演讲稿部分

Part 1

// 刘浩宇

演讲稿

Last Thursday we talked about "software architecture" .Software architecture is the blueprint for designing and developing software systems. It defines the work assignments for design and implementation teams, and is crucial for achieving system qualities like performance, modifiability, and security. The four main views of system architecture are the logical, development, physical, and process views. Additionally, the scenario view helps identify architectural elements and validate design. Software architecture is essential for ensuring that software systems perform as intended and meet user needs. With a solid architecture in place, software development teams can work efficiently and deliver high-quality products that meet customer requirements.

PPT

Part 2

// 陈戒

Regarding software architecture, other scientists have also conducted related research and put forward their views.

Over the past four decades, applications and services have been moving slowly but consistently towards a uniform model based on Service Oriented Approach (SOA). The shift towards abstract models, objects and services however is not efficiently supported by the underlying delivery platforms, especially the legacy Internet architecture. An architectural rethinking is necessary at the network level well to accommodate future services, applications and routing priorities. Thus, ROSA, a service oriented framework with relationship as the orchestration parameter, is proposed as a solution. ROSA is the synthesis of various ideas already existing or proposed into a coherent architecture based on accepted standards.

Part 3

// 陈实

We learned about Relationship-Oriented software Architecture. When it comes to software architecture, it is naturally to talk about the programming thinkings. The current mainstream programming thinking is Object-Oriented. Mainly based on the properties and behavior of things, but there is a lack of abstraction of the relationship between things and things. Therefore, whether the properties and inherent behavior of things themselves are based on object-orientation, and the association relationship between other things will be further abstracted, which is called Relationship-Oriented. The associations between things are addressed through Relation-Oriented programming.

There are various associations in the world, and what the relationship should do is to separate the related behavior of things from other things from the inherent behavior of things and further abstract the associations. It is precisely because the relationship between things is too complex, which will make the implementation of functions or systems to be too large. If we can have a better solution to solve the relationship between things, it may simplify the development of the programming.

PPT部分

Part 1

**What is Software Architecture?**

1.Definition:

System architecture serves as the blueprint for both the system and the project developing it.

It defines the work assignments that must be carried out by design and implementation teams.

2.System Qualities:

Architecture is the primary carrier of system qualities, such as performance, modifiability, and security.

None of these qualities can be achieved without a unifying architectural vision.

**Four Views and Scenario View of System Architecture**

1.Logical View: Focuses on the system's functionality and end-user interactions.

2.Development View: Focuses on software management from a programmer's perspective.

3.Physical View: Focuses on the system's physical layer and the connections between software components.

4.Process View: Focuses on the system's dynamic aspects, such as its runtime behavior and performance.

5.Scenario View: Describes the system's interactions and uses a small set of use cases to identify architectural elements and validate design.

Part 2

Other Views

* The concept of Service Oriented Architecture (SOA) was adopted to enable a standardised and open way for enterprises to open up their IT infrastructure for collaboration. We note the best known implementation of SOA in Web Services (WS-\*) specifications, standardization and its implementations. While, Web Services specifications specifically address Enterprise Application Integration (EAI), we need a similar principle and framework to be applied to future heterogenous communication networks. ‘’
* ROSA - A Service Framework: We propose a frame work termed ‘Relationship Oriented Service Architecture’ (ROSA) to agree upon a broad vocabulary that will be used to model recurring themes in ICT and integration environments. The aim of such a framework is to be able to reference one or more open specification or standards for each identified service, that can be used to implement various version of the service. ‘’

‘’ R. Kumar, A. Haber, A. Yazidi and F. Reichert, "Towards a relation oriented service architecture," 2010 Second International Conference on COMmunication Systems and NETworks (COMSNETS 2010), Bangalore, India, 2010, pp. 1-8, doi: 10.1109/COMSNETS.2010.5431972.

Part 3

Programming Thinkings

* The current mainstream programming thinking is Object-Oriented Programming Thinkings.
* A lack of abstraction of the relationship between things and things.
* Association relationship between things will be further abstracted, which is called Relationship-Oriented.
* If we can have a better solution to solve the relationship between things, it may simplify the development of the programming