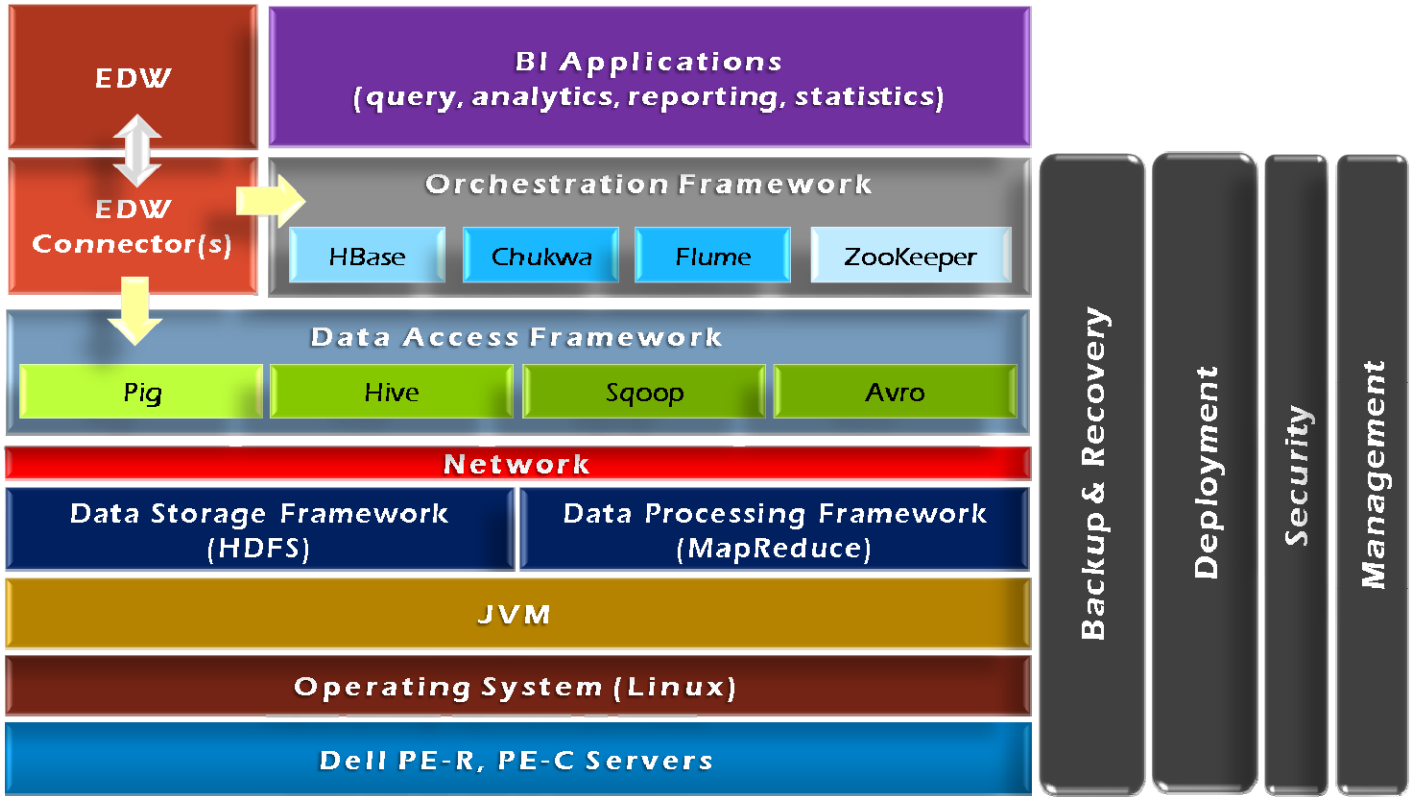
CpSc 875

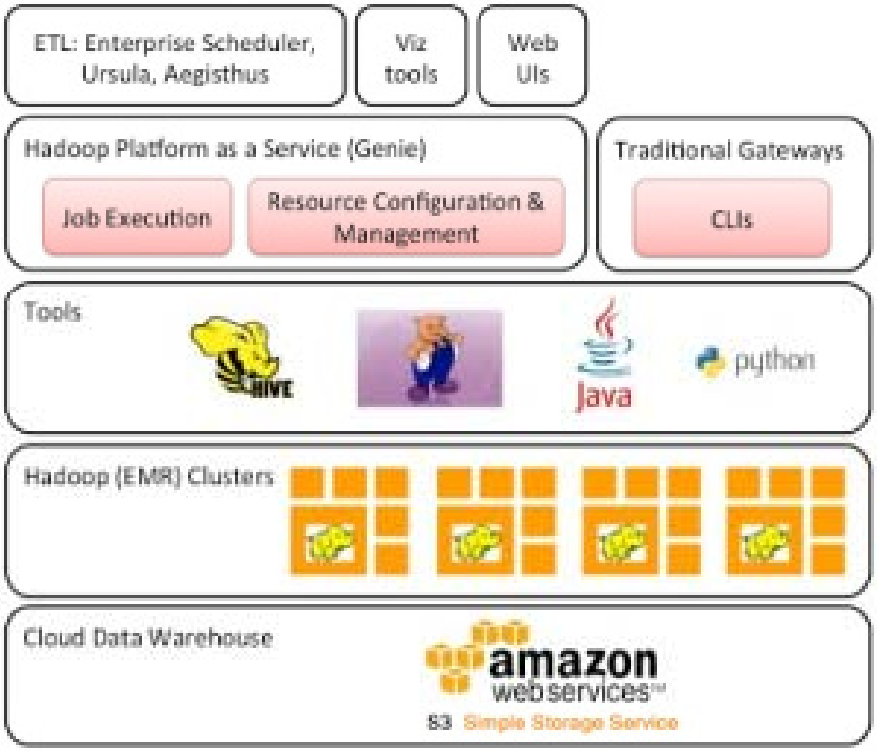
John D. McGregor

C 12 – Architecture Evaluation

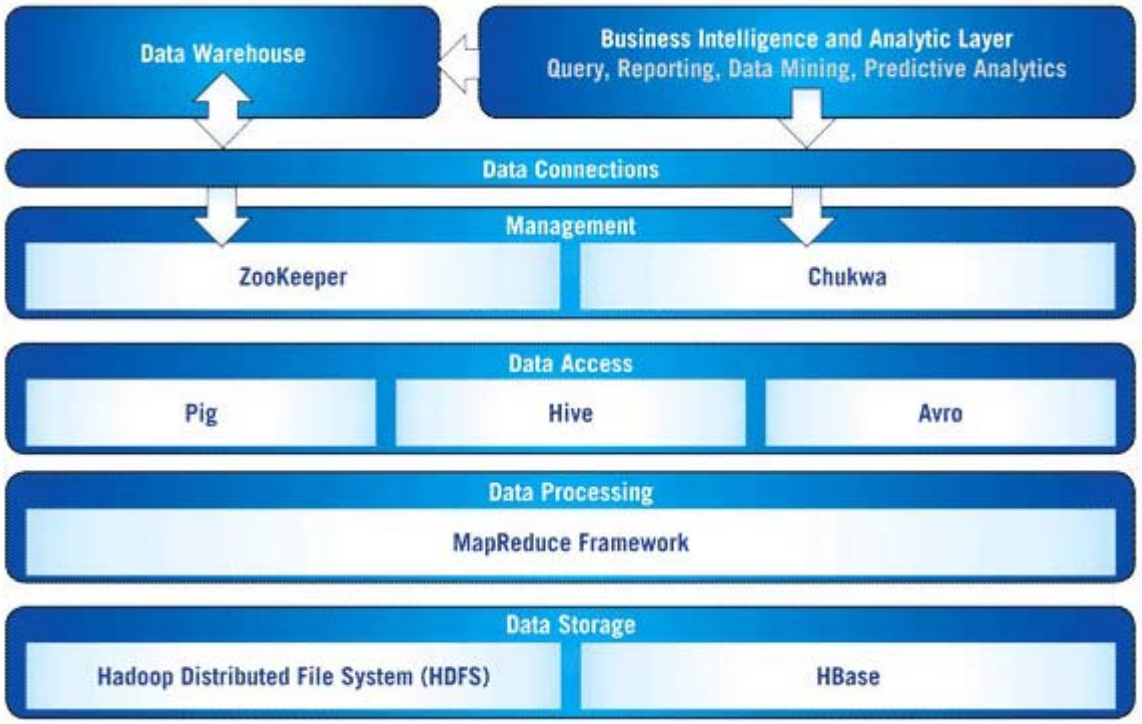
Dell/Hadoop data analytics architecture



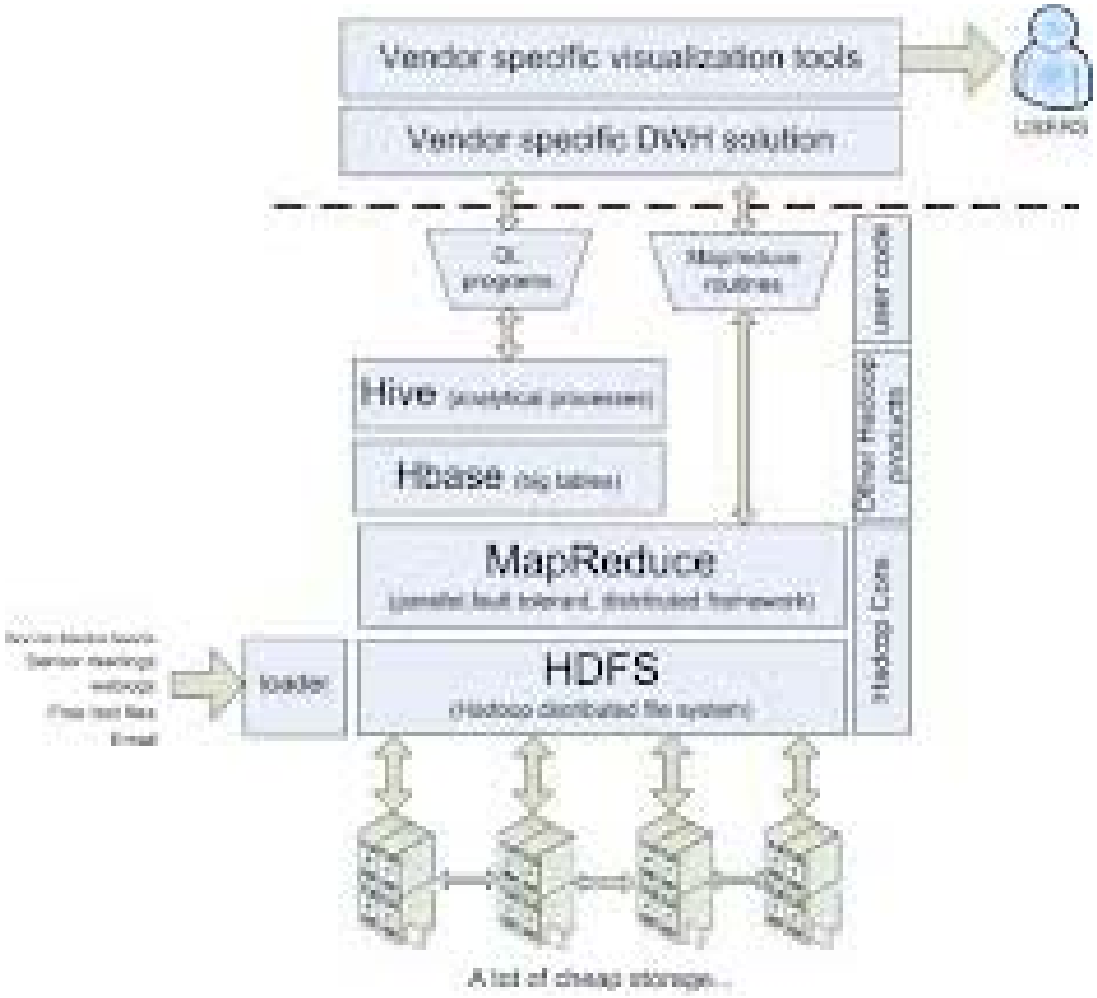
Netflix Hadoop



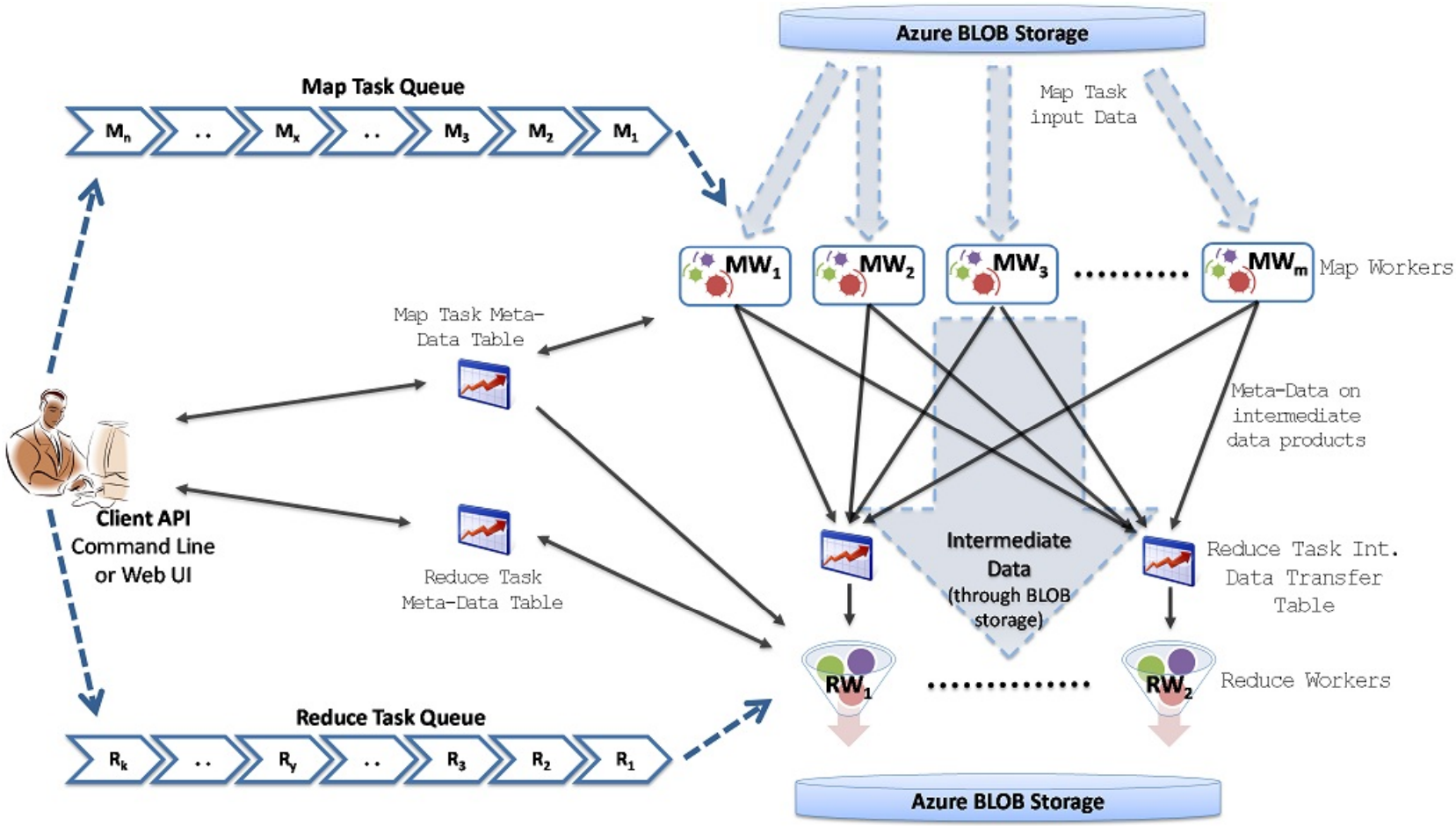
Different view



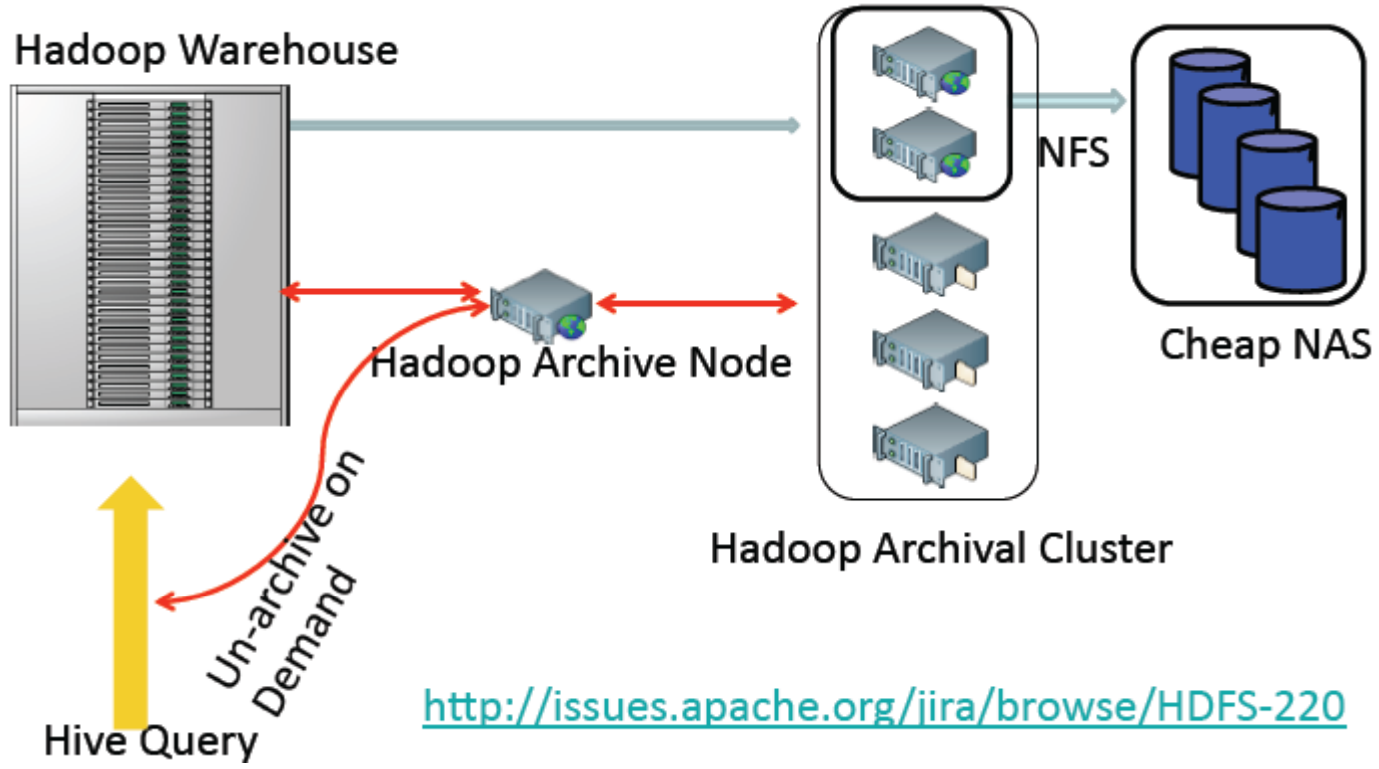
Yet another


Vendor specific visualization tools
Vendor specific DWH solution
Hive
Hbase
MapReduce
HDFS
loader
User code
Hadoop Core

Microsoft Azure Map/Reduce

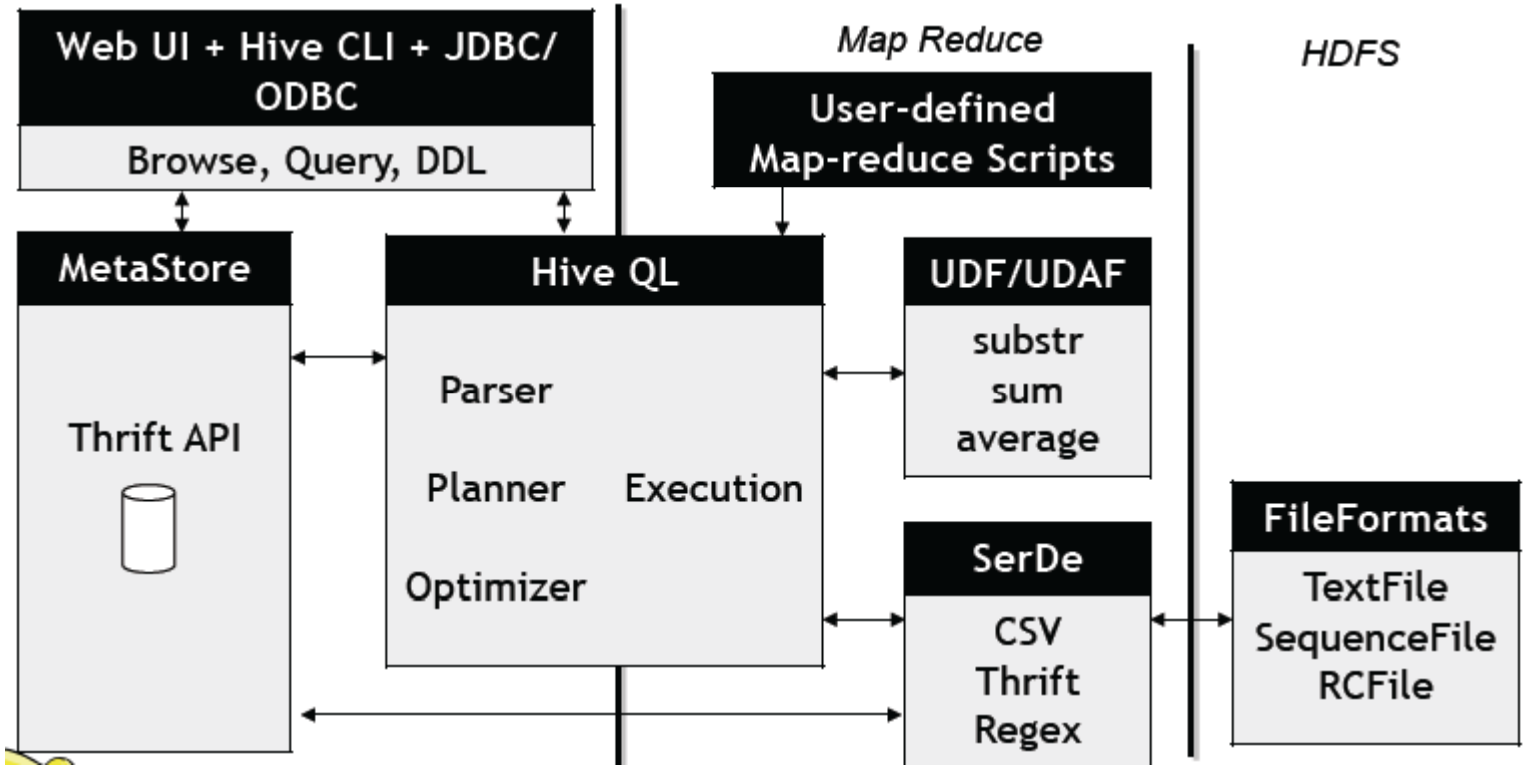


System view



http://issues.apache.org/jira/browse/HDFS-220

Hadoop at Facebook



Connections and Flows

- Connection is between an out port and an in port
- Flow is within a component
- Flow path is across connections and through components
- End-to-end flow is flow source to flow sink and all inbetween
- A flow must be declared in the specification in order to appear in the implementation

Spec and Impl

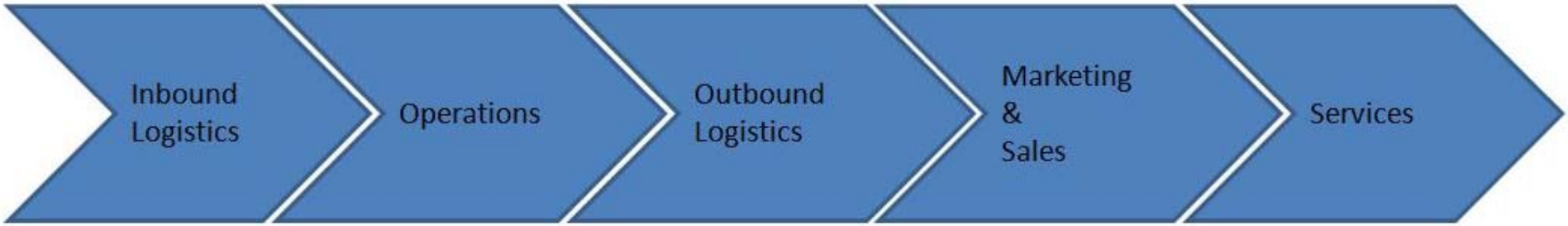
- A specification is the public information about a module
- An implementation is how the specification can be realized
- f1: **flow path init -> failure**
- f1: **flow path init -> normal -> selfTestThread.f1->failedSelfTest->failure**

Architecture adds value

- Value is a synonym for desirableness
- If the value of something increases it is because it has become more desirable for some reason
- A “value chain” represents a sequence of stages, each of which makes the “thing”, for which this is the value chain, more desirable.
- The value chain for a software product is the series of activities that craft a solution.

Porter's Value Chain

A Value Chain Analysis



- **Collection Service**
  - Time
  - Request Processing
- **Self Return**
  - Plant Scheduling
  - Plant capacity
  - Proximity to plant

- **Product Quality**
- **Quality Expectation**
- **Order fulfillment**
- **Customer Visit**

- **Ordering**
- **Order Fulfillment**
- **Emergency Load**
- **On Time Delivery**
- **Customer Pick Up**
- **Inventory Management**
- **Transaction Management**
  - Reporting
  - EDI
  - Reconciliation
  - Audit
  - Invoice Accuracy

- **Account Management**
- **Pricing**
  - Pricing Structure
  - Total Cost of Acquisition
- **Invoicing**
- **Credit process**
- **Understanding the value of our product**
- **Understand what customer needs**

- **Customer Call Center**
- **Problem resolution**
  - Speed
  - Solution
- **Customer Survey**
- **Audit variance reconciliation**
- **Customer Location Audit**

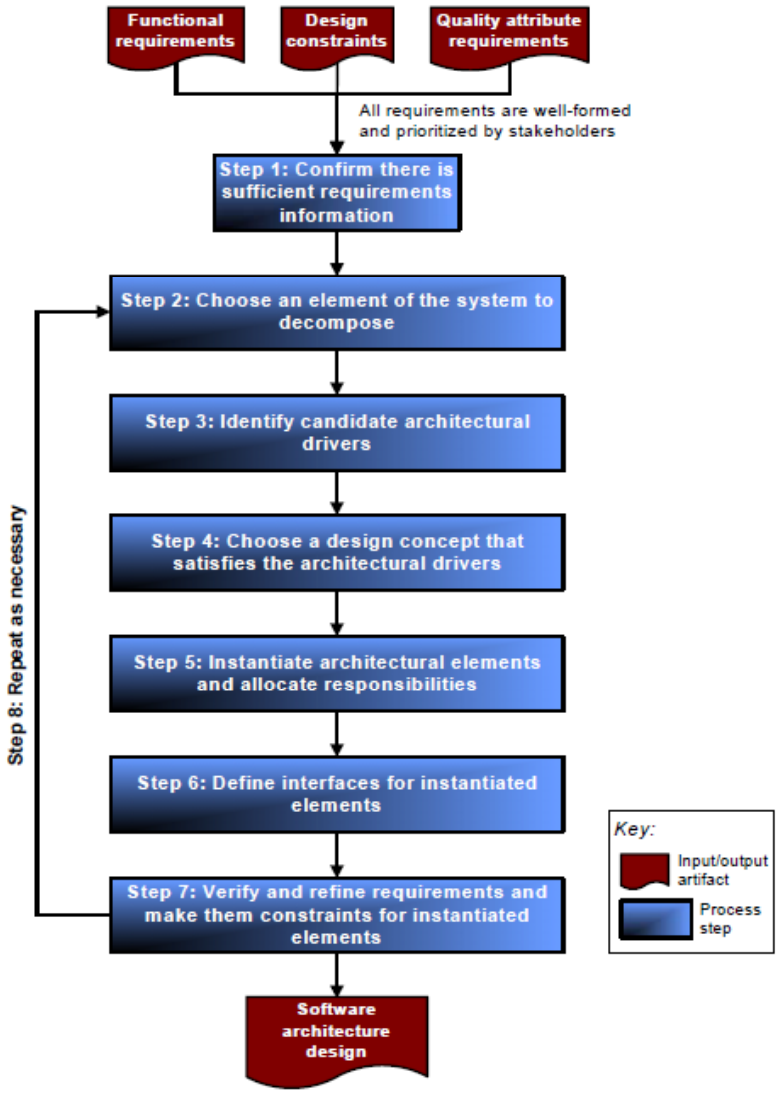
Adding value

- How does architecture add value? (How does it make the product more desirable?)
  - Increased probability that customers like the product
  - Increased probability of highly reliable operation
  - Increased probability that the product will have the qualities desired
  - Increased predictability of implementation

Adding value - 2

- Even architecture evaluation adds value
- It removes defects making the architecture more desirable as a basis for building a product
- Question: How do we measure these increases in value?

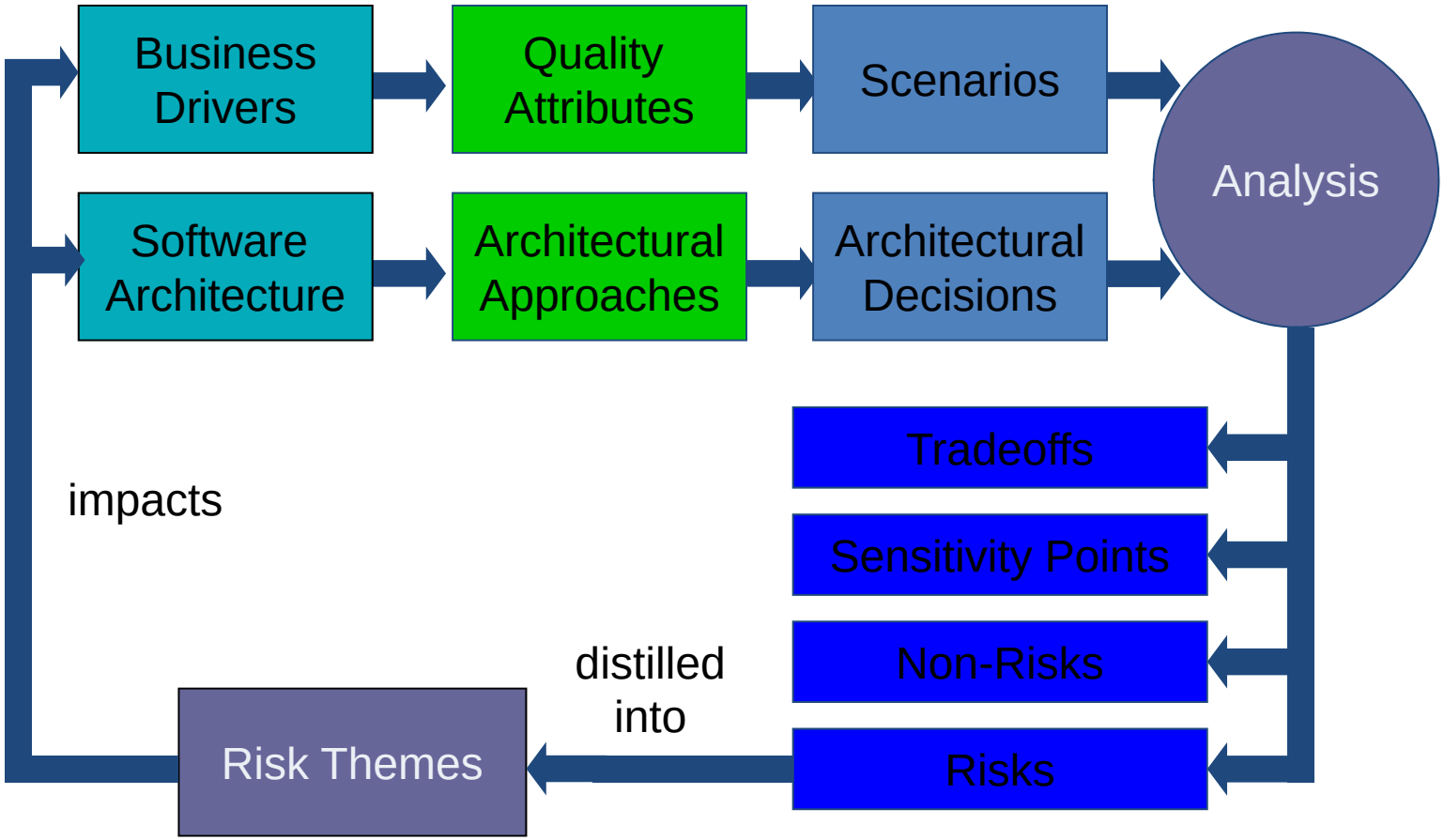
Where are we?


Functional requirements
Design constraints
Quality attribute requirements
All requirements are well-formed and prioritized by stakeholders
Step 1: Confirm there is sufficient requirements information
Step 2: Choose an element of the system to decompose
Step 3: Identify candidate architectural drivers
Step 4: Choose a design concept that satisfies the architectural drivers
Step 5: Instantiate architectural elements and allocate responsibilities
Step 6: Define interfaces for instantiated elements
Step 7: Verify and refine requirements and make them constraints for instantiated elements
Step 8: Repeat as necessary
Software architecture design
Key:
Input/output artifact
Process step

Architecture TradeOff Analysis Method (ATAM)

- **The purpose of the ATAM is to assess the consequences of architectural decisions in light of quality attribute requirements.**
- http://www.sei.cmu.edu/reports/00tr004.pdf

Conceptual Flow of ATAM



- Phase 0
  - Partnership and preparation
- Phase 1
  - Evaluation
- Phase 2
  - Evaluation continued
- Phase 3
  - Follow-up

Phase 0

- Logistics are agreed to
  - Meeting dates
  - Who must attend
  - Team membership
- Agenda is agreed to
- Collect initial information

Overview of Phase 1

- Step 1
  - Present the ATAM
- Step 2
  - Present business drivers
- Step 3
  - Present architecture
- Step 4
  - Identify architectural approaches
- Step 5
  - Generate quality attribute utility tree
- Step 6
  - Analyse architectural approaches

Present Business Drivers

- Describe
  - The system's most important functions
  - Any relevant technical, managerial, economic, or political constraints
  - The business goals and context as they relate to the project
  - The major stakeholders
  - The architectural drivers (the major quality attribute goals)

Present Architecture

- Driving architectural requirements, measurable quantities associated with these, standards/models/approaches for meeting these
- Important architectural information
  - Context diagram
  - Module or layer view
  - Component and connector view
  - Deployment view

Present Architecture - 2

- Architectural approaches, patterns, tactics employed, what quality attributes they address and how they address those attributes
- Use of COTS and their integration
- Most important use case scenarios
- Most important change scenarios
- Issues/risk w.r.t. meeting the diving requirements

Step 4: identify architectural approaches

- Catalog the evident patterns and approaches
  - Based on step 3
  - Serves as the basis for later analysis

Quality Attribute Scenario

Stimulus

Stimulus source

Environment

Artifact

Response

Response measure

Quality Attribute Scenario

Stimulus – the hand control is moved to make a 2 inch long, 1 inch deep incision

Stimulus source – doctor

Environment – patient and robot have been aligned

Artifact – image in the viewfinder

Response – the view is updates with no flicker

Response measure – the doctor experiences no difficulty seeing the incision as it is made

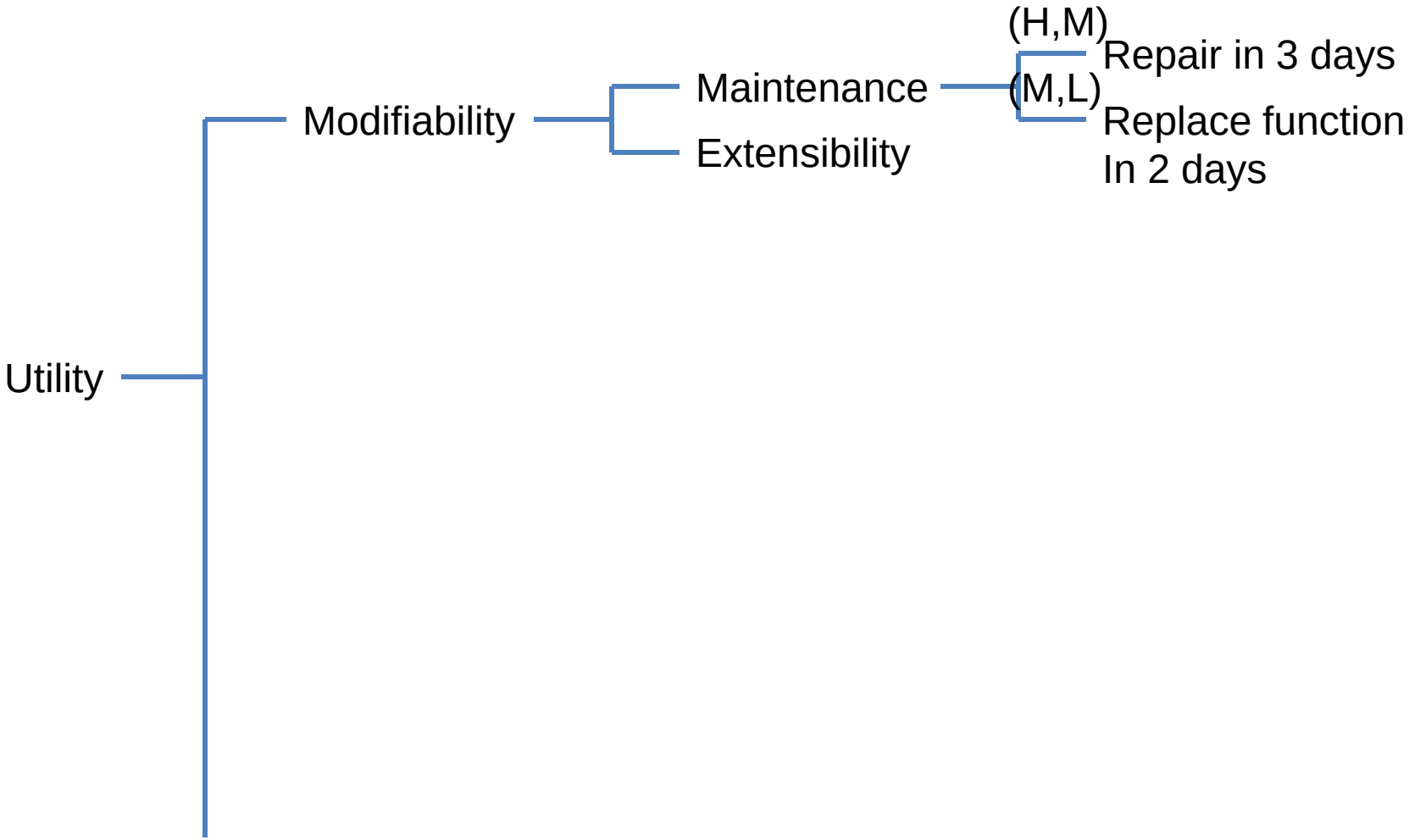
Step 5: Generate quality attribute utility tree

- Utility tree
  - Present the quality attribute goals in detail
- Quality attribute goals are
  - Identified, prioritized, refined
  - Expressed as scenarios
- Utility is an expression of the overall goodness of the system
  - Quality attributes form the second level being components of utility

Step 5: Generate quality attribute utility tree con’t

- Scenarios are prioritized
  - Depending on how important they are and
  - Depending on how difficult it will be for the architecture to satisfy a scenario

Step 5 – Lets draw the tree



Step 6: Analyze architectural approaches

- Examine the highest ranked scenarios
- The goal is for the evaluation team to be convinced that the approach is appropriate for meeting the attribute-specific requirements
- Scenario walkthroughs
- Identify and record a set of sensitivity points and tradeoff points, risks and non-risks
  - Sensitivity and tradeoff points are candidate risks

Phase 2

- Step 7
  - Brainstorm and prioritize scenarios
- Step 8
  - Analyze architectural approaches
- Step 9
  - Present results

Step 7: Brainstorm and prioritize scenarios

- Utility tree shows architects view on the quality attributes
- Here the focus is on the other stakeholders view on the quality attributes and scenarios based on these
  - Which are the mot meaningful and important scenarios w.r.t. users etc.

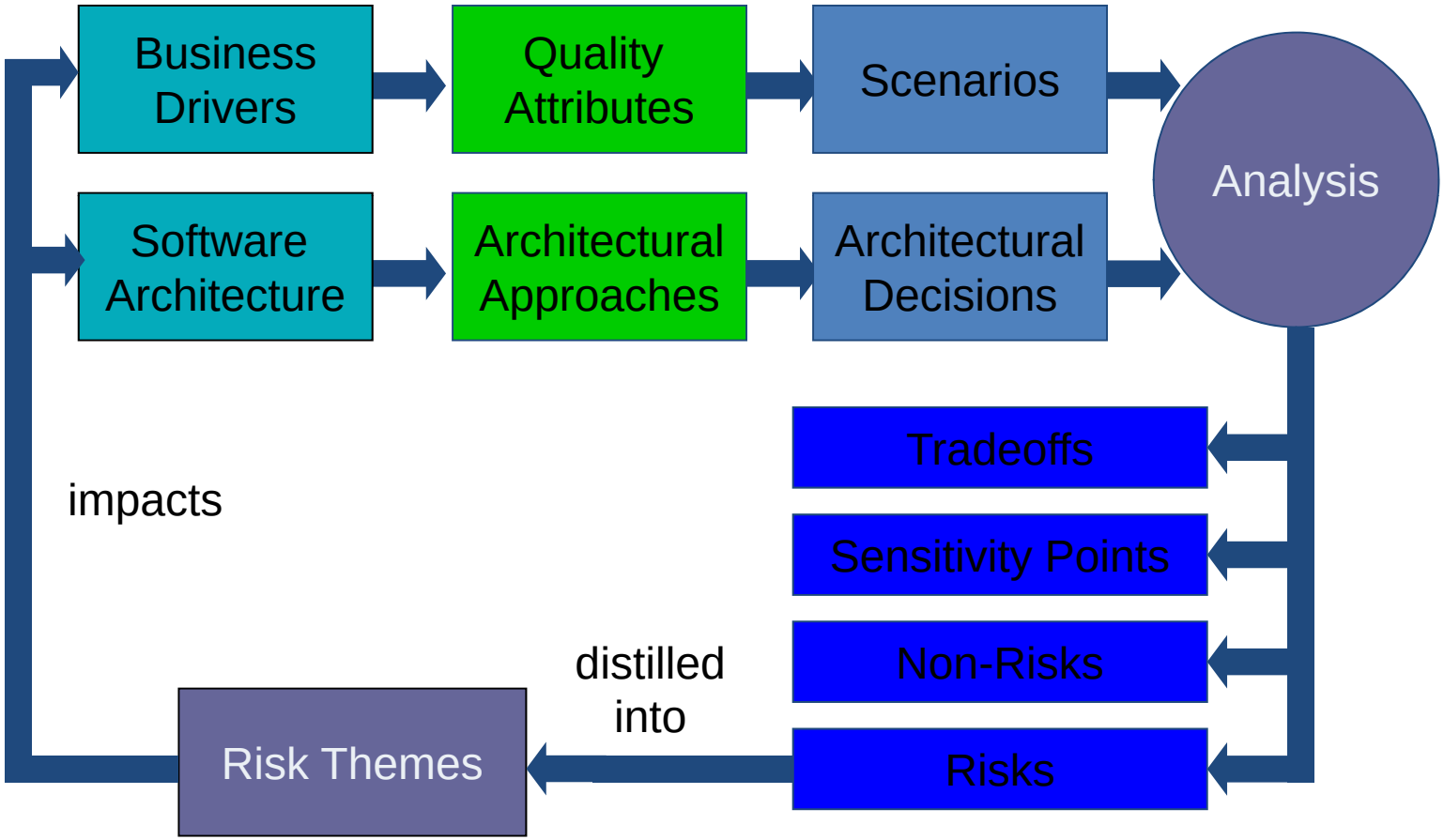
Step 8: Analyse architectural approaches

- Highest ranked scenarios from step 7 are presented to the architect
  - Explain how relevant architectural decisions contribute to realising each one

Step 9: Present results

- Outputs:
  - The architectural approaches documented
  - The set of scenarios and their prioritization from the brainstorming
  - The utility tree
  - The risks discovered
  - The non-risks documented
  - The sensitivity points and tradeoff points found

Conceptual Flow of ATAM



Here's what you are going to do…

- Exchange your architecture with another team
- Use their use cases and utility tree to create three evaluation scenarios
- Conduct an ATAM including meeting with the other team and having them present their architecture to you.
- Then evaluate and write a brief document that addresses the tradeoffs, risks/non-risks, and sensitivity points.