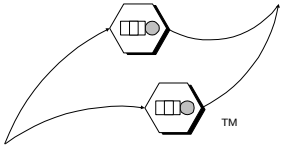


The Modeling Pyramid: From Design to Production

CMG2000 Session 4404, December 13, 2000

http://www.simalytic.com/CMG00/4404ppt.pdf

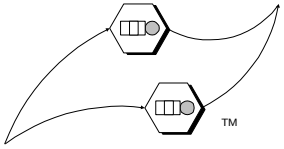
Dr. Tim R. Norton

Simalytic Solutions, LLC

719-635-5825

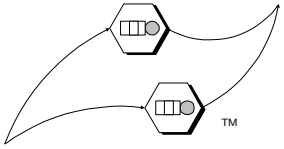
email: tim.norton@simalytic.com

http://www.simalytic.com



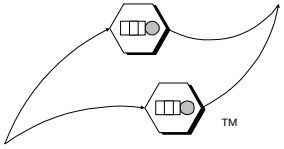
Agenda

- **Introduction**
  - Modeling through the Application Life Cycle
- **Role of Modeling Techniques**
  - General Modeling Tools
  - Specialized Tools for Modeling
- **The Modeling Pyramid**
  - Modeling Pyramid Levels
  - Modeling Pyramid Implementation
- **Conclusion**



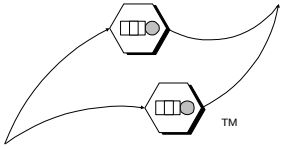
Introduction

- **Value of Modeling to Management**
  - Binary Condition (valuable or worthless)
  - Understanding Political Situation Key
    - Improve the business based on real measurement objectives
    - Frustration with modeling cost and effort when the answer is already known (hidden agendas)
  - Modeling Pyramid assumes existing value at one level
    - Objective: Increase value by relating models from adjacent Application Life Cycle stages



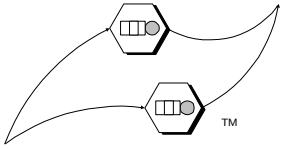
Life Cycle Stages

- **Application Design Models**
  - Understanding application characteristics
- **Implementation Models**
  - Understanding application behavior
- **Deployment Models**
  - Understanding the relationship between the application and the environment

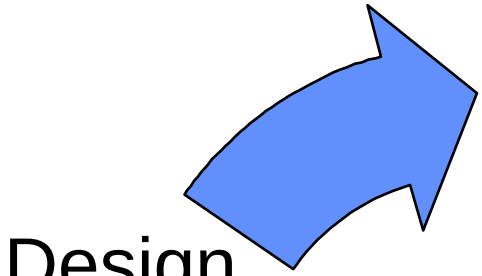


Life Cycle Stages

- **Production Models**
  - Understanding the relationship between application load and resource requirements
- **Planning Models**
  - Predicting future resource requirements
- **Enhancement Models**
  - Predicting the impact of application changes



Application Life Cycle

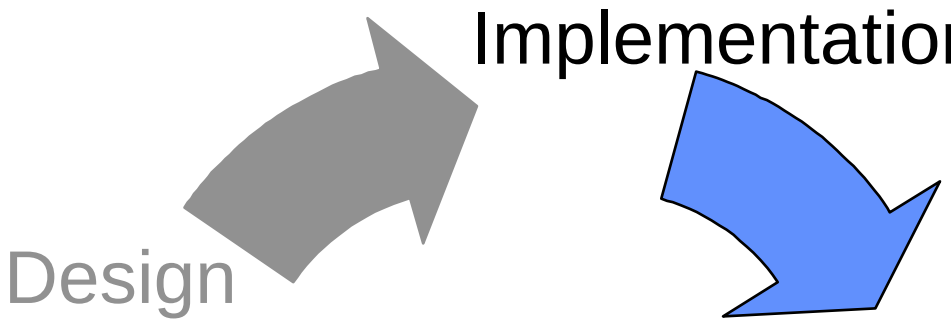


Application Life Cycle

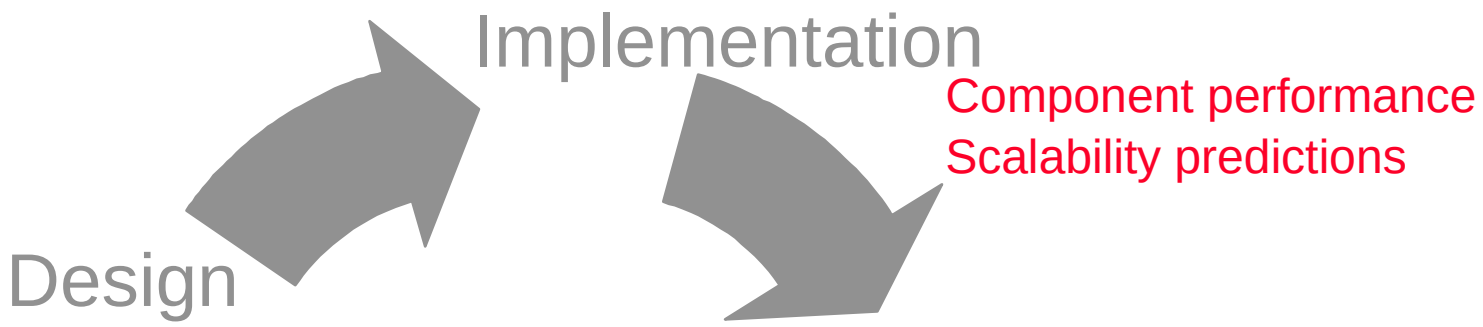
Function - UML
Performance - SPE

Design

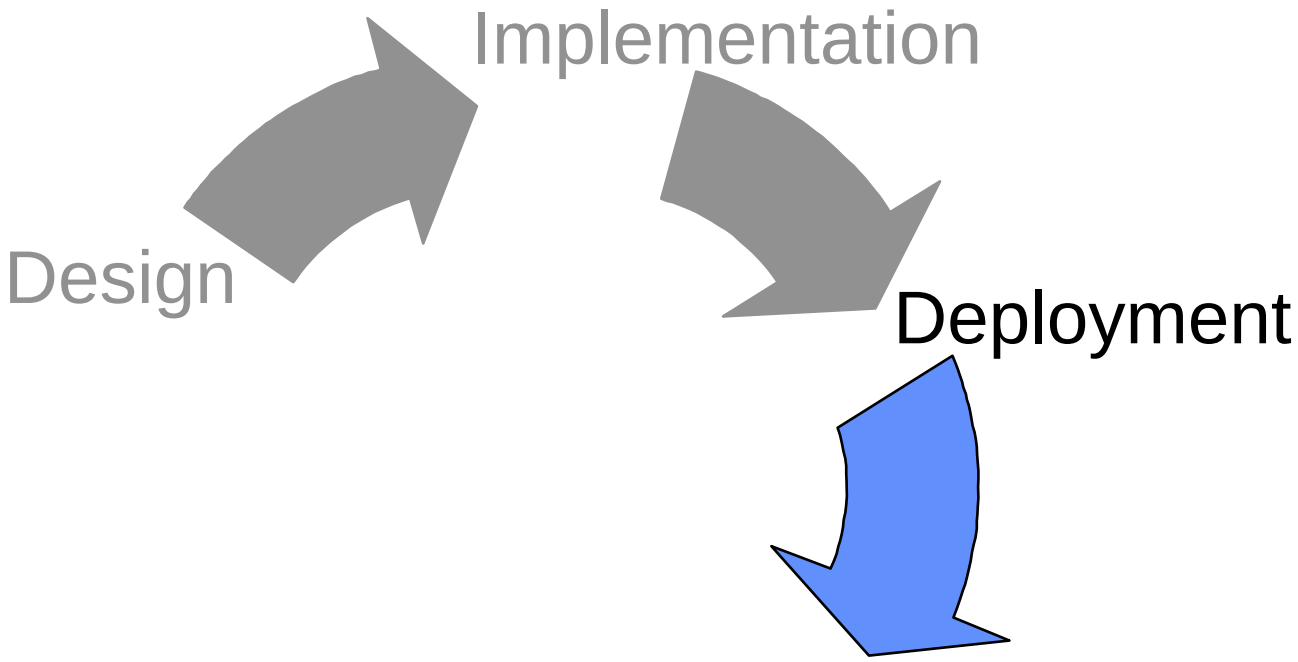
Application Life Cycle



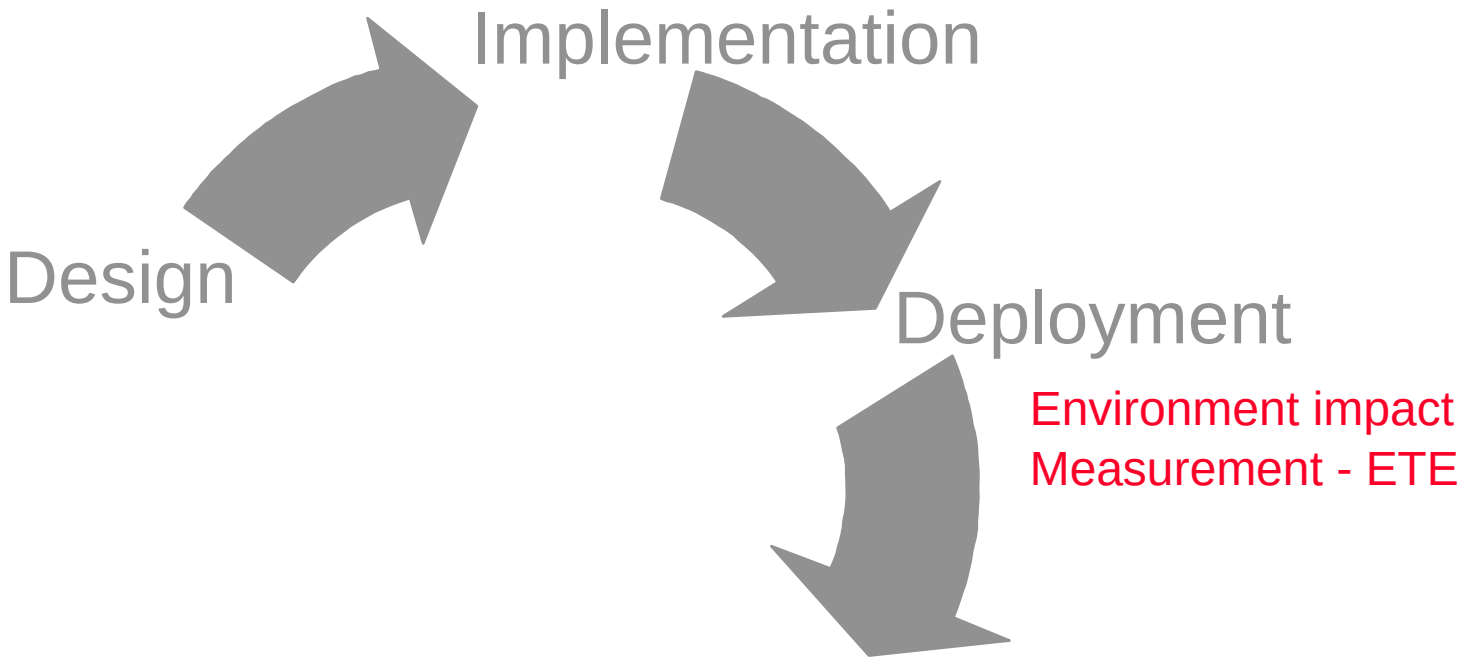
Application Life Cycle



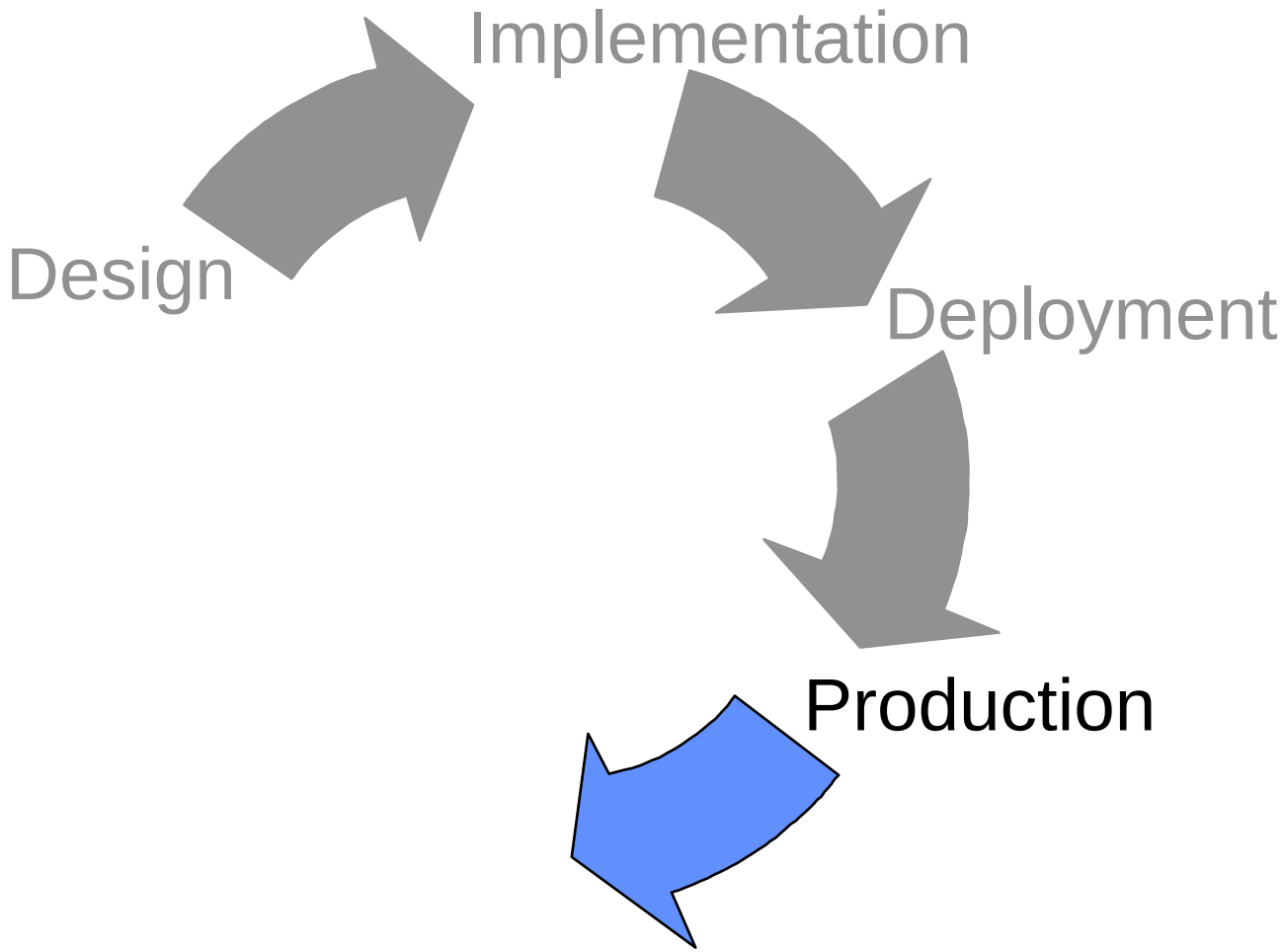
Application Life Cycle



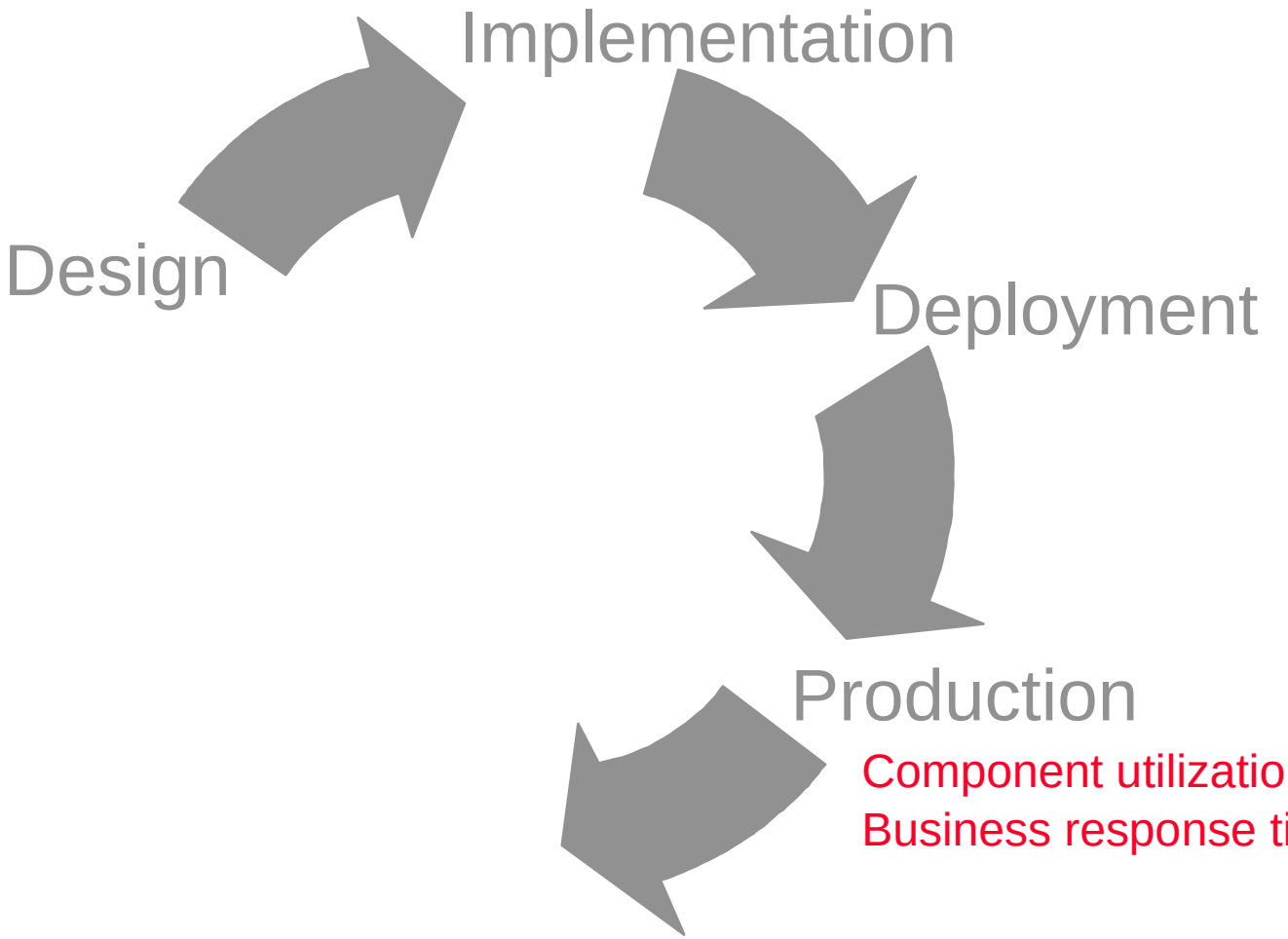
Application Life Cycle



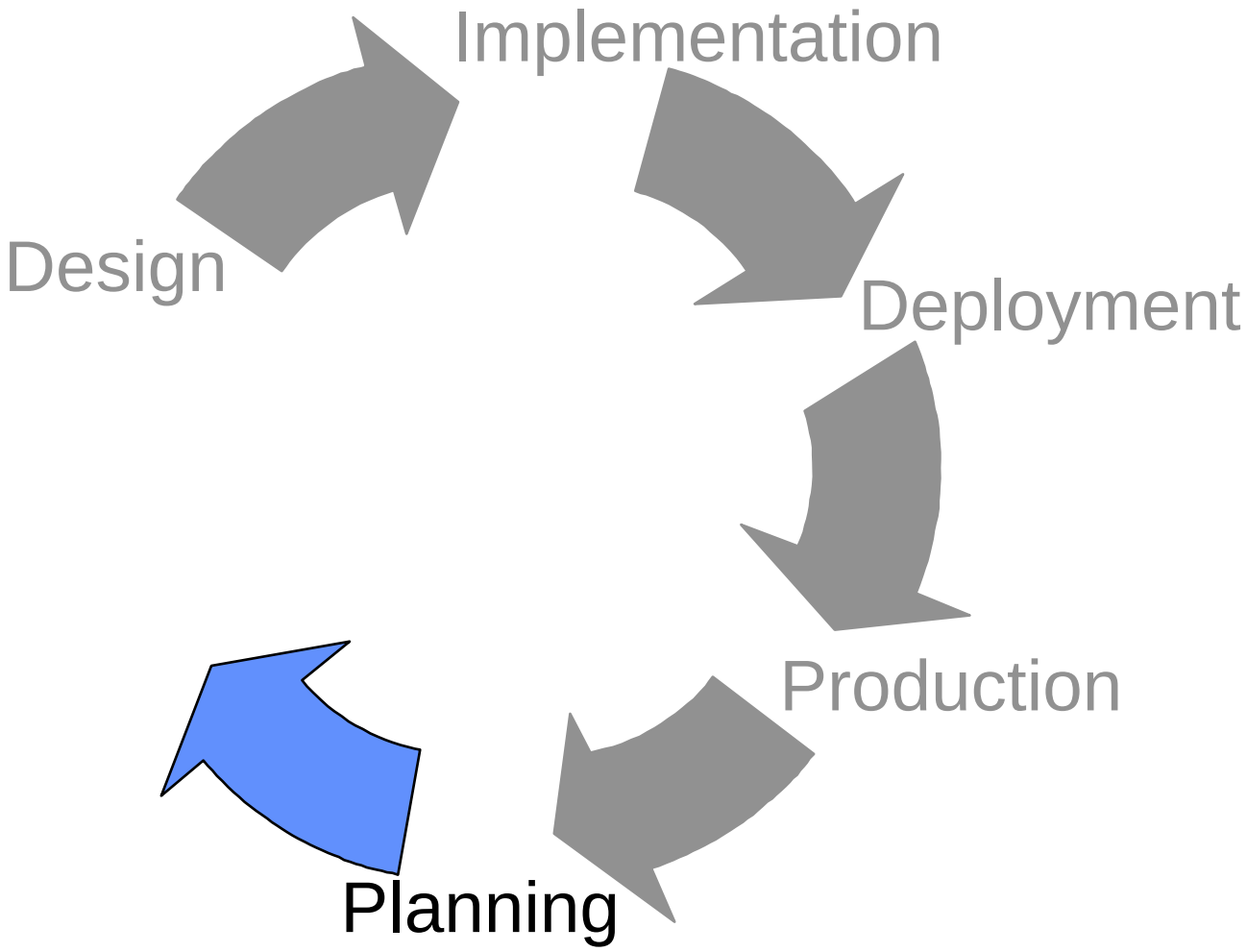
Application Life Cycle



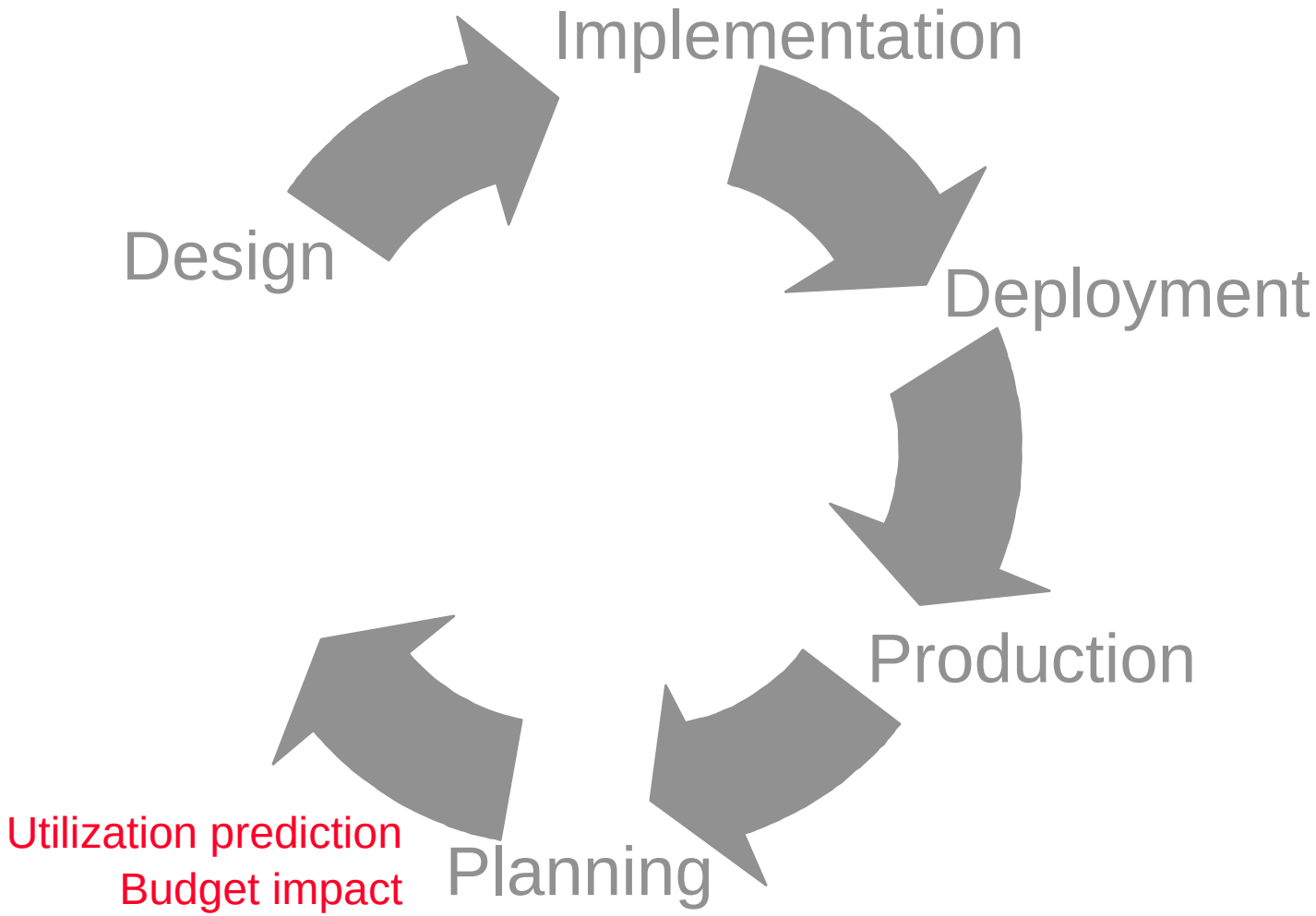
Application Life Cycle



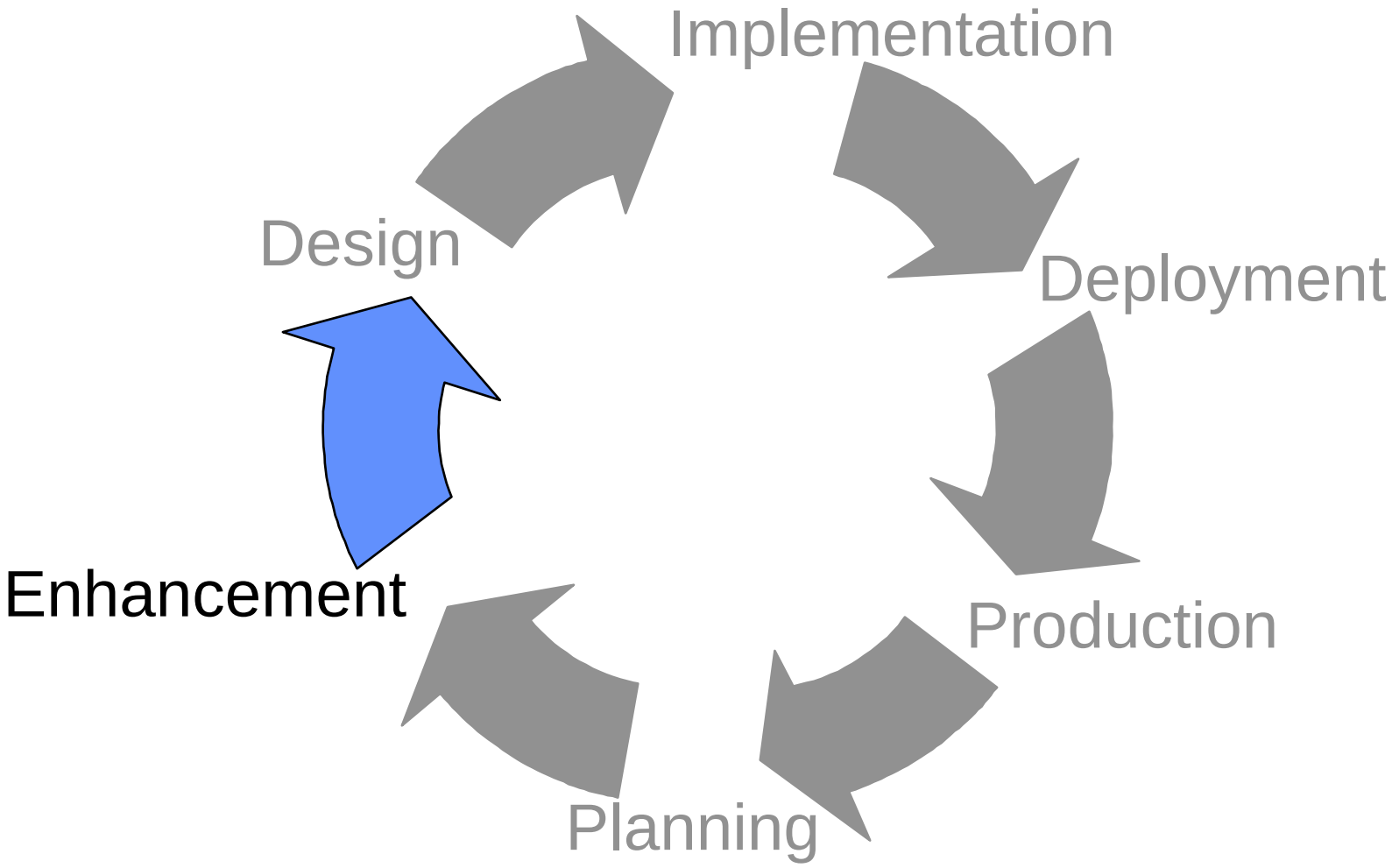
Application Life Cycle



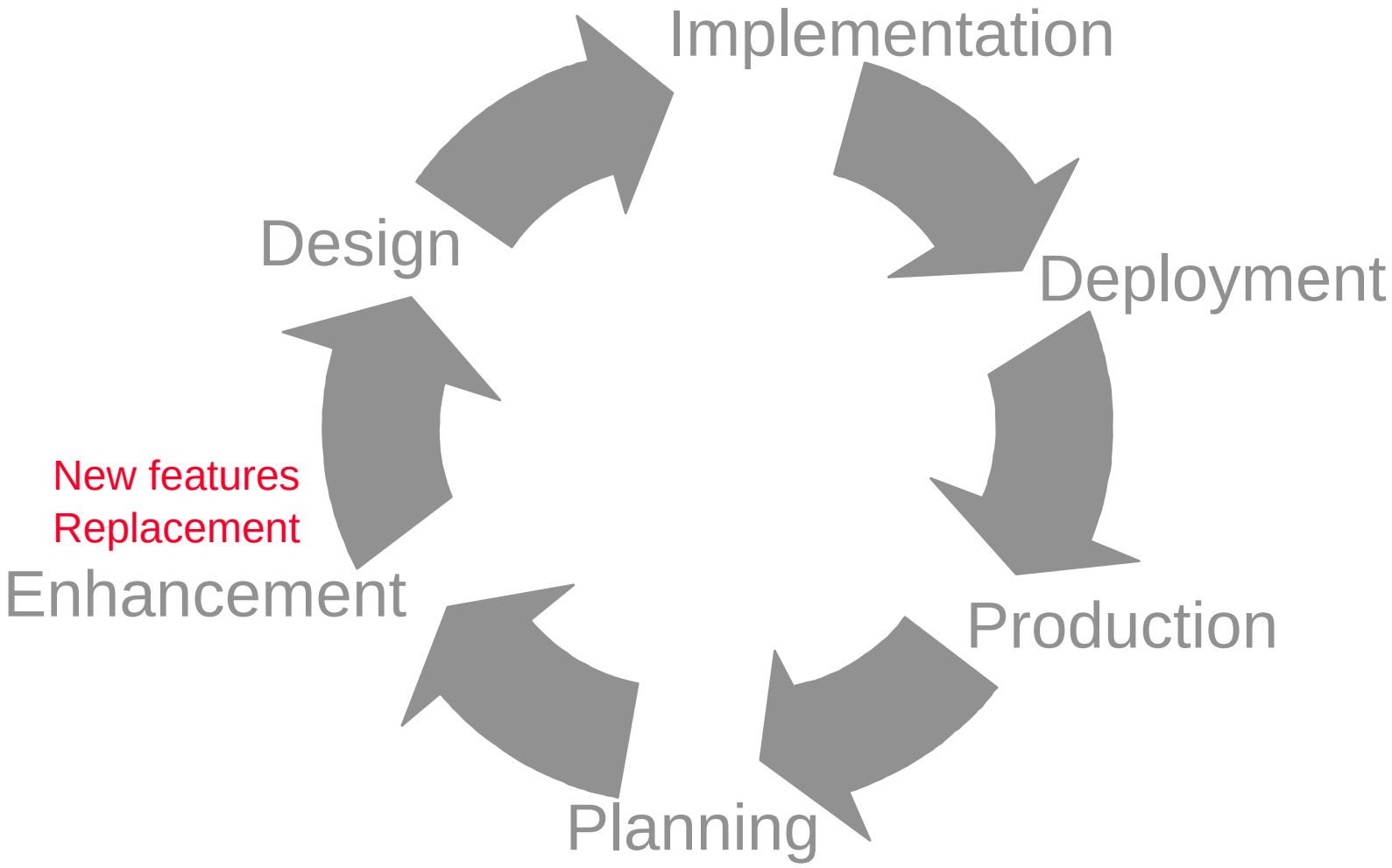
Application Life Cycle



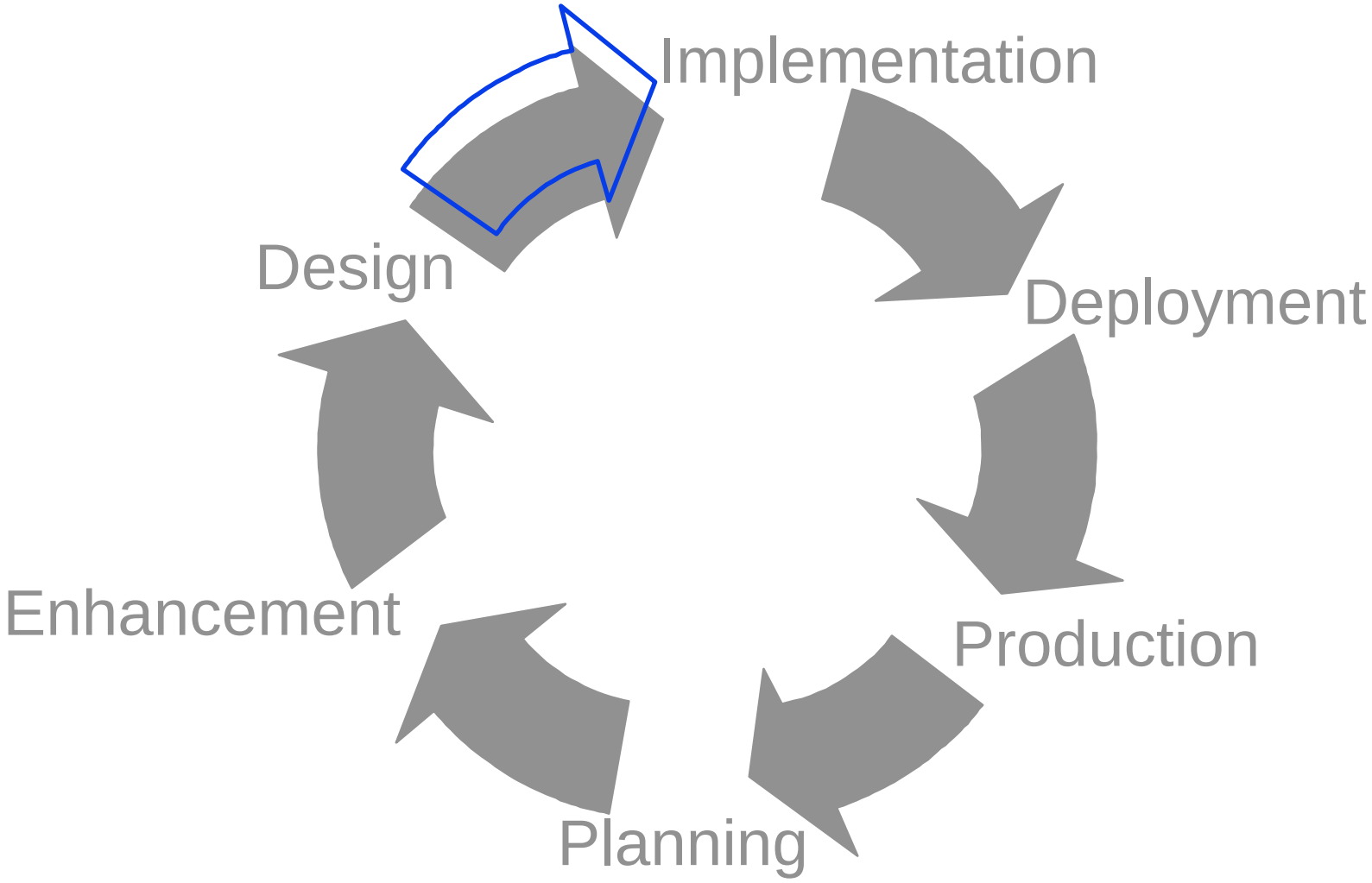
Application Life Cycle



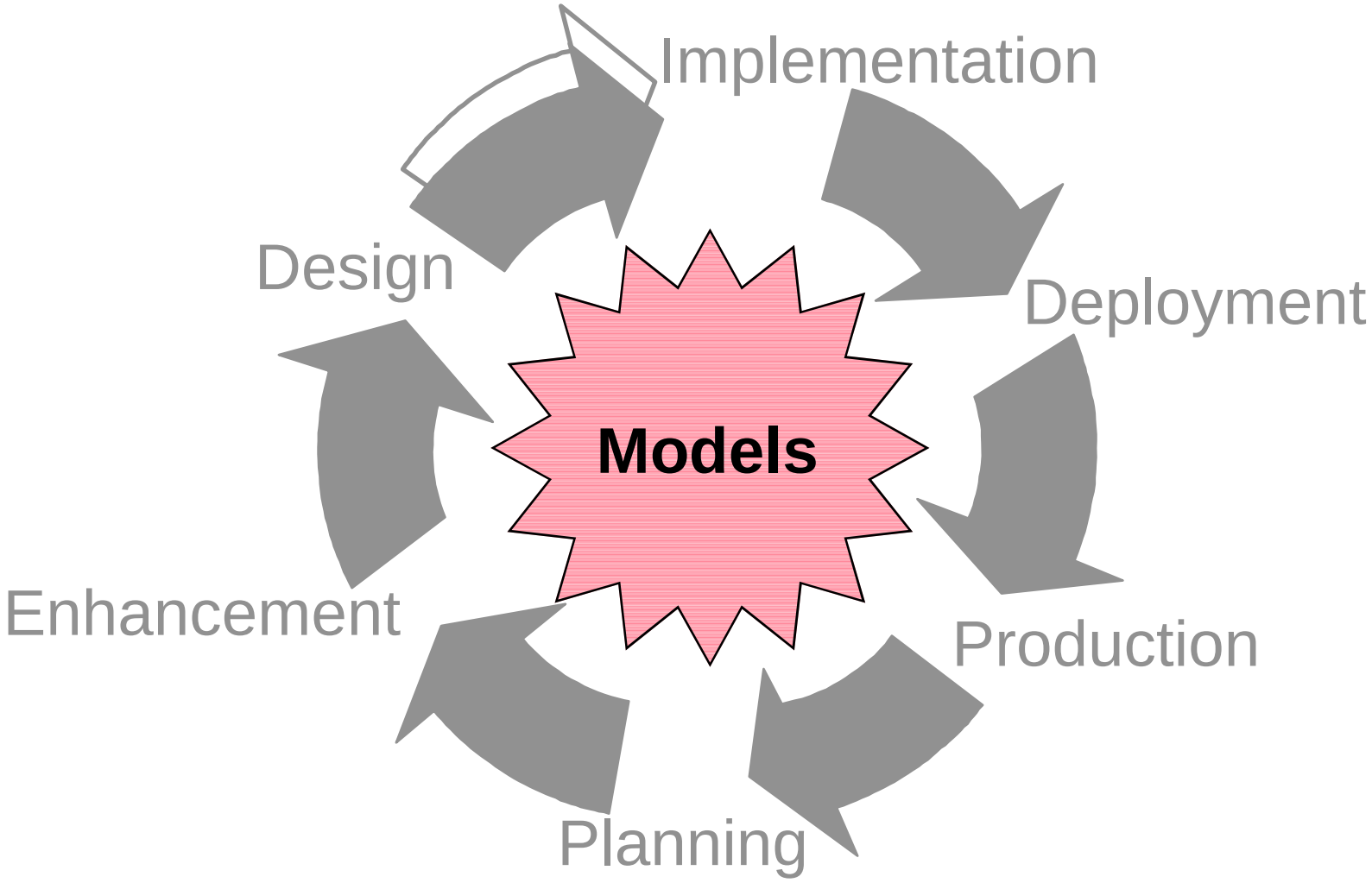
Application Life Cycle

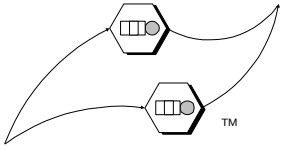


Application Life Cycle



Application Life Cycle

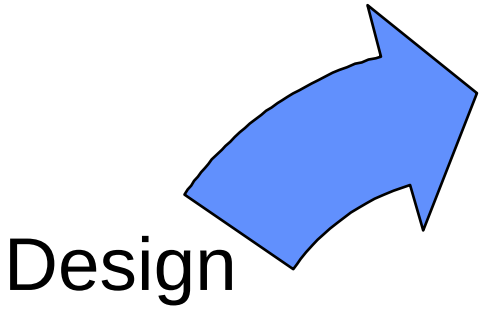




Application Example

- **Modeling an Order Entry Application**
  - Mail-order catalog company
  - Customers call and place orders
  - Operators use Order Entry Application
  - Response time impacts customer satisfaction
  - Resource capacity limits business growth
  - Increased transaction rates based on Marketing plans for new customers and new products
  - New function: replacing paper shipping reports with an on-line transaction

Application Life Cycle Example



Application Life Cycle Example

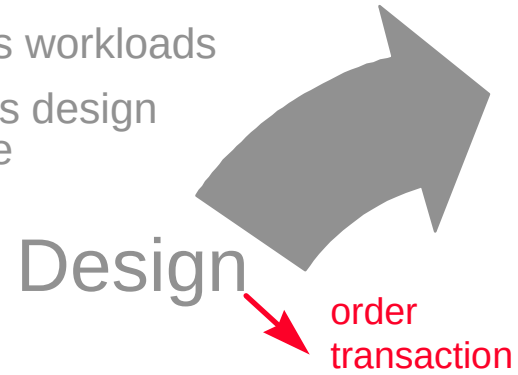
UML defines workloads

SPE predicts design performance

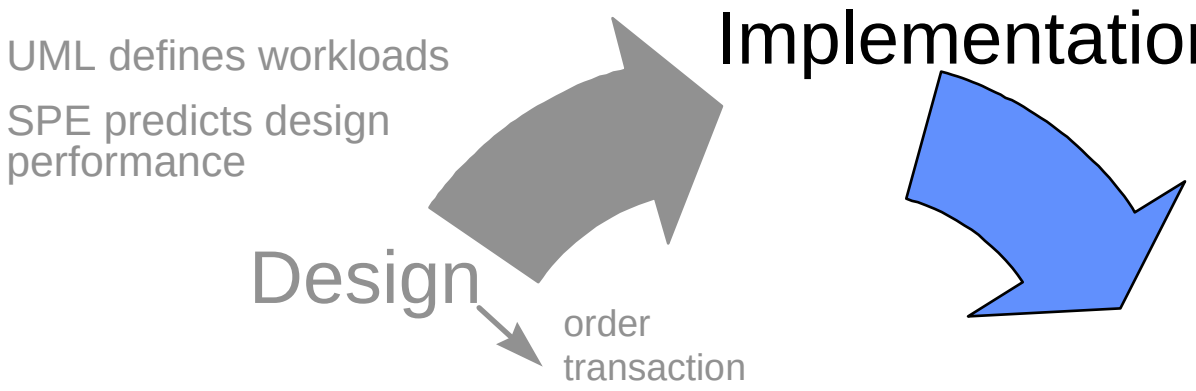
Application Life Cycle Example

UML defines workloads

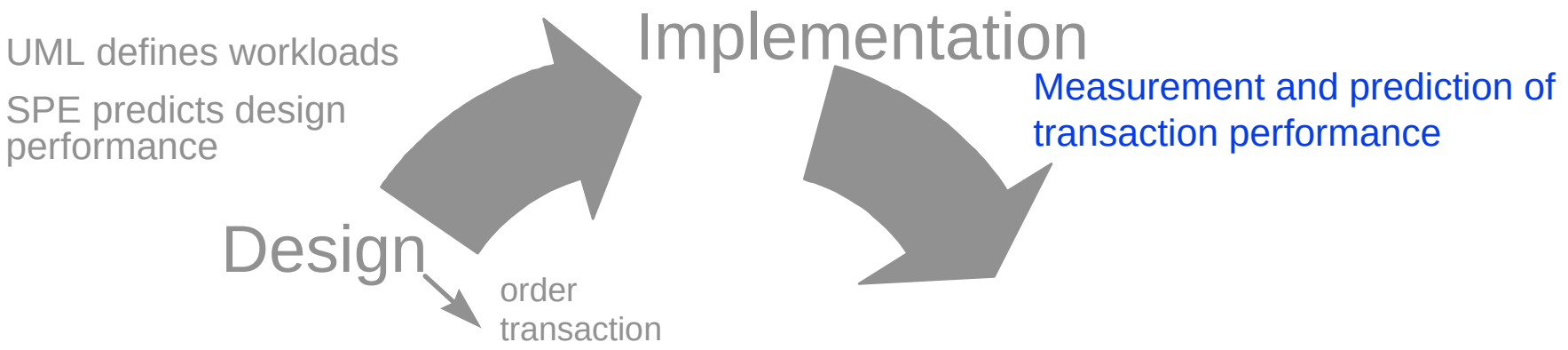
SPE predicts design performance



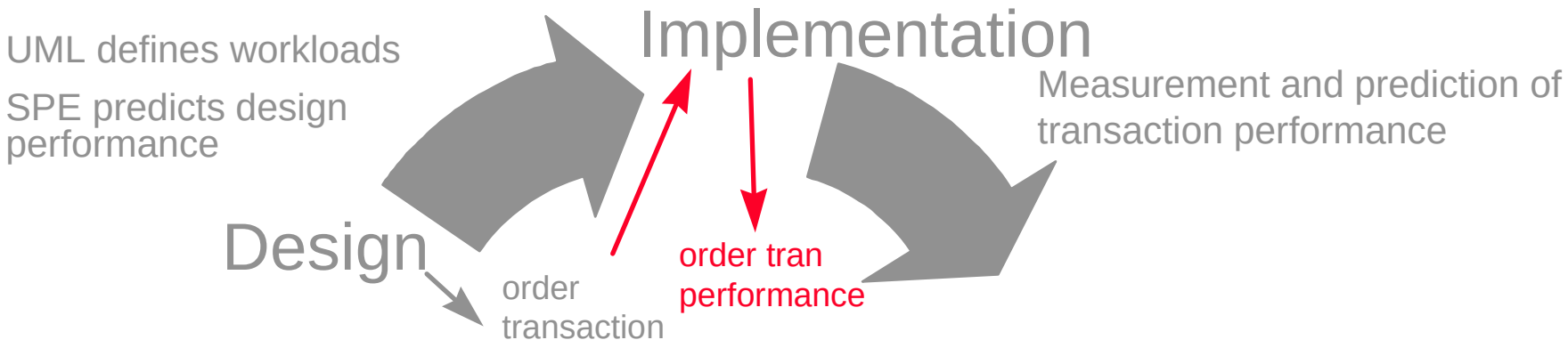
Application Life Cycle Example



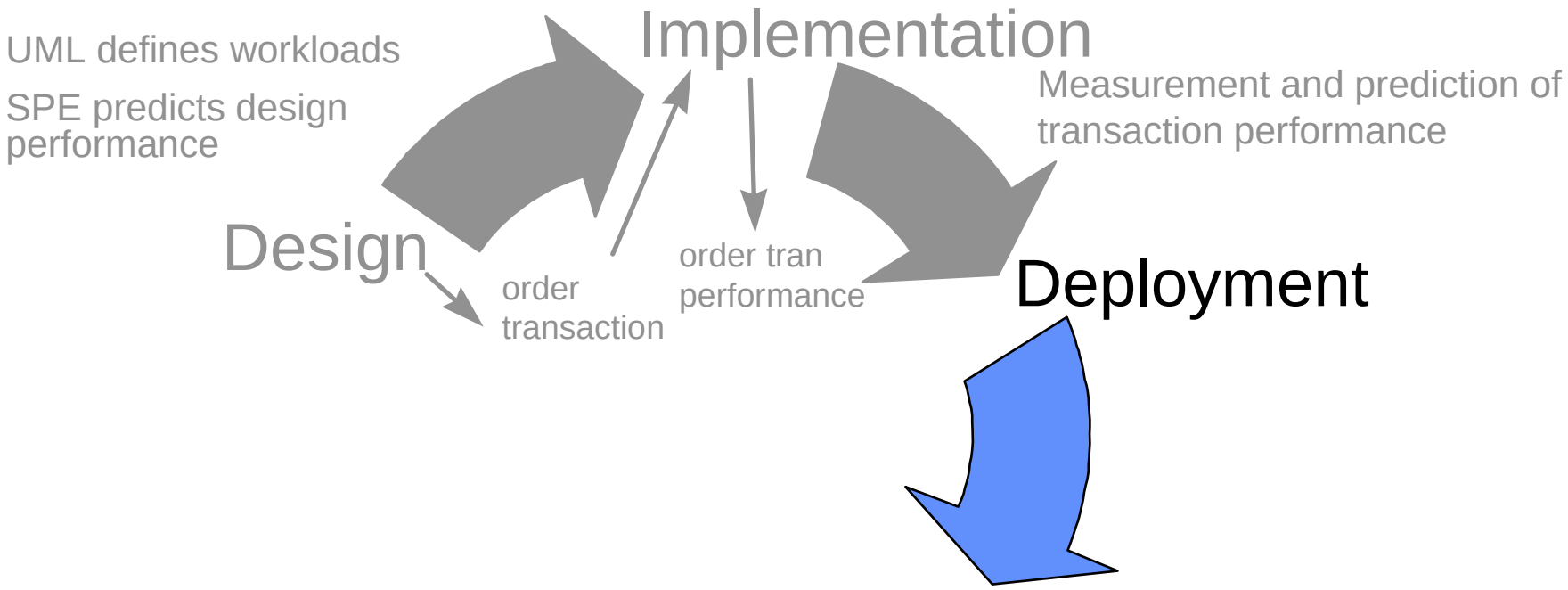
Application Life Cycle Example



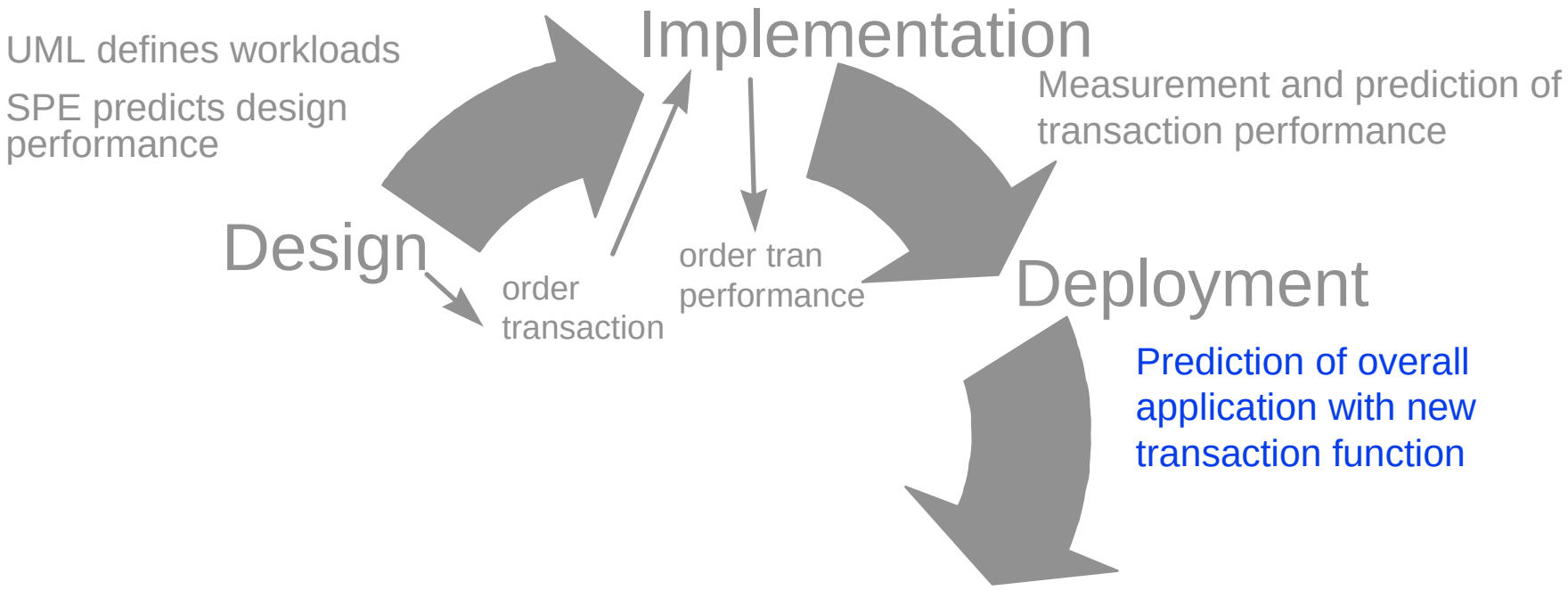
Application Life Cycle Example



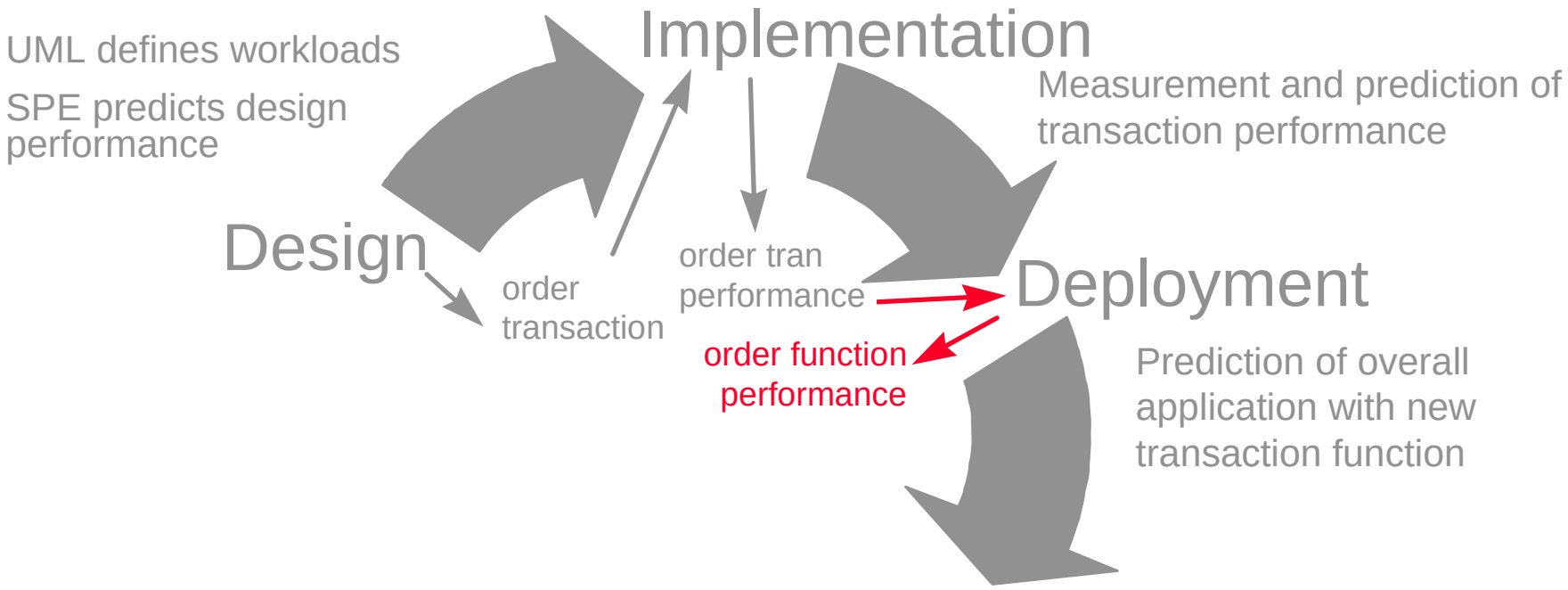
Application Life Cycle Example



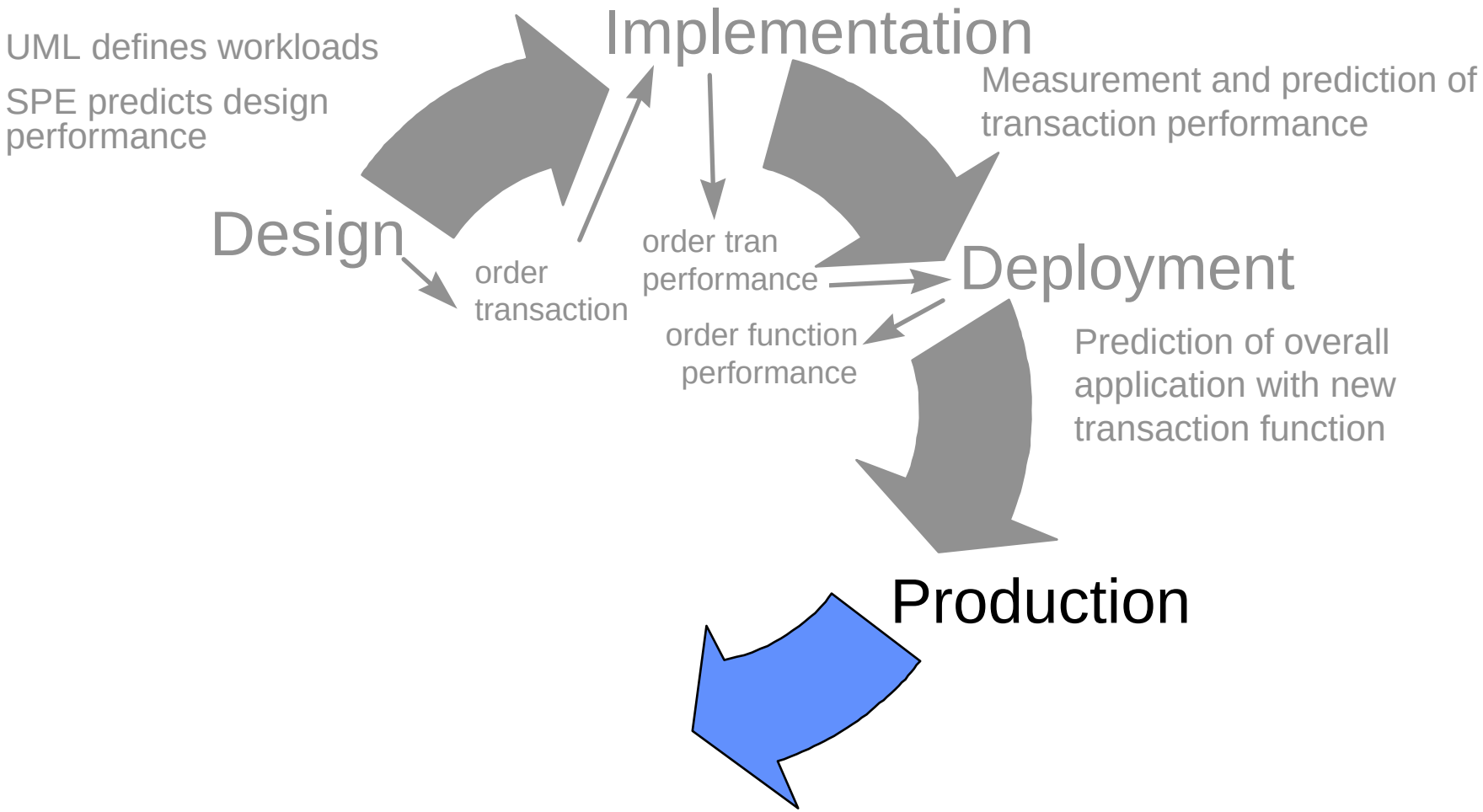
Application Life Cycle Example



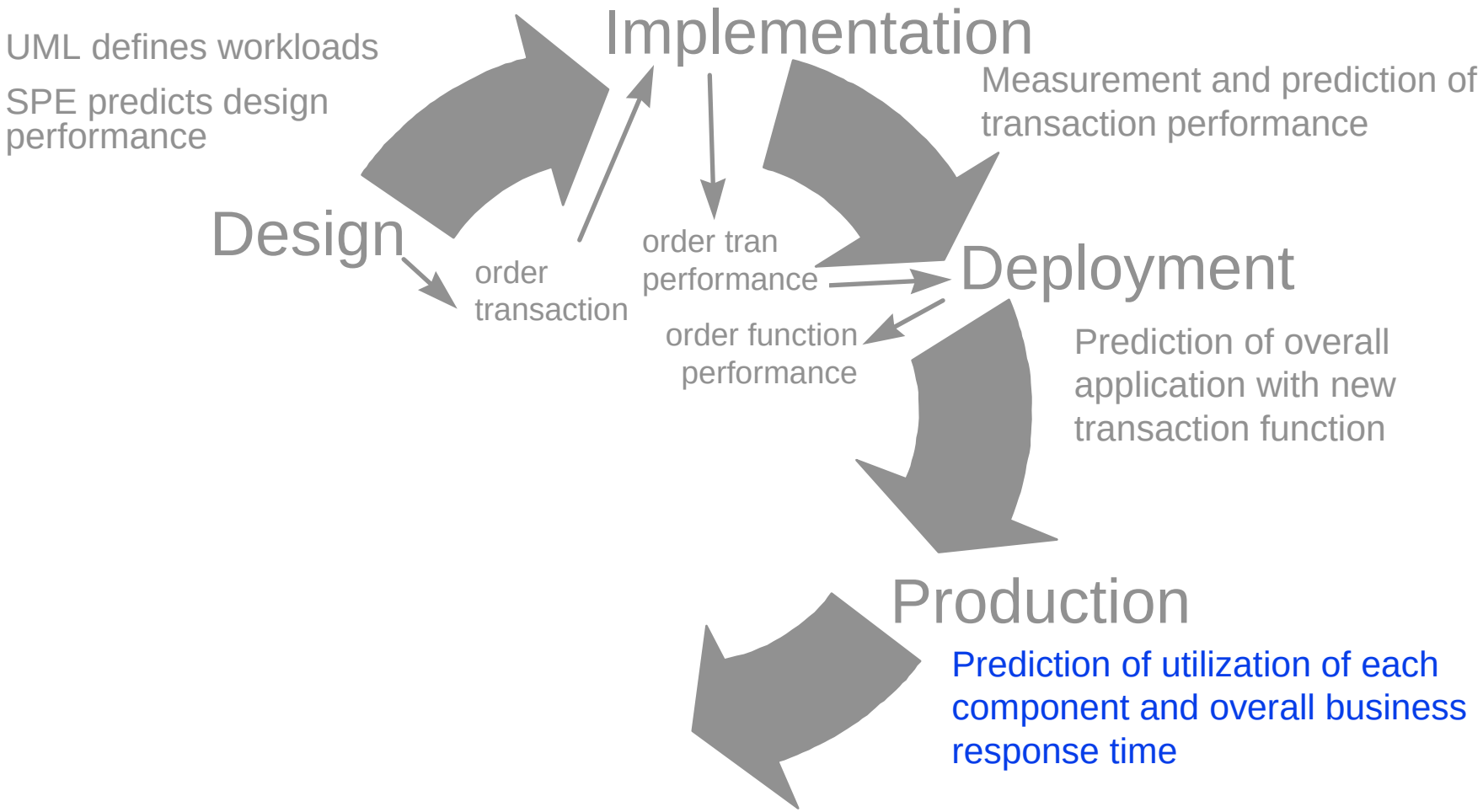
Application Life Cycle Example



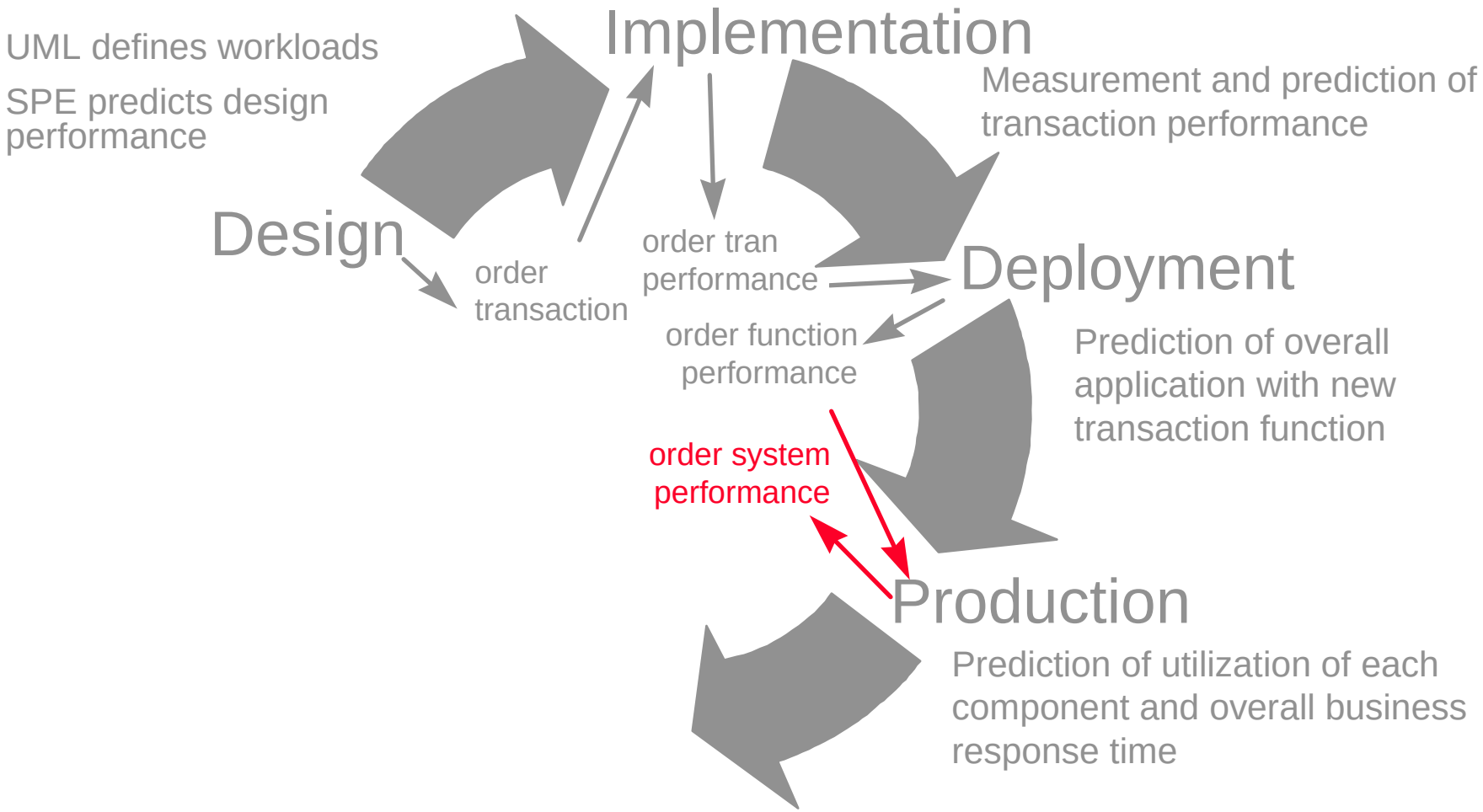
Application Life Cycle Example



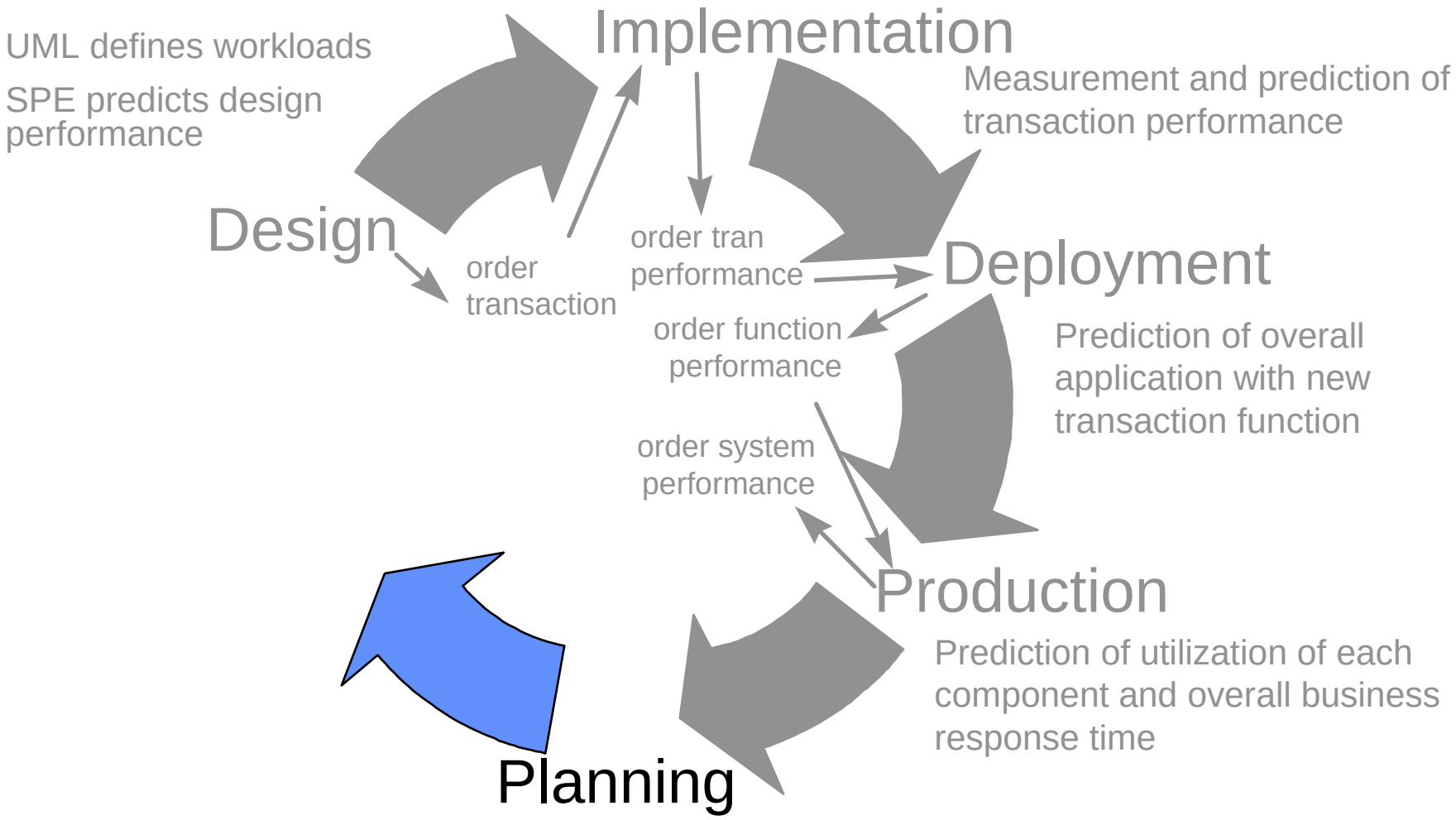
Application Life Cycle Example



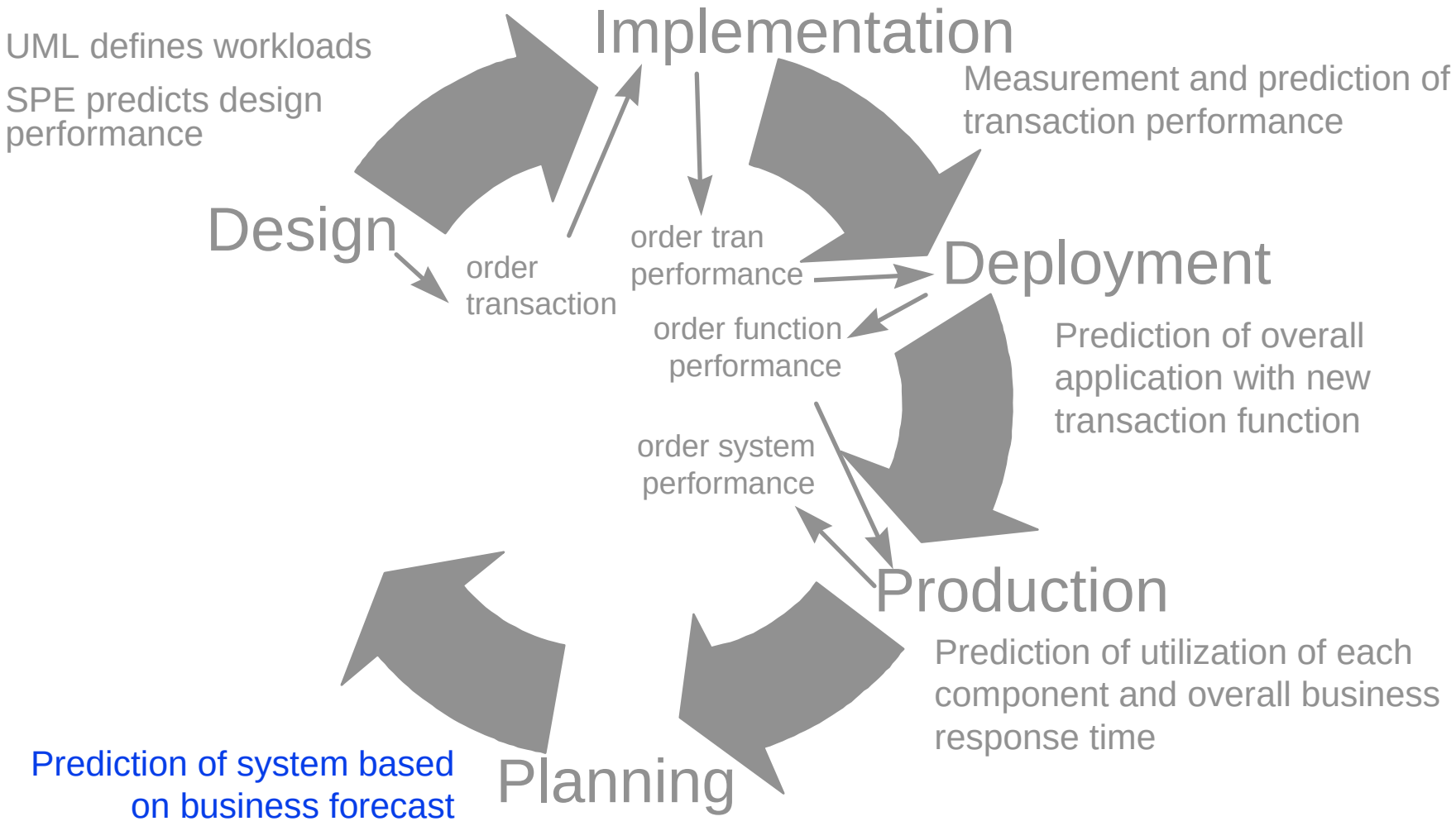
Application Life Cycle Example



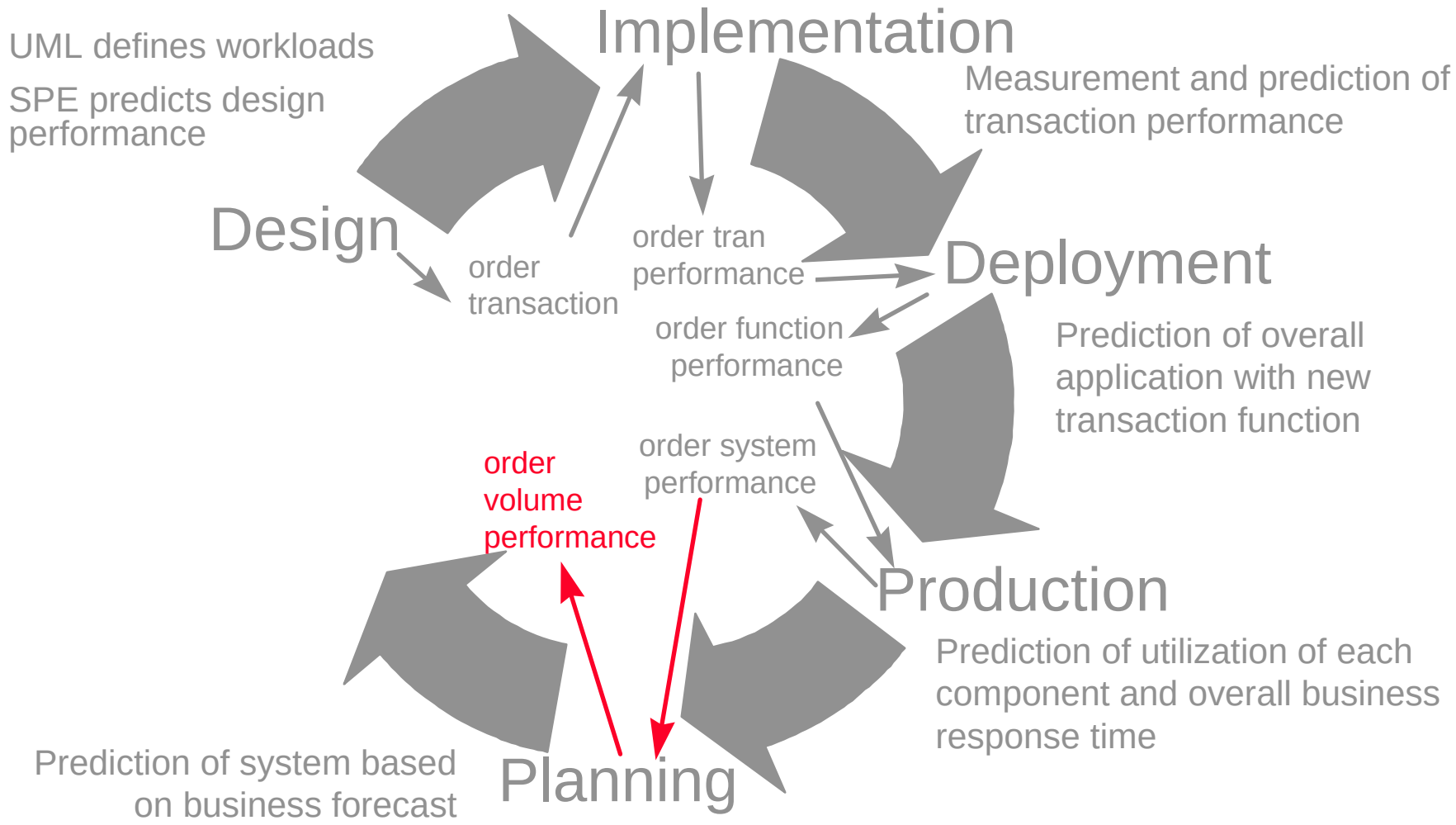
Application Life Cycle Example



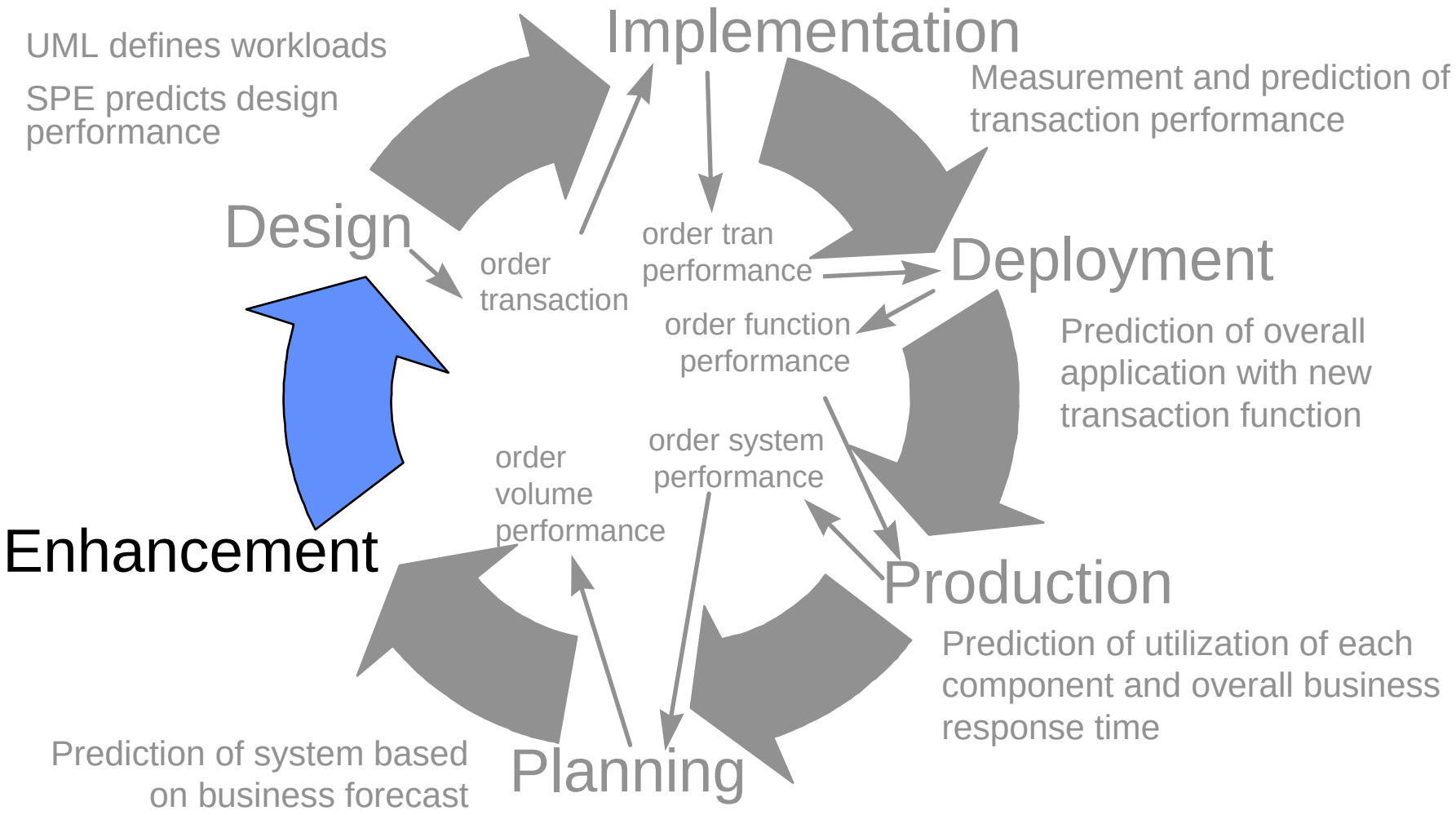
Application Life Cycle Example



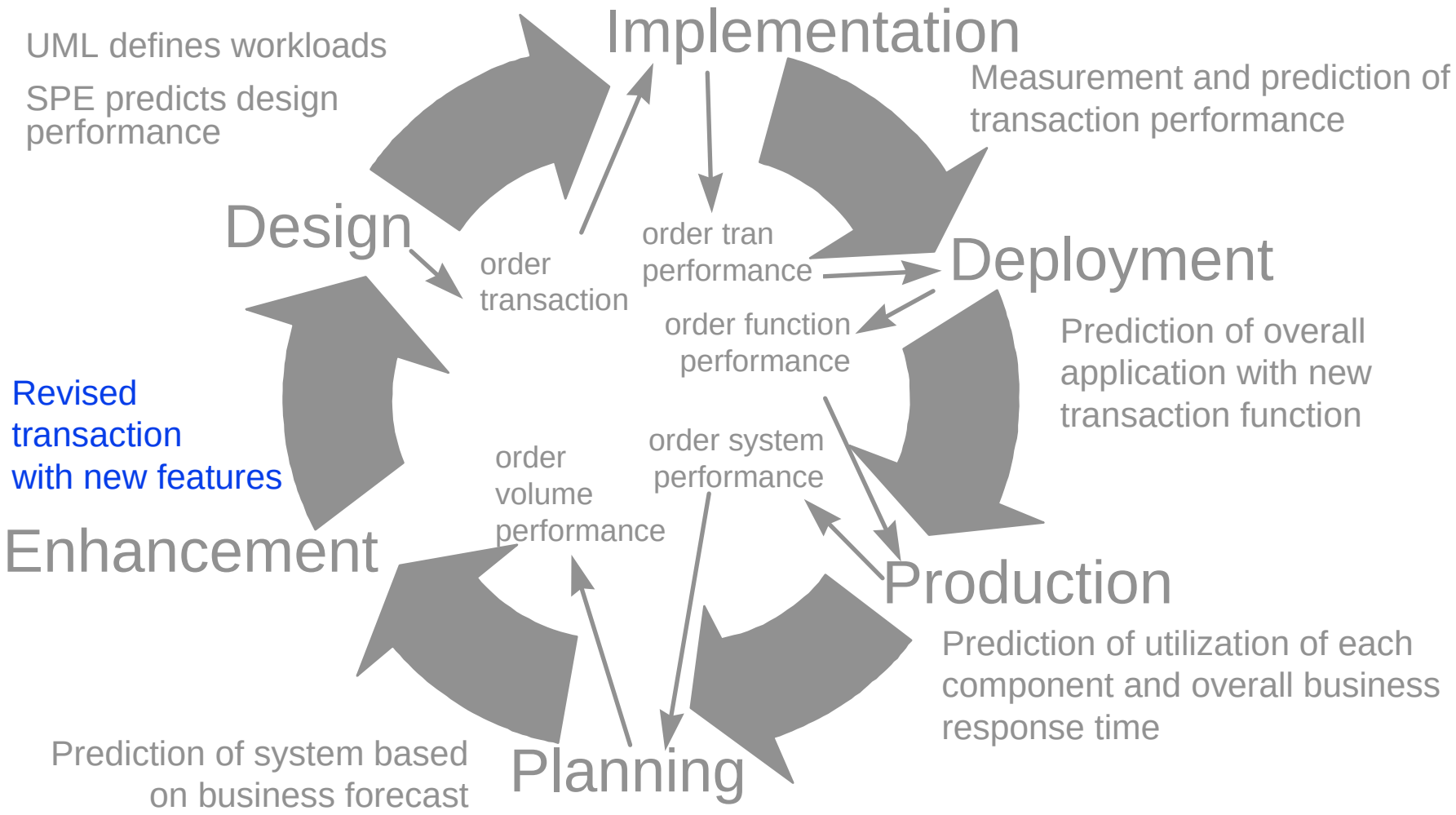
Application Life Cycle Example



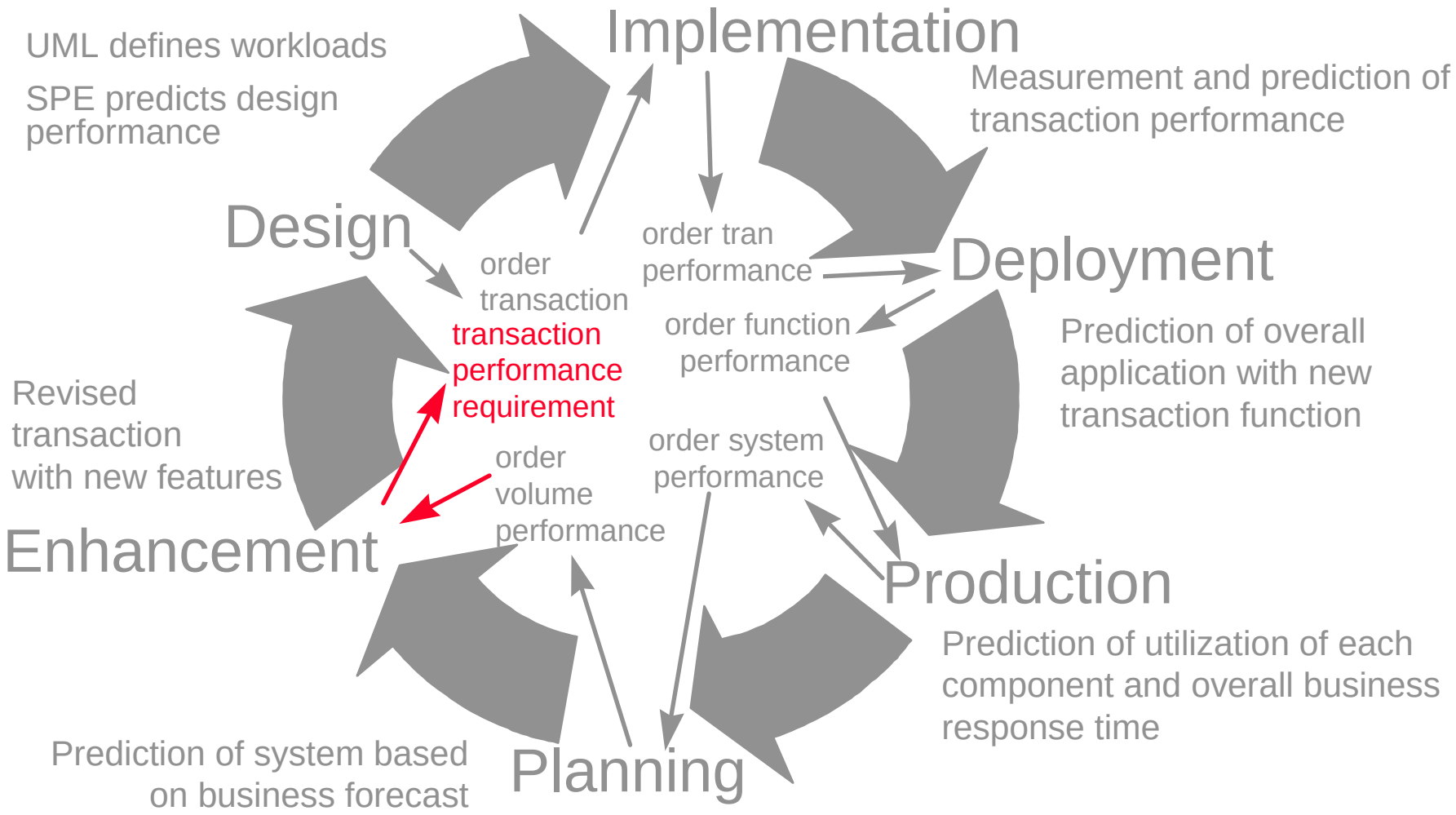
Application Life Cycle Example



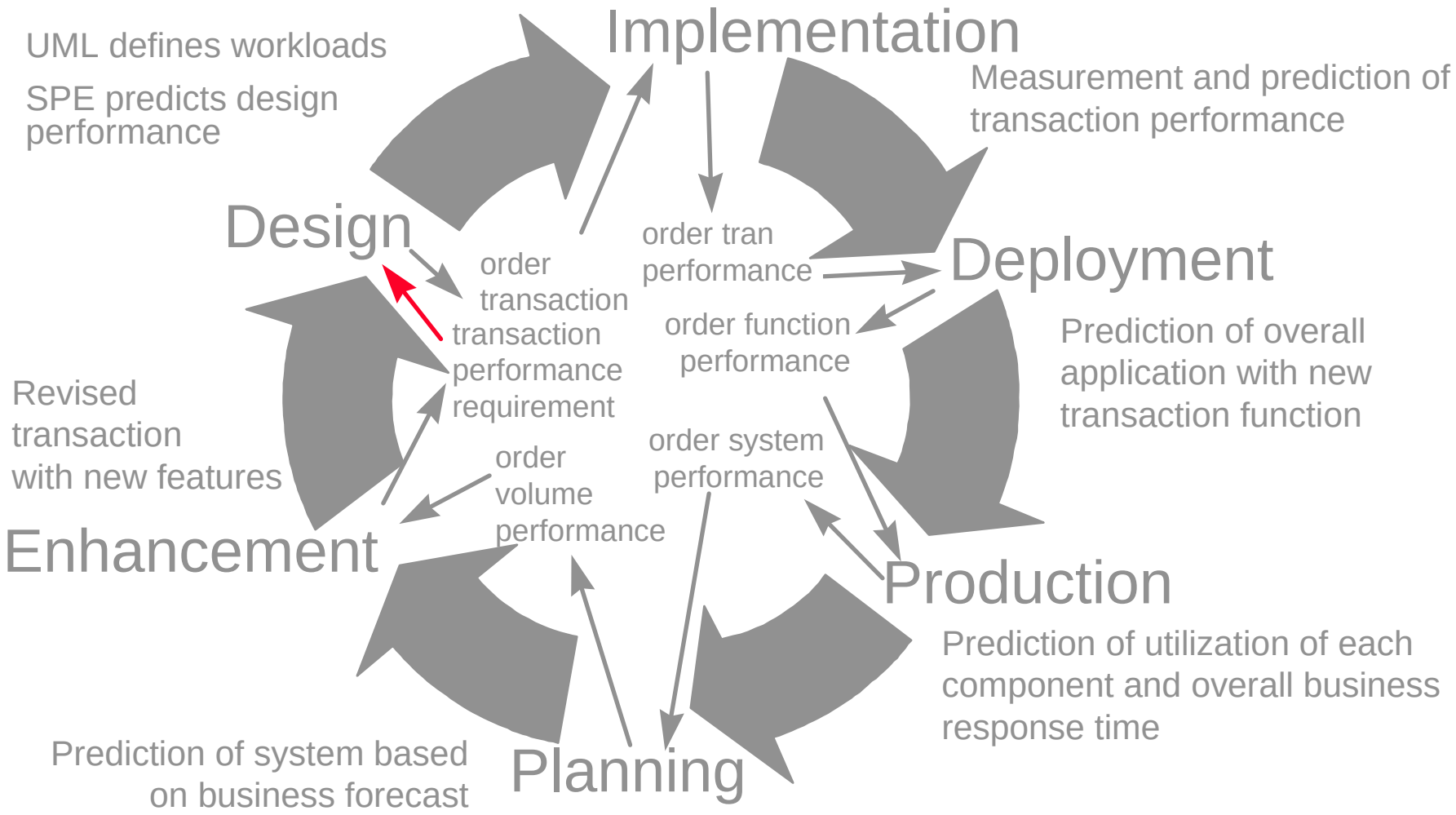
Application Life Cycle Example

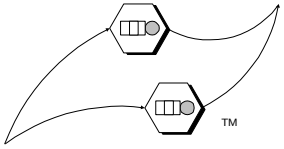


Application Life Cycle Example



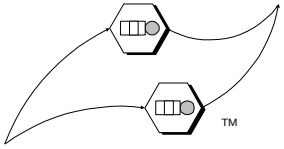
Application Life Cycle Example





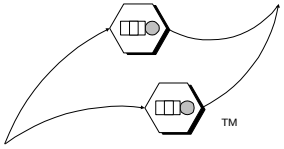
Modeling Techniques

- **General Modeling Tools**
  - Simulation
    - Tokens represent individual transactions
    - Examples:
      - DES - Discrete Event Simulation
      - Continuous Simulation - System Dynamics
  - Analytic
    - Mathematical description of the average transaction
    - Examples:
      - M/M/1Models
      - Queuing Network Models



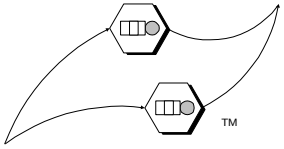
Modeling Techniques

- **Specialized Tools for Modeling**
  - UML - Unified Modeling Language
  - ETE-RT - End-To-End Response Time
  - SPE - Software Performance Engineering
  - Simalytic Modeling Technique
  - Business Models - System Dynamics
  - Node Models
  - Platform-Centric Models



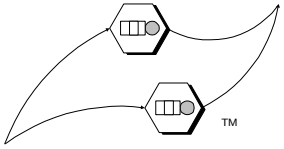
The Modeling Pyramid

- **Strategic Level**
  - Overall view to move an organization forward
    - Marketing or Business Plan
    - Projections by business area
  - Order Entry Example:
    - Customer growth
    - Product line growth
    - Replace paper catalog/orders with online functions
    - Metrics:
      - # customers, # products, # orders, etc.



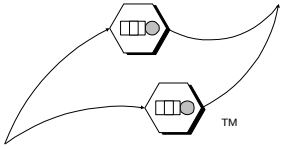
The Modeling Pyramid

- **Business Level**
  - Detailed view of business processes
    - Models predict how the entire process reacts to changes
  - Order Entry Example:
    - Staff to support additional customers
    - IT growth to support additional products/services
    - Metrics:
      - # calls, # operators, # telephone lines, etc.



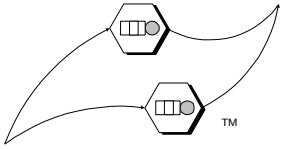
The Modeling Pyramid

- **Application Level**
  - Detailed view of how the application functions
    - Models predict how the application reacts to changes
  - Order Entry Example:
    - Number of OE transactions supported
    - OE transaction response time changes with growth
    - Impact of replacing Shipping paperwork with an on-line function
    - Metrics:
      - # transactions, tran response time, user think time, etc.



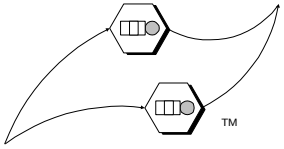
The Modeling Pyramid

- **System Level**
  - Broad view of all applications using specific hardware
    - Models predict system performance and application interactions
  - Order Entry Example:
    - Impact of other loads on OE database
    - Interference caused by system backups
    - Metrics:
      - # DB queries, processor utilization, memory load, etc.



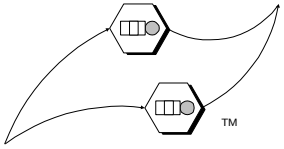
The Modeling Pyramid

- **Infrastructure Level**
  - Detailed view of the interconnections in the system
    - Models predict environment performance and interactions between systems
  - Order Entry Example:
    - RT impact of message latency to Shipping server
    - Impact of transaction growth on message latency
    - Metrics:
      - # DB I/Os, # network I/Os, I/O response time , etc.

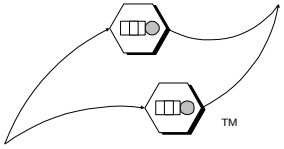


The Modeling Pyramid

- **Component Level**
  - Most detailed view of underlying hardware
    - Models predict hardware performance for component selection
    - Components of one system may be systems themselves (i.e. network gateways or disk subsystems)
  - Order Entry Example:
    - RT impact of router or network segment upgrade
    - Metrics:
      - # packets, # router hops, packet latency, etc.

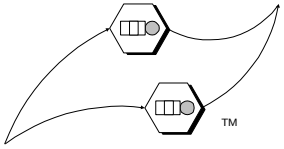


The Modeling Pyramid



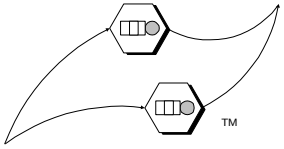
The Modeling Pyramid

More General Models



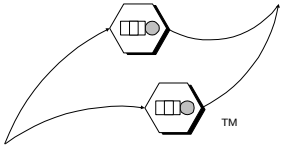
The Modeling Pyramid

More General Models



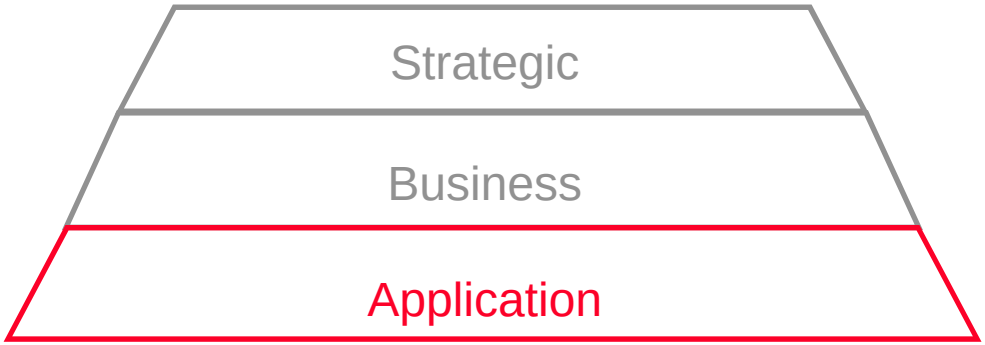
The Modeling Pyramid

More General Models

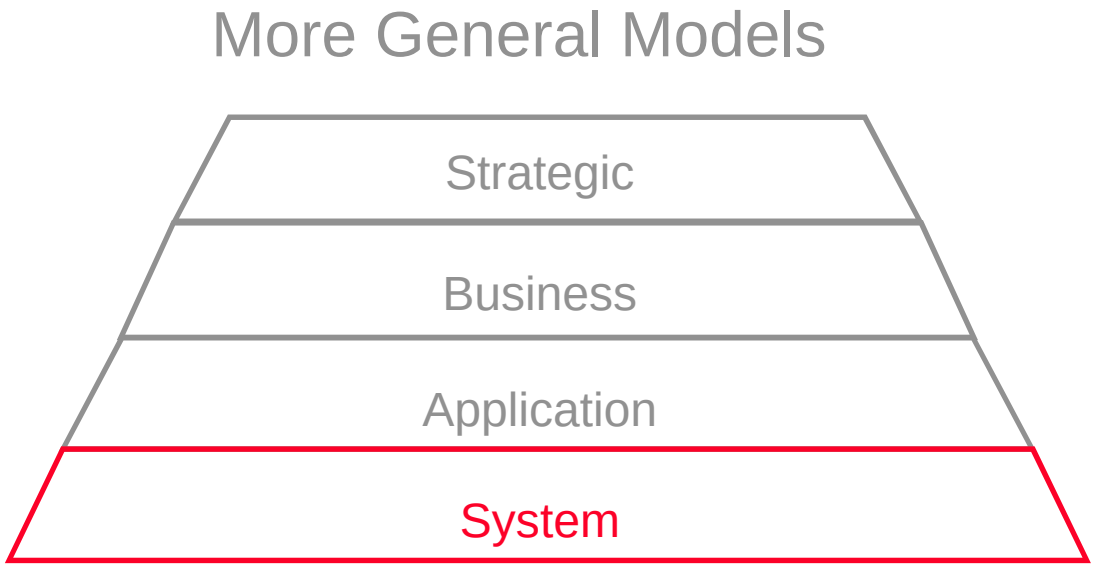


The Modeling Pyramid

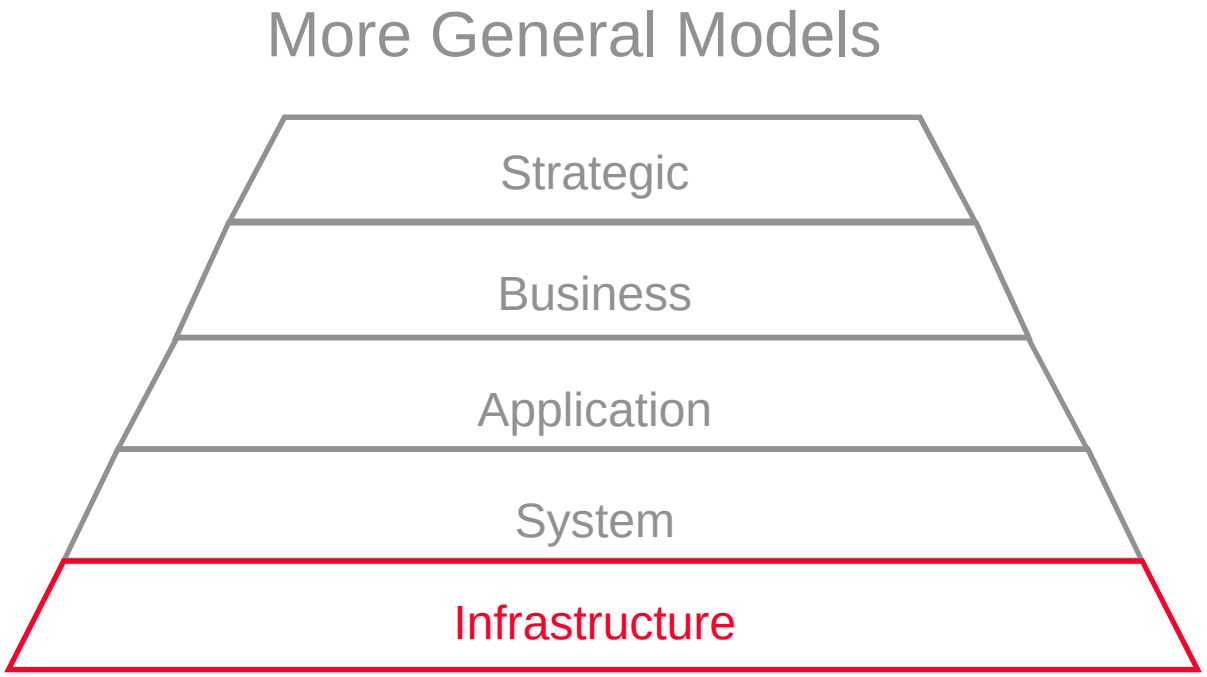
More General Models

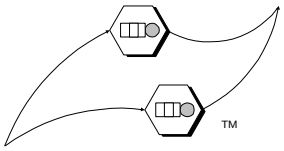


The Modeling Pyramid

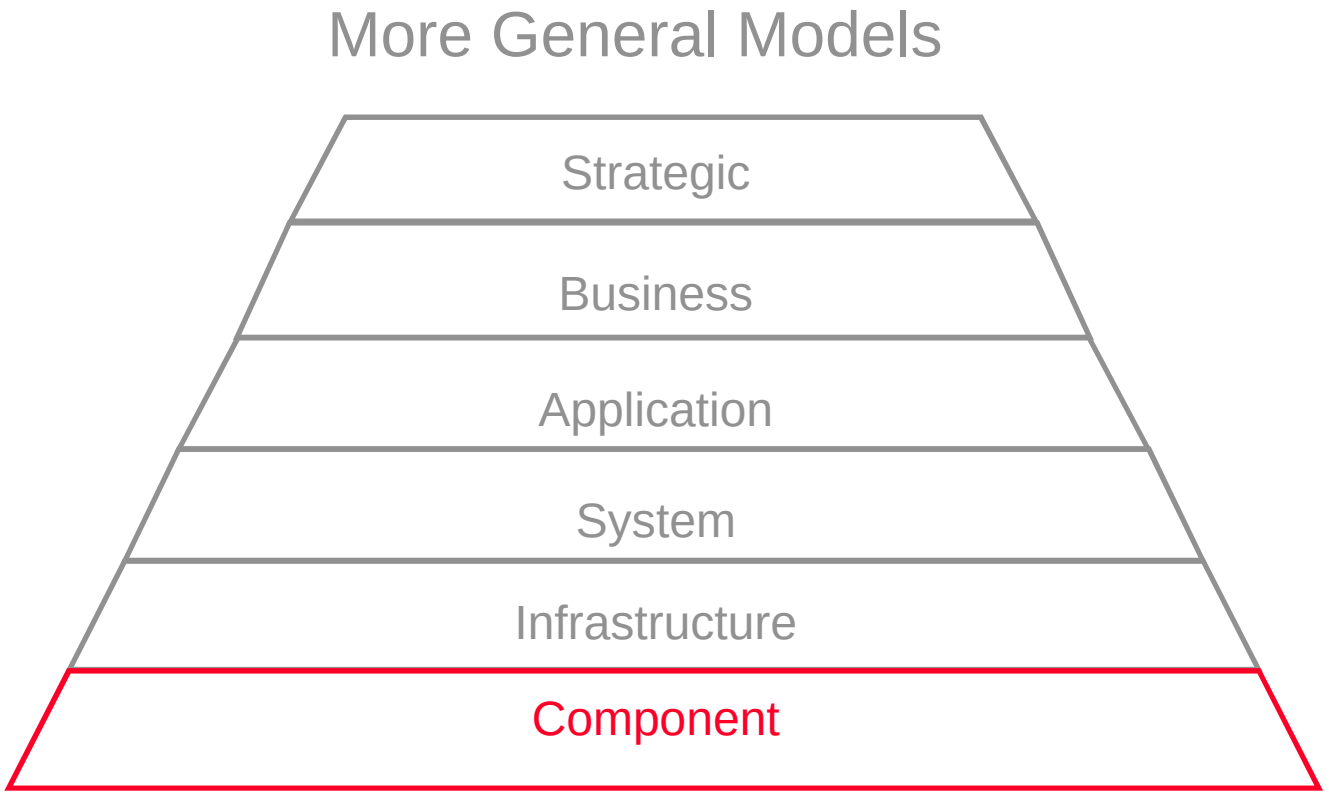


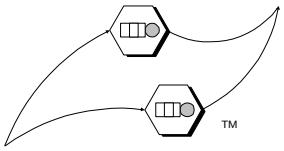
The Modeling Pyramid





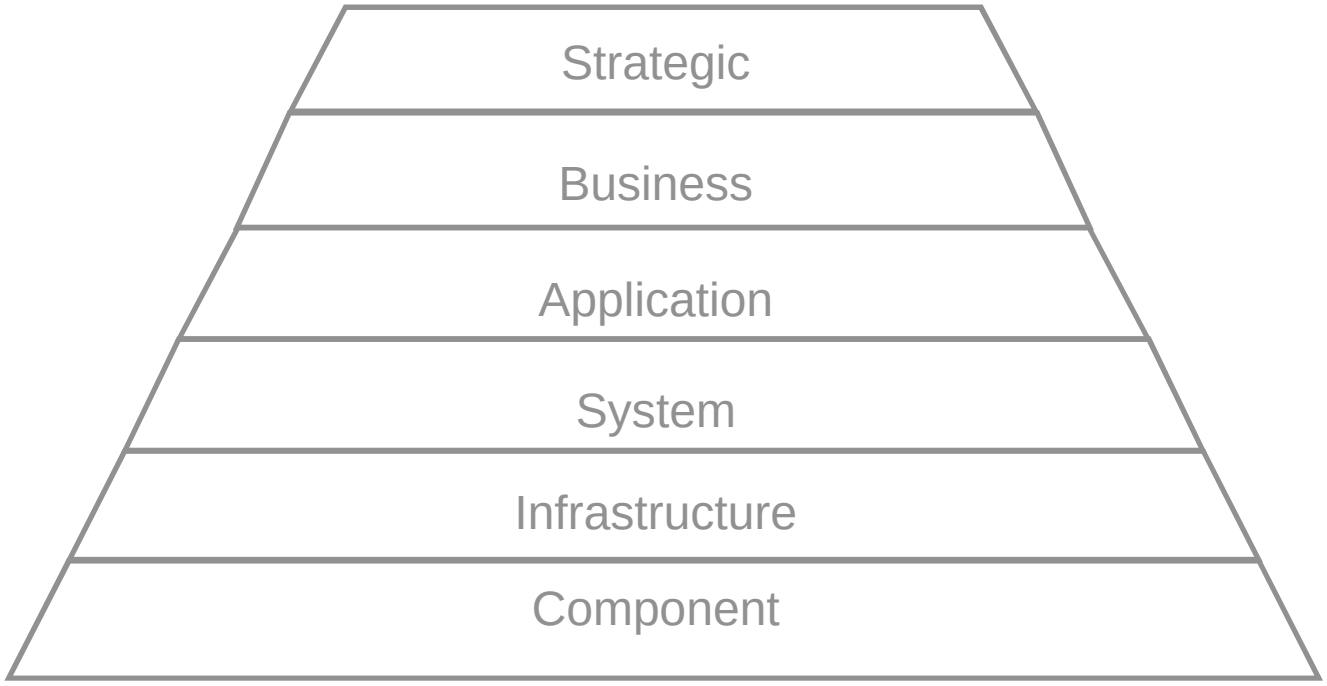
The Modeling Pyramid





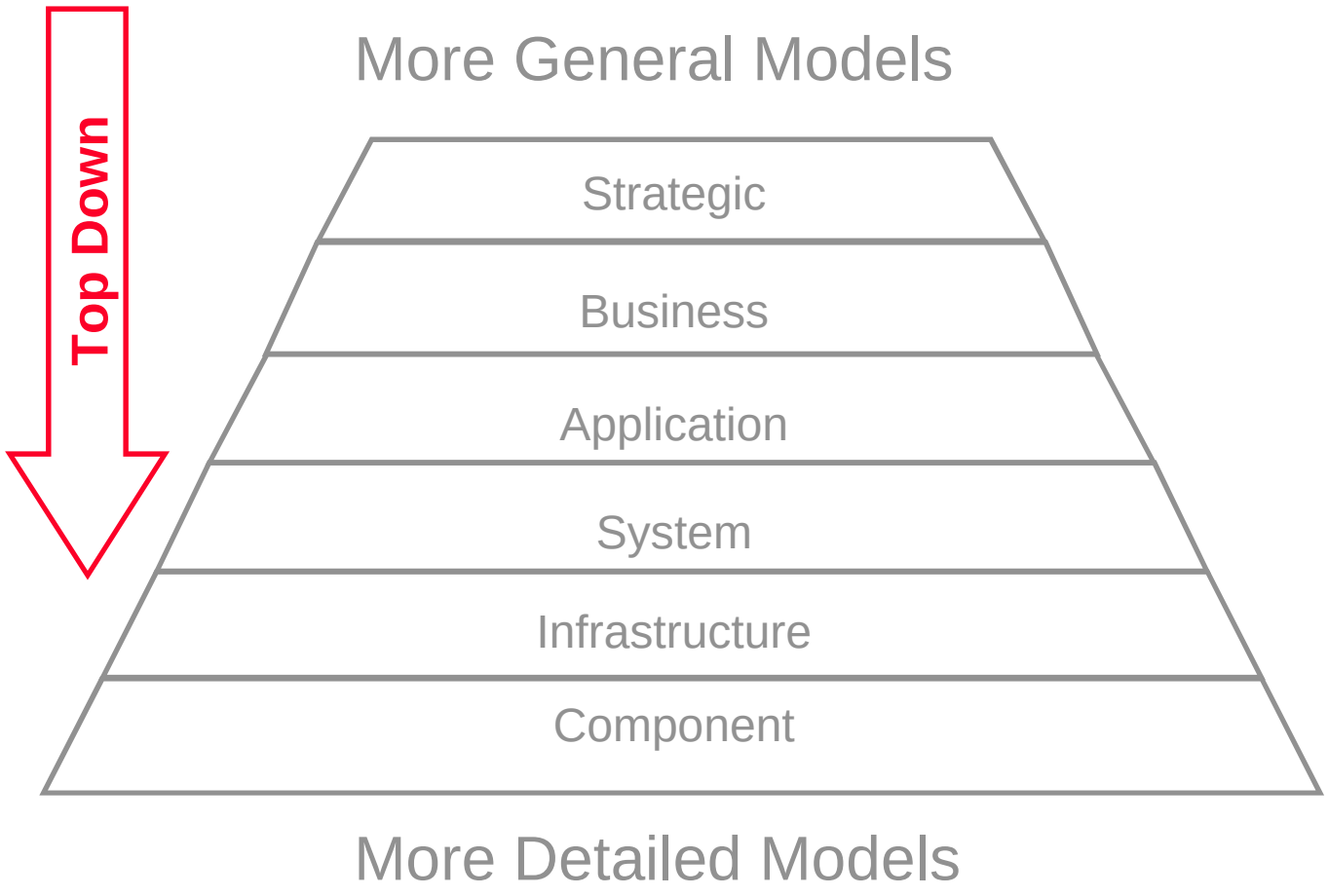
The Modeling Pyramid

More General Models

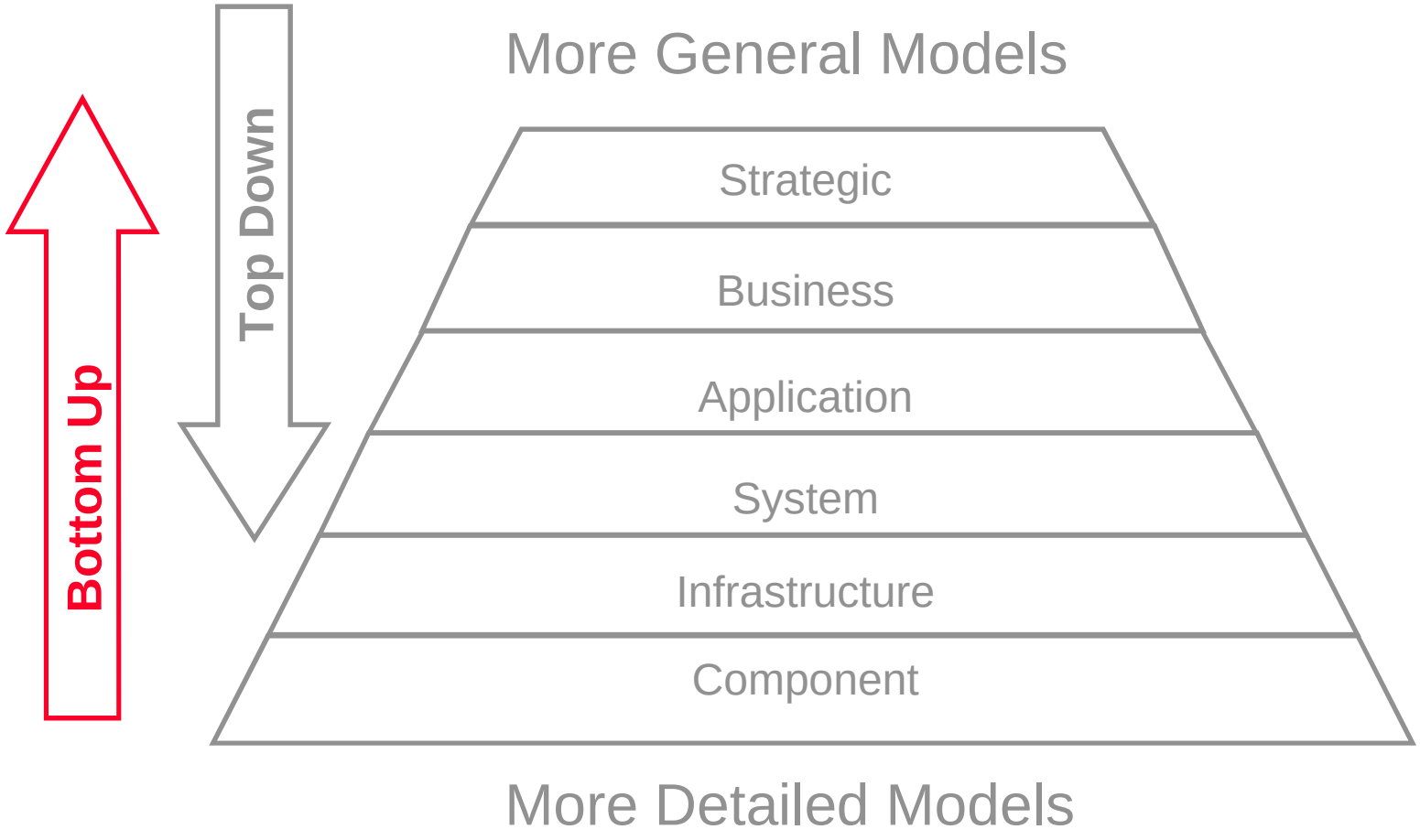


More Detailed Models

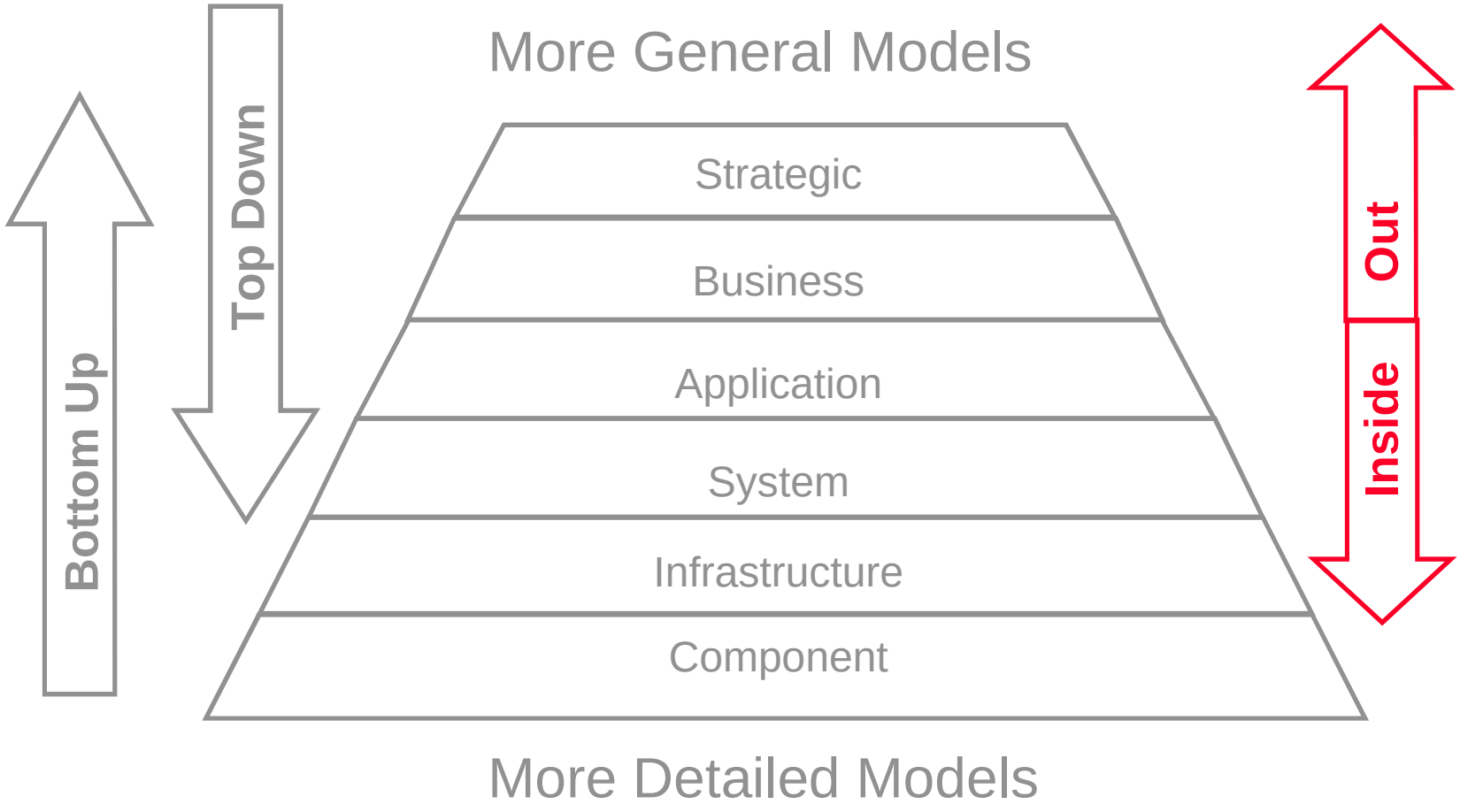
The Modeling Pyramid



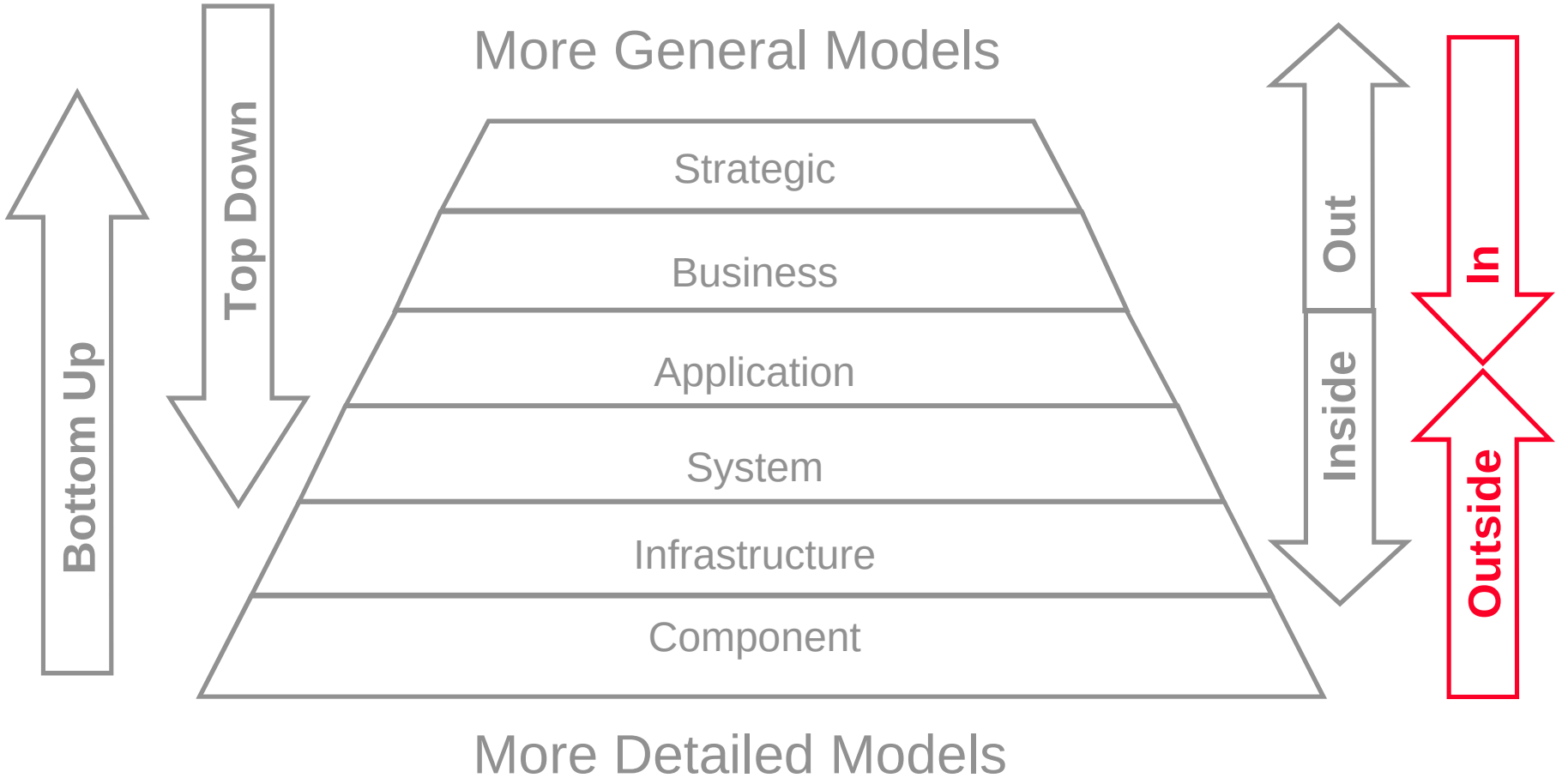
The Modeling Pyramid

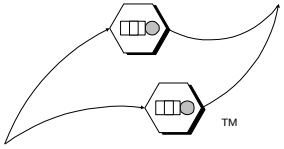


The Modeling Pyramid



The Modeling Pyramid

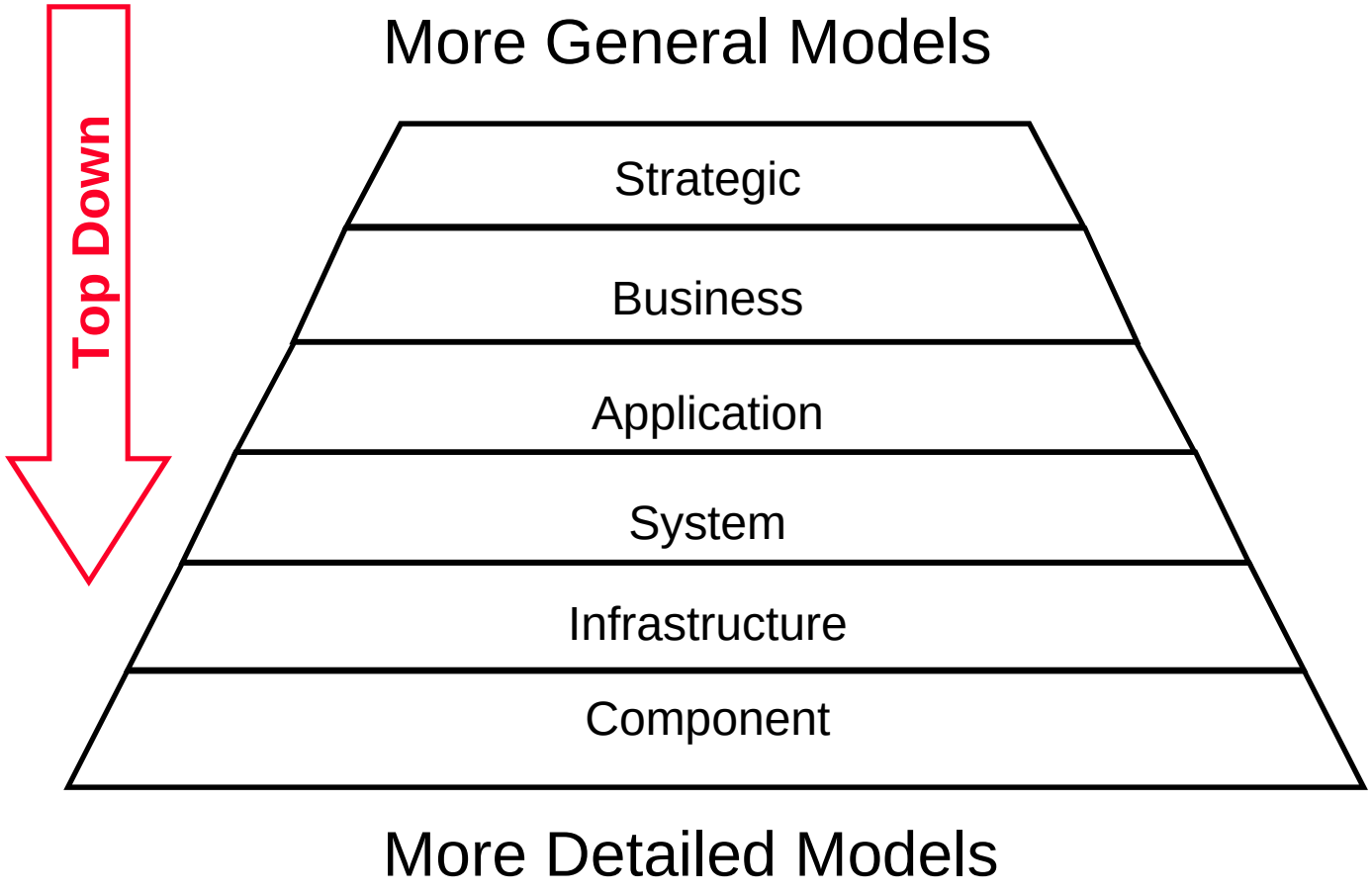




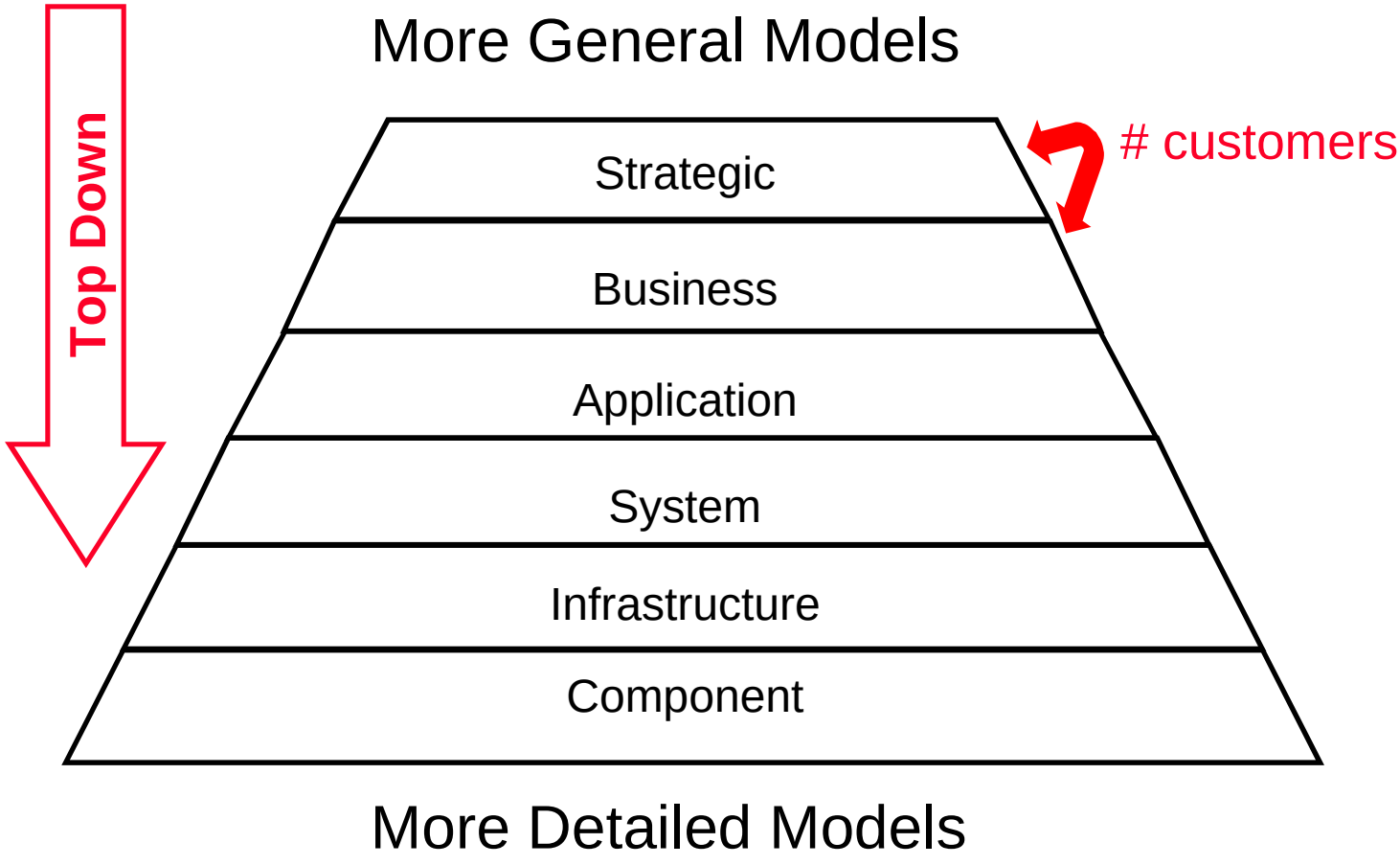
The Modeling Pyramid

- **Modeling Pyramid Implementation**
  - Top Down
    - Start at the most general (Strategic) level
    - Connect the metrics:
      - output of upper level
      - to input of lower level
    - Progress to the detailed (Component) level
      - Stop when the business question is answered
      - Avoid increased detail for its own sake

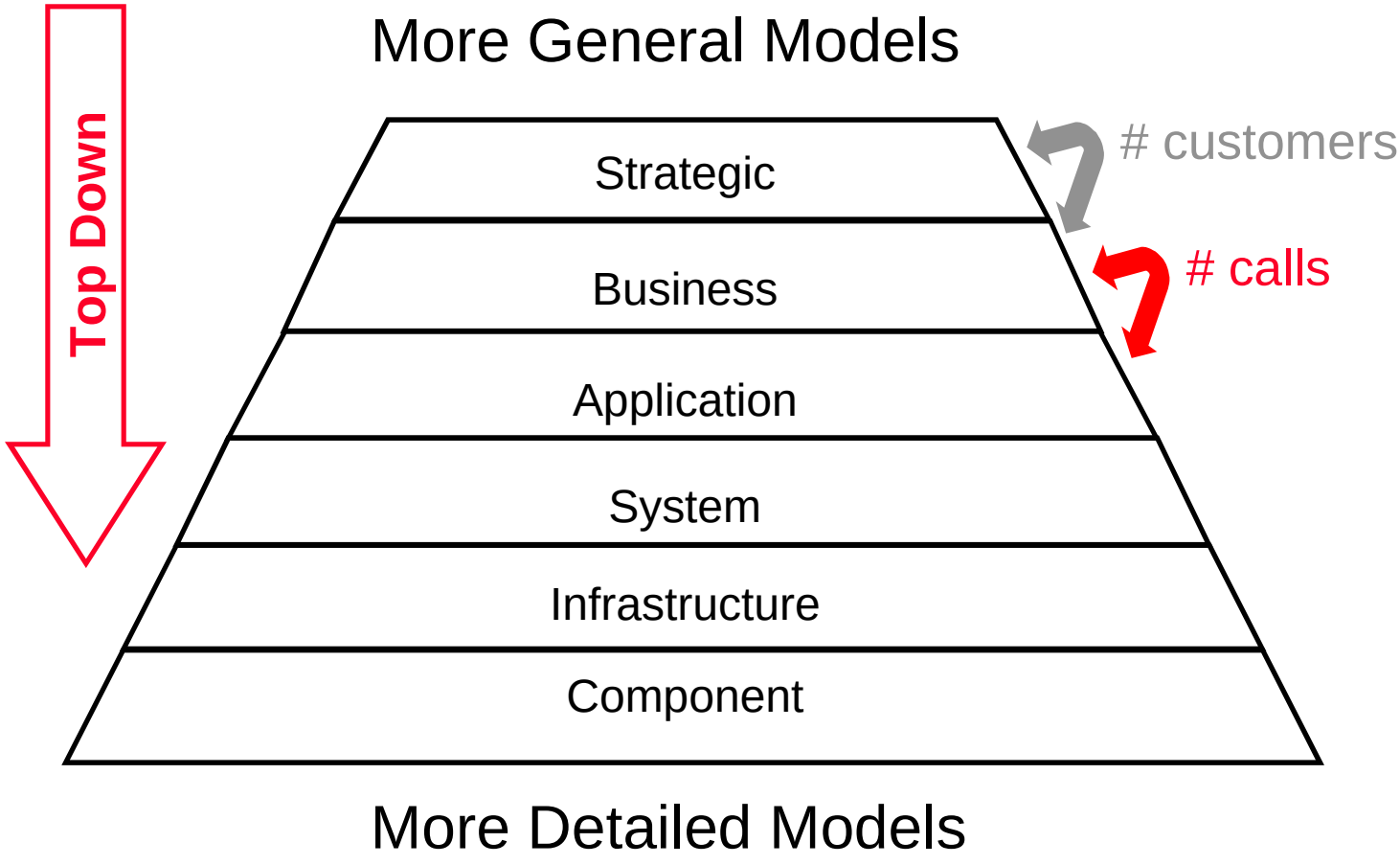
The Modeling Pyramid



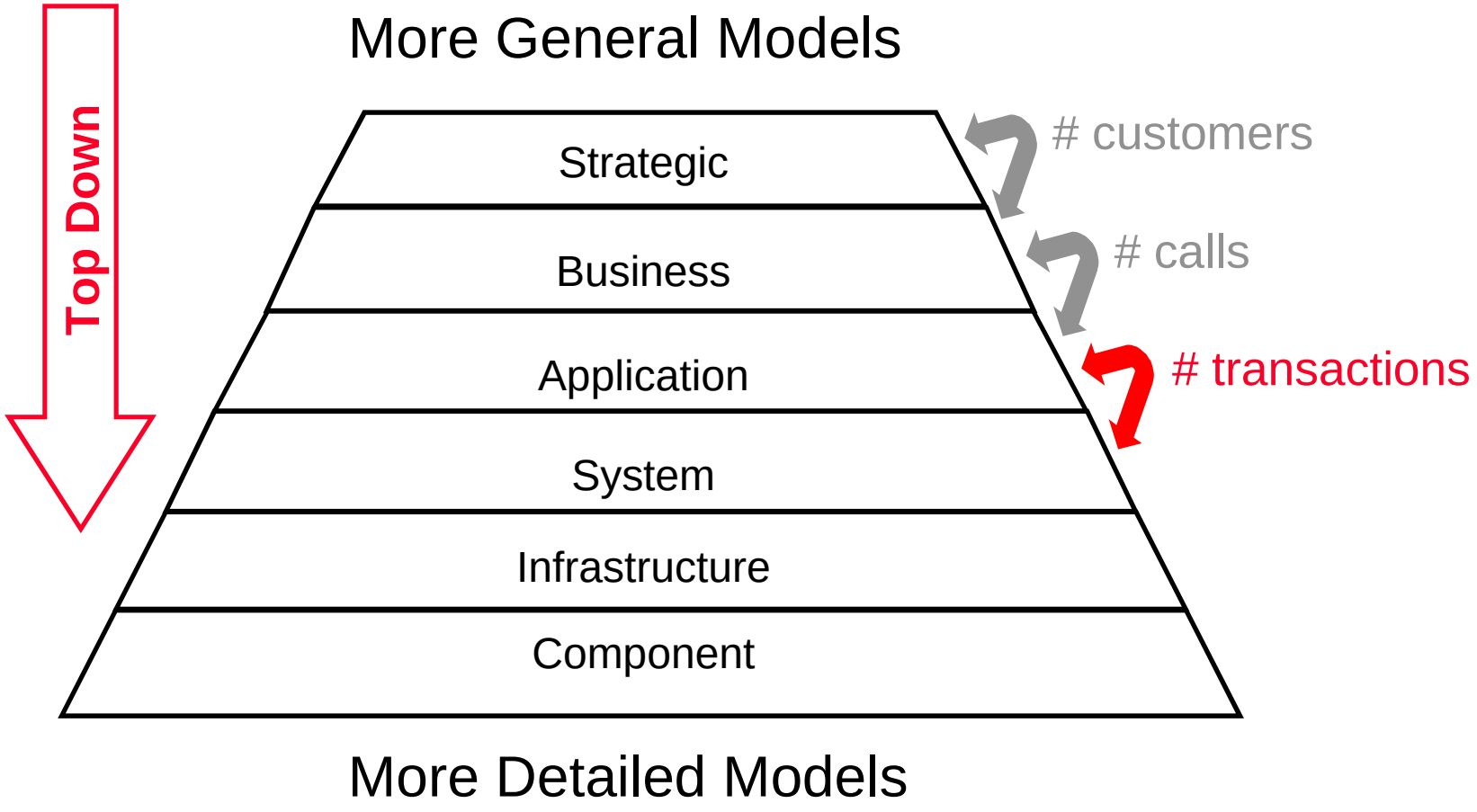
The Modeling Pyramid

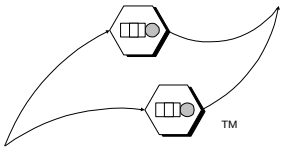


The Modeling Pyramid

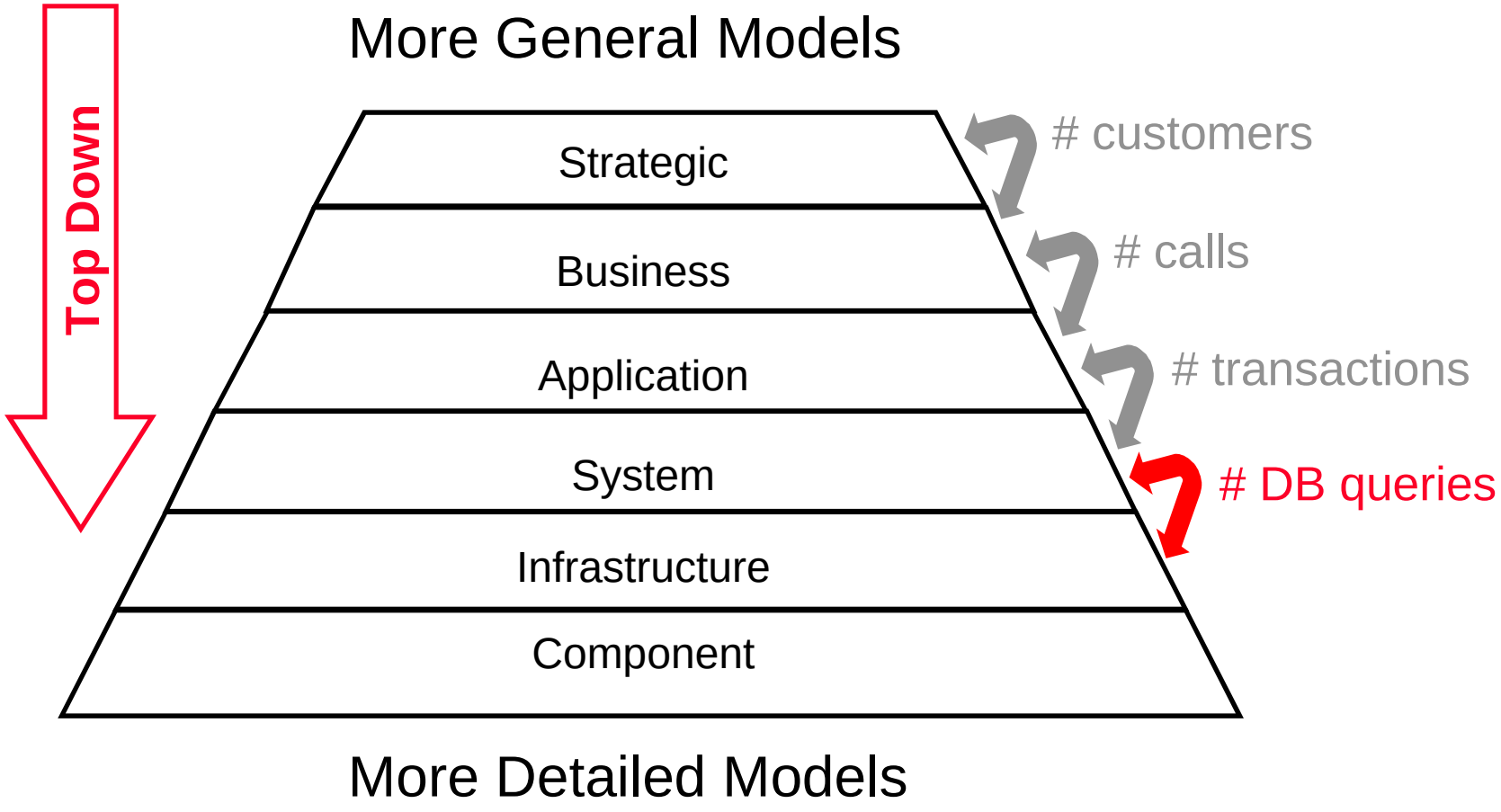


The Modeling Pyramid

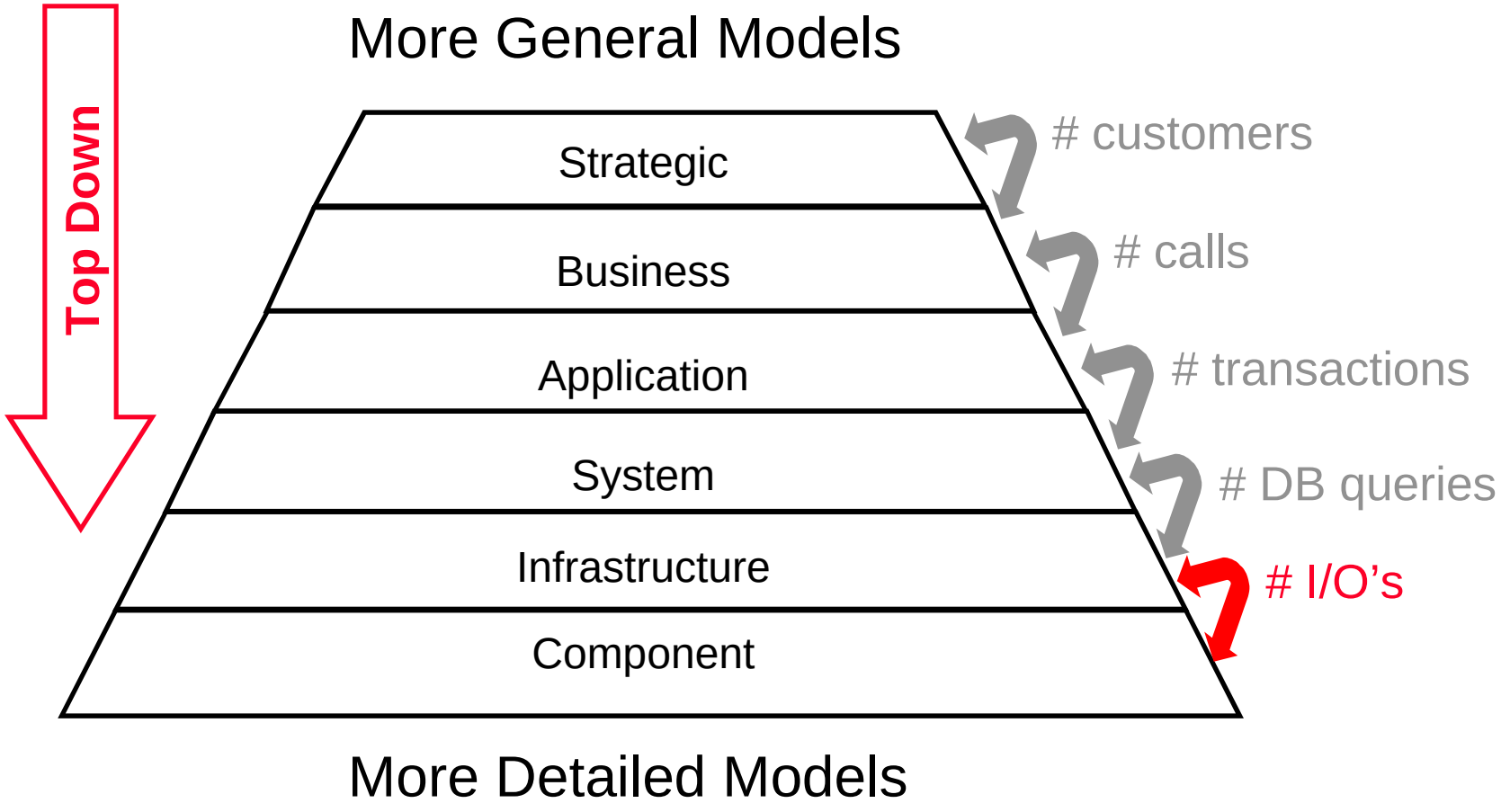


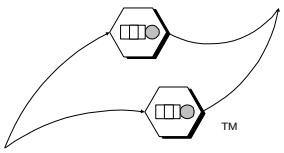


The Modeling Pyramid

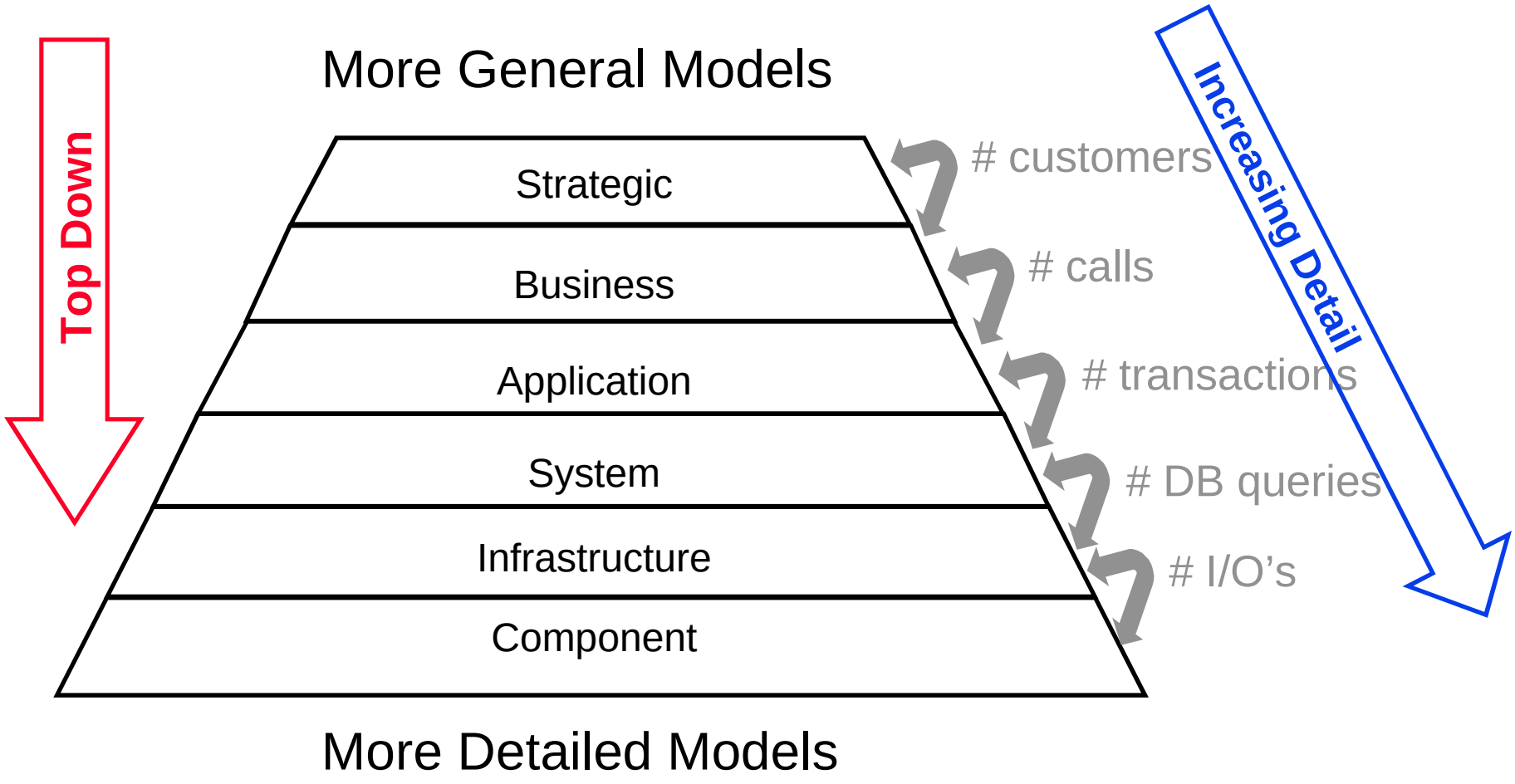


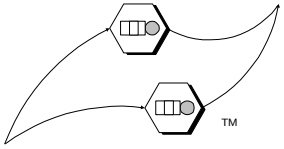
The Modeling Pyramid





The Modeling Pyramid

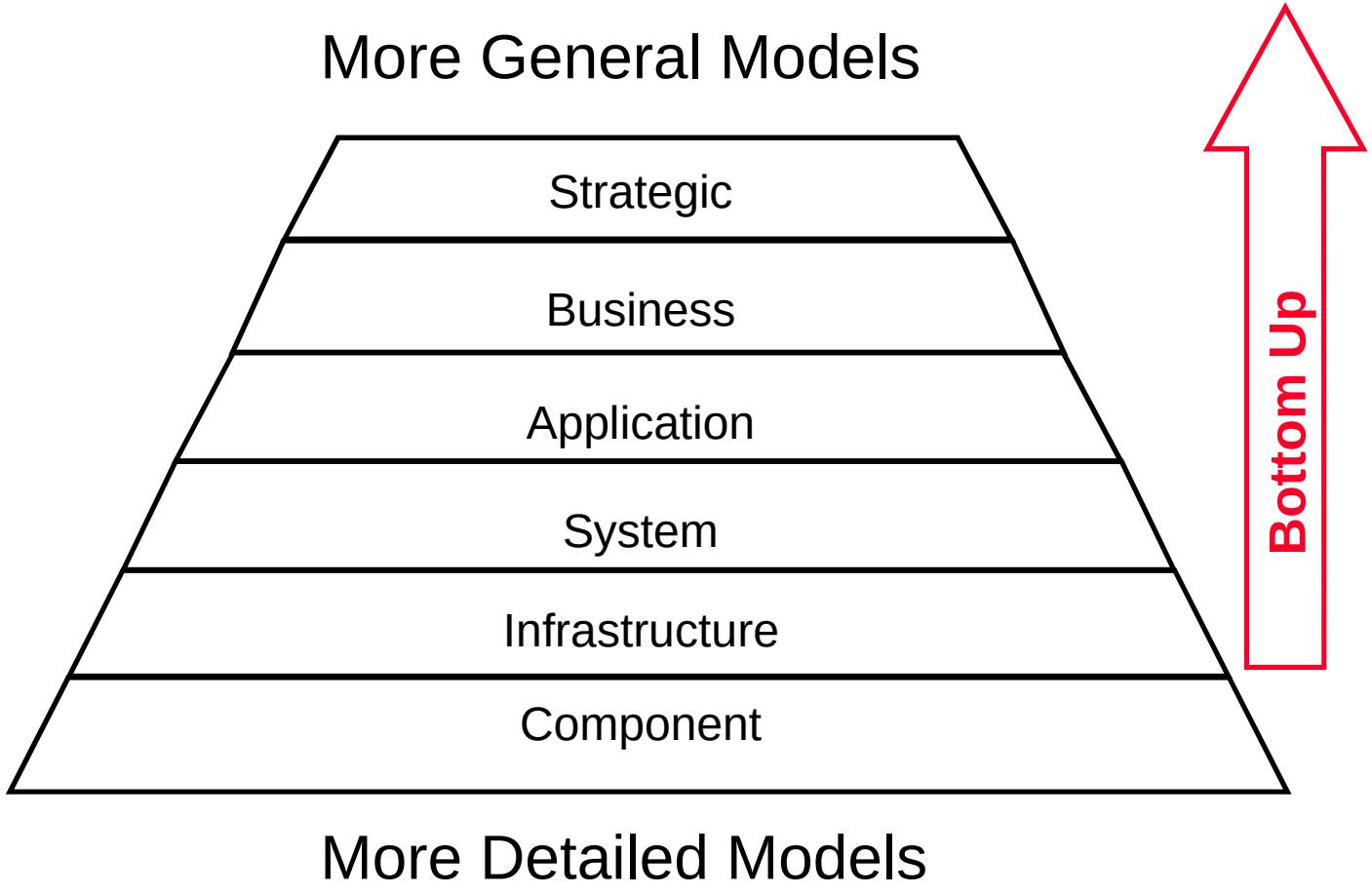




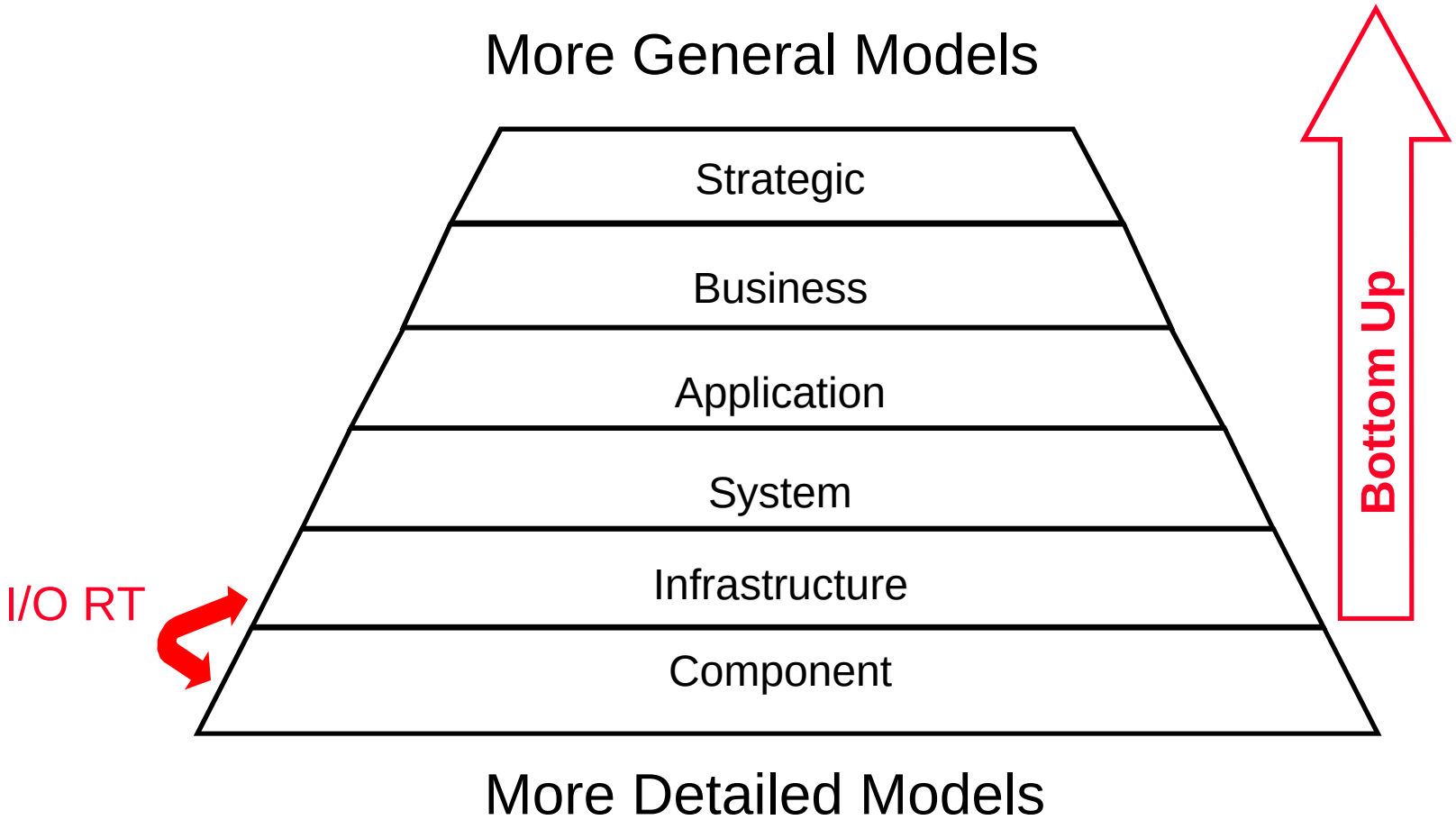
The Modeling Pyramid

- **Modeling Pyramid Implementation**
  - Bottom Up
    - Start at the most detailed (Component) level
    - Connect the metrics:
      - output of lower level
      - to input of upper level
    - Progress to the general (Strategic) level
      - Stop when the technical question is answered
      - Avoid speculation outside of the business objectives
        - A server upgrade would support 100x more customers when the business plan projects 3x growth.

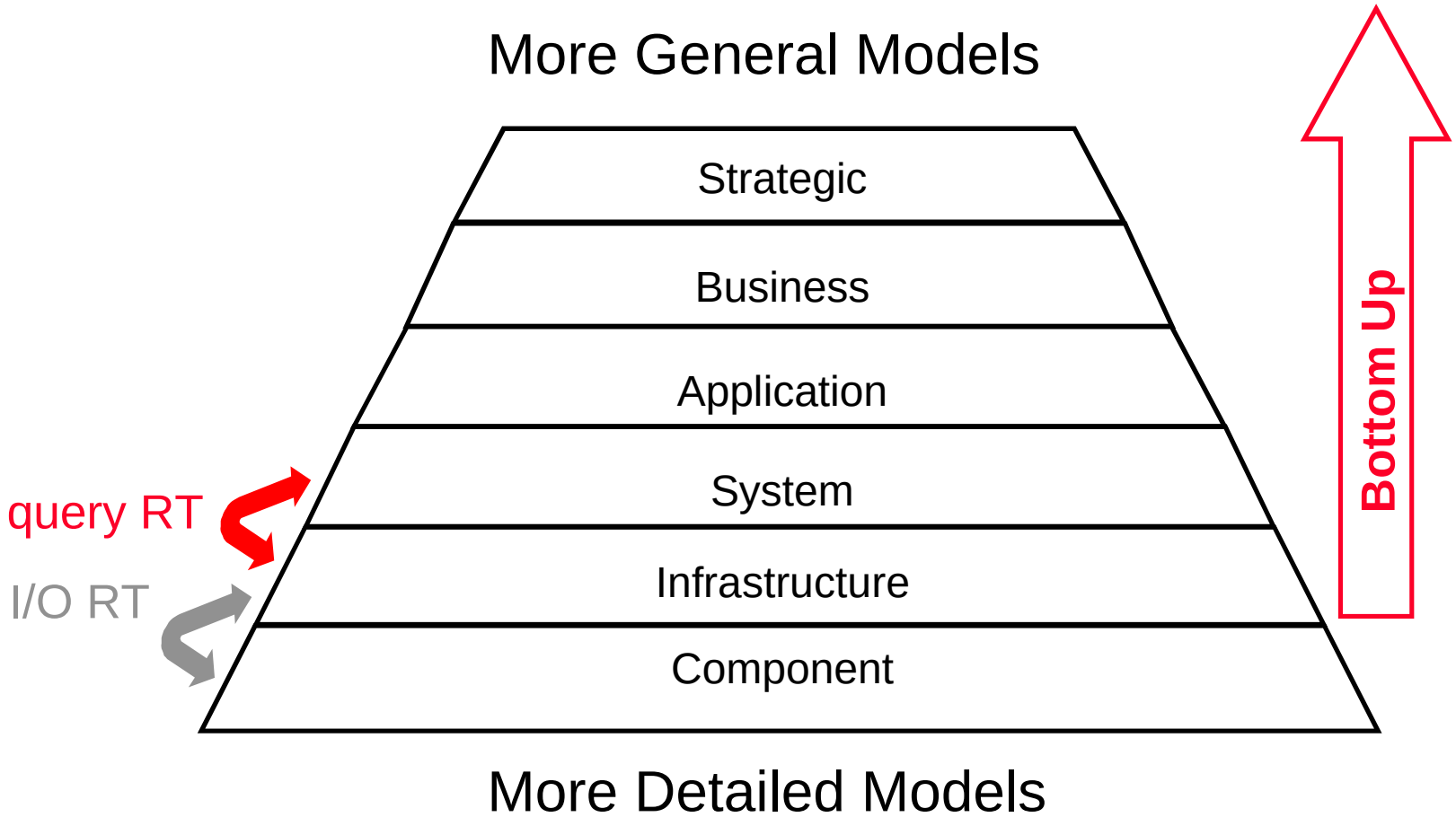
The Modeling Pyramid



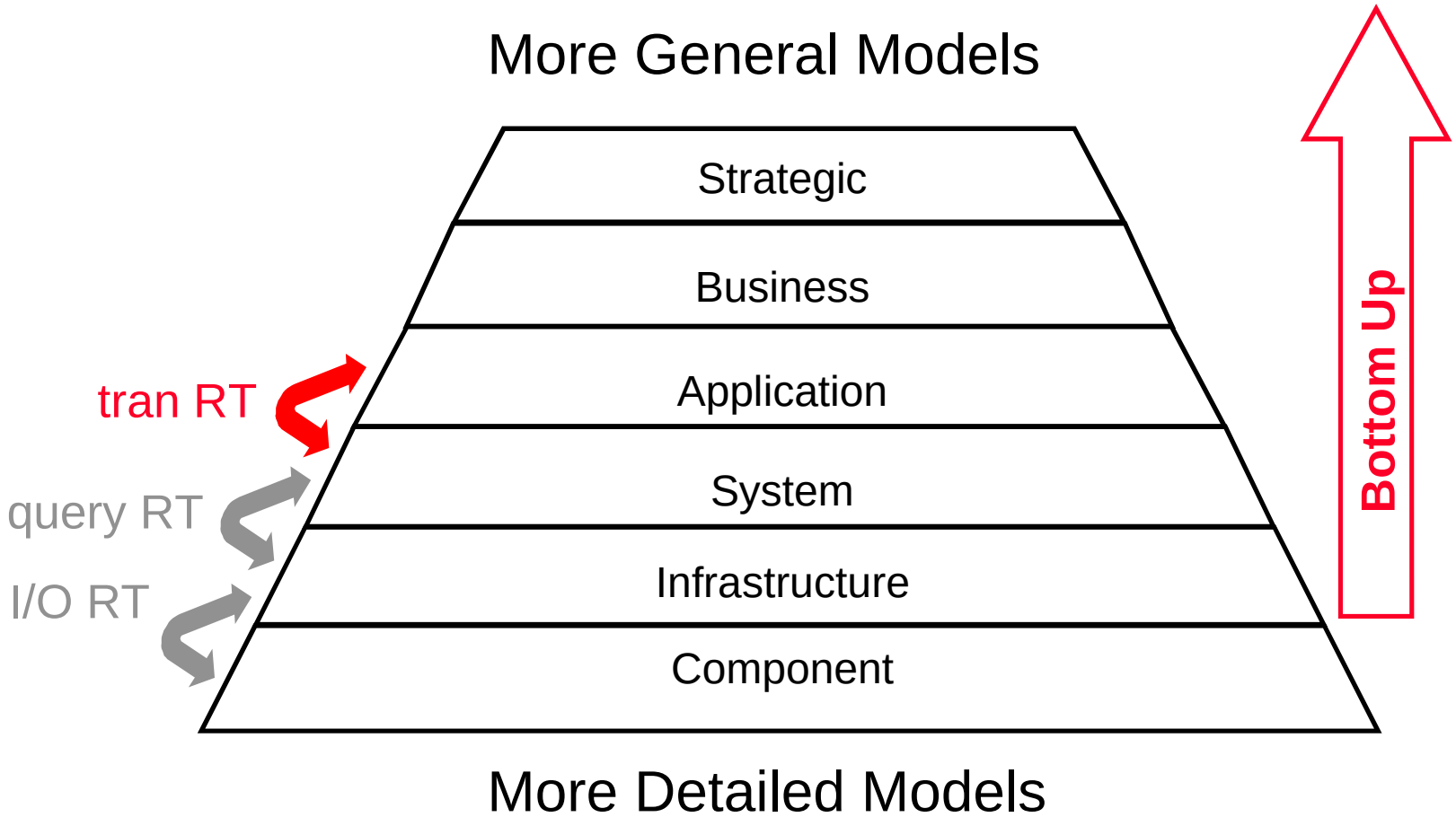
The Modeling Pyramid



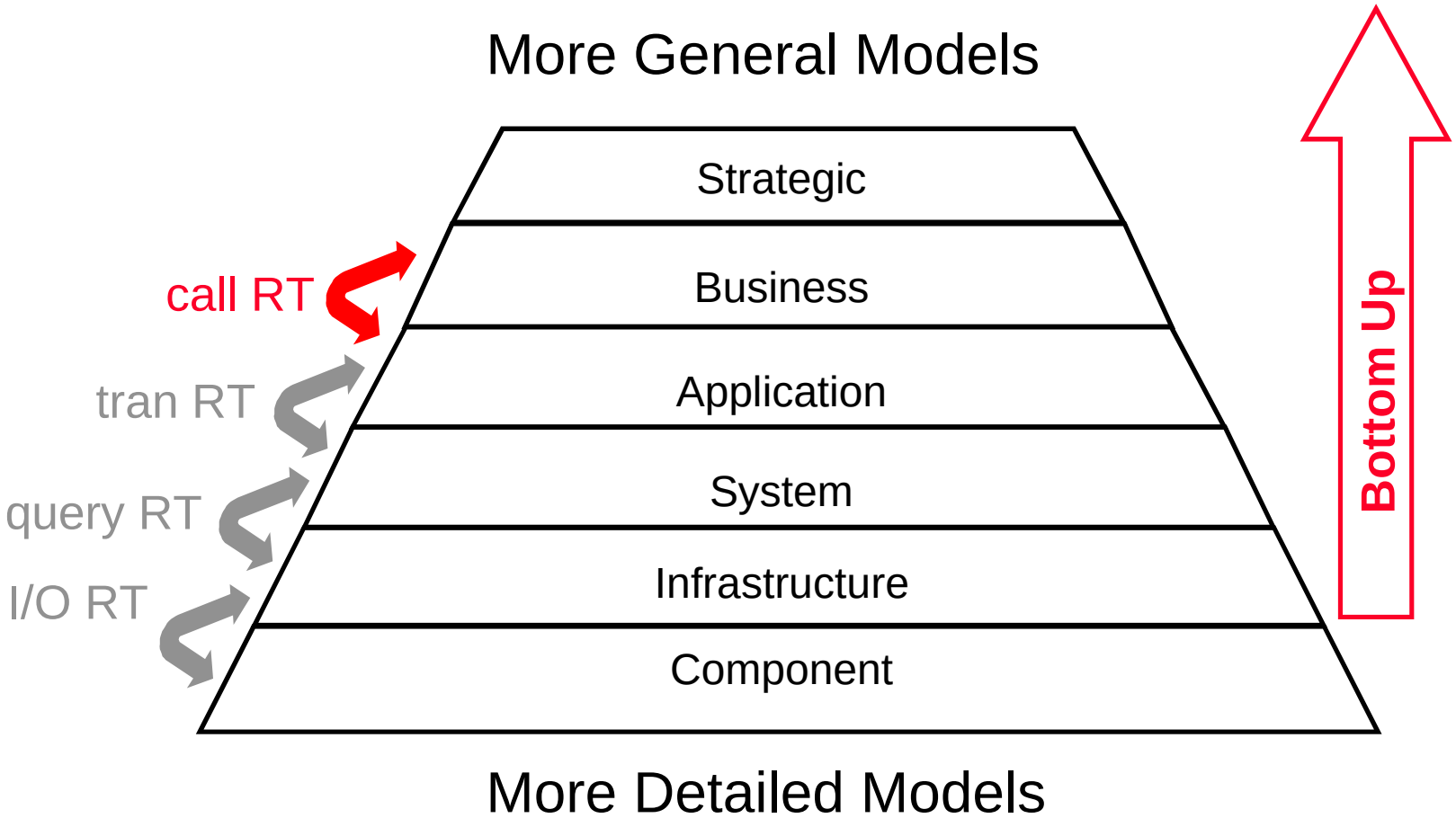
The Modeling Pyramid



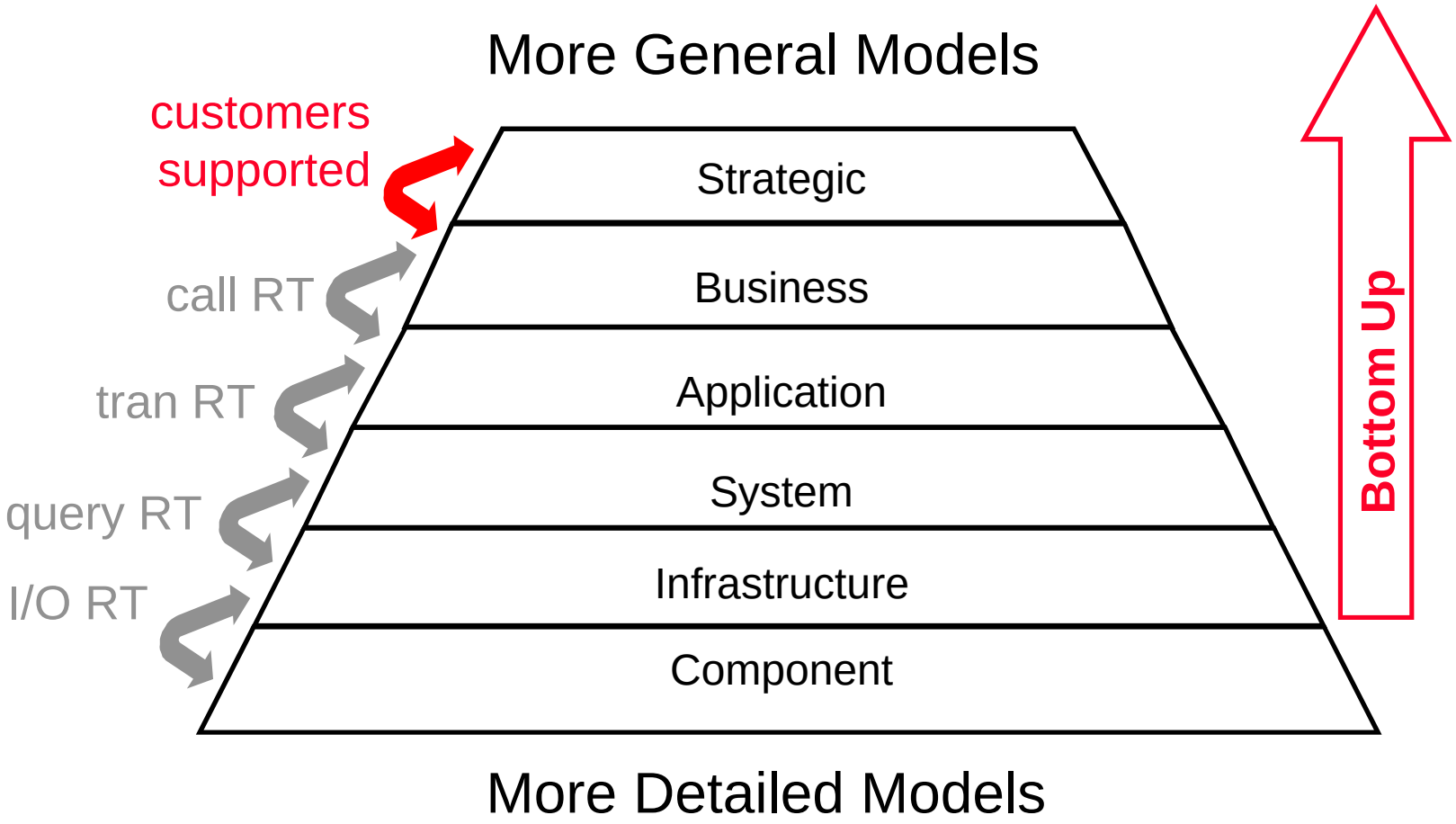
The Modeling Pyramid



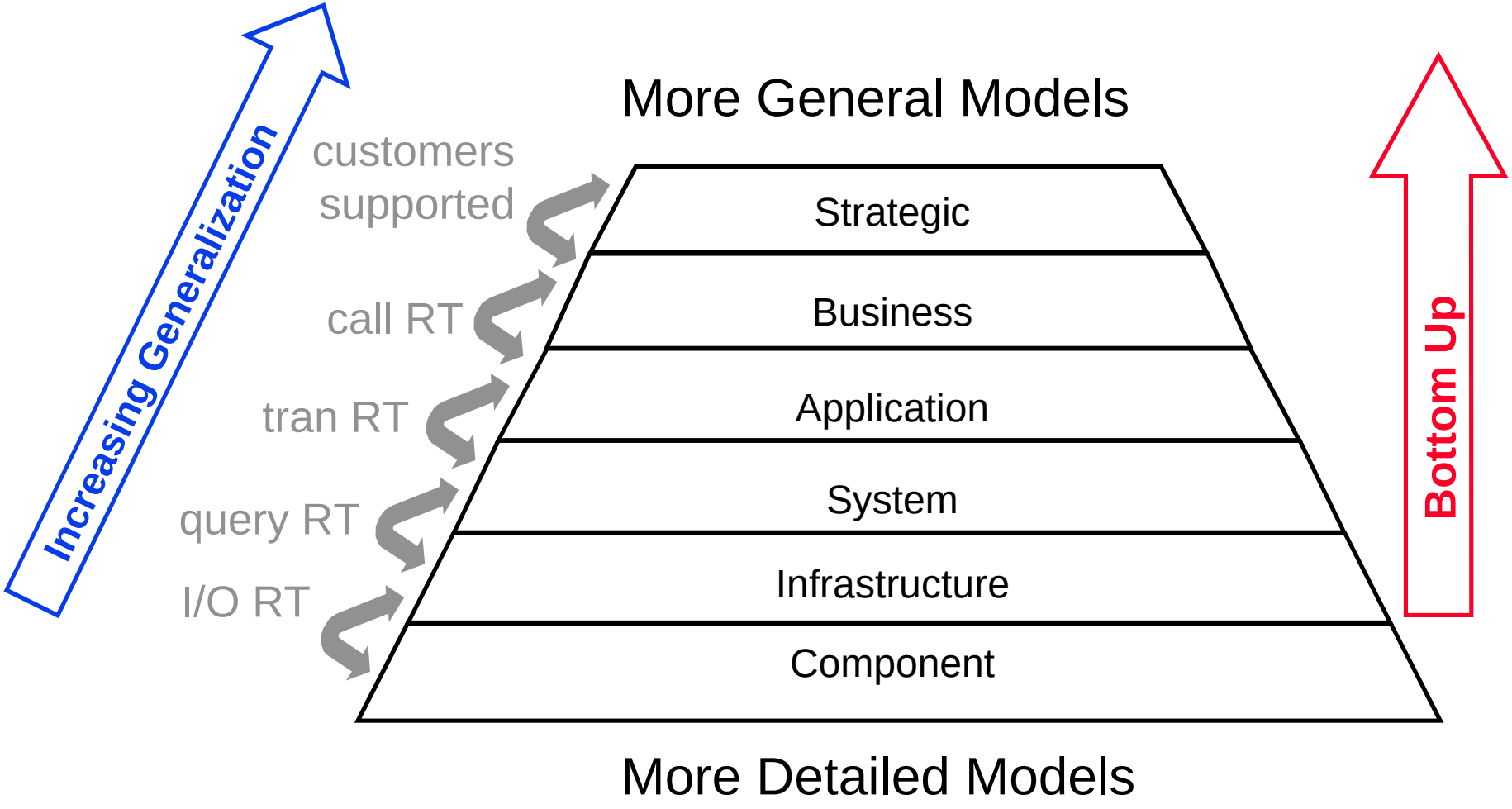
The Modeling Pyramid

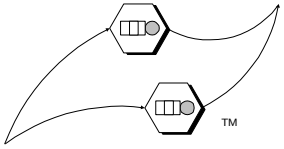


The Modeling Pyramid



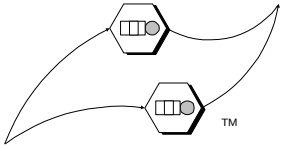
The Modeling Pyramid





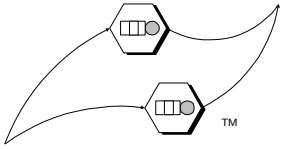
The Modeling Pyramid

- **Modeling Pyramid Implementation**
  - Inside Out
    - Start at a middle (Application) level
    - Quickly addresses Application issues
    - Lacks initial strategic direction or component detail
  - Outside In
    - Start at Top and Bottom levels simultaneously
    - Strategy defined while detailed work gets started
    - Unlikely that the two efforts will actually meet



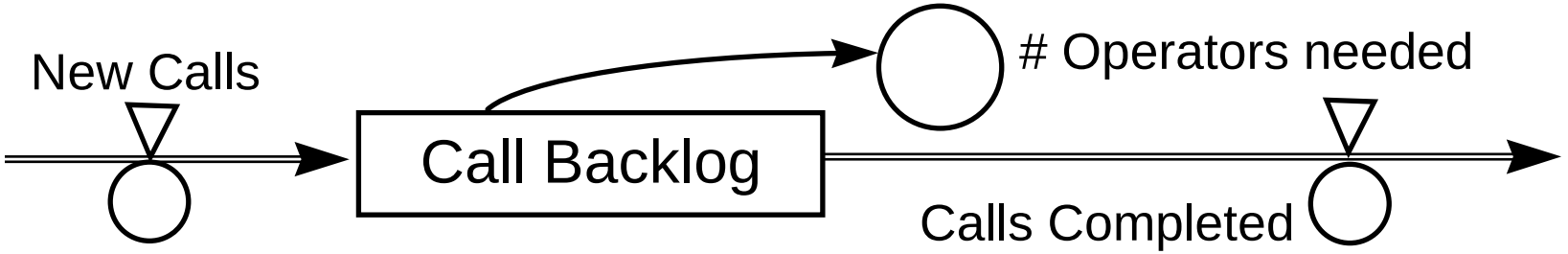
The Modeling Pyramid

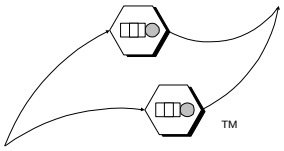
- **Modeling Pyramid Implementation**
  - The Spiral
    - Start with general assumptions for all levels
    - Challenge the assumptions:
      - Use the model at one level to challenge the models at the adjacent levels
      - Roll up to the Strategic Level to keep business focus
    - Refine at each level only when and as needed
    - Overall view (like a street map) where detail is added as needed (city center) and assumptions are used elsewhere (straight line between cities) unless that's a problem (mountain roads).



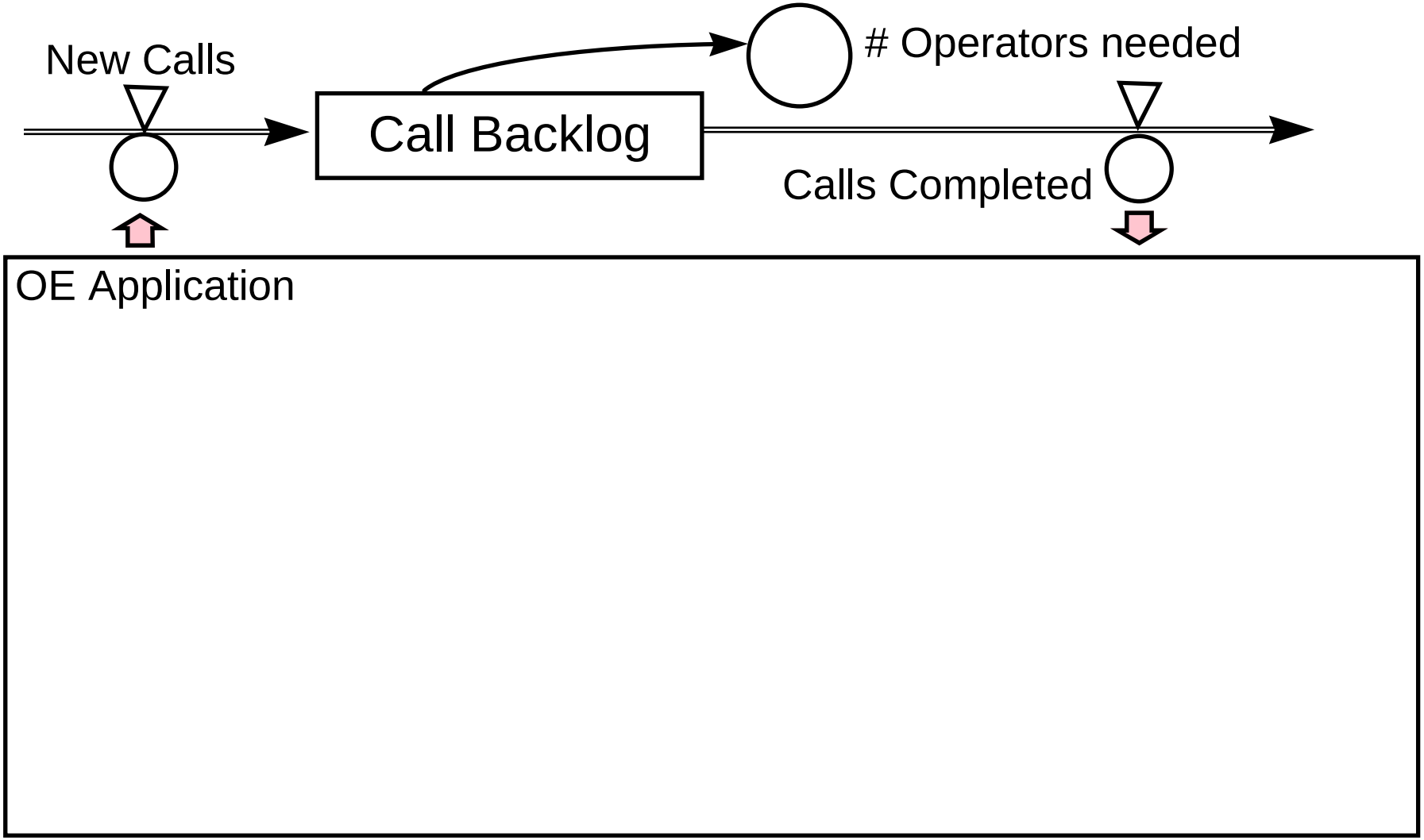
The Order Entry Model

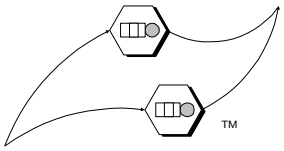
The Order Entry Model



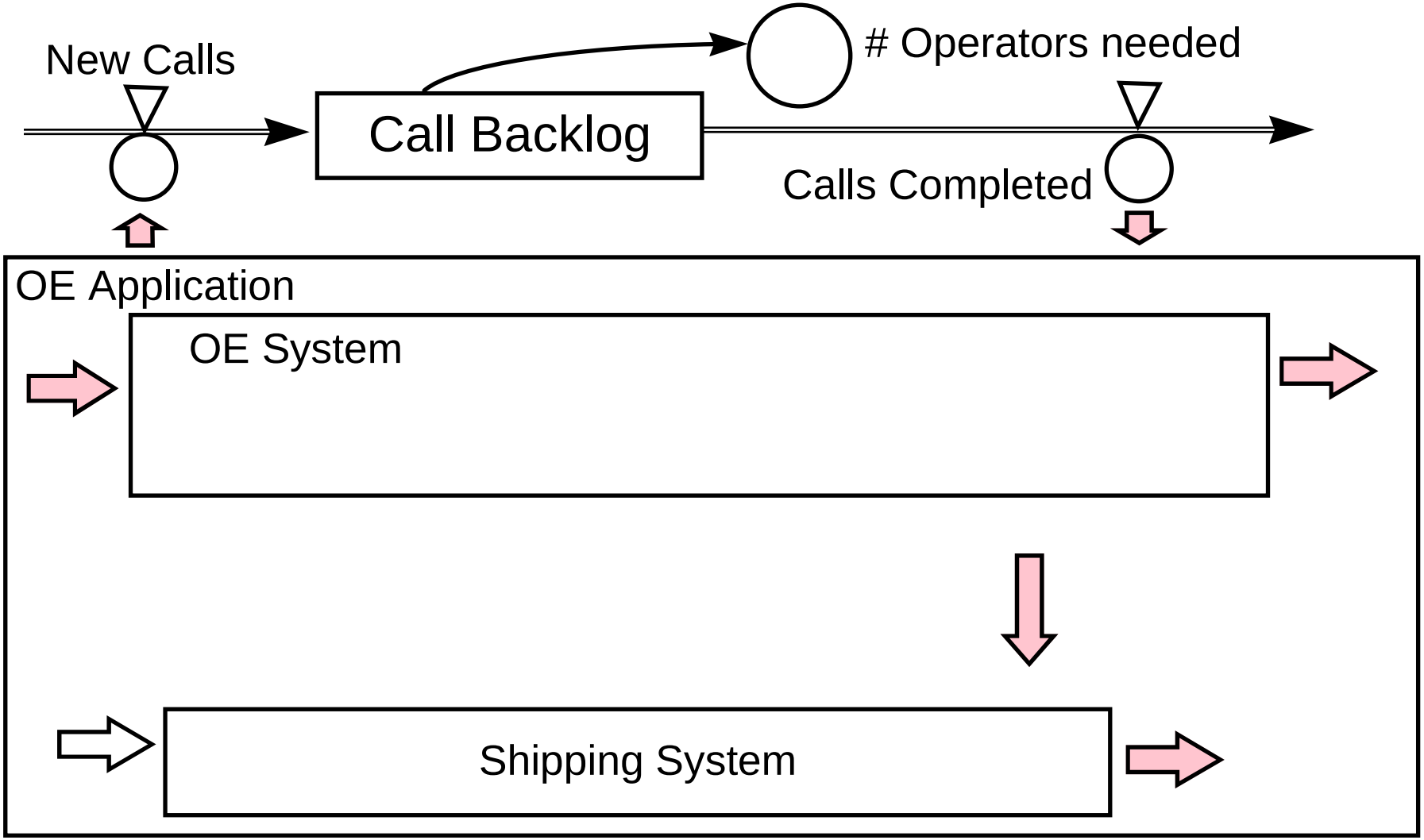


The Order Entry Model

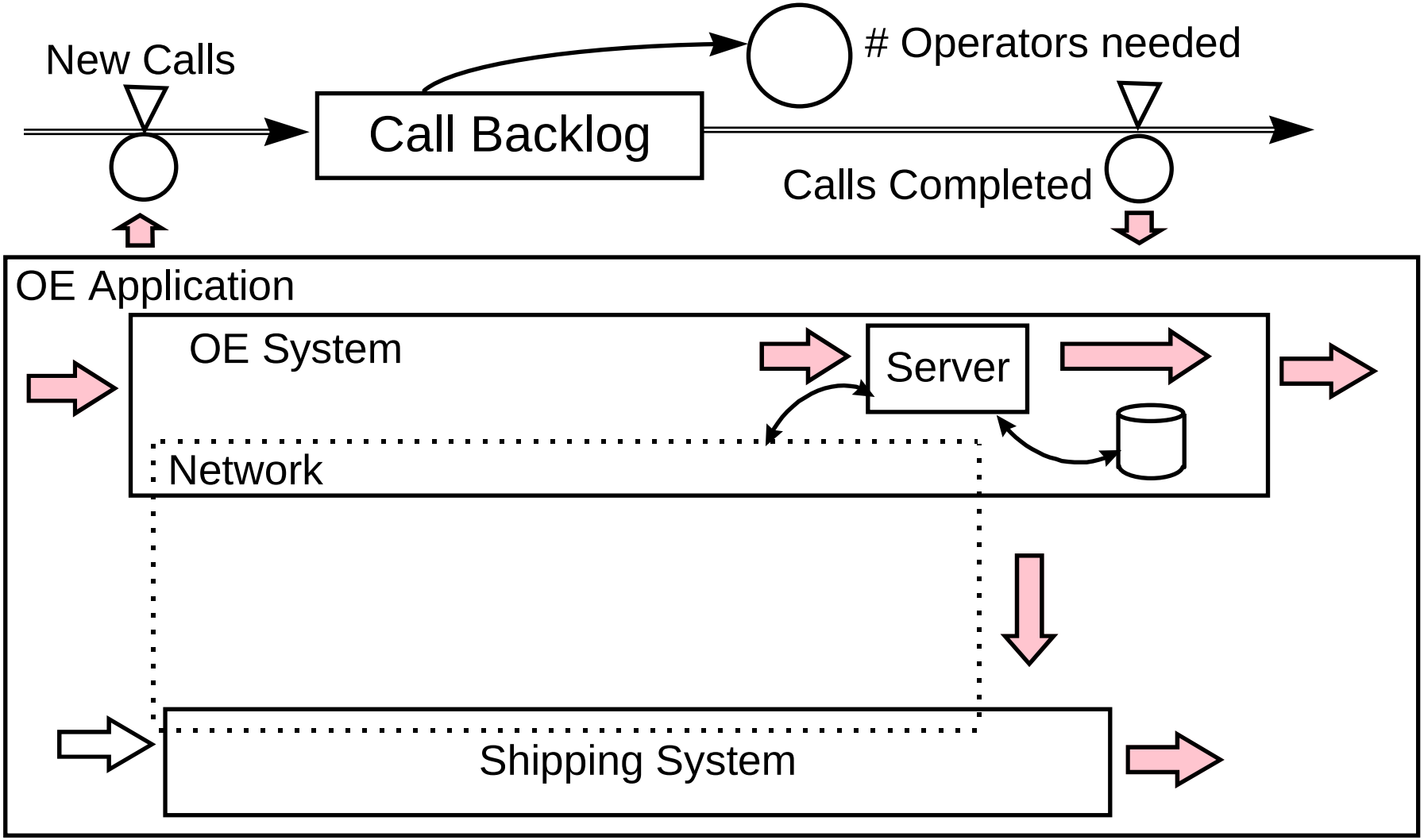


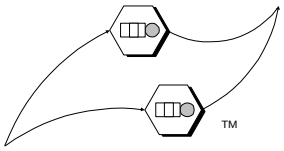


The Order Entry Model

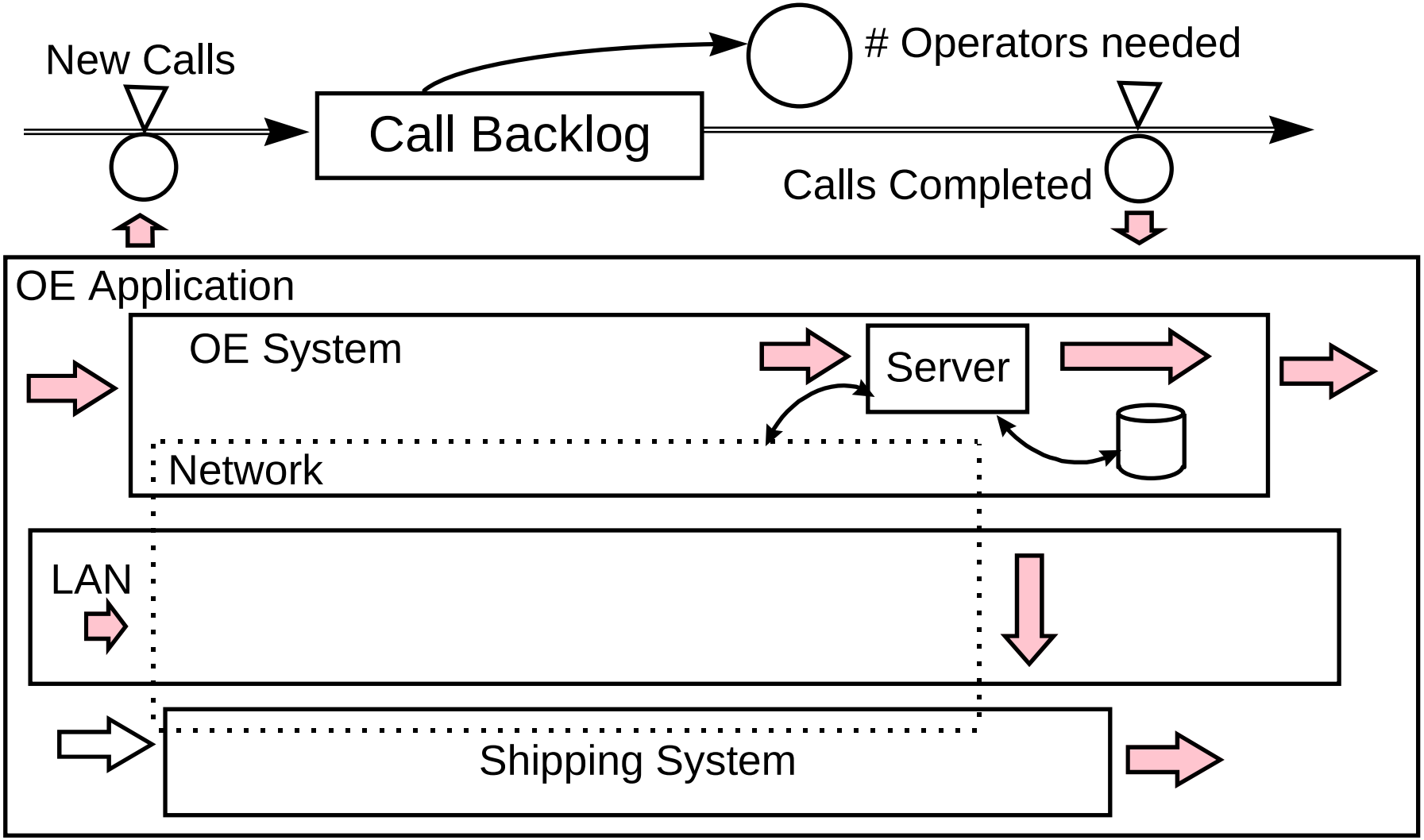


The Order Entry Model

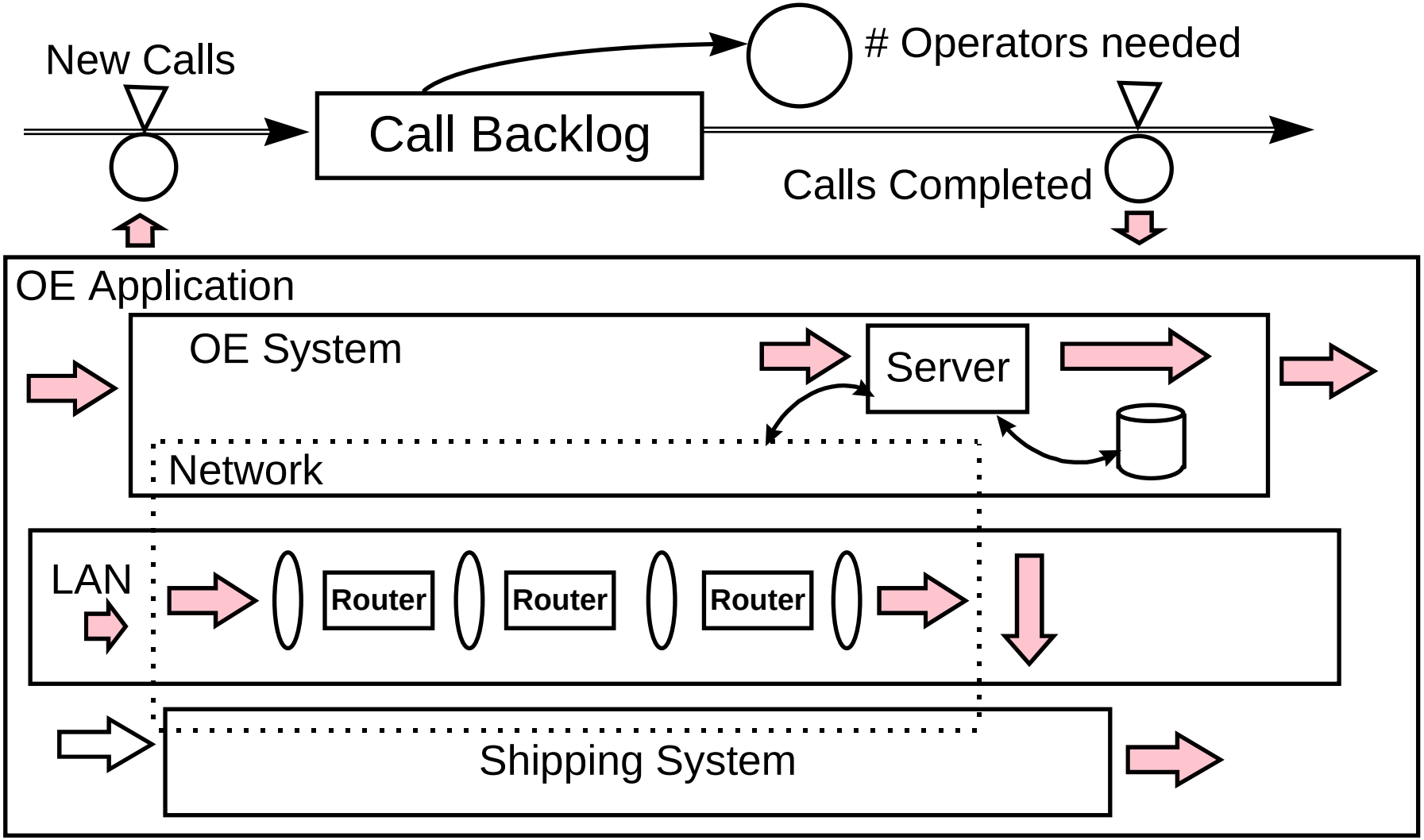


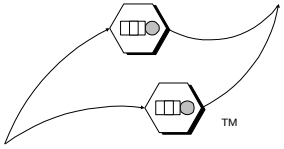


The Order Entry Model



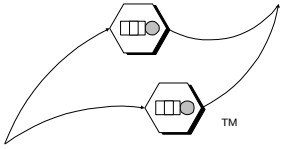
The Order Entry Model





Conclusion

- **Modeling Pyramid is a Technique to:**
  - Apply a uniform strategic direction to all modeling activities
  - See application progression from birth to death
  - Get objective information for business decisions
  - Focus technical efforts on business issues
  - Investigate the relationships between levels
  - Use appropriate tools at each level
  - Enhance each model level only as needed



Questions

?