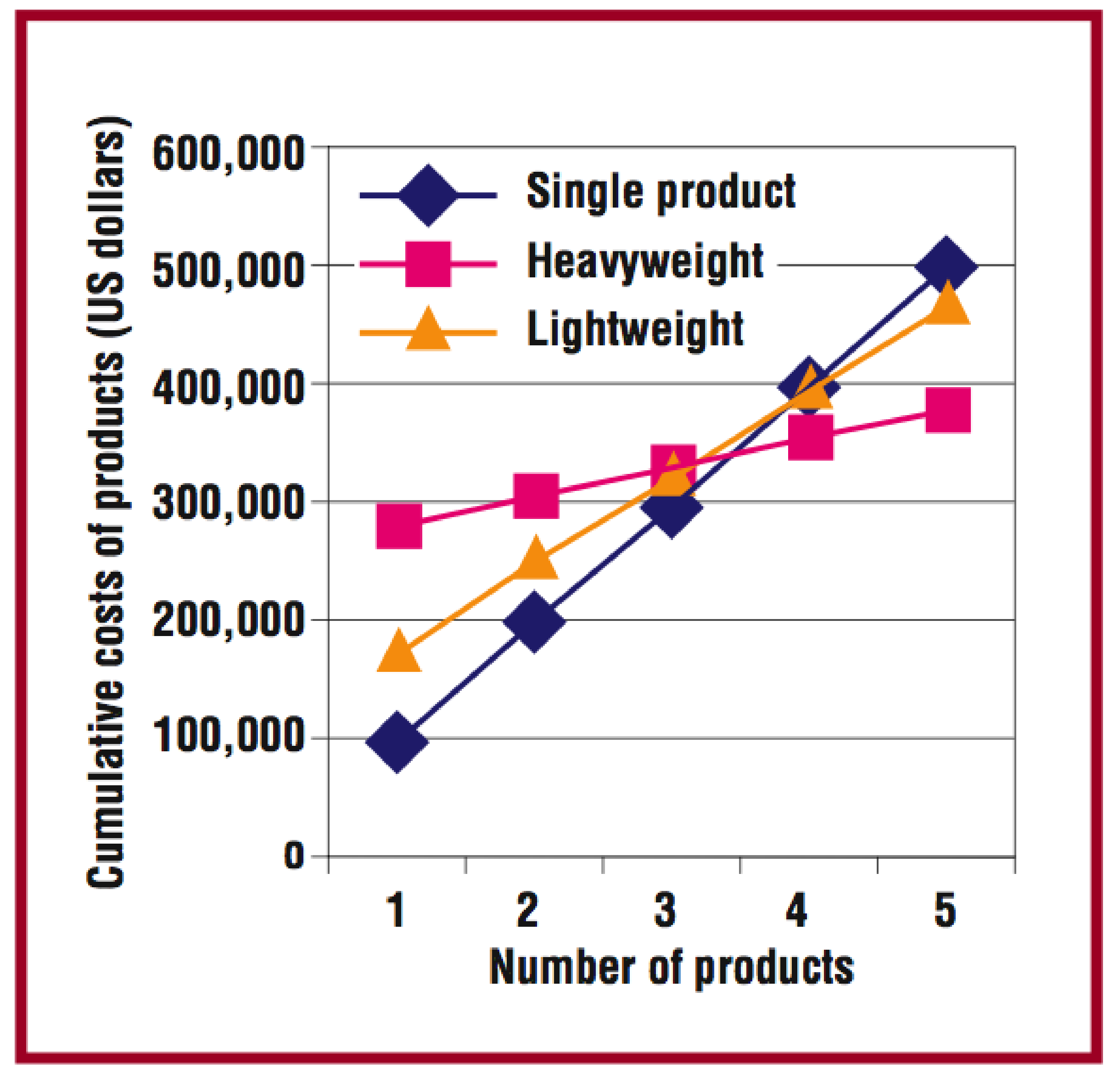
Software Product Line Engineering

L8: Transitioning to SPL

Transitioning/Adopting SPLs

- If we decide to adopt SPLs and transition to SPLE, HOW should we make the transition?
  - If? - PLPA
  - How? - Transition strategies



Incremental Adoption: Engenio case

- Engenio
  - High-performance storage servers
  - Customers: IBM, SGI, Cray, StorageTek, Teradata
  - Customers utilize E. core competence + wants unique features
  - Controller Firmware Dev team
    - Firmware for 82 products
    - ~1 Million LOC per product
    - 80% of code is common between products

Incremental Adoption: Engenio case

- Challenges:
    - Contractually dictated production schedules
    - Business demand outpaced maintenance ability
    - From sequential releases to intertwined/overlapping release cycles
    - Product diversification: low-end hardware platform
    - Variability through CM: 34% of 3300 src files had 2-16 branches
    - SPL adoption barrier: 2.5 products eq. upfront investm.
    - => 900-1350 personmonths, 100 persons

Incremental Adoption: Engenio case

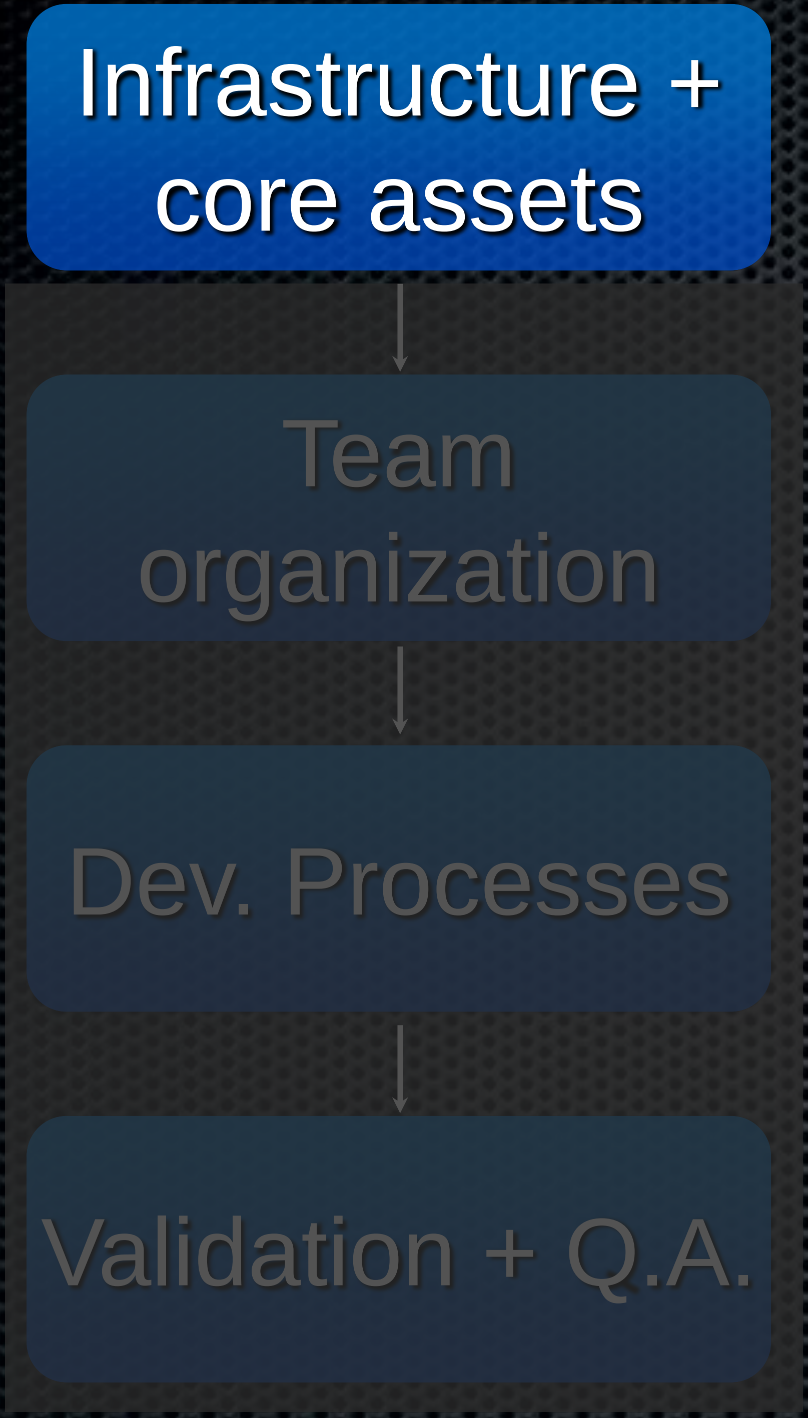
- Solution: Incremental investments
  - 4 personmonth upfront investment => cumulative returns outpaced cumulative investments
  - Focus on current bottlenecks/inefficiencies
    - Excessive File Branching due to Multiple Product-focus was root causes
  - Upfront investment
    - Pilot study using SPL support tool, here BigLever Software Gears
    - 2 existing prods. remodeled => convinced mngmnt.
  - Small incremental SPL investments, no schedule disruptions

SPL Support Tool

- Feature Modeling Language
  - model optional and varying features between products
- Product Feature Profile
  - instantiates feature model for each product
- Configurable SW Assets via Variation Points
  - language for programming variation points
  - v.p.’s configures themselves based on feature profile
- Configurator
  - compiler from (feature profile, assets) -> product

Incremental Adoption: Engenio case

4-stage Transition:



Infrastructure + core assets

Team organization

Dev. Processes

Validation + Q.A.

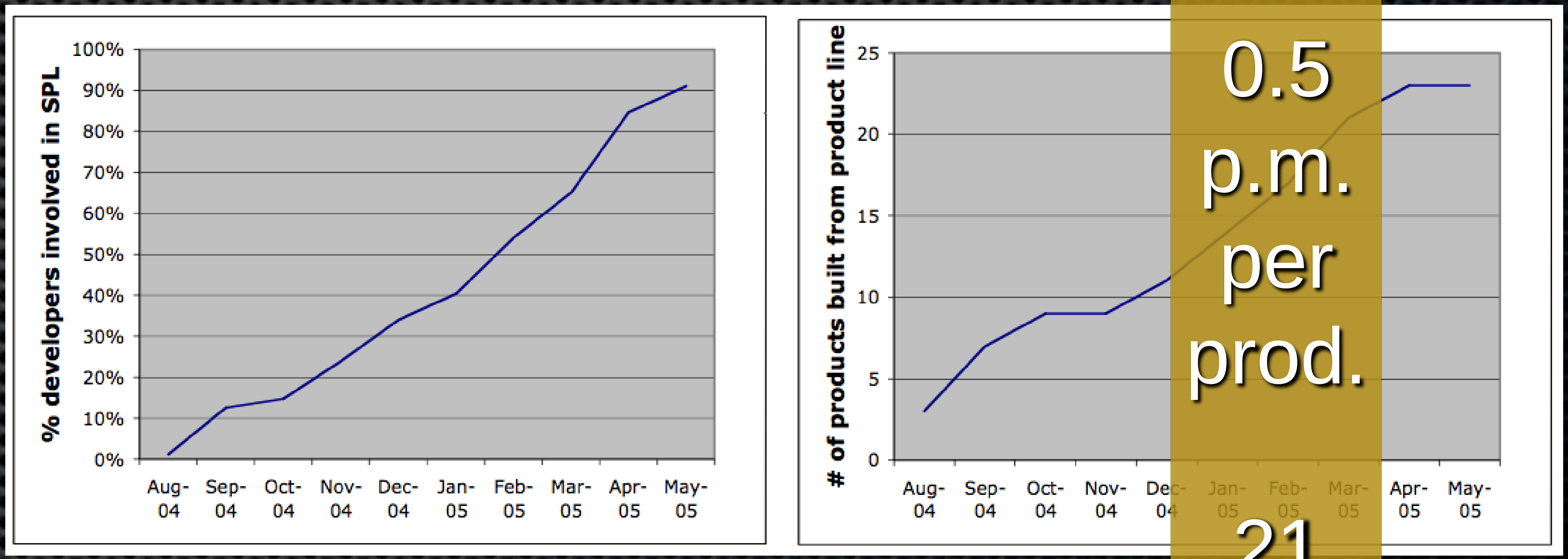
Setup SPL infrastructure

Extract core assets from branches

4 p.m.

2 products in 3300 files -> 3103 files + 51 v.p. files

Transition teams from branching



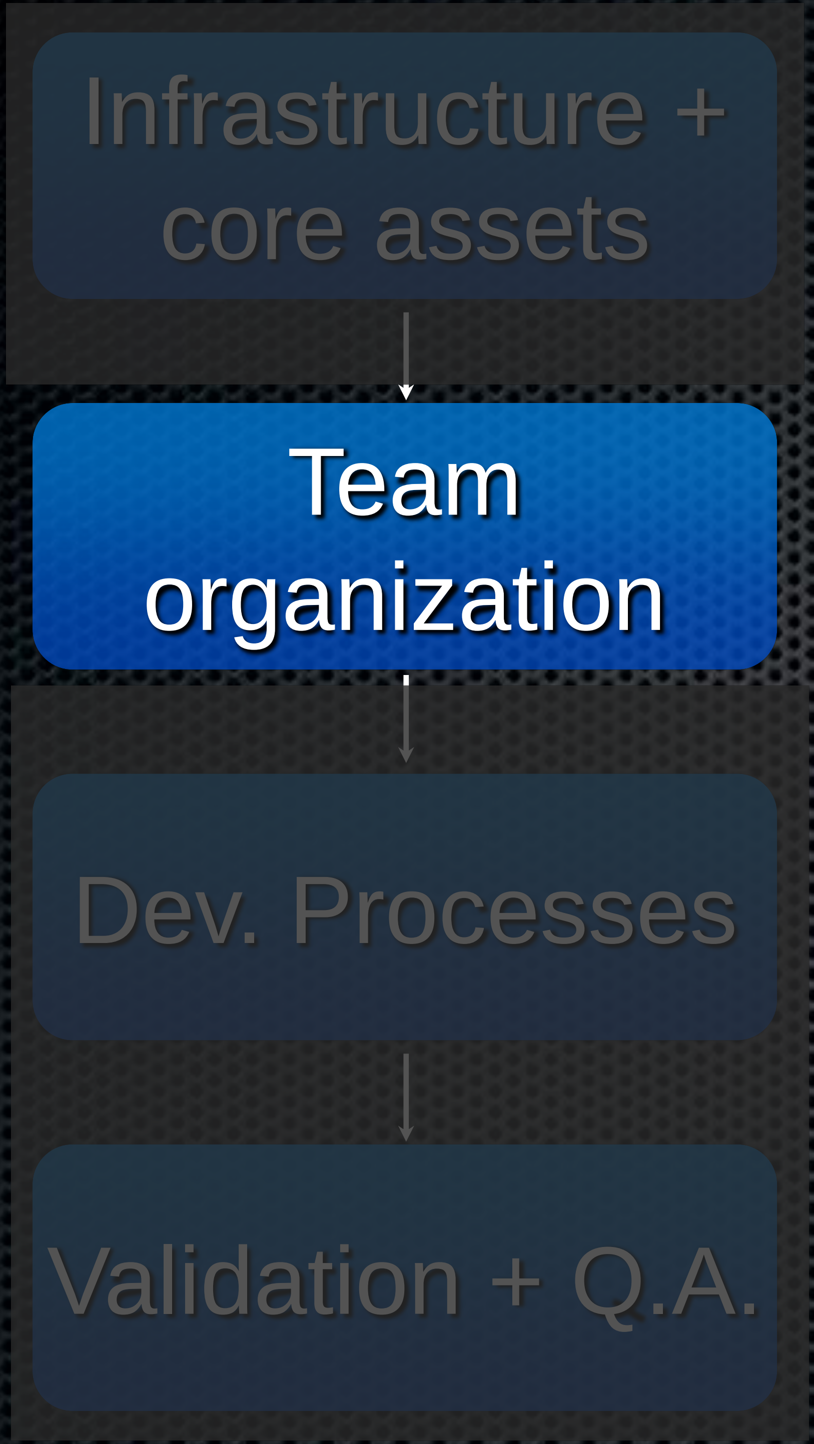
0.5 p.m. per prod.

21 prod.

Refactor core to optimize commonality and variation points

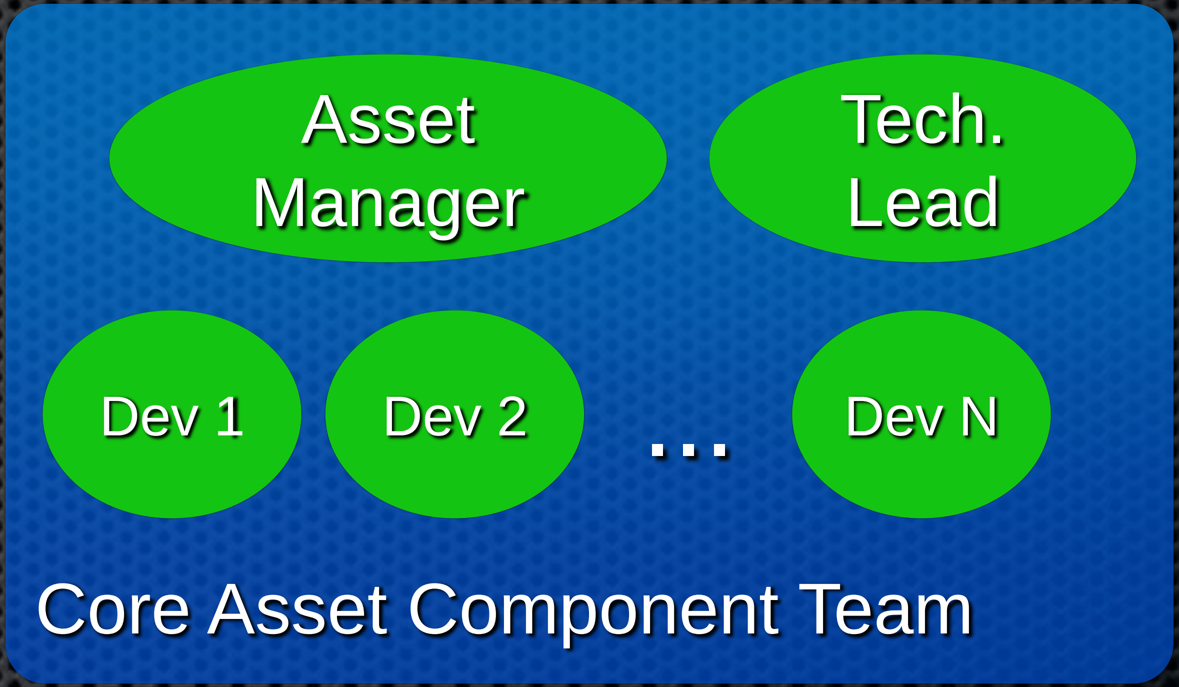
Incremental Adoption: Engenio case

4-stage Transition:



From product teams to core asset component teams

Assets grouped by service layers in architecture

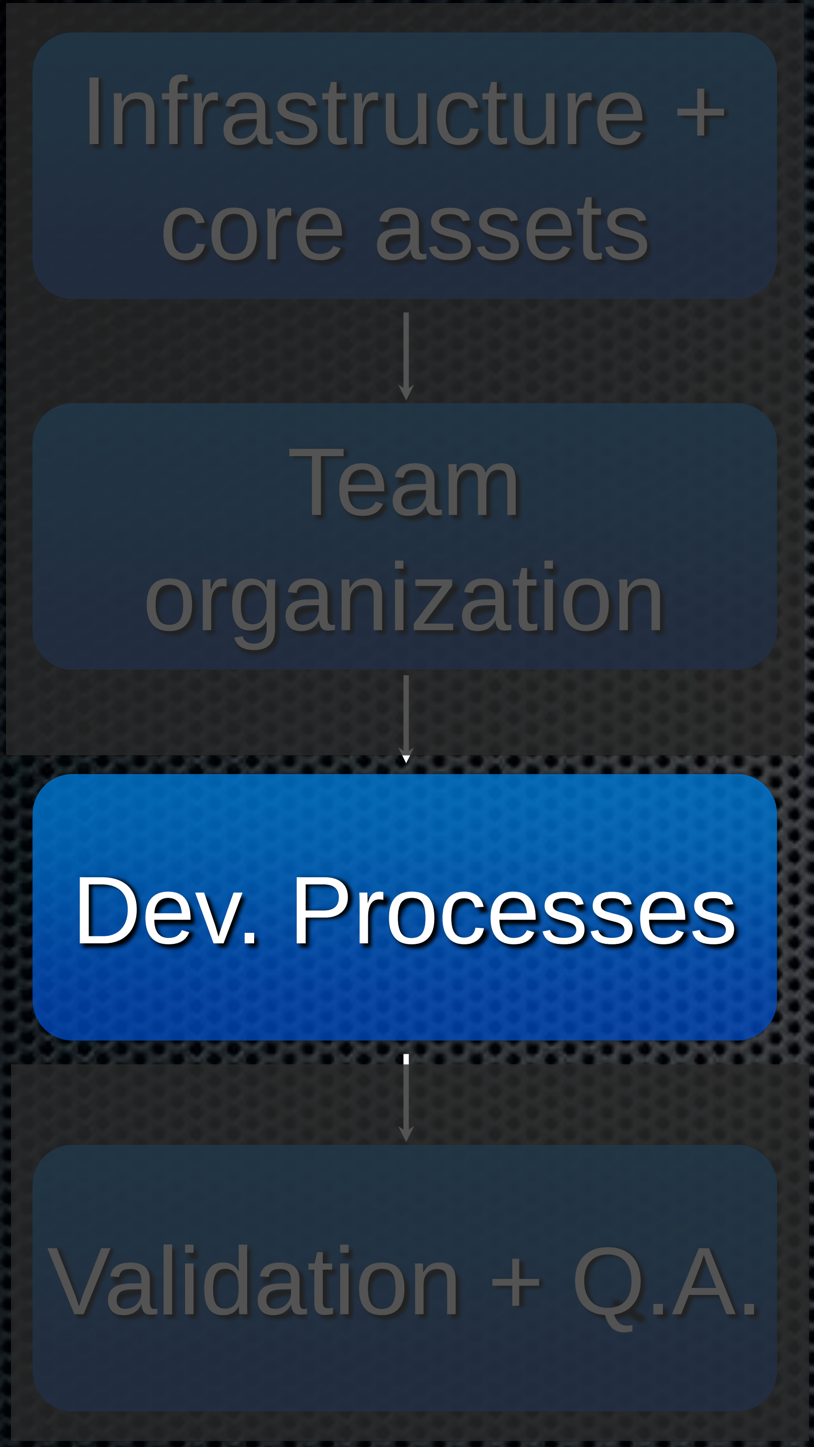


Planning to define teams

Educating team members

Incremental Adoption: Engenio case

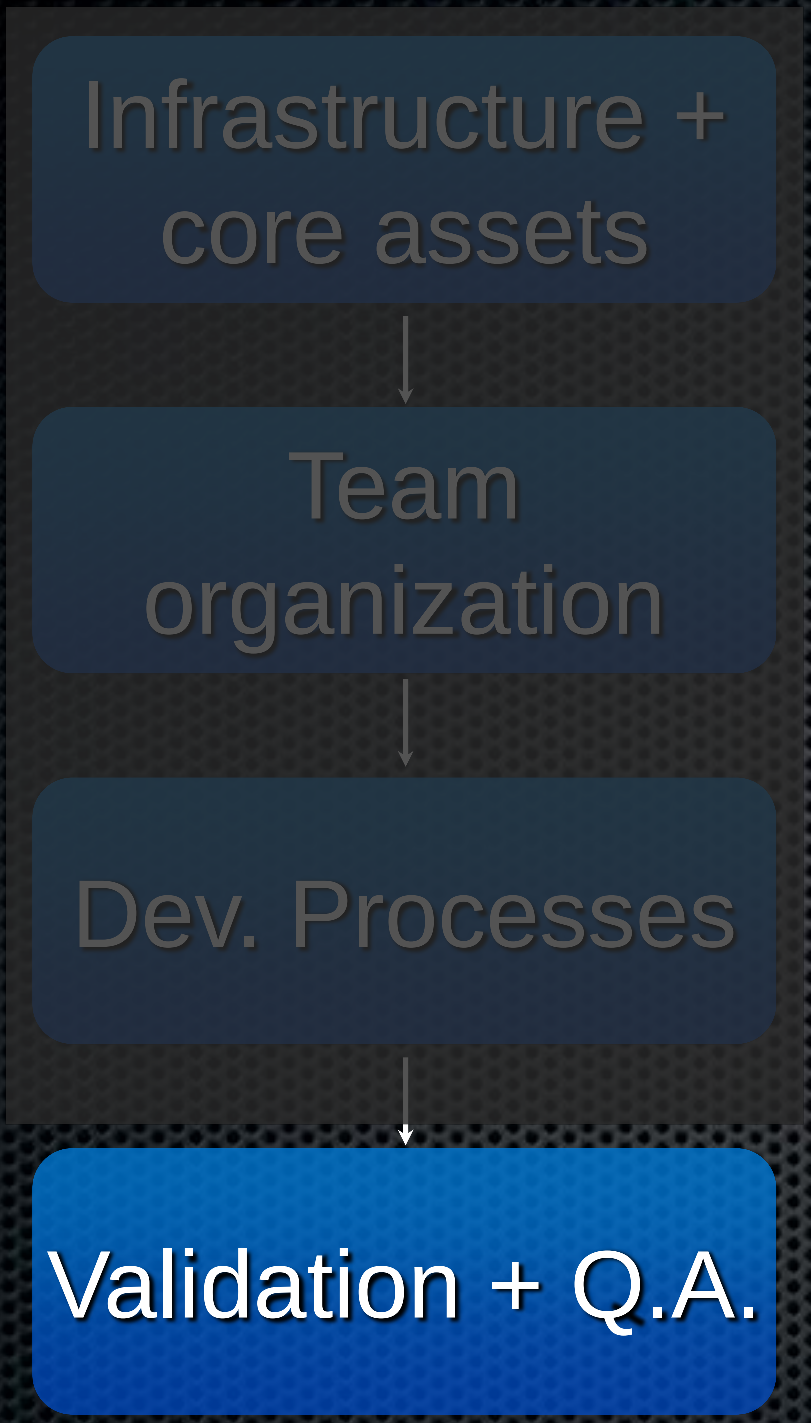
4-stage Transition:



- From Release centric to SW Product Family centric
- Assemble Process Task Force
- Add mapping step from feature reqs to asset reqs
- Same time: Better respond to changing customer reqs

Incremental Adoption: Engenio case

4-stage Transition:



Infrastructure + core assets

Team organization

Dev. Processes

Validation + Q.A.

Iterative Feature req validation

Shift responsibility of certification groups

Incremental Adoption: Engenio case

- Investments:
  - 4 personmonths upfront
  - 12 personmonths total
- Outcomes:
  - 23 products of 1MLOC each, and 135 developers shifted to SPL
  - Increased quality and productivity
  - After first 3 transition stages, they expanded from 23 to 52 products in 5 months
  - By incrementally showing benefit, easier to convince people to actually change, harder for detractors

Four Transition Strategies

- Big Bang introduction Overall cost lower | Plan guides work | Core assets earlier | Harder to undo
  - SPLE for new, key products “at once” Higher costs/time | Stops other dev
- Incremental introduction Change/stop unless good | Limited costs/time
  - Start small then expand in increments Current dev continues | Rework/change
  - Expand: Organisational scope || Investments More time
- Tactical approach Focus on urgent needs | Start w. small team | Wrong direction
  - Partial adoption, driven by technical problems Low start cost | Lack support
- Pilot project strategy Current dev continues | Limited costs/time | More time
  - Develop new product partly via SPL Change/stop unless good | Rework
  - First SPL product || Extension of related, existing prods || Toy product || Prototyping

Successful SPL Adoption

- Decide
  - 1. Define Business Strategy and Vision
  - 2. Learn about SPLE
  - 3. Perform a risk analysis in company context
- Prepare
  - 4. Identify stakeholders & Gain support for new ways of working
  - 5. Set concrete goals for the transition & Create stakeholder business cases
  - 6. Scope the PL to determine boundaries and content
  - 7. Evaluate orgs status and ability to adopt new ways
  - 8. Plan the transition, create adoption plan
- Transition
  - 9. Launch and institutionalize

Business Strategy & Vision (Decide)

- Strategy
  - How is the world changing? How will it affect the company?
  - Customer needs that we cannot support? New markets or segments? etc.
- Vision statement

WHAT!

NOT How!

Pulse-Eco Benefit/Risk analysis (Decide)

- Consider benefits and risks for dimensions:
  - Domain Maturity - sufficient domain understanding?
  - Stability - requirements/market change speed?
  - Resource constraints - Money, Time, Experts/Knowledge
  - Organizational constraints - Cultural differences etc.
  - Market potential - internal (assets used?) + external (enough customers?)
  - Sufficient Commonality & Systematic Variability?
  - Coupling & Cohesion - higher coupling => harder to reuse
  - Existing assets?

Successful SPL Adoption

- Decide
  - 1. Define Business Strategy and Vision
  - 2. Learn about SPLE
  - 3. Perform a risk analysis in company context
- Prepare
  - 4. Identify stakeholders & Gain support for new ways of working
  - 5. Set concrete goals for the transition & Create stakeholder business cases
  - 6. Scope the PL to determine boundaries and content
  - 7. Evaluate orgs status and ability to adopt new ways
  - 8. Plan the transition, create adoption plan
- Transition
  - 9. Launch and institutionalize

Stakeholder Business Case (Prepare)

- Stakeholder: Product Manager
- Current state: Variation supported with file branching. Redundant work between similar customer products.
- Stakeholder goals:
  - Increase revenue, profit, market coverage, quality, time-to-market
- SPL Goal Achievement metrics:
  - Connects goals to SPL. How does SPL help reach goal? Metrics that compare to current situation/single-system dev? Connect to costs?
  - Decrease TTM <- Fewer file branches <- Feature models + V.P. (metric: Average branches per file, # of feature models, # V.P.)
- Deliverables, Resources, Workload

Successful SPL Adoption

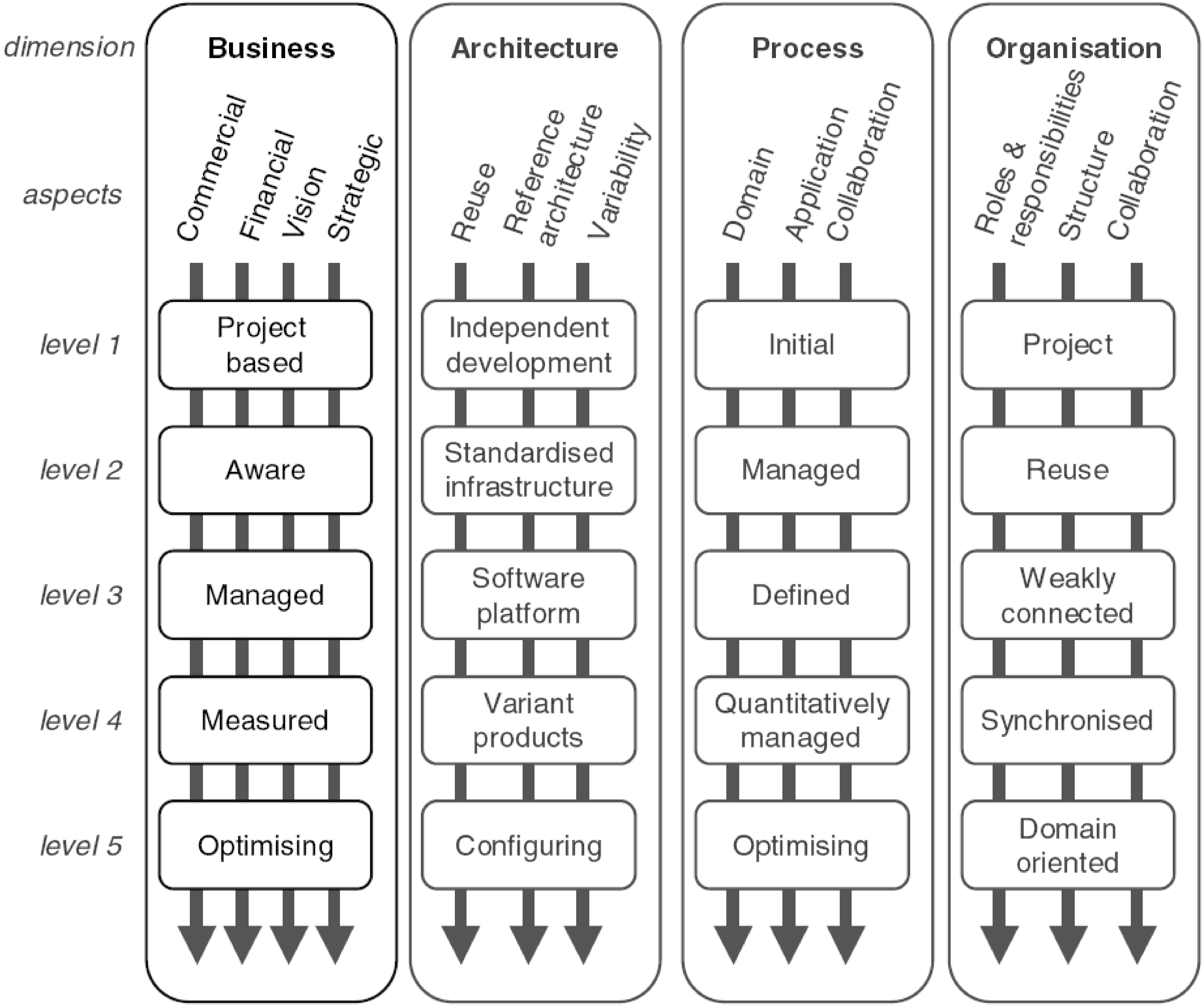
- Decide
  - 1. Define Business Strategy and Vision
  - 2. Learn about SPLE
  - 3. Perform a risk analysis in company context
- Prepare
  - 4. Identify stakeholders & Gain support for new ways of working
  - 5. Set concrete goals for the transition & Create stakeholder business cases
  - 6. Scope the PL to determine boundaries and content
  - 7. Evaluate orgs status and ability to adopt new ways
  - 8. Plan the transition, create adoption plan
- Transition
  - 9. Launch and institutionalize

Launching SPLE (Transition)

- Example, market maker
  - Hired new dev that started SPL dev
  - Close integration with rest, Existing assets to use
  - Firm time deadline to focus
- Exampe, Phillips Consumer Electronics
  - 3 years to set up
  - Two lead products on SPL: high visibility, low risk
  - Then roll out to other products

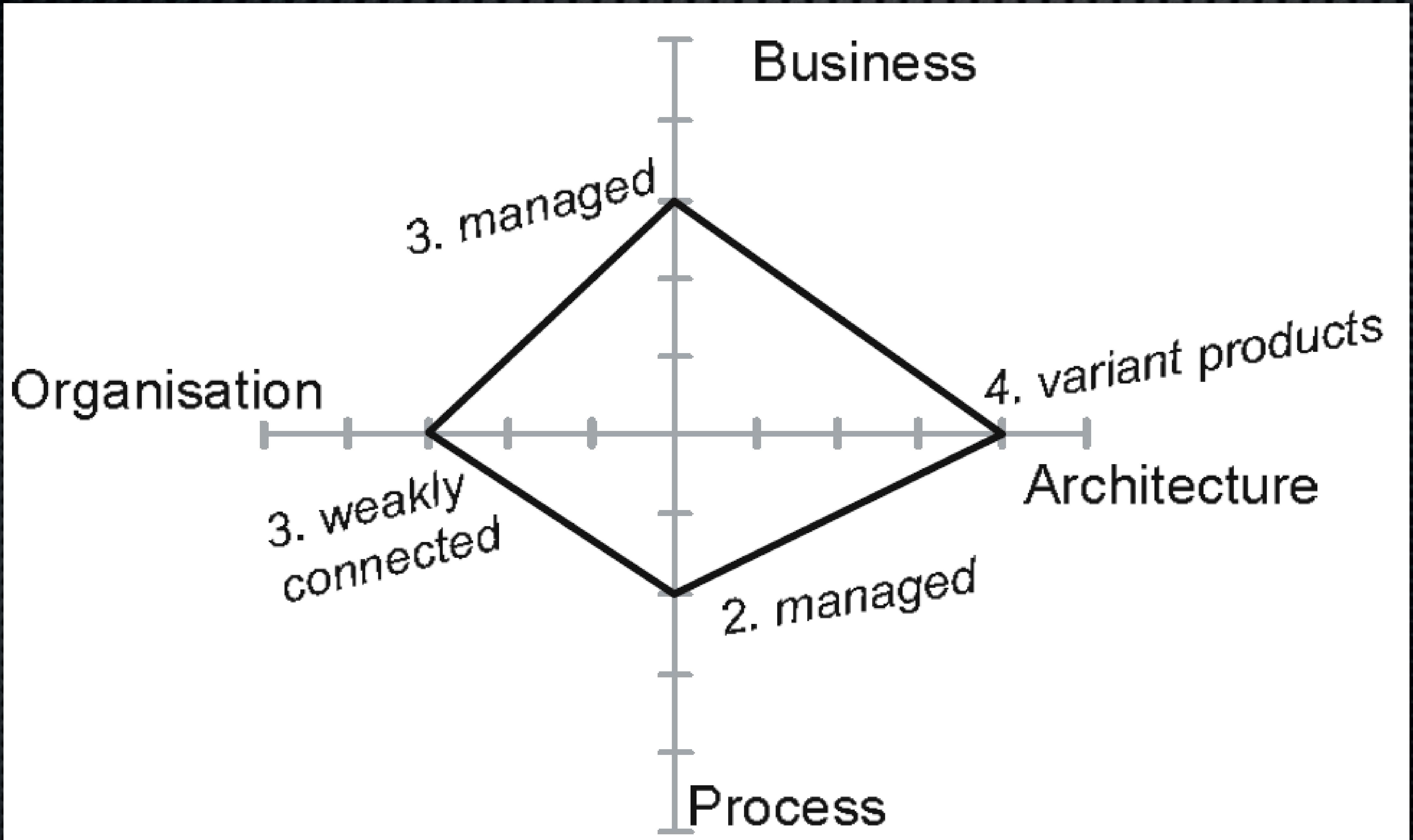
BAPO, PLPA

- Process assessment

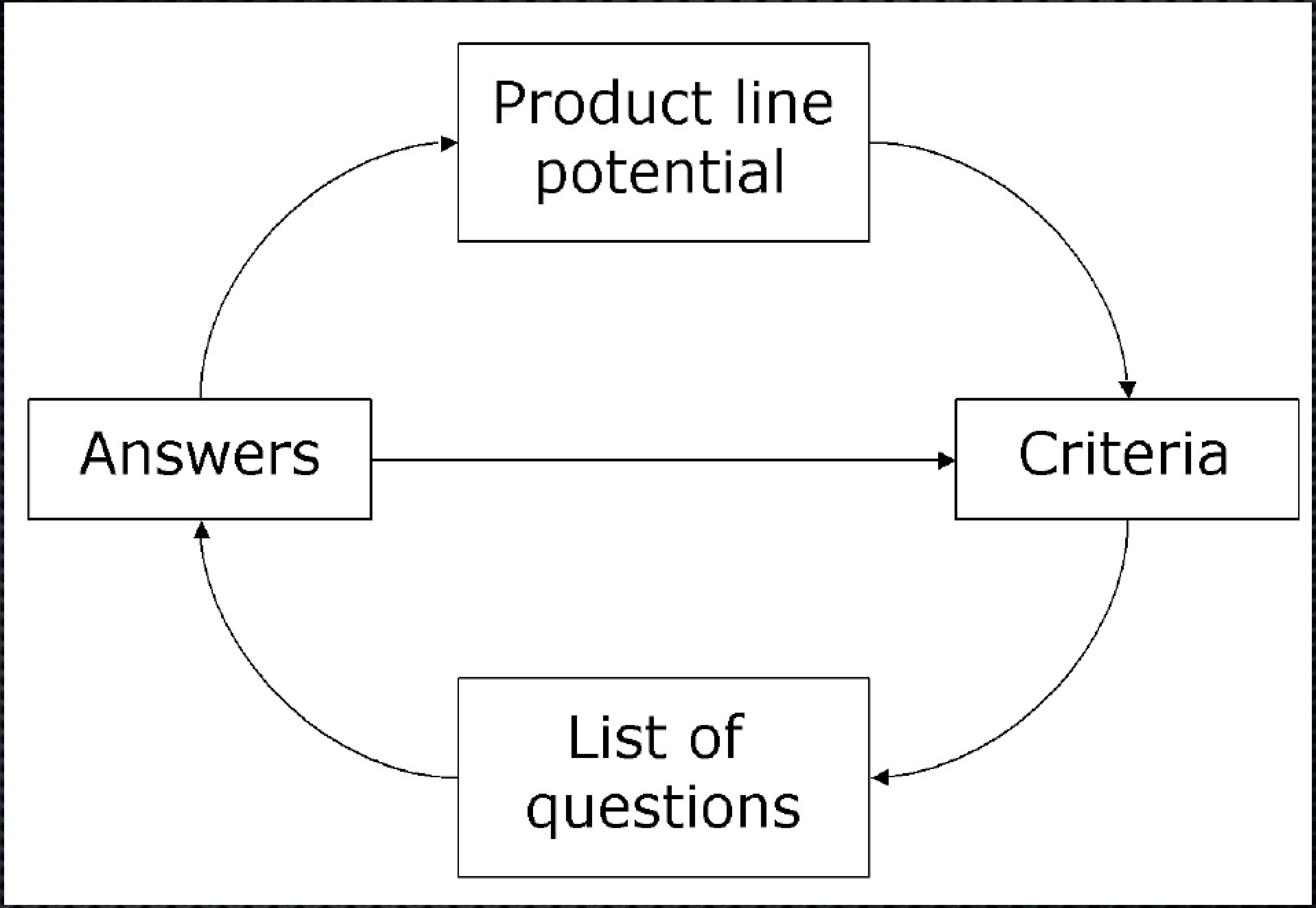


dimension	Business	Architecture	Process	Organisation
aspects	Commercial Financial Vision Strategic	Reuse Reference architecture Variability	Domain Application Collaboration	Roles & responsibilities Structure Collaboration
level 1	Project based	Independent development	Initial	Project
level 2	Aware	Standardised infrastructure	Managed	Reuse
level 3	Managed	Software platform	Defined	Weakly connected
level 4	Measured	Variant products	Quantitatively managed	Synchronised
level 5	Optimising	Configuring	Optimising	Domain oriented

BAPO


Business
3. managed
Organisation
4. variant products
Architecture
3. weakly connected
2. managed
Process

PLPA



Product line potential
Criteria
List of questions
Answers

PLPA

- **Main criteria** are essential for product line development and have to be fulfilled:
  - The business unit develops more than one product.
  - Products have common features.
  - Products have common qualities.
- **Inclusion criteria** indicate that product lines already exist:
  - The same part of software is used in more than one product.
- **Supporting criteria** apply if a business unit has problems that the PLA addresses:
  - The business unit has quality problems.
  - The business unit has complexity problems.
  - The business unit expects increasingly differentiated products.
- **Exclusion criteria** rule out an economically advantageous product line:
  - There is an immature, instable market for the products.
  - There is technological change.
  - The software is small; optimization will not be profitable.
  - The software development effort is negligible. It would be better to focus on other improvements.
  - New product development is too seldom.
  - The business unit develops specific, commissioned custom products.
- **Additional information** is useful data that cannot be assigned to one of the preceding criteria:
  - the competitive situation