Kordamisküsimused IDU0080 2012.a.

1. Hajussüsteemi mõiste.
   Hajutatud süsteem on arvutisüsteem, kus erinevad komponendid teevad koostööd kasutades omavahelist kommunikatsiooni üle arvutivõrgu.
2. Ettevõtte sisene ja ettevõtete vaheline integratsion. Näited.
3. Võimalused integreerimiseks.
   - Käsitsi. Üllatav on, et näiteks Euroopas ainult 5% arvetest käsitletakse automaatse integratsiooni vahenditega. Ülejäänud 95% juhtudel kasutatakse käsitsi tööd.
   - File transfer.
   - RPC
   - Teadete süsteemid. Messaging systems.
   - Koostöö andmebaasid.
   - Veebi teenused
4. Hajutatud süsteemide väljakutsed.
   - *Arvutivõrgud (internet) toob kaasa lisaohu. Andmeid tuleb transportida ühest arvutist teise läbi võrgu ja see toob kaasa lisaprobleeme.*
   - *Arvutivõrgud on aeglasemad kui infovahetus ühe arvuti sees. Ühe arvuti piires me alati ei peagi infot vahetama, vaid võime edastada teisele rakendusele viite andmetele.*
   - *Kõik rakendused võivad olla erinevad, nende loomisel võidi kasutada erinevaid programmeerimiskeeli, operatsioonisüsteeme ja andmeformaate. Integreeritud rakendus peab suutma käsitleda kõiki neid tehnoloogiaid.*
   - *Muudatused rakenduse on paratamatud. See toob kaasa ka seoses oleva rakenduse muudatusteks.*
   - *Süsteemide turvalisus. Kuidas anda turvaliselt oma rakendusele juurdepääs.*
5. Sünkroonne ja asünkroonne integratsioon.
   Põhiline vahe sünkroonse ja asünkroonse kommunikatsiooni vahel
   - Sünkroonne on request response tüüpi
   - Asünkroonne – teadetele orienteeritud keskvara kasutav - Java Messaging Service (JMS), Microsoft MQ, IBM MQ,BEA MQ

     Asynchronous processing enables methods to return immediately without blocking on the calling thread. Consumers request asynchronous processing when initializing a data source object or when opening or populating a rowset. The consumer must explicitly request asynchronous processing. Otherwise, the provider can perform asynchronous operations in the background but must behave synchronously, blocking if necessary, until the underlying asynchronous operation completes.

6. Ettevõtte teenuste siin – ESB
   An **enterprise service bus** (ESB) is a software architecture model used for designing and implementing the interaction and communication between mutually interacting software applications in Service Oriented Architecture. As a software architecture model for

distributed computing it is a specialty variant of the more general client server software architecture model and promotes strictly asynchronous message oriented design for communication and interaction between applications.

7. Teenustele orienteeritud arhitektuur.
   Teenustele orienteeritud arhitektuur on omavahel seotud teenuste kogum. Need teenused vahetavad omavahel teateid. Teated omakorda jagunevad päringteadeteks (request) ja vastusteadeteks (response). Teenuste koordineerimiseks on kasutusel spetsiaalsed vahendid (näiteks Biztalk server).
8. Veebi teenus.
   Teenus on täpselt määratletud funktsioon, mis on kirjeldatud spetsiaalse lepinguga. (WSDL – web service description language).
9. Pilve lahenduste liigid.
   Eristatakse kolme liiki pilve teenuseid. Need on:
   - Taristu kui teenus (IaaS),
   - Tarkvara platvorm kui teenus (PaaS),
   - Tarkvara kui teenus (Saas).
10. Horisontaalne ja vertikaalne integratsioon.
    - Horisontaalse integratsiooni all vaadeldakse rakenduste vahelist ühilduvust ja kooskõla. Seda nii ettevõtte sees, kui ka ettevõtete vahel.
    - Vertikaalne integratsioon on lahenduse sees. Niinimetatud mitmetasemeline lähnemisviis.

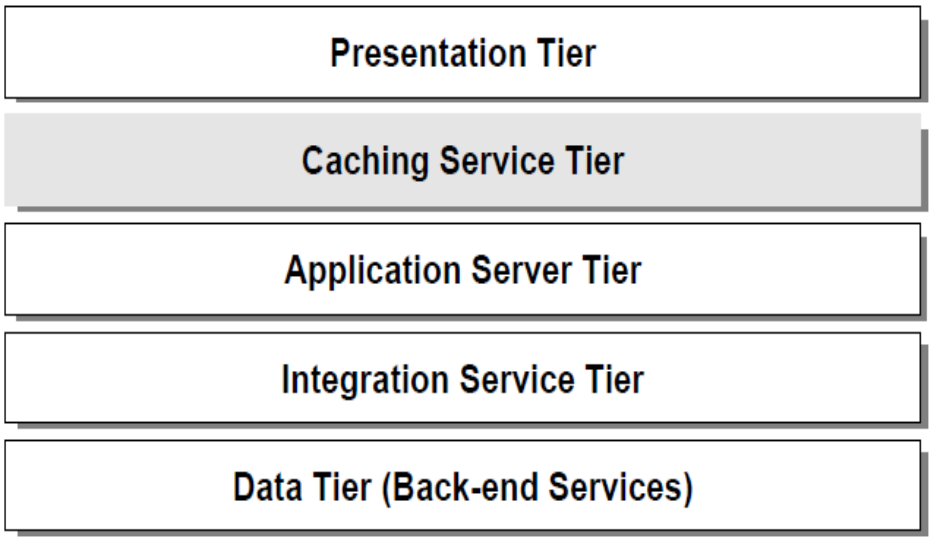


FIGURE 1 A common N-tier architecture layered approach.

11. WSDL

WSDL Document Structure

Abstract Definitions

Types

Machine- and language-independent type definitions.

Messages

Contains function parameters (inputs separate from outputs) or document descriptions.

PortTypes

Refers to message definitions in Messages section to describe function signatures (operation name, input parameters, output pameters).

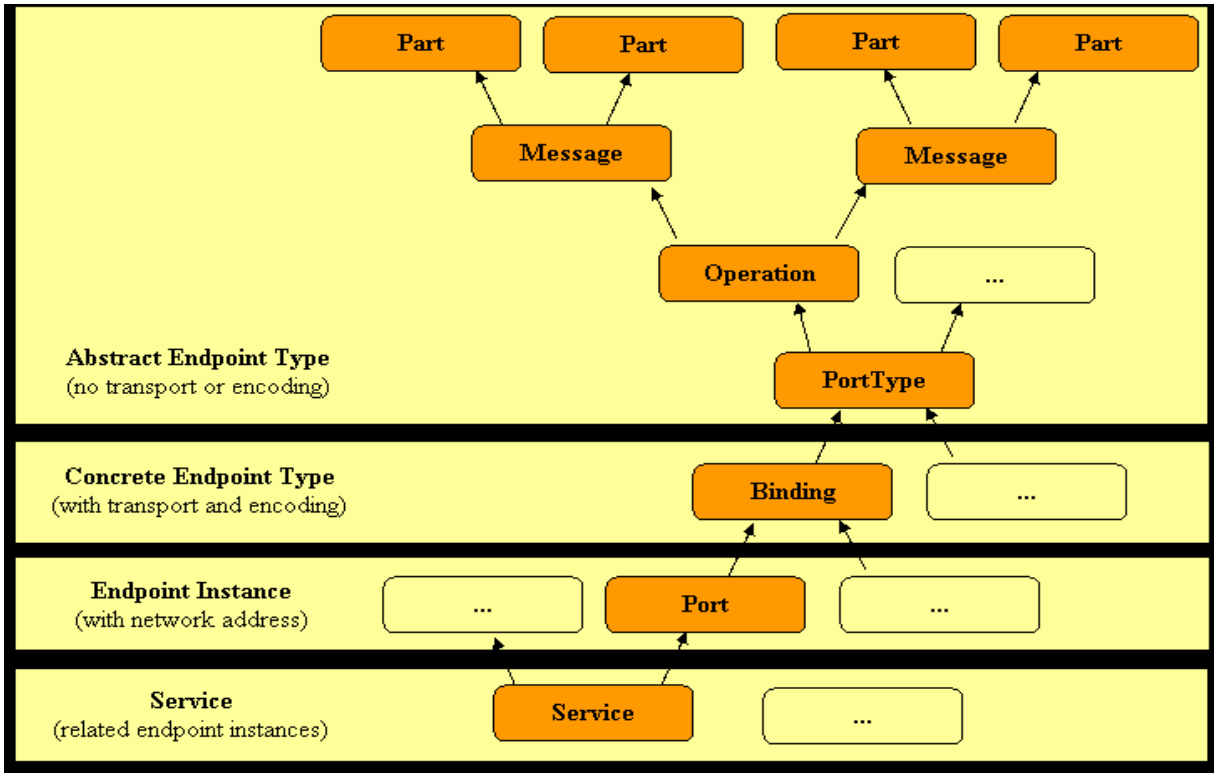
12. · **Concrete Descriptions**

Bindings

Specifies binding(s) of each operation in the PortTypes section.

Services

Specifies port address(es) of each binding.



13. REST- **Representational State Transfer**

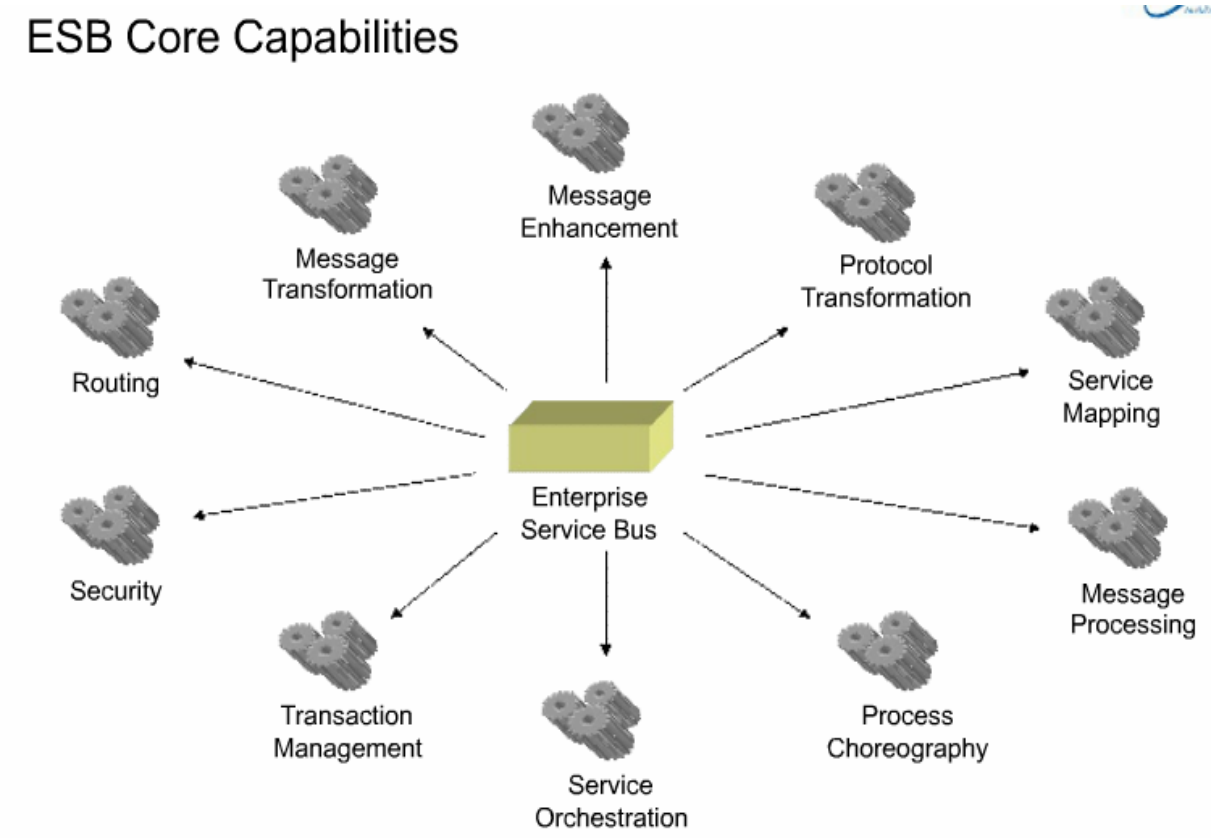
Representational State Transfer (**REST**) is a style of software architecture for distributed hypermedia systems such as the World Wide Web. The terms "Representational State Transfer" and "REST" were introduced in 2000 in the doctoral dissertation of Roy Fielding,[1] one of the principal authors of the Hypertext Transfer Protocol (HTTP) specification. The terms have since come into widespread use in the networking community.

14. REST printsiibid

REST's proponents argue that the Web enjoyed the scalability and growth that it has had as a direct result of a few key design principles:

- Application state and functionality are divided into **resources**
- Every resource is uniquely addressable using a **universal syntax** for use in **hypermedia links**
- All resources share a **uniform interface** for the transfer of state between client and resource, consisting of
  - A constrained set of **well-defined operations**
  - A constrained set of **content types**, optionally supporting **code on demand**
- A protocol that is:
  - **Client-server**
  - **Stateless**
  - **Cacheable**
  - **Layered**

15. REST protokoli eelised
    - Provides improved response times and server loading characteristics due to support for **caching**
    - Improves server scalability by reducing the need to maintain communication state. This means that different servers can be used to handle initial and subsequent requests
    - Requires less client-side software to be written than other approaches, because a single browser can access any application and any resource
    - Depends less on vendor software than mechanisms that layer additional messaging frameworks on top of HTTP
    - Provides equivalent functionality when compared to alternative approaches to communication
    - Does not require a separate resource discovery mechanism, due to the use of hyperlinks in content
    - Provides better long-term compatibility and evolvability characteristics than RPC. This is due to:
    - The capability of document types such as HTML to evolve without breaking backwards- or forwards-compatibility, and
    - The ability of resources to add support for new content types as they are defined without dropping or reducing support for older content types.
16. Veahaldus ja logiraamat, mis need on?
17. ESB põhivõimekused?



18. Biztalk
    - Tasuta kaasas 35 adapterit, lisaks suur hulk open-source adaptereid CodePlex'is
    - Väga lihtne arendada uusi adaptereid
    - Visuaalne arendus
    - BPM funktsionaalsus
    - Business Rule Engie (BRE)
    - Business Activity Monitoring (BAM)
    - Ühtne haldus ja administreerimisliides
    - EDI tugi ja kiirendid (X12, EDIFACT)
    - Kiirendid (RosettaNet, HL7, HIPAA, SWIFT)
    - RFID tugi
    - Konkurentsivõimeline hind

19. Äriprotsesside juhtimine –BPM

    Business Process Management (BPM)

    Gartner defines **business process management (BPM)** as a management discipline that treats business processes as assets that directly improve enterprise performance by driving operational excellence and business agility.