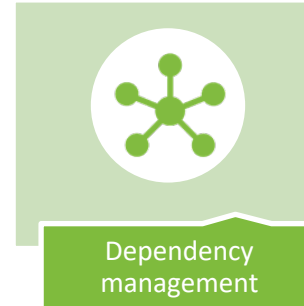
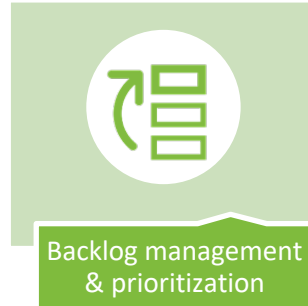


	Project-Oriented Management	Product-Oriented Management
Budgeting	Funding of milestones, pre-defined at project scoping. New budget requires creation of a new project.	Funding of product value streams based on business results. New budget allocation based on demand. Incentive to deliver incremental results.
Time Frames	Term of the project (e.g., one year). Defined end date. Not focused on the maintenance/health after the project ends.	Life cycle of the product (multiple years), includes ongoing health/maintenance activities through end of life.
Success	Cost center approach. Measured to being on time and on budget. Capitalization of development results in large projects. Business incentivised to ask for everything they might need up front.	Profit center approach. Measured in business objectives and outcomes met (e.g., revenue). Focus on incremental value delivery and regular checkpoint.
Risk	Delivery risks, such as product/market fit, is maximized by forcing all learning, specification, and strategic decision making to occur up front.	Risk is spread across the time frame and iterations of the project. This creates option value, such as terminating the project if delivery assumptions were incorrect or pivoting if strategic opportunities arise.
Teams	Bring people to the work: allocated up front, people often span multiple projects, frequent churn and re-assignment.	Bring work to the people: stable, incrementally adjusted, cross-functional teams assigned to one value stream.
Prioritization	PPM and project plan driven. Focus on requirements delivery. Projects drive waterfall orientation.	Roadmap and hypothesis testing driven. Focus on feature and business value delivery. Products drive Agile orientation.
Visibility	IT is a black box. PMOs create complex mapping and obscurity.	Direct mapping to business outcomes, enabling transparency.

Table 2.1: Project-Oriented Management vs. Product-Oriented Management

Project to Product, Mik Kersten, IT Revolution, 2018.

7 dimensions to practically assess where you are and plan what's next



Project to Product

How are teams organized and resourced?

- Funding projects
- People brought to the work
- Build and run are separate

- Funding products (external, internal and platform)
- Work is brought to the people
- Stable, long-lived, build-and-run teams

How strong is customer-centricity?

- Loose affinity with customers
- A shallow understanding of customer needs
- No feedback loops

- Good understanding of customers and their value
- Programmatic mechanisms to gather, respond and react to customer feedback

How clear is the definition of value?

- Emphasis is on being on time and on-budget
- Most people don't understand how they contribute to the business

- Emphasis is on achieving clear business outcomes
- Value is a focal point for everyone on the teams
- Focus is on incremental value delivery

How are backlogs managed and prioritized?

- Project plan drives priorities
- Long-term implications (like tech debt) are not considered
- Gantt charts drive waterfall-like behavior

- The product manager drives product vision and roadmap; cares about long-term viability
- Business imperatives drive roadmap, tradeoffs, and prioritization

How are dependencies managed?

- Execution path involves SLAs with functional areas (QA, Security)
- Heavy coordination required

- Dependencies are consumed as self-service capabilities
- Handoffs and coordination is minimal

How rapidly can feedback revise planning?

- Highly governed release windows
- Difficult to apply principles of feedback and continual learning

- Products have automated independent paths to production
- Feedback can be incorporated within weeks

How are delivery teams measured?

- Siloed metrics (Agile, engineering, QA, DORA)

- Leading metrics: Flow Metrics measure across horizontal delivery processes
- Lagging metrics: Business outcomes

Join the movement!

Take the Project to Product Maturity Assessment

- 5 minutes: 7 easy multiple-choice questions
- Find out your stage, 1–5
- Get recommendations on how to advance
- Get an eBook that details the entire journey to maturity

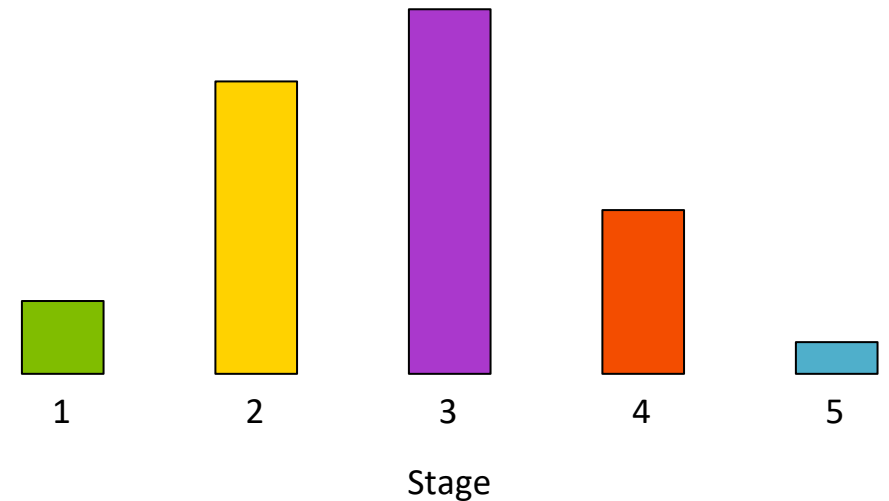


**How is the
shift going?**



How is the shift going?

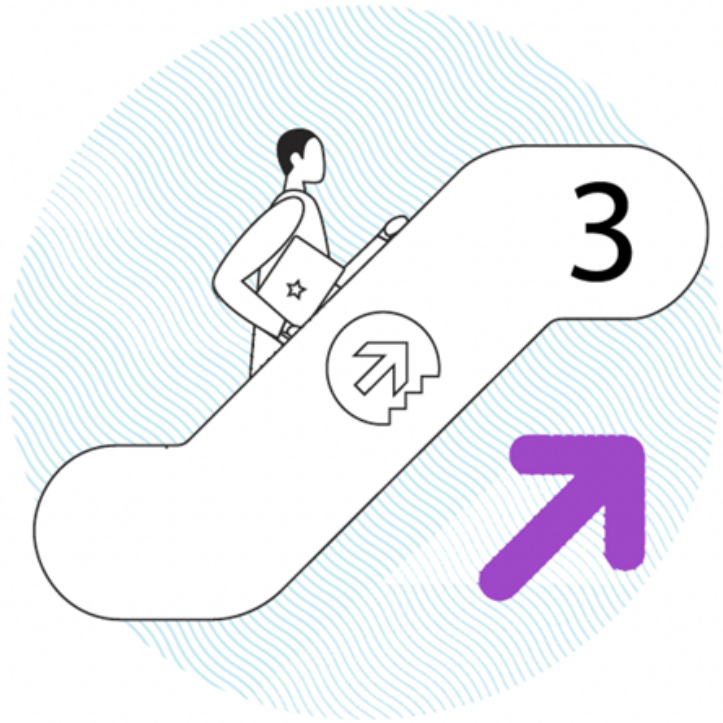
- Early findings show the majority of organizations (78%) have not yet fully operationalized the shift, where the true ROI is captured
- Two hypotheses:
 - Expansion is proving difficult
 - Expansion simply takes a long time



Tips for Experimentation

1. Develop success criteria
2. Carve out one experience or product to start the journey
3. Carve out some money for a cross-functional build-and-run team
4. Include a product manager/owner
5. Measure your baseline from the customer's perspective
6. Follow your process and product dependencies to investigate your constraints
7. Experiment to improve speed and throughput
8. Make the improvement itself work visible

Don't rush into Expansion



1. Maximize learnings and don't begin expansion before you've developed good patterns
2. Go deep: apply product patterns to internal and platform products
3. Carefully select expansion to areas with supportive IT and business leaders
4. Use stories to drive culture change across portfolios
5. Be open to continuous learning as you go

Empower the Product Owner and the product teams



1. Even in annual funding, strategy is revisited quarterly to adjust funding levels across products, justified by targeted outcomes
2. Empower the Product Owner to prioritize work that reduces time-to-market or secures long-term velocity
3. Extend boards to the left of the dev team's backlog
4. Embed business roles as part of the team
5. Teams are full-stack to ease planning and resourcing

Anti-patterns to avoid

1. Starting with big re-orgs
2. Having no clear metrics to measure success or progress
3. Time-boxing
4. Setting conflicting OKRs and incentives
5. Ignoring internal and platform products
6. Not considering the human element

You can technically do
all the right things...
and still lose the people



6 tips to manage the culture change

1. Not just for the cool kids:
demonstrate you are invested
in all your associates
2. Prepare for fixed mindsets:
make the journey as non-
threatening as possible
3. Empower teams: give license to
creativity and experimentation,
reward the effort — not just
results

6 tips to manage the culture change

1. Not just for the cool kids: demonstrate you are invested in all your associates
2. Prepare for fixed mindsets: make the journey as non-threatening as possible
3. Empower teams: give license to creativity and experimentation, reward the effort — not just results
4. Remove the red tape and slow approval cycles
5. Don't dismiss good ideas; create a backlog of ideas and experiments for when the environment is ripe
6. Forget the big bang! Appreciate the cumulative baby steps of progress

Join the movement!

Take the Project to Product Maturity Assessment

- 5 minutes: 7 easy multiple-choice questions
- Find out your stage, 1–5
- Get recommendations on how to advance
- Get an eBook that details the entire journey to maturity

