Script   generated by TTT

Title: Distributed_Applications (22.04.2013)

Date: Mon Apr 22 09:14:34 CEST 2013

Duration: 46:52 min

Pages: 12

Xerox PARC experimented in the 1970's with distributed applications (Alto workstation, Ethernet).

book of Ken Birman (chap 27) gives a brief overview of a number of distributed systems, e.g. Amoeba, NavTech, Totem, Argus, etc.

Mach

Sun Network File System (NFS)

Java 2 Platform Enterprise Edition (J2EE)

Google

Generated by Targeteam

network extension to Unix and other operating systems for distributed file management.

Characteristics

NFS implementation

Generated by Targeteam

File catalogs are exported (by server subsystems) and mounted (by the client machines).
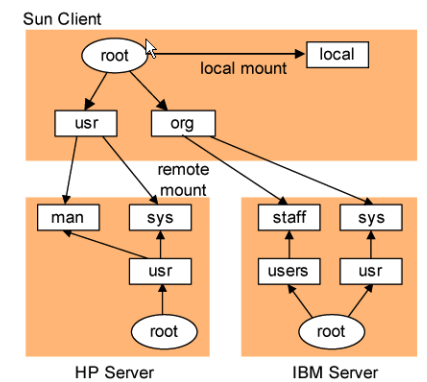
Support of a *mount service* :

file `/etc/exports` on NFS server lists names of local filesystem available for remote mounting.

mounting request by client with: remote host, directory pathname and local name with which it is to be mounted.

automounter : dynamically mounting of a remote directory whenever an 'empty' mount point is referenced by a client.
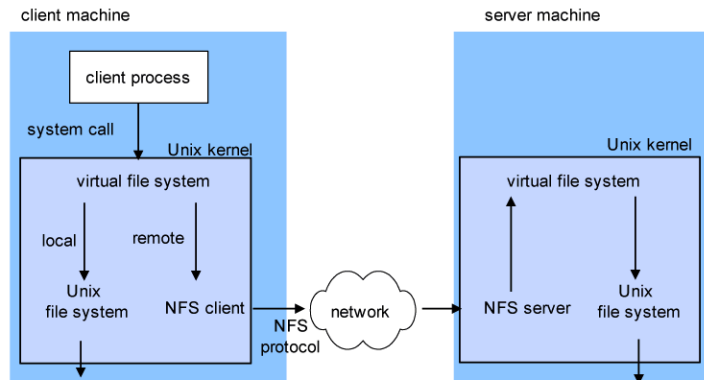
NFS supports access transparency

NFS implementation is based on RPC calls between the involved operating systems. It can be configured to use UDP or TCP.

earlier version of NFS was a *stateless* file server, i.e. a server subsystem does not store state information about its clients and their past operations.

current version of NFS is a *stateful* file server, i.e. a server subsystem supports locking and delegation of actions to client to improve client-side caching.

Generated by Targeteam

---

network extension to Unix and other operating systems for distributed file management.

Characteristics

NFS implementation

Generated by Targeteam

The idea of J2EE is to provide a standardized programming model for the realization of distributed applications at the organizational level.

Java-based, but with interfaces to legacy applications, for example through Corba.

component-based.

network-oriented: supporting Web Services.

J2EE consists of 2 components

a *runtime infrastructure* for applications.

a set of *Java extension APIs* to build applications. Examples are Enterprise Java Beans (EJB), Java Servlets, JavaServer Pages (JSP), RMI via Internet-Inter-ORB Protocol (RMI-IIOP), Java Naming and Directory Interface (JNDI), Java Transaction API and Java Mail.
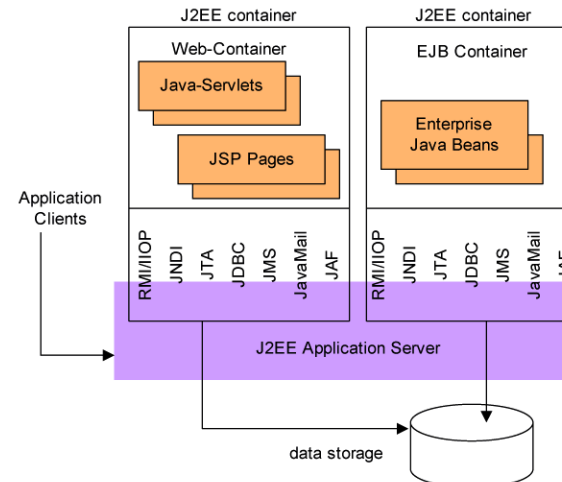
Generated by Targeteam

A J2EE platform consists of the J2EE application server (runtime environment), one or several J2EE containers, and the data storage.

EJB is a specification of a server-side, managed component architecture.

a bean offers one or more business interfaces to clients.

especially suited for 3-tier architectures.

A typical J2EE platform has one or several containers. A J2EE container has two principal tasks:

runtime environment for managing application components.

to provide access to J2EE APIs.

available APIs of the J2EE platform

RMI/IIOP: Remote Method Invocation (via IIOP)

JNDI: Java Naming and Directory Interface

JTA: Java Transaction API

JDBC: Java Database Connectivity Extension

JMS: Java Message Service

Java Mail

JAF: Java Beans Activation Framework.

Examples for application components: JavaServlets, JavaServer Pages, Enterprise JavaBeans.

J2EE supports the following general containers

Web container: Java Servlets, JSP pages

EJB container : Enterprise Java Bean components

A J2EE application consists of several modules, each of which again contains several application components. Modules and application components are listed in an archive file:

EAR (Enterprise archive), WAR (Web archive) or JAR (Java archive)

Generated by Targeteam

JavaServer Pages technology uses XML-like tags and scriptlets written in the Java programming language to encapsulate the logic that generates the content for the Web page.

Comment: <%--Comment --%>

Declaration: <%! int x = 0; %>

Expression: <%= expression %>

Scriptlets -contain Java Code

```
<% code fragments %>
    <% if (value.getName().length != 0) { %>
      <H2>The value is: <%= value.getName()%></H2>
    <%} else {%>
      <H2>Value is empty</H2>
    <% }%>
```
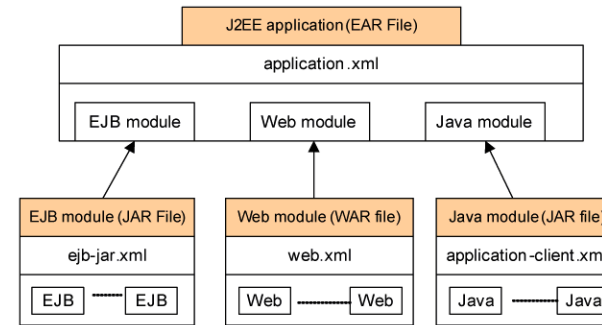
implicit objects available to JSP

request, response, session, out, page

Generated by Targeteam

The J2EE platform (now called Java Platform, Enterprise Edition - Java EE) is essentially a distributed application server environment. It is a Java environment that provides the following:

a runtime infrastructure for hosting applications.

a set of Java extension APIs to build applications.

Objectives of J2EE

J2EE architecture

J2EE container

J2EE application

Java Server Pages

Example implementations

JBoss : Open Source

advanced middleware for J2EE based distributed applications

IBM WebSphere : proprietary

integration and application infrastructure software; provides J2EE support

J2EE is continuously extended by new technologies, e.g. integrating the support for Web Services.

Generated by Targeteam

Google is one of the largest distributed systems in use today.

> besides providing a search engine Google is now a major player in cloud computing.

> end of 2010 more than 88 billion queries a month; never experienced a major outage since the beginning in 1998.

Google provides a significant number of applications, such as

> GMail, Google Docs, Google Calendar, Google Wave, Google Maps, Google Earth, Google Wave, Google News, Google App Engine, etc.

physical infrastructure: commodity PCs organized in racks which are organized into clusters with very large storage capacity.

> Clusters are housed in Google data centers.

middleware: communication paradigm based on protocol buffers and publish-subscribe.

distributed computation based on MapReduce and Sawzall. Key principle of MapReduce

> break input data into a number of chunks.

> carry out initial processing on these chunks of data to produce intermediary results.

> combine intermediary results to produce the final output.