Project Management: Software product transfers

REF: D. Smite and C. Wohlin, "Strategies Facilitating Software Product Transfers", IEEE Software, Vol. 28, No. 5, pp. 60-66

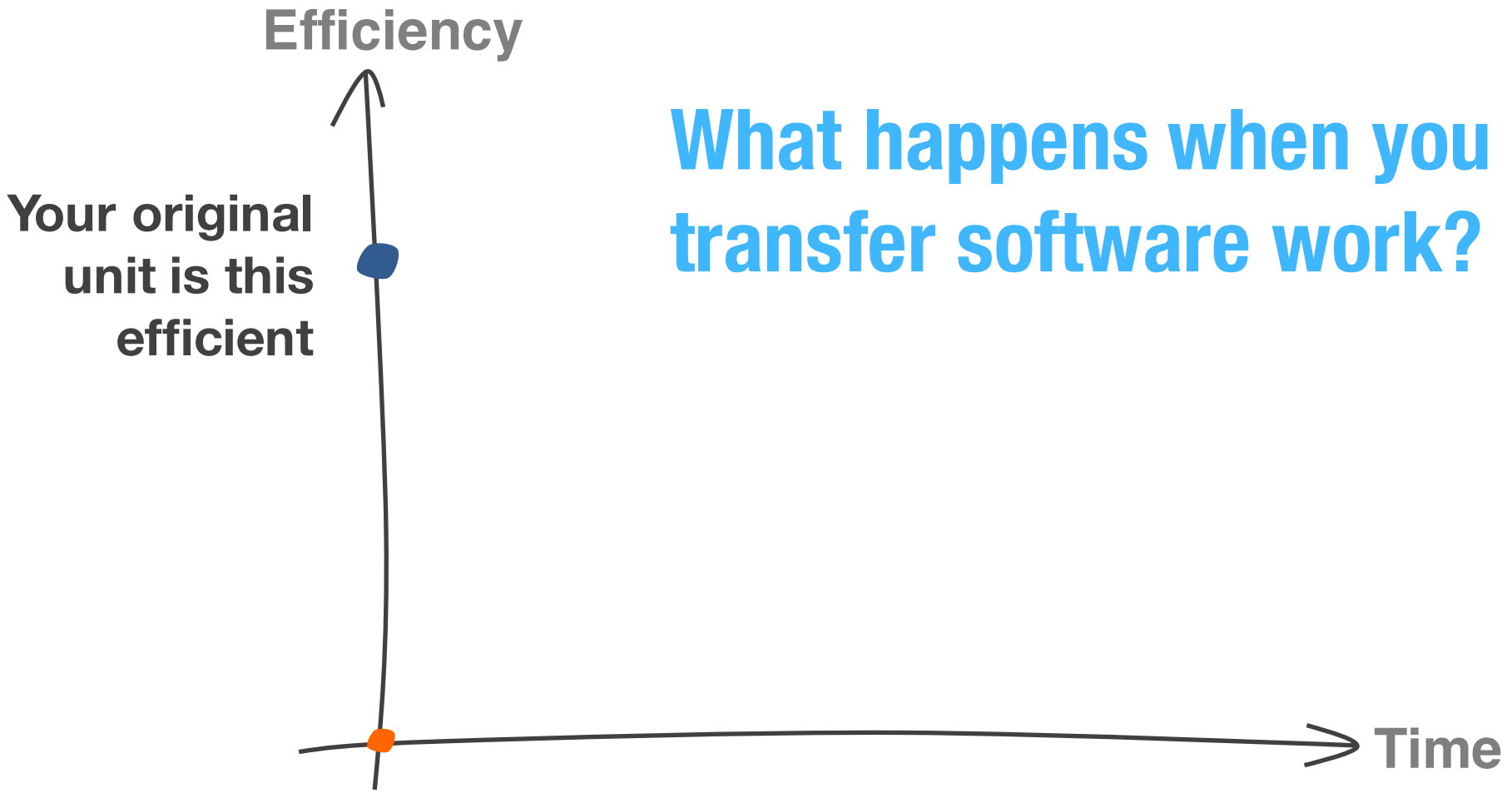
Moving software work — Transfers

Relocation of software development activities from an “original” development site (the sending site) to an offshore site (the receiving site)

Different transfers

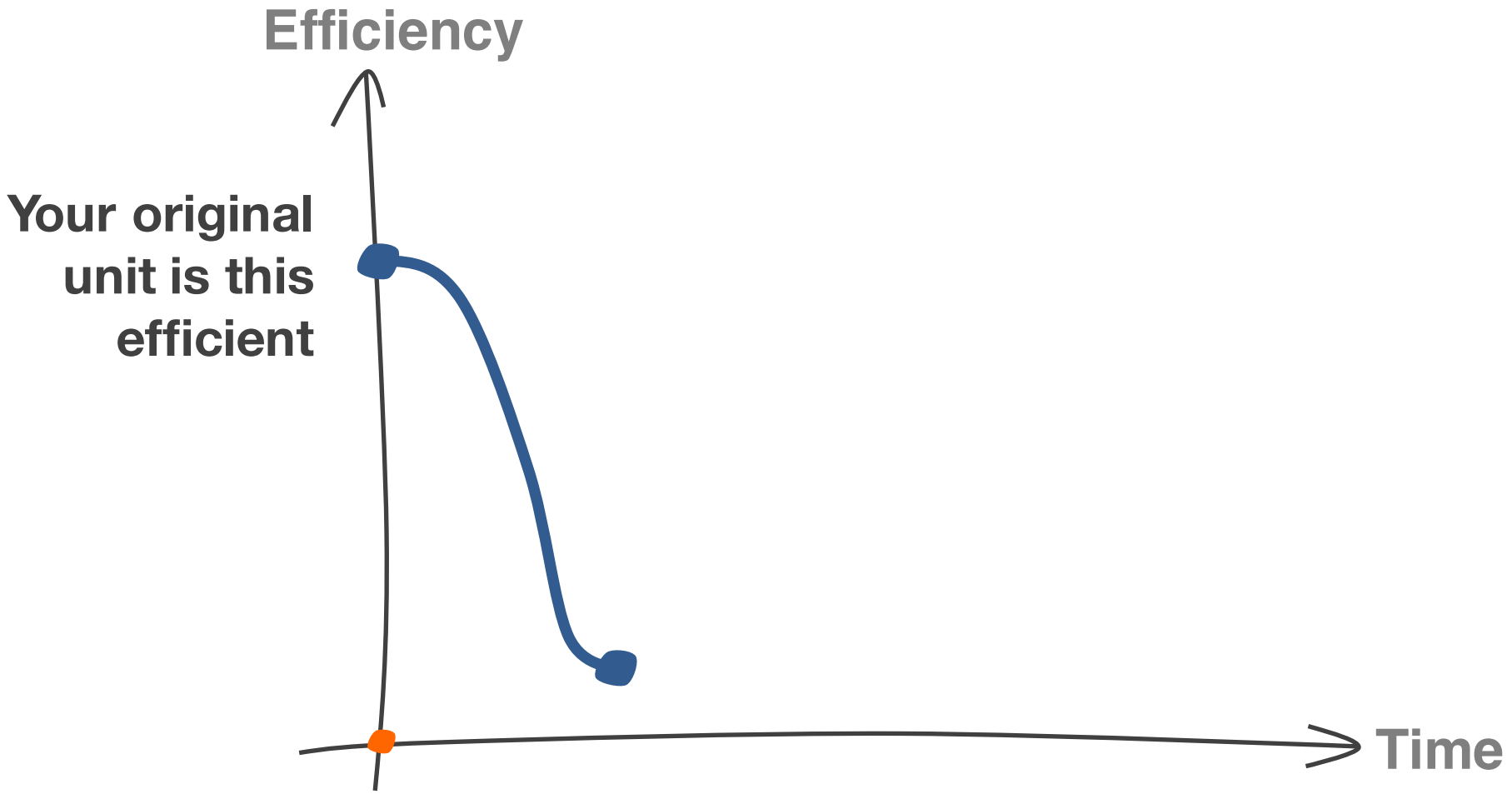
- **Entire project or product** (existing development)
  - Development
  - Customization
  - Maintenance
- **Selected functionality**
  - Subsystem
  - Module
  - Node
  - Component
- **Selected development phase**
  - Coding
  - Testing

Consequences

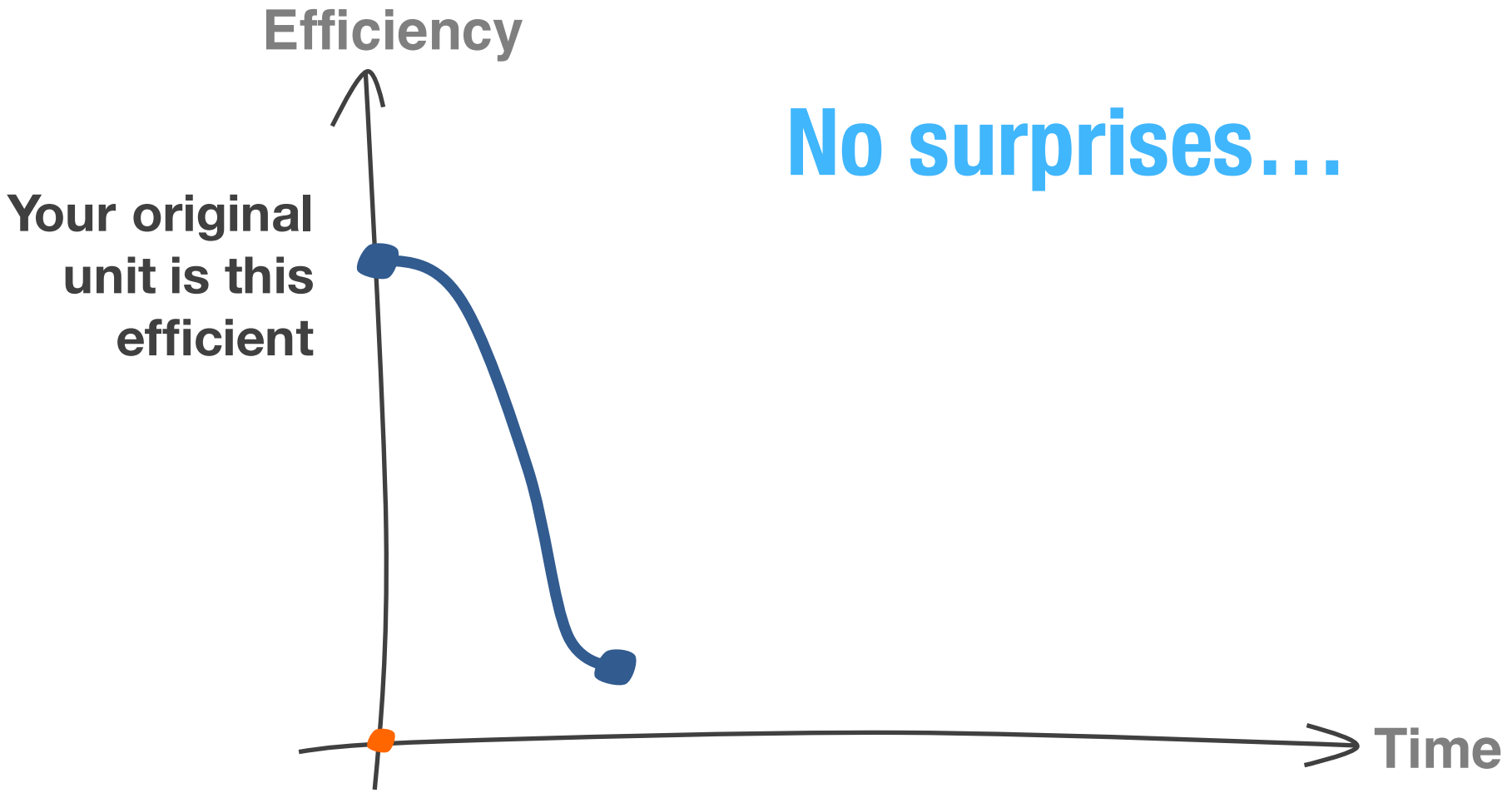


What happens when you transfer software work?

Consequences

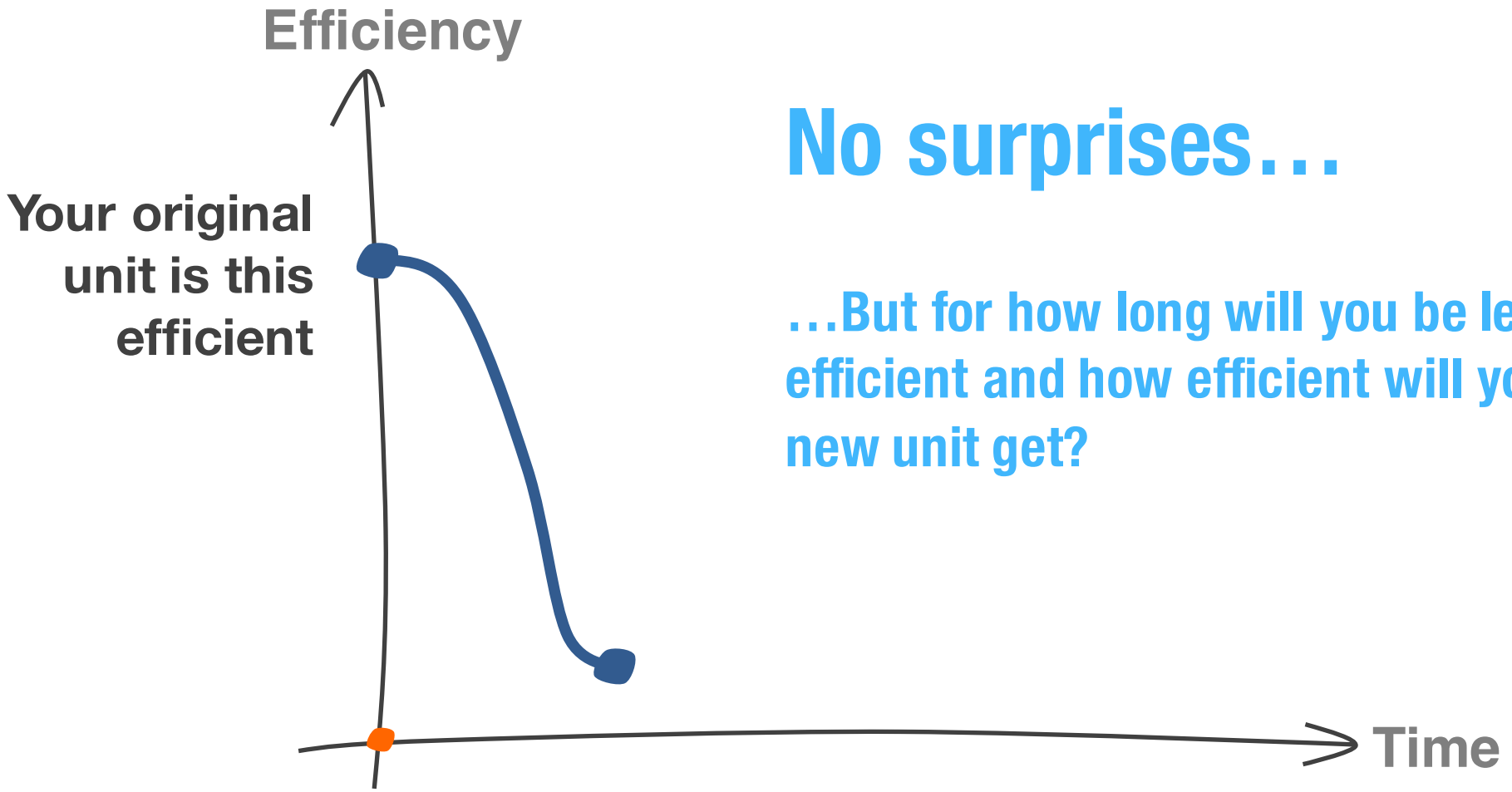


Consequences



No surprises…

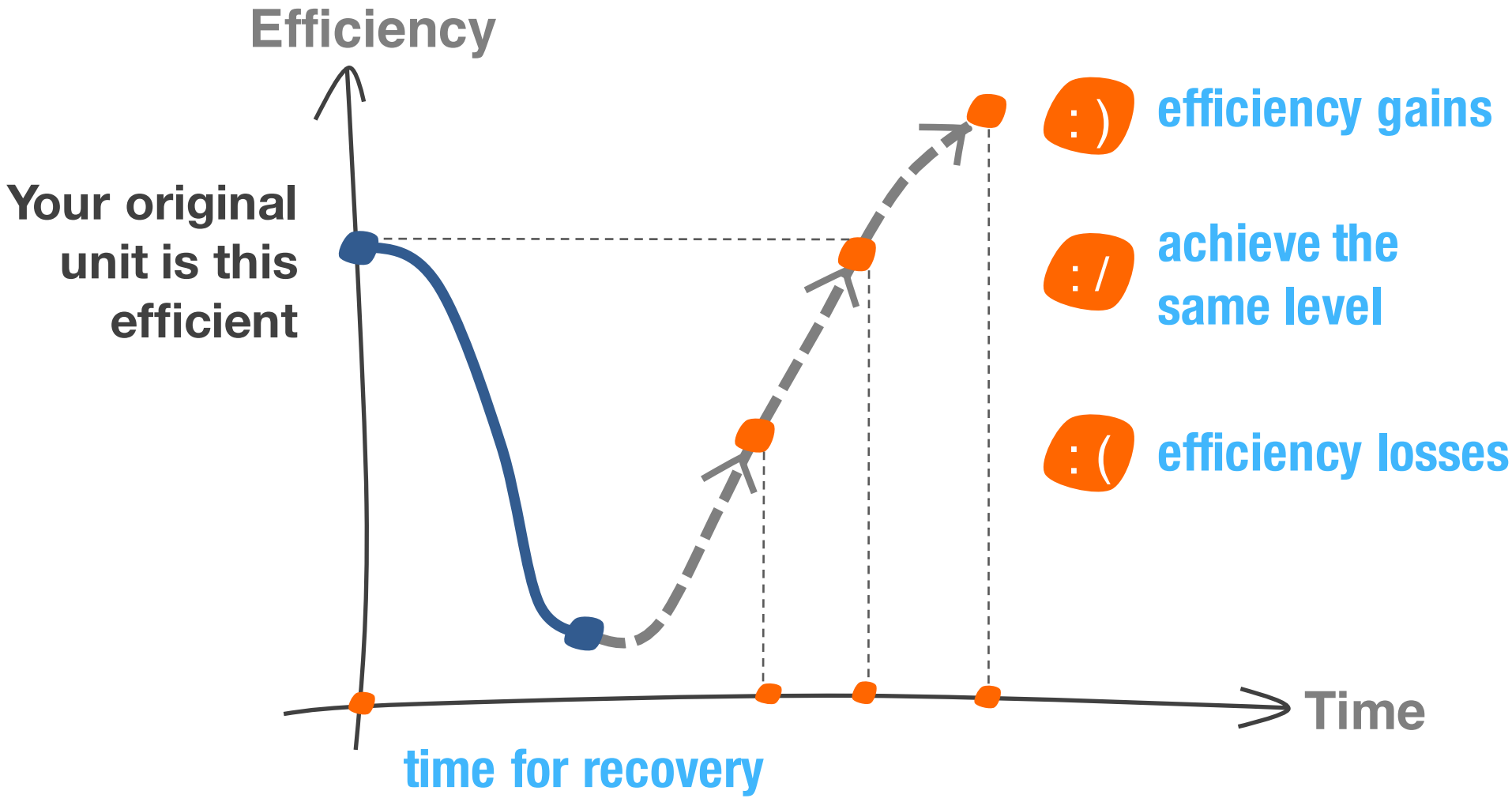
Consequences



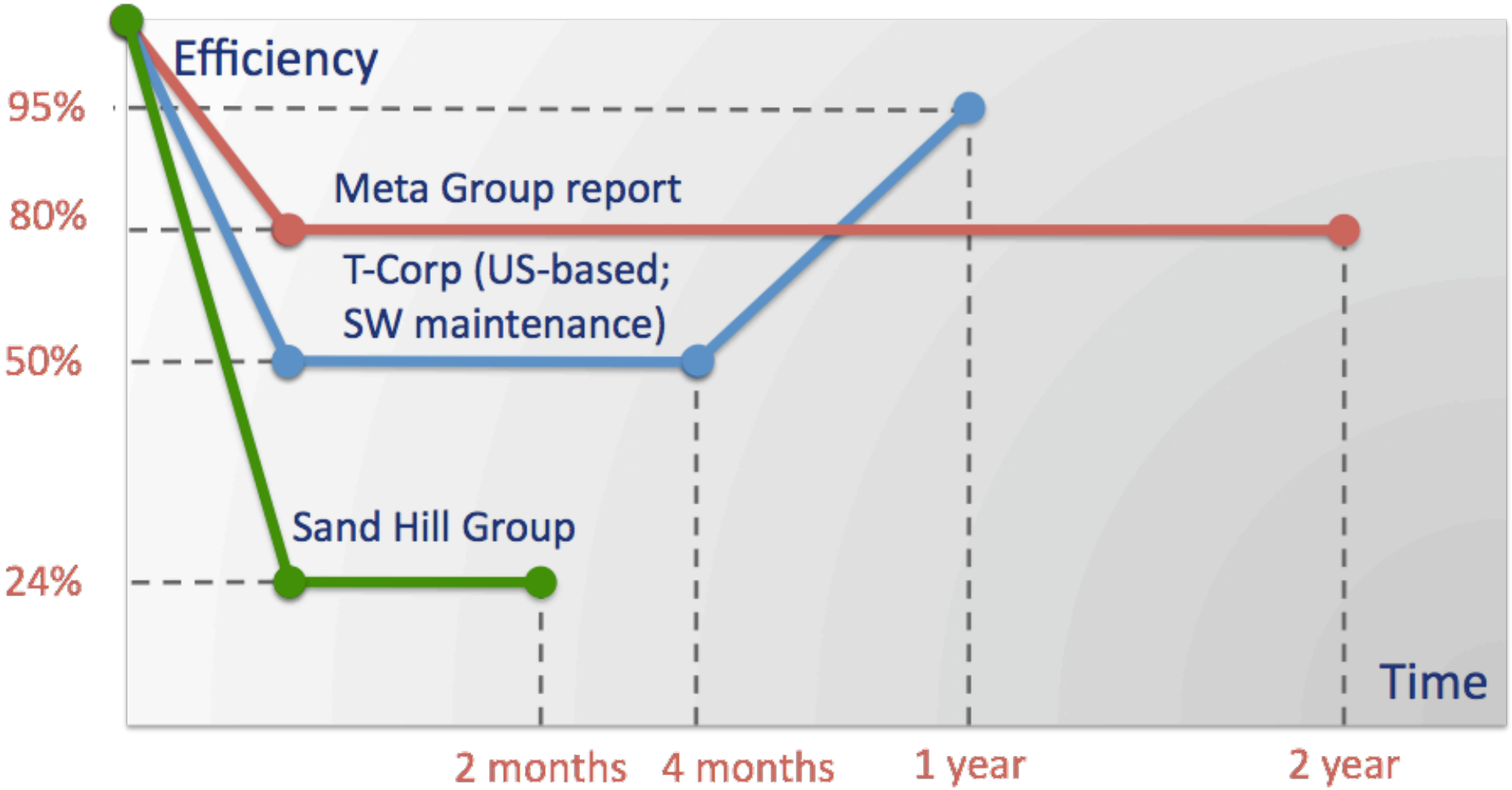
No surprises…

…But for how long will you be less efficient and how efficient will your new unit get?

Consequences



Efficiency decrease in transfers



Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce by E.Carmel and P.Tjia. Cambridge University Press, NY, 2005
The hidden costs of offshore outsourcing by S. Overby . CIO Magazine, Sep. 1, 2003.

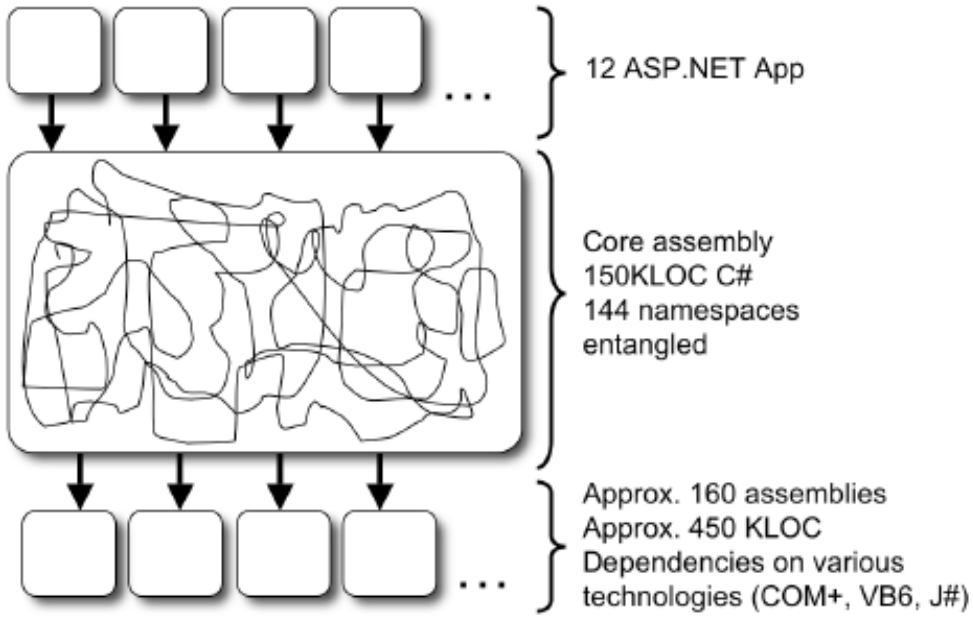
What to expect in a transfer?

- **Negative**
  - Experiences shows that efficiency can decrease down to 20% of the relative efficiency of the original unit and rarely achieves full recovery
  - Consequences of a full transfer are usually experienced long after the transfer is formally finished
  - The learning curve may take up to 5 years
- **Positive**
  - A transfer may rescue projects with low quality if being transferred to developers with more experience and expertise
  - Allocation to offshore sites may demonstrate goodwill
  - Transfers may also create positive option for future transfers

› **Conclusion:** Transfers are not always a good business and products must have at least 7-8 years estimated remaining time of its lifecycle to allow to reach the economic benefits

Loss of knowledge within the product

Source: Software entropy in agile product evolution.
By Hanssen, Yamashita, Conradi & Moonen, 2010



Maintaining legacy code turns into maintaining the "Blob"

Findings from Sweden:

- Product growth in size over time makes it difficult to isolate defective source code components
- Refactoring is seldom performed
- Lack of experience with the product means that developers are not aware of dependencies and ripple effect of changes
- Experience allows improving the product. Products moved around would never get into a state where the product will continuously improve

Maintenance challenges: skills

Basic, general

- Basic computer science principles
- Technical skills
- Coding conventions
- Domain expertise
- Understanding of solutions to domain problems
- Knowledge about program properties
- Knowledge about existing software architecture
- Concept location within the code

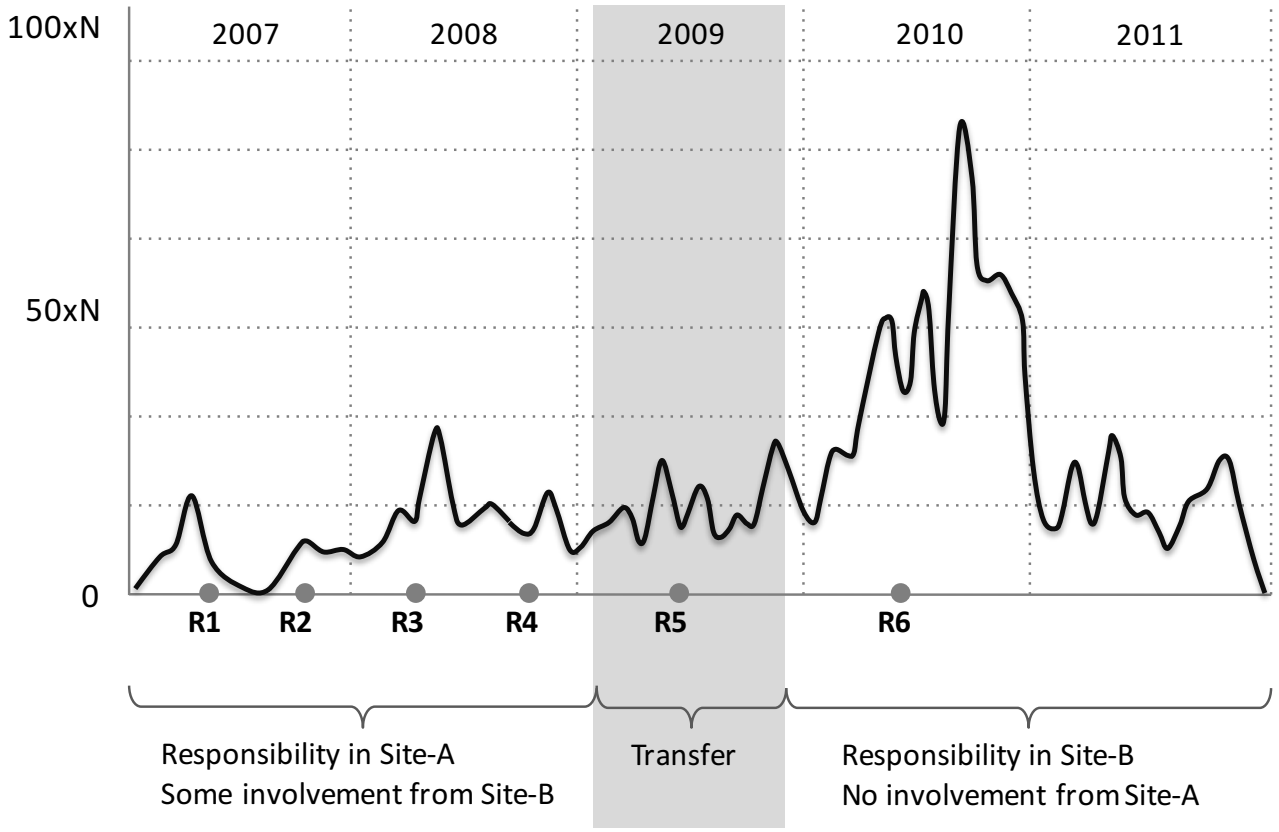
Experience with the product

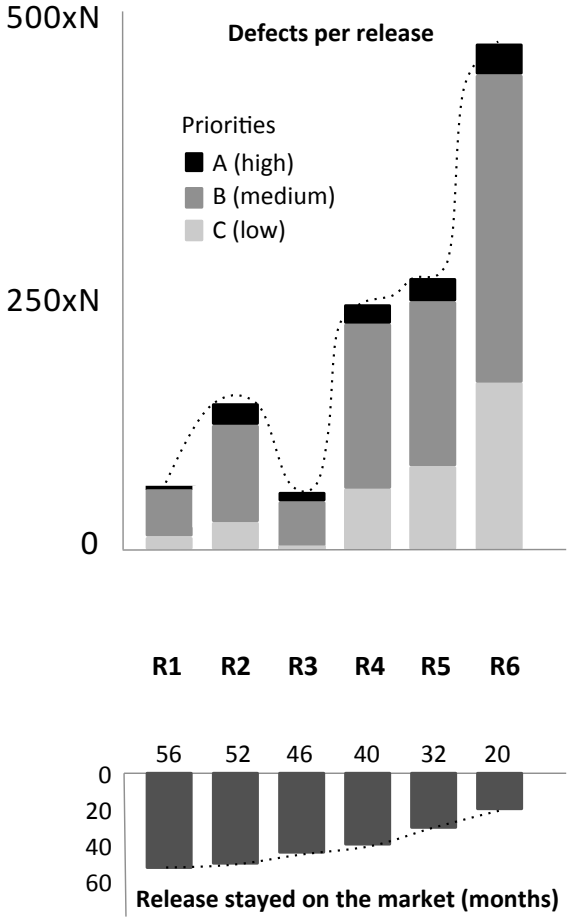
Product specific

Consequences of a software transfer

- Loss of productivity due to loss of expertise within the product
- Decreased quality
- Decreased release frequency
- Decreased release scope
- Transition takes 1 year

Quality decrease



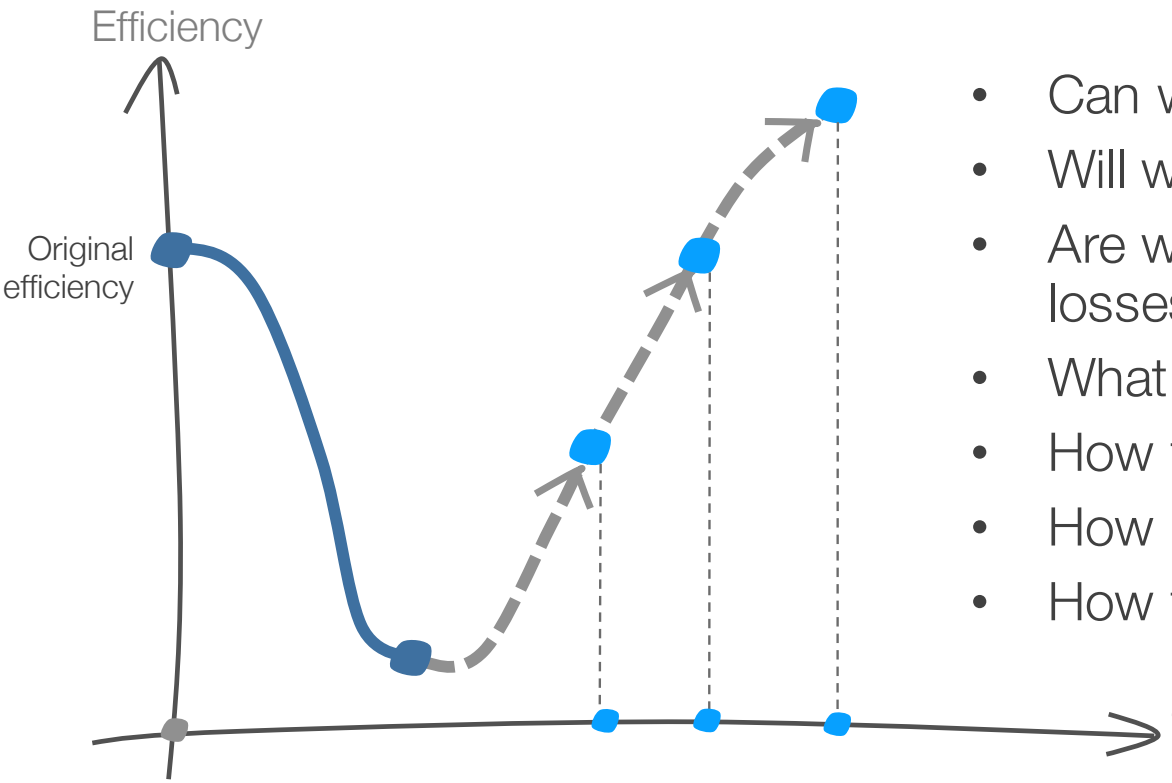


Source: R. Jabangwe, D.Šmite "An Exploratory Study of Software Evolution and Quality: Before, During and After a Transfer", In proceedings of the IEEE International Conference on Global Software Engineering (ICGSE), 2012, pp. 41-50

Lessons learned

- Interviewees noted, that the average defect rate across releases was rather stable, and that introduction of new developers caused various challenges
- It was observed that over time necessary knowledge and competences were built up and defects related to such cases decreased

Relevant questions



- Can we expect efficiency gains?
- Will we achieve the same efficiency?
- Are we doomed to have efficiency losses even after training?
- What influences efficiency?
- How to mitigate decrease in efficiency?
- How long time will recovery require?
- How to speed up recovery?

Loss of knowledge ultimately results in reduced performance — independent of transfer location

Key challenges

- Finding the right people
- Transferring the competence
  - Transferring the implicit knowledge
  - Finding effective training approaches
  - Limiting the scope of a transfer
  - Ensuring support after the product is transferred
  - Ensuring motivation
  - Providing sufficient documentation
- Maintaining on-going development
  - Balancing between transfer and operation
  - Ensuring non-interrupted secure transition
- Overcoming cultural differences
  - Overcoming employee turnover
  - Ensuring effective communication

Critical factors that alleviate transfers

Product

- Mature
- Simple or small
- Independent, decoupled
- Long product life cycle
- Low market pressure
- Small number of customers
- Sufficient documentation
- Easily maintainable product architecture

People

Receiving site:

- Available (employed)
- Competent or trained
- Active, motivated (pull)
- Mature organization

Sending site:

- Competent
- Motivated (no fear of being fired)
- Available for post-transfer support

Process

- Deliberate and discussed decision
- Clear and communicated vision of the end state
  - for the project
  - for the sending site
  - for the product
- Well-established transfer process
- Sufficient transfer time
- Step-wise execution

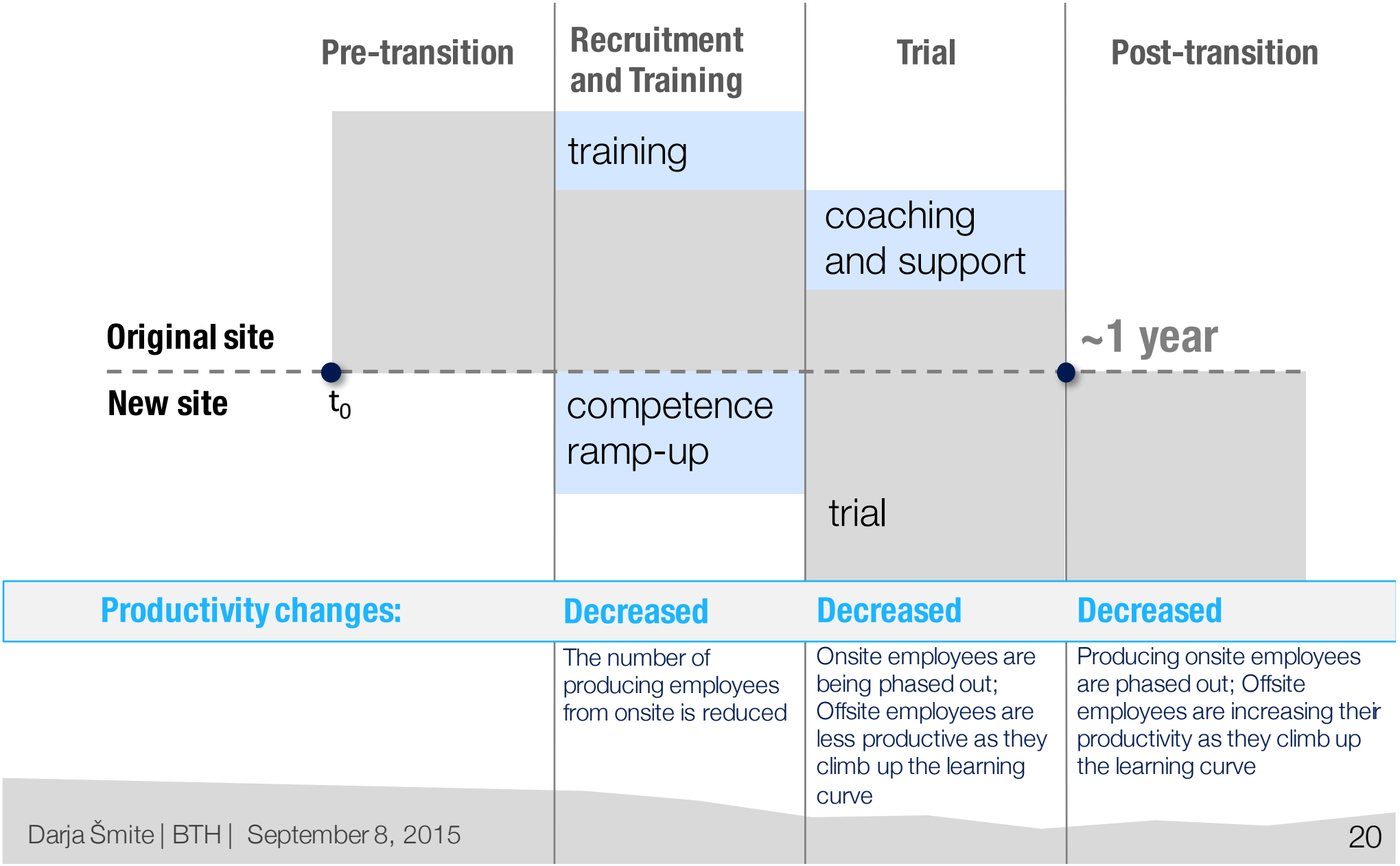
Other recommendations

- Architectural improvements
- Strategic recruitment or involvement of highly competent staff
- Extensive travel between the sites
- Relocation of experts to the new site

  Example from real projects: 1 architect was relocated for 6 months to clarify the challenging questions and mentor the developers at the new site
- Reduction of scope for deliveries
- Creating good documentation

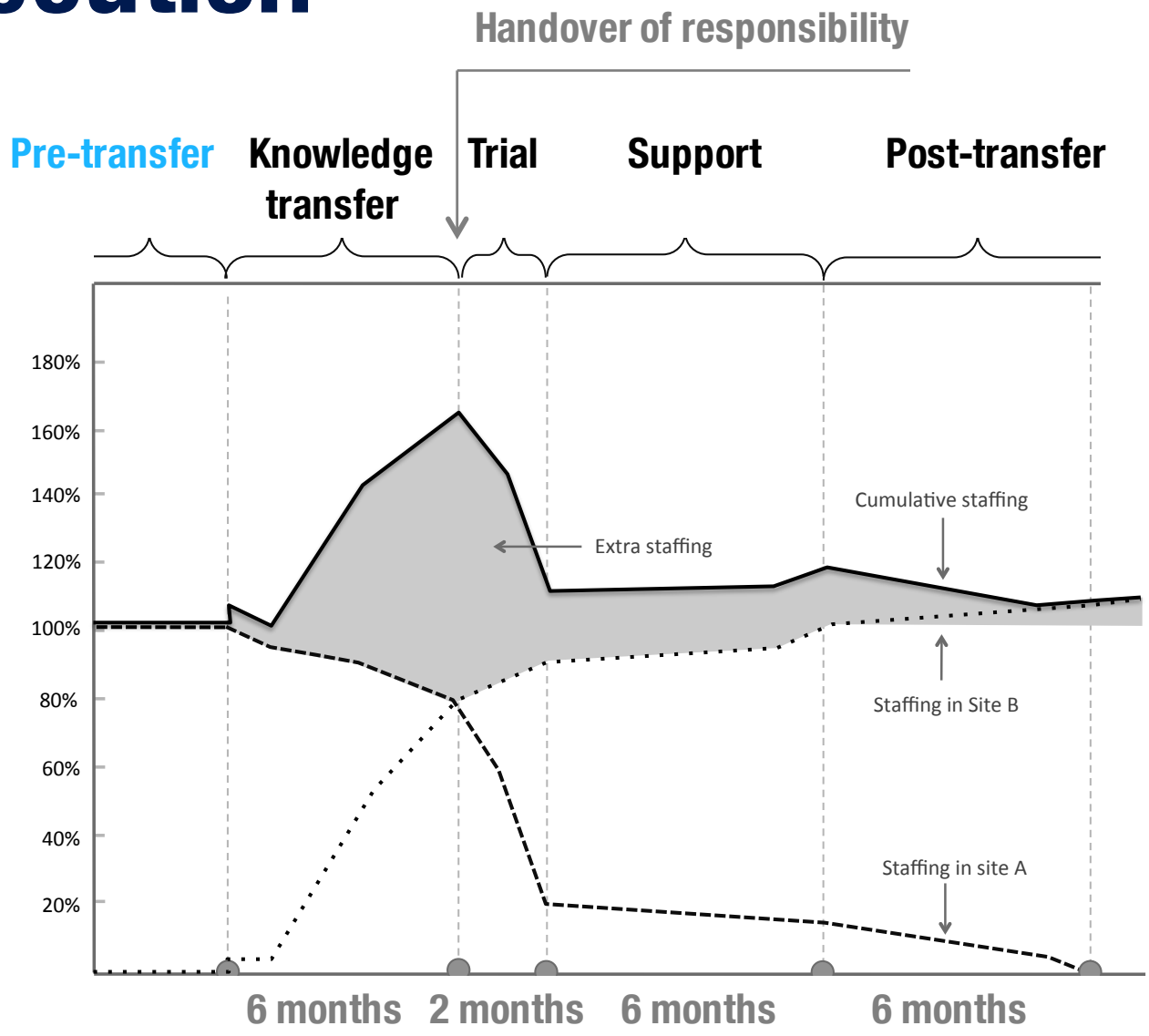
  Example from real projects: 1 consultant was employed for 9 months full time to create documentation for a transferred product

Transition to a global strategy



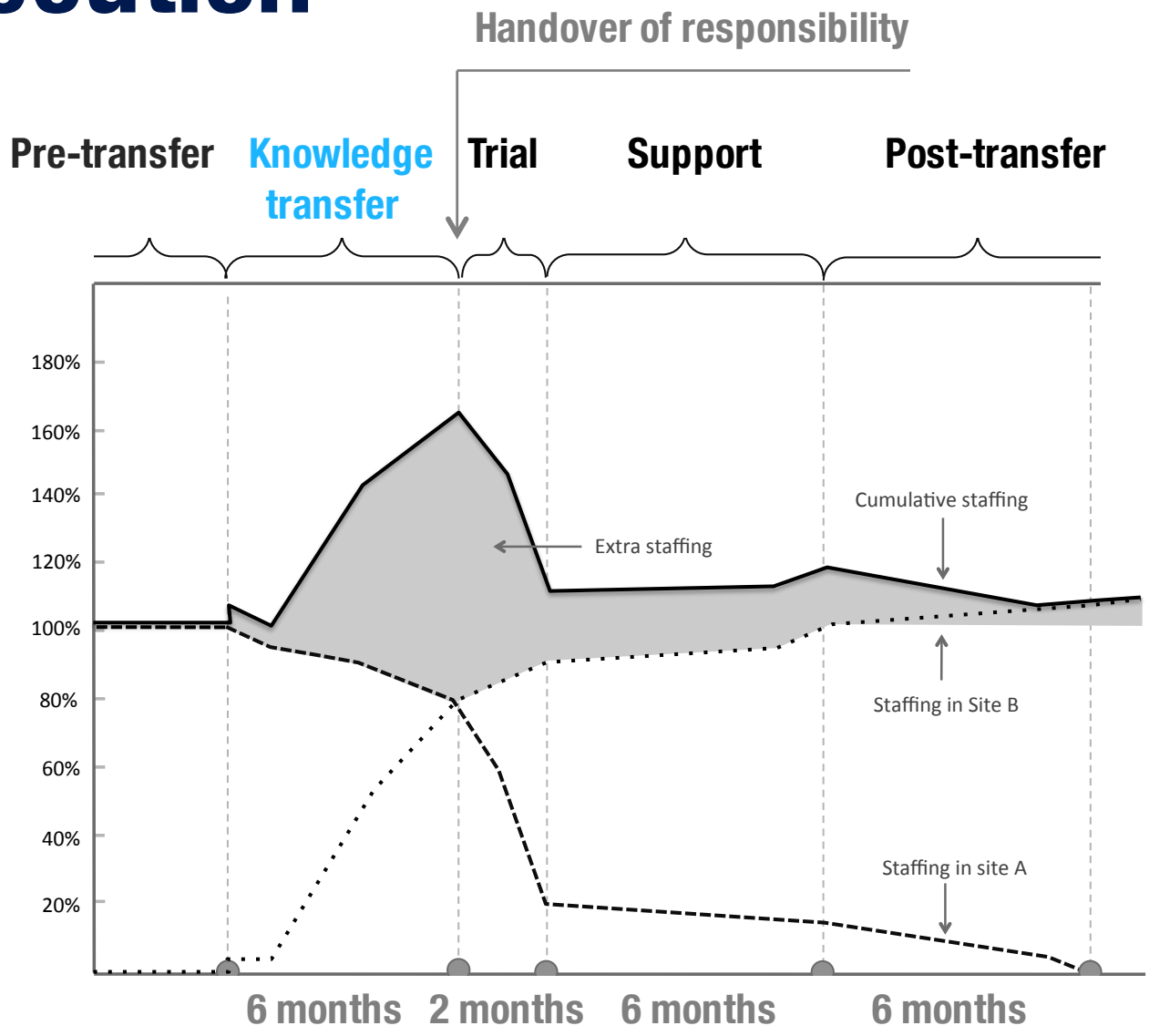
Transfer execution

- Plan recruitment timely and wisely
- Promote people within the Site B
- Inform Site A about future assignments
- Explain decisions
- Plan transfer activities step by step, account for 18-20 months
- Establish detailed transfer plans



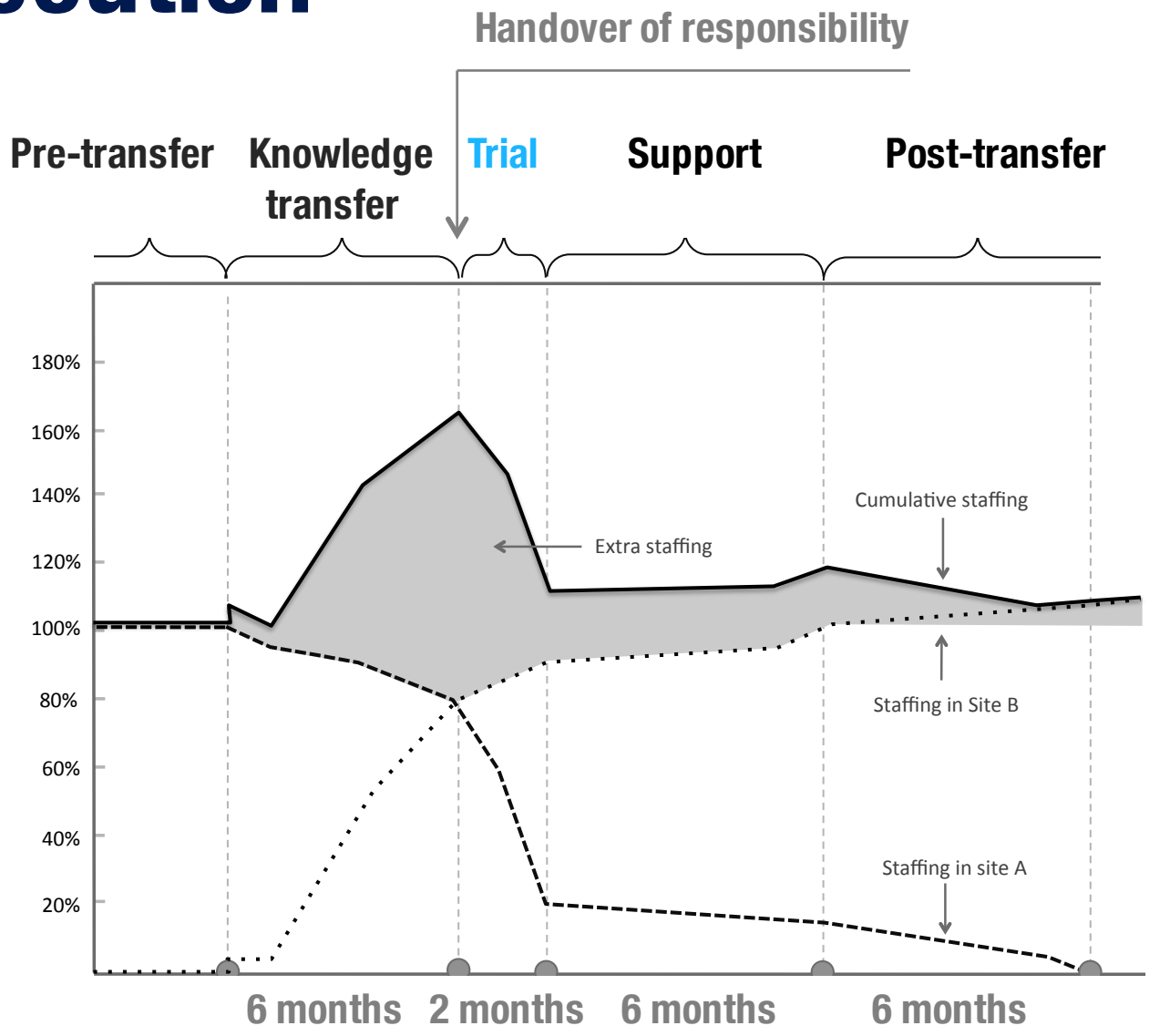
Transfer execution

- Organize co-located hands-on training
- Involve experts, they train faster and better
- Lower the pressure of ongoing development
- Focus on key resources/items
- Ensure product documentation
- Ensure cultural awareness early!



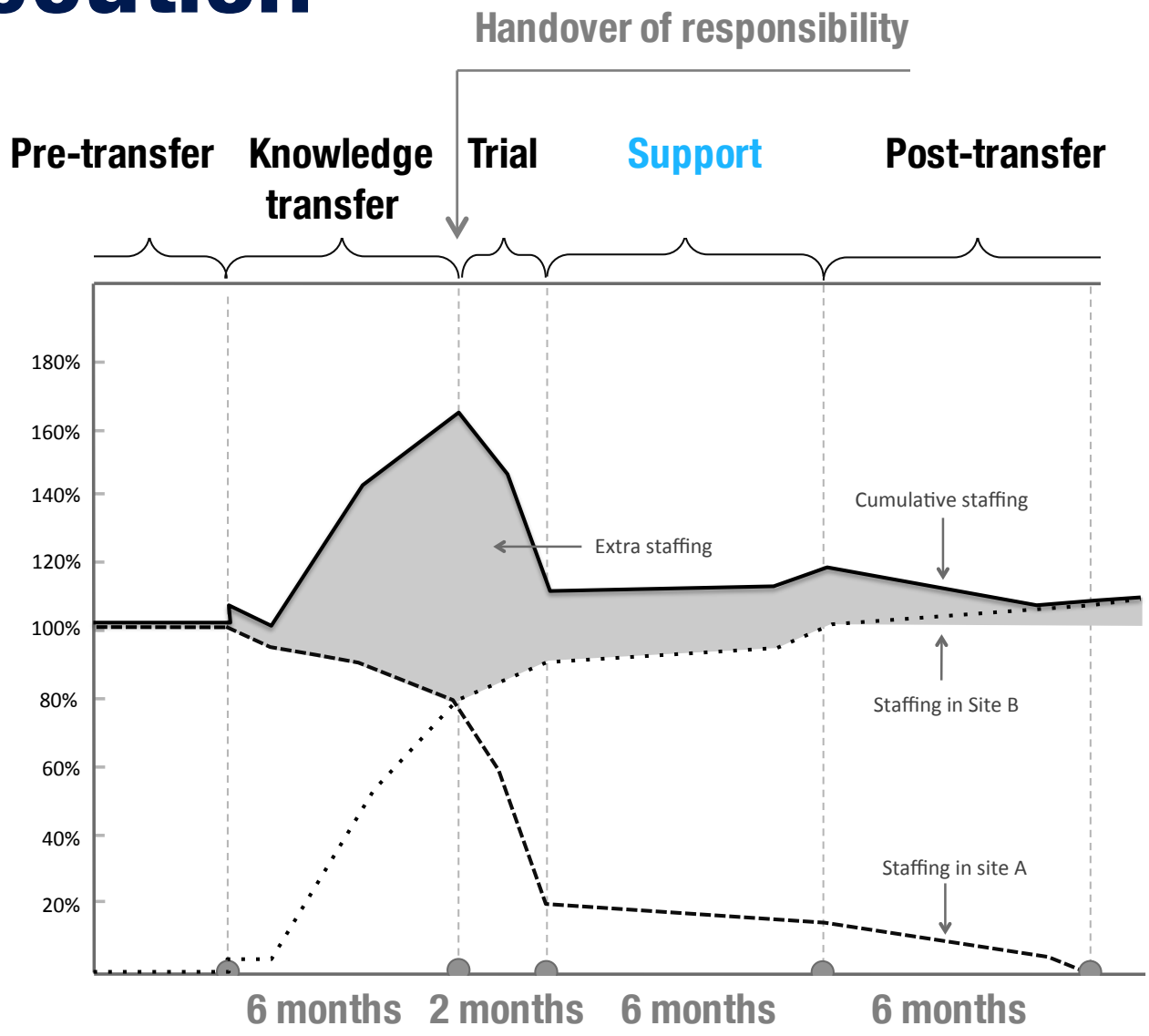
Transfer execution

- Trial before the cut-off having Site A people available as a back-up
- Downsize in Site A after the trial



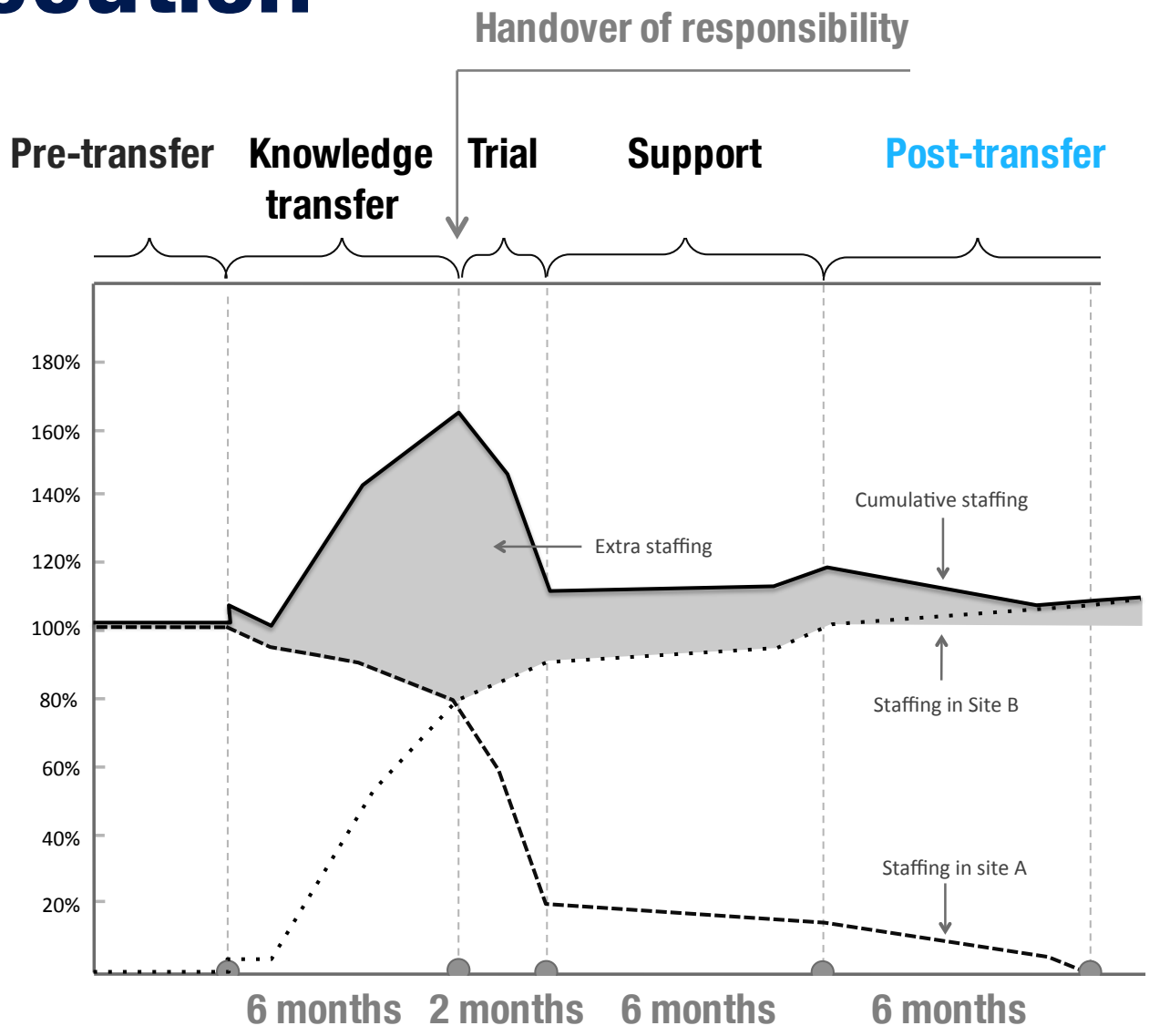
Transfer execution

- Support through coaching
- Keep some people from Site A for fire fighting and escalations
- Do not expect high productivity, foresee reduction in delivery size and frequency



Transfer execution

- Transfer a few people with the product
- Communicate expectations
- Set threshold values for important goals and ensure continuous monitoring

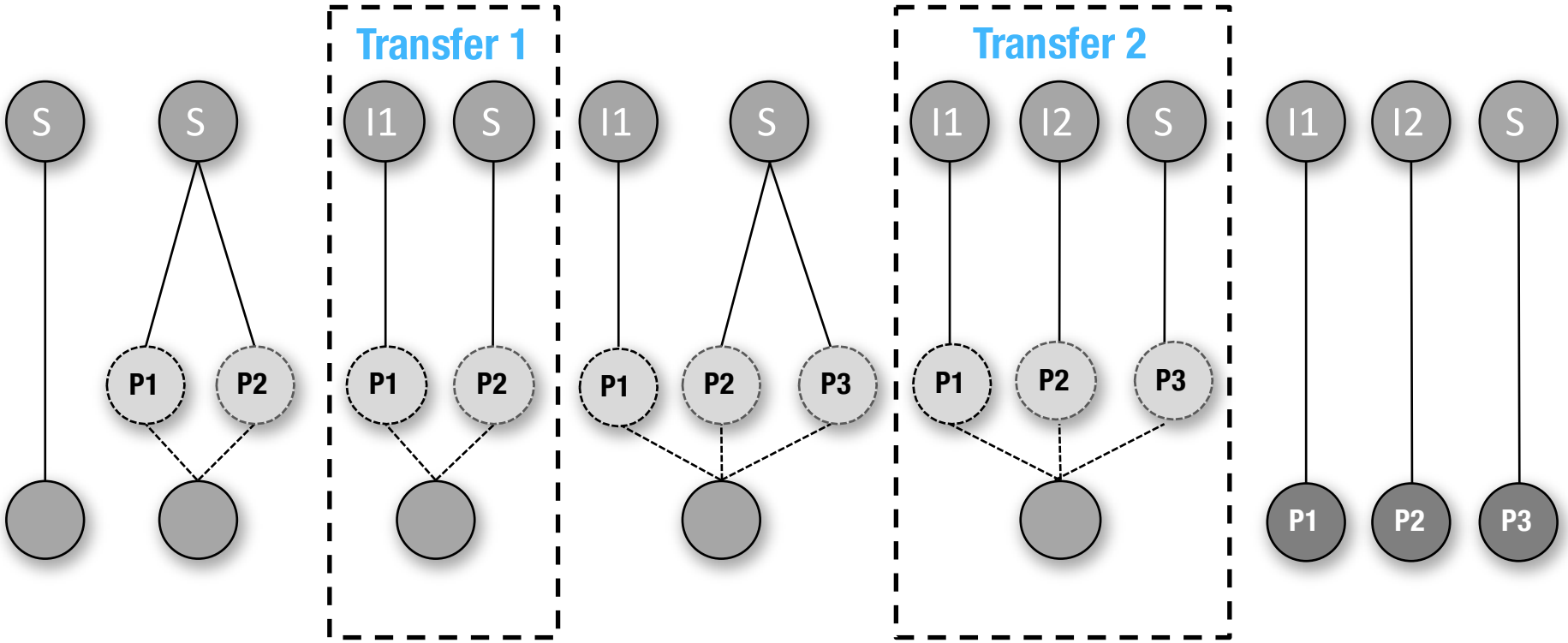


Conclusions: choosing products

- Not all products are equally suitable for a transfer
- Risks:
  - Complex products with specific domain knowledge
  - Products with a large number of interdependencies
  - Products with a large number of customers
  - Products with frequently changing requirements
  - Immature products with unstable architecture or quality
  - New products that have not been released (i.e. unknown quality)

Risky products cannot be transferred, unless specifically addressing the risks before the transfer (sometimes also after the transfer long-term)

Which were products suitable for a transfer and why?



Conclusions: choosing locations

- #1: Availability of resources
- #2: Domain and product expertise

- Process maturity has a positive impact
- Selection of locations can be also influenced by
  - Ease of hiring and training new employees
  - Turnover factors
  - Difficulty to get a visa and availability of direct flights
  - Import and export rules, regulations for customs clearance
  - Security concerns

Questions?