

JNDI

Java Naming and Directory Interface

1. Erklären Sie die Funktionsweise von Stub und Skeleton

Der Stub

Wenn der Client eine Methode eines CORBA Objektes aufruft, so muss dieser Methodenaufruf an den Server weitergesendet werden. Dazu muss eine Nachricht erzeugt werden, welche die Werte der zu übergebenden Parameter enthält. Da der Server in einem anderen Prozessraum oder sogar in einer anderen Maschine ist, müssen alle Werte, die übergeben werden sollen, in diese Nachricht verpackt werden. Dazu werden die Informationen über die Parameter verwendet (Datentyp und Richtung). Die Erstellung der Nachricht übernimmt der Stub. Er kopiert alle Parameterwerte, die an den Server zu übergeben sind, in die Nachricht und übergibt diese an den ORB. Das Verpacken des Methodenaufrufes in eine Nachricht bezeichnet man auch als *Marshalling*.

Der Skeleton

Der Server-Stub heisst in CORBA Skeleton. Hier wird die Nachricht, die der Server über den ORB vom Client erhalten hat, ausgepackt. Mit Hilfe der Information in der erhaltenen Nachricht wird dann die angeforderte Methode des CORBA Objektes aufgerufen. Das Auspacken der Nachricht nennt man *Unmarshalling*.
(Quelle: SWK, SW2-Skript_Komponentenarchitektur.pdf, Kapitel 9.2)

2. Wie nennt man das Interface das die Gegenstelle zum API bildet und die eine Problemlose Kommunikation mit dem Java API ermöglichen soll?

Service Provider Interface (SPI)

3. Welche 2 Services stellt das JNDI zur Verfügung?

- *Namensdienste (z.B. für Objekte)*
- *Verzeichnisdienste (erweiterter Namensdienst)*

4. Was muss als allererstes erzeugt werden bei einer möglichen Verwendung eines Naming Services um ein Java Objekt zu registrieren?

Initial Context (Wo finde ich was)

5. Nennen Sie einige der Basismethoden mit denen Sie auf den Namensdienst zugreifen resp. navigieren?

- *bind*
- *unbind*
- *rebind*
- *create Subcontext*

6. Was ist ein distinguished Name (dn) und wie setzt er sich zusammen?

Eindeutiger Name um Objekte zu lokalisieren. Hierarchische Zusammensetzung aus bestimmte Knoten.

Der distinguished Name im JNDI Namens-Modell identifiziert ein Object Entry („Knoten“ in der LDAP-Baumstruktur) eindeutig vom entsprechenden Eintrag bis zur Wurzel („Root“). Diese Referenzierungen folgen der X.500-Spezifikation von LDAP (Abkürzungen).

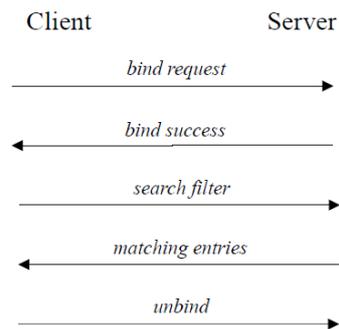
7. Welche 4 Modelle implementiert das LDAP Protokoll?

- **Information Model:** *Object Entries („Knoten“) in der Baumstruktur mit Attributen und Werten*
- **Naming Model:** *Jedes Object Entry ist über den dn (siehe Frage 6) eindeutig referenziert*
- **Functional Model:** *LDAP beinhaltet Interrogations Operations (Abfragen), Update Operations sowie Authentication and Control Operations*
- **Security Model:** *ermöglicht eine flexible Zugriffskontrolle auf die gesamte Baumstruktur, wobei die Verbindung zum LDAP-Server über SSL / TLS zusätzlich noch abgesichert werden kann*

8. Welche Operation nennt man Interrogation Operations (Abfrage) und welches Äquivalent hat sie in der Welt der Datenbanken?

Mit dem „search filter“ werden die Abfragen ausgeführt.

Datenbank: select



9. Was ist besonderes an einer LDAP Anfrage (ganz im Gegensatz zu einer DB Anfrage?)

LDAP unterscheidet sich wesentlich im Lese- / Schreib-Verhältnis zu einer Datenbank: Durch die standardisierte „baumartige“ Struktur ist die Lesegeschwindigkeit (und nur diese!) wesentlich höher.

Weiter kann die LDAP-Struktur nicht frei gewählt, sondern lediglich erweitert werden. LDAP ist zudem verteilbar und replizierbar.