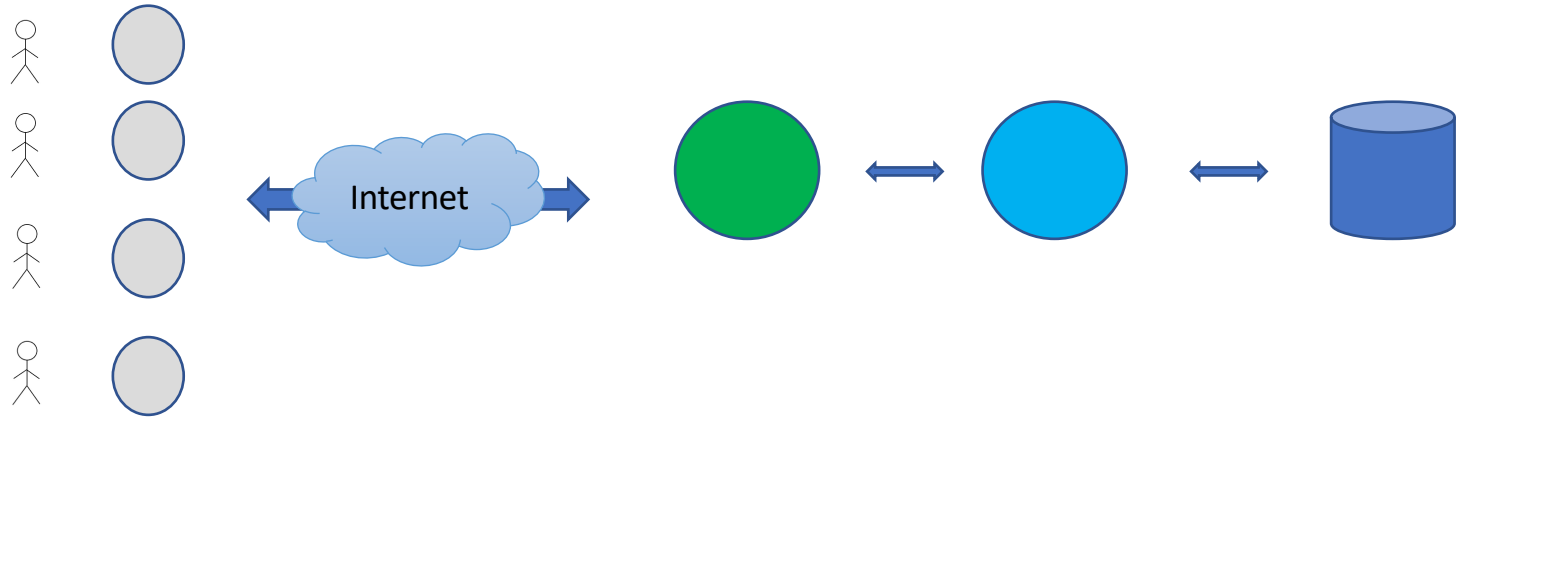
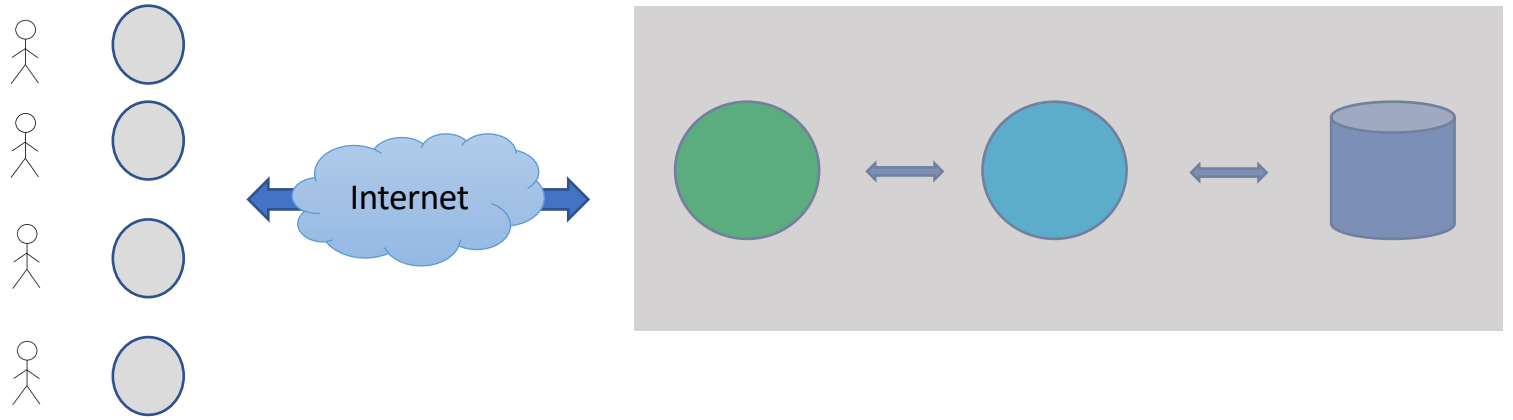
What is Architecture?

What is Software Architecture


Internet
Clients
Web Server
App Server
Database Engine

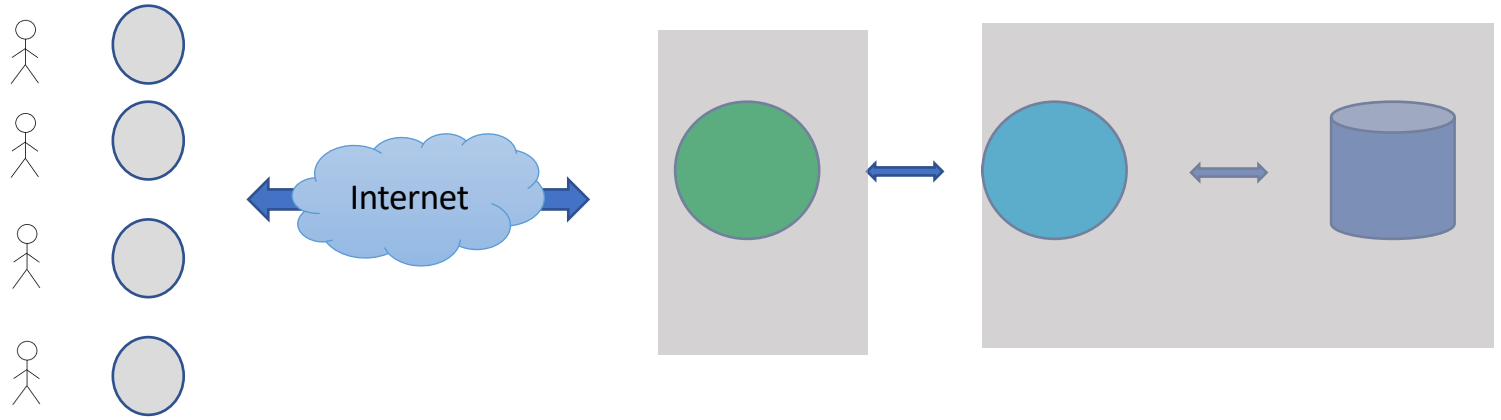


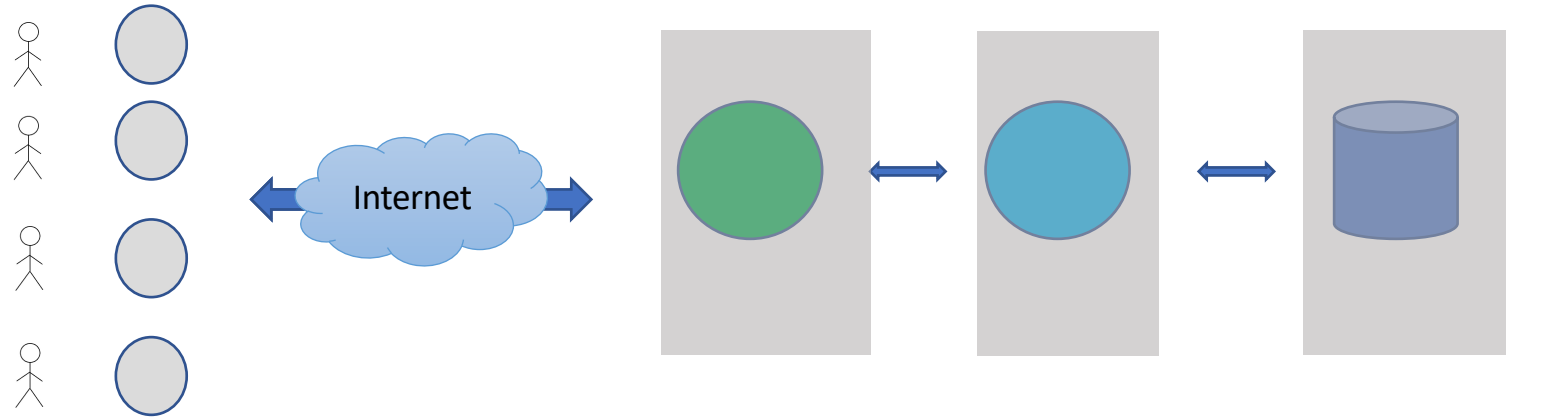
Clients

Web Server

App Server

Database Engine


Internet
Clients
Web Server
App Server
Database Engine


Internet
Clients
Web Server
App Server
Database Engine

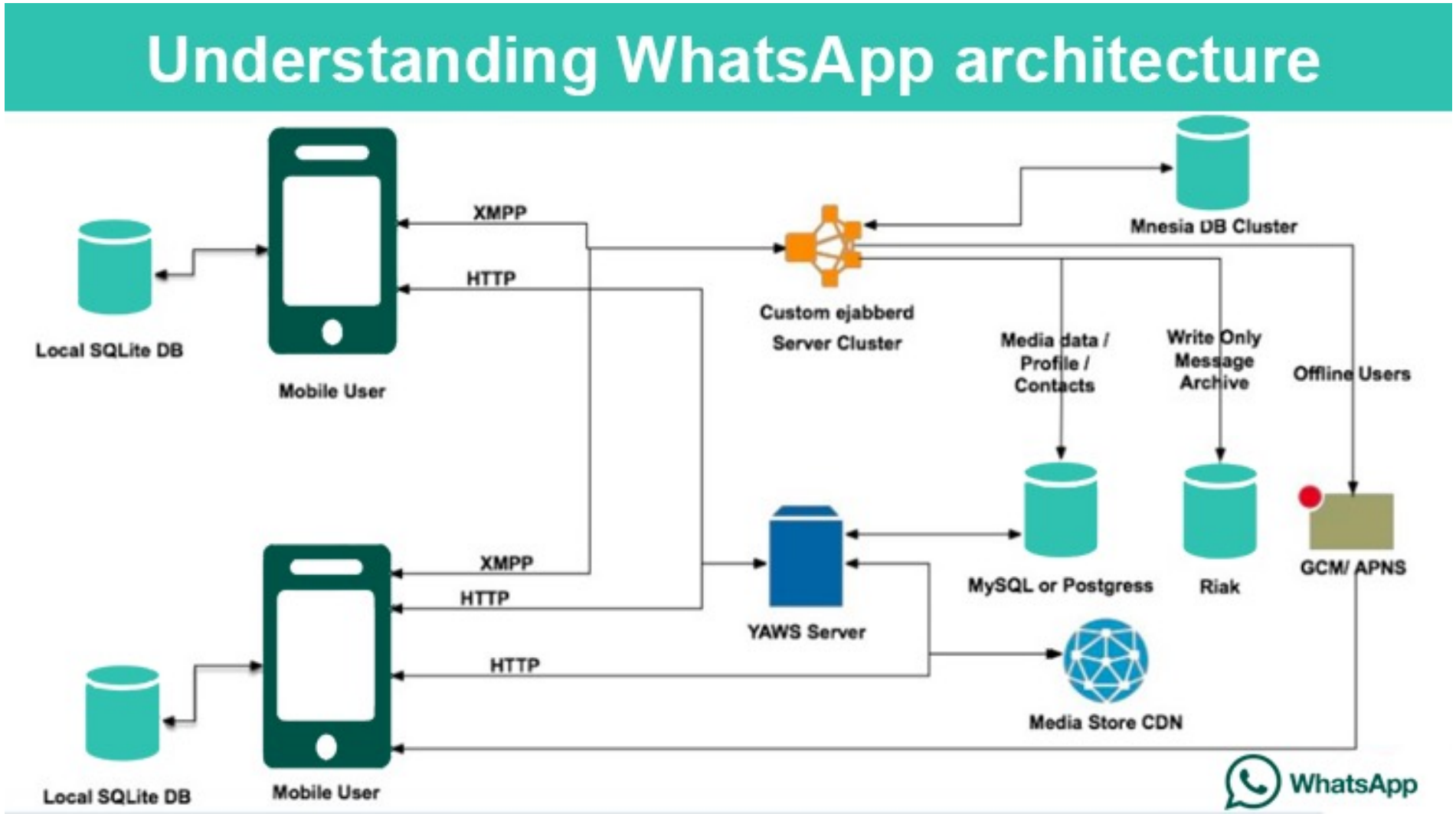
What is the difference in these deployments?

- No difference to the end user!
- But
  - Response times
  - Scalability
  - Disaster Recovery

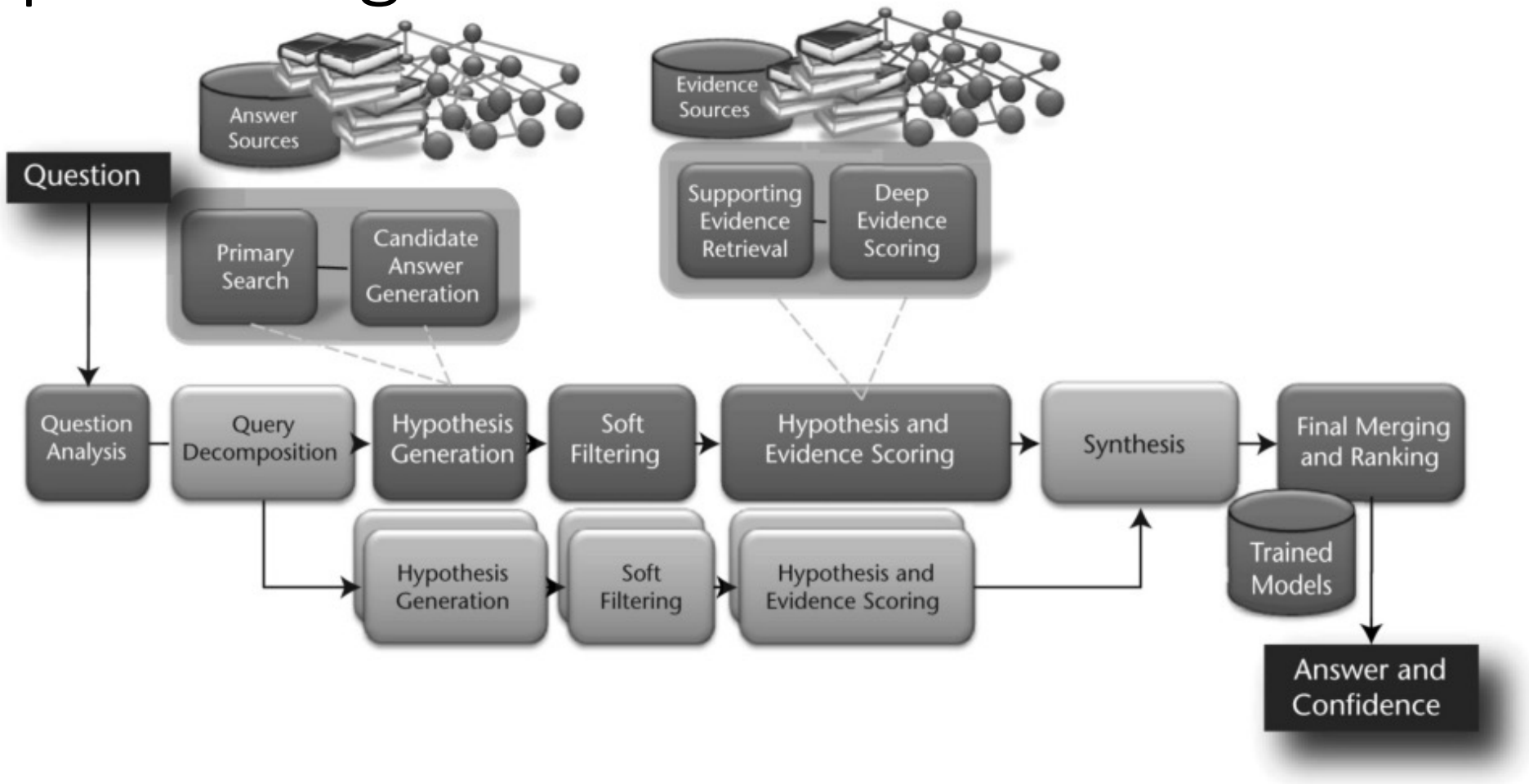
May be different

Architecture

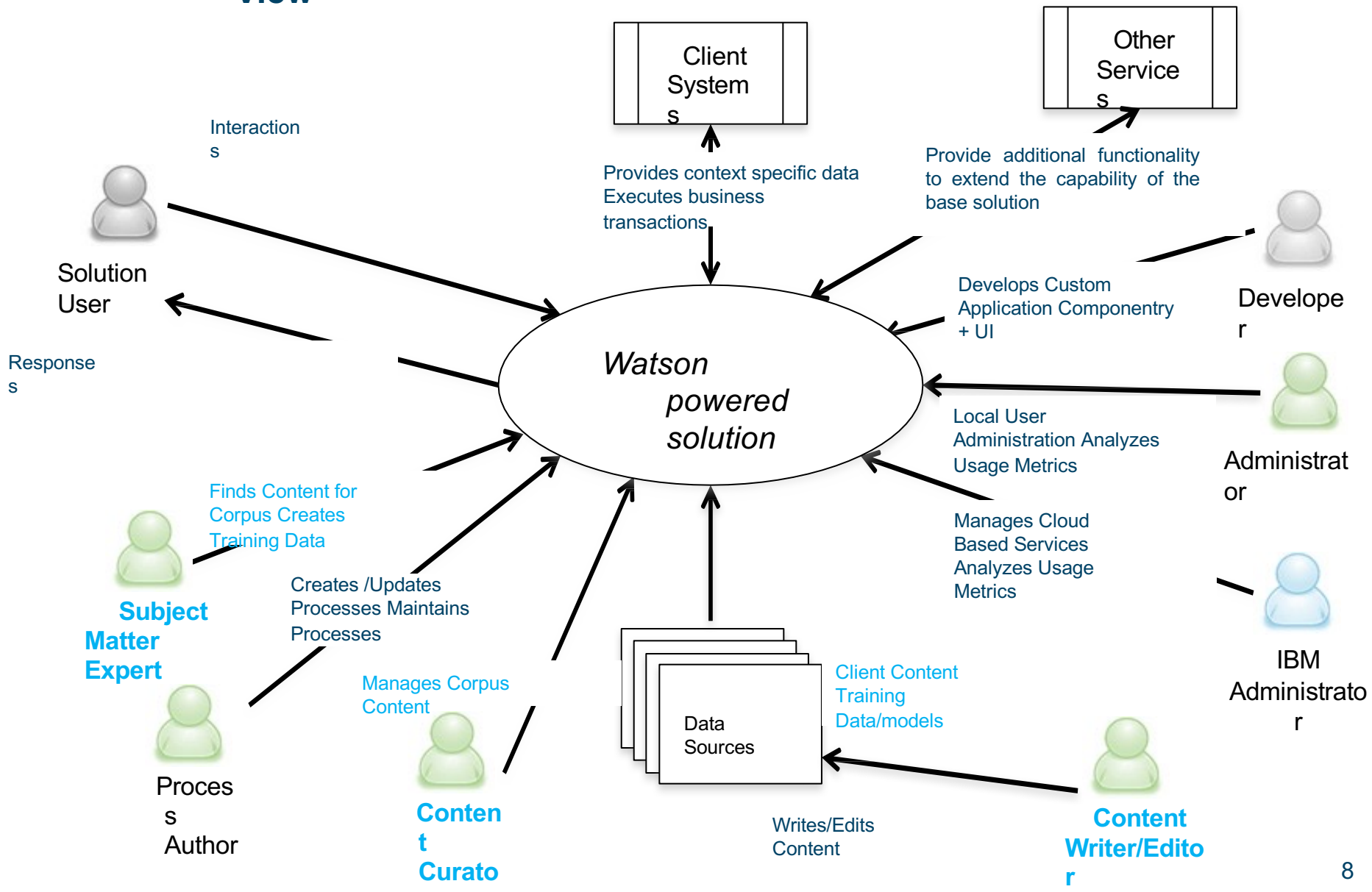
"Box and line" diagrams



DeepQA – Highlevel Architecture



Watson High Level Reference Architecture – System Context View



Architecture Definitions

Documenting Software Architectures: Views and Beyond (2nd Edition), Clements et al, 2010

- The set of structures needed to reason about the system, which comprises software elements, relations among them, and properties of both
- Structures consist of elements, relations among the elements, and the important properties of both. So documenting a structure entails documenting those things.

Bass, Clements, Kazman 2012 (3rd Edition)

- "The Software architecture of a system is the set of structures needed to reason about the system, which comprise software elements, relations among them, and properties of both"

Bass, Clements, Kazman. 2003 (2nd Edition)

The software architecture of a program or computing system is the structure or structures of the system, which comprise software elements, the externally visible properties of those elements, and the relationships among them.

Externally visible properties

- assumptions other elements can make of an element, such as
  - its provided services,
  - performance characteristics,
  - fault handling, shared resource usage, and so on.
- Implications
  - *architecture defines elements*.
  - the definition makes clear that *systems can and do comprise more than one structure* and that no one structure holds the irrefutable claim to being *the* architecture.

ANSI/IEEE Std 1471-2000, ***Recommended Practice for Architectural Description of Software- Intensive Systems***

the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution.

IEEE Standard..

- a variety of uses of the term architecture
- In some cases, these elements are the physical components of the system and their relationships.
- In other cases, these elements are not physical, but instead, logical components.
- In still other cases, these elements are enduring principles or patterns that create enduring organizational structures.

Software Architecture for Product Families: Principles and Practice, Mehdi Jazayeri, Alexander Ran, Frank van der Linden. Addison Wesley Longman, 2000.

Software architecture is a set of concepts and design decisions about the structure and texture of software that must be made prior to concurrent engineering to enable effective satisfaction of architecturally significant explicit functional and quality requirements and implicit requirements of the product family, the problem, and the solution domains.

Perry and Wolf, 1992

- A set of architectural (or, if you will, design) elements that have a particular form. Perry and Wolf distinguish between processing elements, data elements, and connecting elements, and this taxonomy by and large persists through most other definitions and approaches

(Software Engineering Notes, ACM SIGSOFT, 1992)

Garlan and Shaw, 1993:

...beyond the algorithms and data structures of the computation; designing and specifying the overall system structure emerges as a new kind of problem. Structural issues include gross organization and global control structure; protocols for communication, synchronization, and data access; assignment of functionality to design elements; physical distribution; composition of design elements; scaling and performance; and selection among design alternatives."

Bass, et al., 1994:

- Writing about a method to evaluate architectures with respect to the quality attributes they instill in a system , Bass and his colleagues write that
- *...the architectural design of a system can be described from (at least) three perspectives -- functional partitioning of its domain of interest, its structure, and the allocation of domain function to that structure.*

HAYES-ROTH, 1994

...an abstract system specification consisting primarily of functional components described in terms of their behaviours and interfaces and component-component interconnections.

ARPA Domain-Specific Software Architecture (DSSA) program

Garlan and Perry, 1995

- David Garlan and Dewayne Perry have adopted the following definition for their guest editorial to the April 1995 IEEE Transactions on Software Engineering devoted to software architecture:
- *The structure of the components of a program/system, their interrelationships, and principles and guidelines governing their design and evolution over time.*
- (The source of this definition was a weekly discussion group devoted to software architecture at the Software Engineering Institute.)

Soni, Nord, and Hofmeister, 1995
Siemens Corporate Research

- based on structures found to be prevalent and influential in the development environment of industrial projects they studied, software architecture has at least four distinct incarnations:
- Within each category, the structures describe the system from a different perspective:
  - The conceptual architecture describes the system in terms of its major design elements and the relationships among them.
  - The module interconnection architecture encompasses two orthogonal structures: functional decomposition and layers.
  - The execution architecture describes the dynamic structure of a system.
  - The code architecture describes how the source code, binaries, and libraries are organized in the development environment

Rational Unified Process, 1999, Kruchten

Set of significant decisions about the

- The organization of a software system,
- the selection of the structural elements and their interfaces by which the system is composed, together with their behaviour as specified in the collaborations among those elements,
- the composition of these structural and behavioural elements into progressively larger subsystems, and
- the architectural style that guides this organization---these elements and their interfaces, their collaborations, and their composition

Senthil Kumar Subramaniam (System Engineer, TCS, Denver Colorado USA, http://objectweb.cjb.net)

Architecture is the process by which the customer requirements are met by designing a well defined infrastructure with hardware/software and considering the cost, schedule and existing systems.

Sathyanarayana Panduranga (Software Design Engineer, eCapital Solutions)

- Architecture is the way to specify the structure of a software system to satisfy the functional and non-functional requirements of the system. Non functional requirements may include scalability, availability and manageability. Also the architecture dipicts some compromises that might have been made in order to reduce cost, development time etc. Architecture represents the software and supporting hardware components of the system at as abstract entities.

Raghuraman Ramasubbu(Senior Software Engineer, Mastech Corporation):

Software architecture is a framework that provides the basis for manifestation of all software objects within an enterprise. Software architecture can take a form that has a strong conceptual integrity. This architecture can then be transformed into a stable physical architecture to support organization mission and objectives.

Sandeep Rao (Technical Architect, Infosys Tech. Ltd., Bangalore, India)

Software Architecture is a statement that addresses the key concerns of the software stakeholders. Stakeholders could be sponsors, investors, builders, approvers etc.

Sridhar Palacholla (Lead Consultant, Wipro Technologies, Bangalore, India):

Software Architecture is a conceptual vehicle, usually represented in the form of different views (Logical View, Layered view, Component View, Physical/Deployment View, Systems View) that guides and governs the design, development, deployment and maintenance of the Software Application in order to meet the stated objectives, requirements (functional, non-functional) under a given set of constraints (such as cost, time, system, platform and users).

Niraj Trivedi (Assistant Vice President, Polaris Software Lab Limited)

- Software Architecture is a way of developing software systems. It facilitates migration from problem space (Requirements) to solution space (Working system).
- This migration is achieved by representing the system from various focus and perspectives, the focus being (as defined by Zachman) planner, owner, designer, builder and sub-contractor and the perspectives being what, how, where, who, when and why.

Why Software Architecture

What can we use architecture for?

- to drive system's quality attributes.
- to predict system's qualities.
- for communication among stakeholders.
- to reason about costs
- used for training

- BCK, Software Architecture in Practice

Thank you