1. What is Cloud Computing (CC)? 3
   1.1. Utility Computing First 3
   1.2. The New Features of Cloud Computing (CC) 4
   1.3. Cloud Service Provider (CSP) 6
      1.3.1 Thin Provisioning 7
   1.4. CC via a CSP (not via own cloud OS) 8
      1.4.1 Public Cloud 9
      1.4.2 Private Cloud 9
      1.4.3 Hybrid Cloud 9
      1.4.4 Infrastructure-as-a Service (IaaS) 10
         1.4.4.1 Resource Administration by the User 11
         1.4.4.2 The Role of the CSP 11
         1.4.4.3 Nested Virtualization 12
      1.4.5 Platform-as-a-Service (PaaS) 12
      1.4.6 Software-as-a-Service (SaaS) 13
      1.4.7 Special Access-Offers 14
         1.4.7.1 Community Cloud 14
         1.4.7.2 Virtual Private Cloud 15
         1.4.7.3 Multi Cloud 15
   1.5. CC with own Cloud OS (not via CSP) 16
2. What are the Pros and Cons of CC? 17
   2.1. CC via CSP 19
      2.1.1 Pros 19
         2.1.1.1 Server Consolidation via VM Migration 20
      2.1.2 Cons 21
         2.1.2.1 Vendor-Lock-in 22
   2.2. CC with own Cloud OS 22
      2.2.1 Pros for own Cloud OS 23
      2.2.2 Cons for own Cloud OS 25
3. Technologies of CC 27
   3.1. Common Technologies for Both, CSP Clouds and Own Cloud 27
   3.2. Tenant and Multi-Tenancy 27
      3.2.1 Tenant 27
      3.2.2 Multi-Tenancy 28
   3.3. Linux Containers (LXCs) 29
      3.3.1 Cgroups 30
      3.3.2 Namespace Isolation 30
      3.3.3 LXC Functionalities 31
   3.4. Virtualization by means of VMs 32
   3.5. Auto-Scaling 33
   3.6. Disjoint Software Technologies 34
   3.7. Specific Technologies for CC via CSPs 34
      3.7.1 Language and Platform-specific APIs 35
3.8. Specific Technologies for CC via own Cloud OS 35
3.8.1 Location of own Cloud OS 36
3.8.2 Basic Services in own Cloud OS 38

4. Virtual Machines (VMs) 39

4.1. What is a VM? 40
4.2. Advantages of Virtualization 42
4.2.1 Setup of a VM 45
4.3. Disadvantages of Virtualization 46
4.4. Host OS and Guest OS 47
4.5. Hypervisor 48
4.5.1 ESXi, Hyper-V and KVM/QEMU 51
4.6. Paravirtualization 53
4.6.1 Virtio for Paravirtualization 54

5. Server Virtualization 55

5.1. Comparison between Classical Computer and Virtualized Server 56
5.2. How the Host OS sees Guest OSes, QEMUs and User Applications 58
5.3. Privileged CPU instructions in Guest OS 59
5.4. Server Virtualization in a Cloud 60
5.5. Virtualization of Memory, CPU and IO 62
5.5.1 Memory Virtualization 62
      5.5.1.1 Virtual Shared Memory 64
5.5.2 CPU Virtualization 64
5.5.3 IO Virtualization 65
      5.5.3.1 Example of IO Virtualization by Sending an Email from a VM 66
5.6. Hardware Accelerators for Server Virtualization 66
5.6.1 Accelerators for Memory Virtualization 67
5.6.2 Accelerators for CPU Virtualization 67
5.6.3 Accelerators for IO Virtualization 67
      5.6.3.1 SR-IOV 68
5.6.4 Status of Hardware Accelerators for Server Virtualization 69
5.6.5 Summary of Hardware Accelerators 70

5.7. Inter-vCPU and Inter-VM Communication on the Same Server 70
5.7.1 Inter-VM Communication Without Ivshmem 72
      5.7.1.1 Tap Device 72
      5.7.1.2 Macvlan Driver 73
      5.7.1.3 Macvtap 73
      5.7.1.4 Inefficient Inter-VM Communication as Result 74
5.7.2 Inter-VM Communication With Ivshmem 74

5.8. Inter-VM Communication between Servers without Cloud OS 78
5.9. Inter-VM Communication between Servers with Cloud OS 78
5.9.1 Inter-VM Communication between Servers with OpenStack 80
6. Overview on OpenStack and its Services 85

6.1. REST Protocol 86

6.2. Calling and Controlling an OpenStack Service 87

6.2.1 Extended UUID Usage for Information Items 88

6.2.2 Calling via REST API 89

6.2.3 Calling from Host OS Shell 90

6.2.4 Calling via GUI 91

6.2.5 Caveats 92

6.2.6 Calling via CLI 92

6.2.7 Calling via UUID 92

6.3. Overview on OpenStack Services 92

6.4. The 10 Most-Important OpenStack Services 96

6.5. The Smallest Possible OpenStack System 101

6.6. Controller Node and Other Nodes 102

6.7. Minimum Software Stack in a Compute Node 104

6.8. Intra-Service and Inter-Service Communication 105

6.8.1 oslo.messaging 105

6.8.1.1 Advanced Message Queuing Protocol (AMQP) 106

6.8.1.2 ZeroMQ Message Transfer Protocol (ZMTP) 107

6.8.1.3 Notifications 108

6.9. The Horizon Service 108

6.9.1 Horizon GUI 109

6.9.2 List of Tabs and Sub-tabs for the User GUI 112

6.9.2.1 Compute Tab 113

6.9.2.2 Volume Tab 113

6.9.2.3 Network Tab 114

6.9.2.4 Object Store Tab 115

6.9.3 List of Tabs and Sub-tabs for the Admin GUI 115

6.9.3.1 Overview Tab 115

6.9.3.2 Compute Tab 115

6.9.3.3 Volume Tab 117

6.9.3.4 Network Tab 117

6.9.3.5 System Tab 118

6.9.4 Horizon Projects (Tenants) And User Authorization 119

6.9.5 Extensions to Horizon 120

6.9.6 Abstract Service API of Horizon 121

6.9.7 Horizon GUI Terminology 121

6.10. General OpenStack Terminology 123

6.11. The Nova Service 125

6.11.1 Nova Compute Cells 125

6.11.2 Nova Internal Setup 127

6.11.3 Parallel Processing Inside of Nova 128

6.11.4 Communication Between Nova Components 129

6.12. The Neutron Service 130

6.12.1 Neutron Task List 130

6.12.2 Neutron Software Defined Networks 132
6.12.3 Neutron Components for ISO Layer 2 Operation 133
6.12.4 Neutron Components for ISO layer 3 Operation 133
6.12.5 Further Neutron Components 133
6.12.6 Types of Data to be Exchanged in OpenStack 135
6.12.7 Example of Data-Type Exchanges 136
6.12.8 Example Neutron Network Topology 138

6.13. The Keystone Service 140
6.13.1 Keystone Terminology 140
6.13.2 Securing API Requests 142
6.13.3 Fernet Tokens 142
  6.13.3.1 The Algorithms in a Fernet Token 144
  6.13.3.2 Creating a Fernet Token 145
  6.13.3.3 Using a Fernet Token 146
  6.13.3.4 Checking the Fernet Token 147
  6.13.3.5 Example Interplay Between Caller, Callee and Horizon 148
  6.13.3.6 Advantages of Fernet Tokens 148
  6.13.3.7 Disadvantages of Fernet Tokens 150
6.13.4 Token Revocation 150
6.13.5 Keystone REST Request Examples 151
6.13.6 Internal Keystone Setup 153

6.14. The Cinder Service 154
6.14.1 Cinder Terminology 155
6.14.2 Cinder's External Setup 156
6.14.3 Cinder's Internal Setup 159
6.14.4 Cinder Virtual Hard Drive 160
  6.14.4.1 Attachment of a Virtual Hard Drive to a VM 162
  6.14.4.2 Accessing Records in Cinder Volumes 163
6.14.5 Cinder Snapshot 164
6.14.6 Cinder Backup 164
6.14.7 Cinder Volume Encryption 164

6.15. The Swift Service 165
6.15.1 Swift Terminology 167
6.15.2 Object Replication 168
6.15.3 Asynchronous Eventual Consistency for Replicas 168
6.15.4 Erasure Codes 169
  6.15.4.1 Swift Use Case for Erasure Codes 170
6.15.5 Swift Data Hierarchy 170
  6.15.5.1 Swift Accounts 171
  6.15.5.2 Swift Containers 172
  6.15.5.3 Swift Objects 172
6.15.6 Internal Swift Setup 173
  6.15.6.1 Account Components 173
  6.15.6.2 Container Components 173
  6.15.6.3 Object Components 174
  6.15.6.4 Proxy Server 174
6.15.7 Functional Concept of Swift 175
  6.15.7.1 Manifest Objects 175
  6.15.7.2 Swift Proxies 176
  6.15.7.3 Replica Policy 177
  6.15.7.4 Erasure Code Policy 177
6.15.7.5 Load Balancing 178
6.15.7.6 Hash Ring 179
6.15.7.7 Ring Partition Number 180
6.15.7.8 Replica-to-Partition-to-Device Table 180
6.15.7.9 2nd Table In the Hash Ring 182
6.15.7.10 The Swift-Ring-Builder 182

6.16. The Glance Service 182
6.16.1 Glance Internal Setup 183
6.16.2 Glance Tasks 184

6.17. The Trove Service 184
6.17.1 Trove Terminology 185
6.17.2 Classification of Trove Datastores 185
6.17.3 Storage Locations 186
6.17.4 Trove Configuration Groups 186
6.17.5 Trove Internal Setup 186
6.17.6 Database-VM Minimum-Requirements 189
6.17.7 Trove Guest-Agent Configuration 189
6.17.8 Save Inter-Component Communication 190
6.17.9 Managing DBMSes and their Datastores 190
6.17.10 The Trove API 191
   6.17.10.1 Create and Manage Database VMs 192
   6.17.10.2 Create and Manage and Databases 192
   6.17.10.3 Manage Datastores 192
   6.17.10.4 Create and Manage Replicas of Database VMs 192
   6.17.10.5 Create and Manage Backups of Database VMs 193
   6.17.10.6 Create and Manage Configuration Groups 193

6.18. The Ceilometer Service 193
6.18.1 Ceilometer Terminology 194
6.18.2 Types of Meters 195
6.18.3 Ceilometer Components 196
6.18.4 Ceilometer Operations 197
   6.18.4.1 Collecting Usage Data 197
   6.18.4.2 Collecting Events 198
   6.18.4.3 Collecting Alarms 198
6.18.5 Supported Hypervisors 198
6.18.6 Other Supported Tools 199

6.19. The Heat Service 199
6.19.1 Orchestrating 199
6.19.2 AWS CloudFormation Templates 201
6.19.3 Heat Orchestration Templates 202
6.19.4 Difference between Heat Stack and HOT 204
6.19.5 Heat Setup 206
6.19.6 Heat Terminology 207
6.19.7 The native Heat API 208

7. Overview 210

7.1. Using AWS Services 211
7.2. Calling AWS Services 212
7.2.1 Calling Via an SDK 212
7.2.2 Regional Endpoints For Calls 213
7.3. AWS Accounts and Users 213
7.4. AWS Account IDs 214
7.5. AWS Resource Names (ARNs) 215
7.6. AWS Identity and Access Management (IAM) 215
    7.6.1 Account Access Keys 216
    7.6.2 IAM User Access Key 217
    7.6.3 Temporary IAM Credentials 217
7.7. Data Protection and AA in REST Requests 218
8. Description of Selected AWS Services 219
    8.1. Simple Storage Service (S3) 219
    8.2. Buckets 220
        8.2.1 Creation of a Bucket 221
        8.2.2 Folders in a Bucket 222
        8.2.3 Retrieving Objects in a Bucket 222
        8.2.4 Automatic Versioning of Objects 222
        8.2.5 Website Hosting 222
        8.2.6 Additional AA for Buckets 223
            8.2.6.1 Bucket-Access Control-Lists (ACLs) 223
            8.2.6.2 Bucket Access Policies 224
        8.2.7 Other Access Policies and ACLs 225
            8.2.7.1 Object-Access Control-Lists 225
            8.2.7.2 User Access Policies 226
        8.2.8 Personal Opinion 227
    8.3. Entity Tag for Data Protection 227
    8.4. Eventual Consistency and Simultaneous Writing 228
        8.4.1 Storage Classes 229
    8.5. Calling S3 229
    8.6. S3 REST API 229
        8.6.1 Operations On Buckets 230
        8.6.2 Operations On Objects 230
    8.7. Elastic Compute Cloud (EC2) 230
    8.8. DynamoDB 233
    8.9. Data Storage and Retrieval 235
        8.9.1 Primary Key 236
        8.9.2 Secondary Key 237
            8.9.2.1 Global Secondary Key 237
            8.9.2.2 Local Secondary Key 238
            8.9.2.3 Multiple Secondary Indices 238
    8.10. DynamoDB Streams 238