

## TClouds ACaaS Module

Generated by Doxygen 1.8.3.1

Mon Aug 19 2013 19:17:56



# Contents

<b>1</b>	<b>Hierarchical Index</b>	<b>1</b>
1.1	Class Hierarchy	1
<b>2</b>	<b>Class Index</b>	<b>5</b>
2.1	Class List	5
<b>3</b>	<b>Class Documentation</b>	<b>7</b>
3.1	<a href="#">nova.scheduler.filters.acaas_filter.ACaaSFilter Class Reference</a>	7
3.1.1	Detailed Description	7
3.1.2	Member Function Documentation	7
3.1.2.1	host_passes	7
3.2	<a href="#">nova.api.openstack.compute.servers.ActionDeserializer Class Reference</a>	7
3.2.1	Detailed Description	8
3.3	<a href="#">nova.db.sqlalchemy.models.AgentBuild Class Reference</a>	8
3.3.1	Detailed Description	8
3.4	<a href="#">nova.db.sqlalchemy.models.Aggregate Class Reference</a>	9
3.4.1	Detailed Description	9
3.5	<a href="#">nova.compute.api.AggregateAPI Class Reference</a>	9
3.5.1	Detailed Description	10
3.5.2	Member Function Documentation	10
3.5.2.1	add_host_to_aggregate	10
3.5.2.2	create_aggregate	10
3.5.2.3	delete_aggregate	10
3.5.2.4	get_aggregate	10
3.5.2.5	get_aggregate_list	10
3.5.2.6	remove_host_from_aggregate	10
3.5.2.7	update_aggregate	11
3.5.2.8	update_aggregate_metadata	11
3.6	<a href="#">nova.db.sqlalchemy.models.AggregateHost Class Reference</a>	11
3.6.1	Detailed Description	11
3.7	<a href="#">nova.db.sqlalchemy.models.AggregateMetadata Class Reference</a>	11
3.7.1	Detailed Description	12

3.8	<a href="#">nova.compute.api.API Class Reference</a>	12
3.8.1	<a href="#">Detailed Description</a>	14
3.8.2	<a href="#">Member Function Documentation</a>	14
3.8.2.1	<a href="#">add_fixed_ip</a>	14
3.8.2.2	<a href="#">attach_volume</a>	14
3.8.2.3	<a href="#">backup</a>	14
3.8.2.4	<a href="#">confirm_resize</a>	14
3.8.2.5	<a href="#">create</a>	14
3.8.2.6	<a href="#">create_db_entry_for_new_instance</a>	14
3.8.2.7	<a href="#">delete</a>	15
3.8.2.8	<a href="#">delete_instance_metadata</a>	15
3.8.2.9	<a href="#">detach_volume</a>	15
3.8.2.10	<a href="#">force_delete</a>	15
3.8.2.11	<a href="#">get</a>	15
3.8.2.12	<a href="#">get_active_by_window</a>	15
3.8.2.13	<a href="#">get_all</a>	15
3.8.2.14	<a href="#">get_console_output</a>	15
3.8.2.15	<a href="#">get_diagnostics</a>	15
3.8.2.16	<a href="#">get_instance_bdms</a>	16
3.8.2.17	<a href="#">get_instance_faults</a>	16
3.8.2.18	<a href="#">get_instance_metadata</a>	16
3.8.2.19	<a href="#">get_instance_type</a>	16
3.8.2.20	<a href="#">get_lock</a>	16
3.8.2.21	<a href="#">get_vnc_console</a>	16
3.8.2.22	<a href="#">inject_file</a>	16
3.8.2.23	<a href="#">inject_network_info</a>	16
3.8.2.24	<a href="#">live_migrate</a>	16
3.8.2.25	<a href="#">lock</a>	16
3.8.2.26	<a href="#">pause</a>	16
3.8.2.27	<a href="#">reboot</a>	17
3.8.2.28	<a href="#">rebuild</a>	17
3.8.2.29	<a href="#">remove_fixed_ip</a>	17
3.8.2.30	<a href="#">rescue</a>	17
3.8.2.31	<a href="#">reset_network</a>	17
3.8.2.32	<a href="#">resize</a>	17
3.8.2.33	<a href="#">restore</a>	17
3.8.2.34	<a href="#">resume</a>	17
3.8.2.35	<a href="#">revert_resize</a>	17
3.8.2.36	<a href="#">set_admin_password</a>	17
3.8.2.37	<a href="#">snapshot</a>	18

3.8.2.38	<a href="#">snapshot_volume_backed</a>	18
3.8.2.39	<a href="#">soft_delete</a>	18
3.8.2.40	<a href="#">start</a>	18
3.8.2.41	<a href="#">stop</a>	18
3.8.2.42	<a href="#">suspend</a>	18
3.8.2.43	<a href="#">trigger_provider_fw_rules_refresh</a>	18
3.8.2.44	<a href="#">unlock</a>	18
3.8.2.45	<a href="#">unpause</a>	18
3.8.2.46	<a href="#">unrescue</a>	19
3.8.2.47	<a href="#">update</a>	19
3.8.2.48	<a href="#">update_instance_metadata</a>	19
3.9	<a href="#">nova.db.sqlalchemy.models.BandwidthUsage Class Reference</a>	19
3.9.1	<a href="#">Detailed Description</a>	20
3.10	<a href="#">nova.db.sqlalchemy.models.BlockDeviceMapping Class Reference</a>	20
3.10.1	<a href="#">Detailed Description</a>	20
3.10.2	<a href="#">Member Data Documentation</a>	20
3.10.2.1	<a href="#">instance</a>	20
3.10.2.2	<a href="#">instance_uuid</a>	21
3.11	<a href="#">nova.db.sqlalchemy.models.Certificate Class Reference</a>	21
3.11.1	<a href="#">Detailed Description</a>	21
3.12	<a href="#">nova.api.openstack.compute.servers.CommonDeserializer Class Reference</a>	21
3.12.1	<a href="#">Detailed Description</a>	22
3.13	<a href="#">nova.compute.rpcapi.ComputeAPI Class Reference</a>	22
3.13.1	<a href="#">Detailed Description</a>	23
3.13.2	<a href="#">Member Function Documentation</a>	24
3.13.2.1	<a href="#">add_aggregate_host</a>	24
3.13.2.2	<a href="#">host_maintenance_mode</a>	25
3.13.2.3	<a href="#">remove_aggregate_host</a>	25
3.14	<a href="#">nova.compute.manager.ComputeManager Class Reference</a>	25
3.14.1	<a href="#">Detailed Description</a>	27
3.14.2	<a href="#">Constructor &amp; Destructor Documentation</a>	27
3.14.2.1	<a href="#">__init__</a>	27
3.14.3	<a href="#">Member Function Documentation</a>	27
3.14.3.1	<a href="#">add_aggregate_host</a>	27
3.14.3.2	<a href="#">add_fixed_ip_to_instance</a>	27
3.14.3.3	<a href="#">attach_volume</a>	27
3.14.3.4	<a href="#">change_instance_metadata</a>	27
3.14.3.5	<a href="#">check_can_live_migrate_destination</a>	27
3.14.3.6	<a href="#">check_can_live_migrate_source</a>	28
3.14.3.7	<a href="#">confirm_resize</a>	28

3.14.3.8 detach_volume . . . . .	28
3.14.3.9 finish_resize . . . . .	28
3.14.3.10 finish_revert_resize . . . . .	28
3.14.3.11 get_console_output . . . . .	28
3.14.3.12 get_console_topic . . . . .	28
3.14.3.13 get_diagnostics . . . . .	28
3.14.3.14 get_host_uptime . . . . .	29
3.14.3.15 get_vnc_console . . . . .	29
3.14.3.16 host_maintenance_mode . . . . .	29
3.14.3.17 host_power_action . . . . .	29
3.14.3.18 init_host . . . . .	29
3.14.3.19 inject_file . . . . .	29
3.14.3.20 inject_network_info . . . . .	29
3.14.3.21 live_migration . . . . .	29
3.14.3.22 pause_instance . . . . .	29
3.14.3.23 post_live_migration_at_destination . . . . .	30
3.14.3.24 power_off_instance . . . . .	30
3.14.3.25 power_on_instance . . . . .	30
3.14.3.26 pre_live_migration . . . . .	30
3.14.3.27 prep_resize . . . . .	30
3.14.3.28 reboot_instance . . . . .	30
3.14.3.29 rebuild_instance . . . . .	30
3.14.3.30 refresh_instance_security_rules . . . . .	31
3.14.3.31 refresh_provider_fw_rules . . . . .	31
3.14.3.32 refresh_security_group_members . . . . .	31
3.14.3.33 refresh_security_group_rules . . . . .	31
3.14.3.34 remove_aggregate_host . . . . .	31
3.14.3.35 remove_fixed_ip_from_instance . . . . .	31
3.14.3.36 remove_volume_connection . . . . .	31
3.14.3.37 rescue_instance . . . . .	31
3.14.3.38 reset_network . . . . .	31
3.14.3.39 resize_instance . . . . .	32
3.14.3.40 resume_instance . . . . .	32
3.14.3.41 revert_resize . . . . .	32
3.14.3.42 rollback_live_migration_at_destination . . . . .	32
3.14.3.43 set_admin_password . . . . .	32
3.14.3.44 set_host_enabled . . . . .	32
3.14.3.45 snapshot_instance . . . . .	32
3.14.3.46 start_instance . . . . .	32
3.14.3.47 stop_instance . . . . .	33

3.14.3.48 suspend_instance . . . . .	33
3.14.3.49 terminate_instance . . . . .	33
3.14.3.50 unpause_instance . . . . .	33
3.14.3.51 unrescue_instance . . . . .	33
3.14.3.52 update_available_resource . . . . .	33
3.15 nova.db.sqlalchemy.models.ComputeNode Class Reference . . . . .	33
3.15.1 Detailed Description . . . . .	34
3.15.2 Member Data Documentation . . . . .	34
3.15.2.1 service . . . . .	34
3.16 nova.db.sqlalchemy.models.ComputeNodeStat Class Reference . . . . .	34
3.16.1 Detailed Description . . . . .	35
3.16.2 Member Data Documentation . . . . .	35
3.16.2.1 primary_join . . . . .	35
3.16.2.2 stats . . . . .	35
3.17 nova.db.sqlalchemy.models.Console Class Reference . . . . .	35
3.17.1 Detailed Description . . . . .	36
3.18 nova.db.sqlalchemy.models.ConsolePool Class Reference . . . . .	36
3.18.1 Detailed Description . . . . .	36
3.19 nova.db.sqlalchemy.api.Constraint Class Reference . . . . .	37
3.20 nova.api.openstack.compute.servers.Controller Class Reference . . . . .	37
3.20.1 Detailed Description . . . . .	38
3.20.2 Member Function Documentation . . . . .	38
3.20.2.1 create . . . . .	38
3.20.2.2 delete . . . . .	38
3.20.2.3 detail . . . . .	38
3.20.2.4 index . . . . .	38
3.20.2.5 show . . . . .	38
3.20.2.6 update . . . . .	38
3.20.3 Member Data Documentation . . . . .	38
3.20.3.1 B64_REGEX . . . . .	38
3.21 nova.api.openstack.compute.servers.CreateDeserializer Class Reference . . . . .	39
3.21.1 Detailed Description . . . . .	39
3.21.2 Member Function Documentation . . . . .	39
3.21.2.1 default . . . . .	39
3.22 nova.db.sqlalchemy.models.DNSDomain Class Reference . . . . .	39
3.22.1 Detailed Description . . . . .	40
3.23 nova.db.sqlalchemy.api.EqualityCondition Class Reference . . . . .	40
3.24 nova.db.sqlalchemy.models.FixedIp Class Reference . . . . .	40
3.24.1 Detailed Description . . . . .	41
3.25 nova.db.sqlalchemy.models.FloatingIp Class Reference . . . . .	41

3.25.1 Detailed Description . . . . .	41
3.26 nova.compute.api.HostAPI Class Reference . . . . .	42
3.26.1 Member Function Documentation . . . . .	42
3.26.1.1 get_host_uptime . . . . .	42
3.26.1.2 host_power_action . . . . .	42
3.26.1.3 set_host_enabled . . . . .	42
3.26.1.4 set_host_maintenance . . . . .	42
3.27 nova.api.openstack.compute.contrib.hostsecurityproperties.Hostsecurityproperties Class Reference	43
3.27.1 Detailed Description . . . . .	43
3.27.2 Member Data Documentation . . . . .	43
3.27.2.1 namespace . . . . .	43
3.28 nova.api.openstack.compute.contrib.hostsecurityproperties.HostSecurityPropertiesController Class Reference . . . . .	43
3.28.1 Detailed Description . . . . .	44
3.28.2 Member Function Documentation . . . . .	44
3.28.2.1 delete . . . . .	44
3.28.2.2 index . . . . .	44
3.28.2.3 show . . . . .	44
3.29 nova.db.sqlalchemy.api.InequalityCondition Class Reference . . . . .	44
3.30 nova.db.sqlalchemy.models.Instance Class Reference . . . . .	45
3.30.1 Detailed Description . . . . .	46
3.31 nova.db.sqlalchemy.models.InstanceFault Class Reference . . . . .	46
3.31.1 Member Data Documentation . . . . .	47
3.31.1.1 instance_uuid . . . . .	47
3.32 nova.db.sqlalchemy.models.InstanceIdMapping Class Reference . . . . .	47
3.32.1 Detailed Description . . . . .	47
3.33 nova.db.sqlalchemy.models.InstanceInfoCache Class Reference . . . . .	47
3.33.1 Detailed Description . . . . .	48
3.33.2 Member Data Documentation . . . . .	48
3.33.2.1 instance . . . . .	48
3.33.2.2 instance_uuid . . . . .	48
3.34 nova.db.sqlalchemy.models.InstanceMetadata Class Reference . . . . .	48
3.34.1 Detailed Description . . . . .	49
3.34.2 Member Data Documentation . . . . .	49
3.34.2.1 instance . . . . .	49
3.34.2.2 instance_uuid . . . . .	49
3.35 nova.db.sqlalchemy.models.InstanceSystemMetadata Class Reference . . . . .	49
3.35.1 Detailed Description . . . . .	50
3.35.2 Member Data Documentation . . . . .	50
3.35.2.1 instance . . . . .	50



3.35.2.2	instance_uuid	50
3.35.2.3	primary_join	50
3.36	nova.db.sqlalchemy.models.InstanceTypeExtraSpecs Class Reference	51
3.36.1	Detailed Description	51
3.36.2	Member Data Documentation	51
3.36.2.1	instance_type	51
3.36.2.2	instance_type_id	51
3.37	nova.db.sqlalchemy.models.InstanceTypeProjects Class Reference	51
3.37.1	Detailed Description	52
3.37.2	Member Data Documentation	52
3.37.2.1	instance_type	52
3.37.2.2	instance_type_id	52
3.38	nova.db.sqlalchemy.models.InstanceTypes Class Reference	52
3.38.1	Detailed Description	53
3.38.2	Member Data Documentation	53
3.38.2.1	instances	53
3.39	nova.db.sqlalchemy.models.IscsiTarget Class Reference	53
3.39.1	Detailed Description	54
3.39.2	Member Data Documentation	54
3.39.2.1	volume	54
3.40	nova.db.sqlalchemy.models.KeyPair Class Reference	54
3.40.1	Detailed Description	55
3.41	nova.compute.api.KeypairAPI Class Reference	55
3.41.1	Detailed Description	55
3.41.2	Member Function Documentation	55
3.41.2.1	create_key_pair	55
3.41.2.2	delete_key_pair	55
3.41.2.3	get_key_pair	56
3.41.2.4	get_key_pairs	56
3.41.2.5	import_key_pair	56
3.42	nova.db.sqlalchemy.models.Migration Class Reference	56
3.42.1	Detailed Description	56
3.42.2	Member Data Documentation	57
3.42.2.1	instance_uuid	57
3.43	nova.api.openstack.compute.servers.MinimalServersTemplate Class Reference	57
3.44	nova.db.sqlalchemy.models.Network Class Reference	57
3.44.1	Detailed Description	58
3.45	nova.db.api.NoMoreNetworks Class Reference	58
3.45.1	Detailed Description	58
3.46	nova.db.api.NoMoreTargets Class Reference	59

3.46.1 Detailed Description . . . . .	59
3.47 nova.db.sqlalchemy.models.NovaBase Class Reference . . . . .	59
3.47.1 Detailed Description . . . . .	60
3.47.2 Member Function Documentation . . . . .	60
3.47.2.1 delete . . . . .	60
3.47.2.2 iteritems . . . . .	60
3.47.2.3 save . . . . .	60
3.47.2.4 update . . . . .	60
3.48 nova.db.sqlalchemy.models.ProviderFirewallRule Class Reference . . . . .	61
3.48.1 Detailed Description . . . . .	61
3.49 nova.db.sqlalchemy.models.Quota Class Reference . . . . .	61
3.49.1 Detailed Description . . . . .	62
3.50 nova.db.sqlalchemy.models.QuotaClass Class Reference . . . . .	62
3.50.1 Detailed Description . . . . .	62
3.51 nova.db.sqlalchemy.models.QuotaUsage Class Reference . . . . .	62
3.51.1 Detailed Description . . . . .	63
3.52 nova.db.sqlalchemy.models.Requirement Class Reference . . . . .	63
3.52.1 Detailed Description . . . . .	64
3.53 nova.api.openstack.compute.contrib.requirements.Requirements Class Reference . . . . .	64
3.53.1 Detailed Description . . . . .	64
3.54 nova.api.openstack.compute.contrib.requirements.RequirementsController Class Reference . . . . .	64
3.54.1 Detailed Description . . . . .	65
3.54.2 Member Function Documentation . . . . .	65
3.54.2.1 delete . . . . .	65
3.54.2.2 index . . . . .	65
3.54.2.3 show . . . . .	65
3.55 nova.api.openstack.compute.contrib.requirements.RequirementsTemplate Class Reference . . . . .	65
3.56 nova.api.openstack.compute.contrib.requirements.RequirementTemplate Class Reference . . . . .	66
3.57 nova.db.sqlalchemy.models.Reservation Class Reference . . . . .	66
3.57.1 Detailed Description . . . . .	66
3.57.2 Member Data Documentation . . . . .	67
3.57.2.1 usage . . . . .	67
3.58 nova.db.sqlalchemy.models.S3Image Class Reference . . . . .	67
3.58.1 Detailed Description . . . . .	67
3.59 nova.scheduler.manager_integrity.SchedulerManager Class Reference . . . . .	67
3.59.1 Detailed Description . . . . .	68
3.60 nova.db.sqlalchemy.models.SecurityGroup Class Reference . . . . .	68
3.60.1 Detailed Description . . . . .	68
3.60.2 Member Data Documentation . . . . .	68
3.60.2.1 instances . . . . .	68

3.61 nova.compute.rpcapi.SecurityGroupAPI Class Reference	69
3.61.1 Detailed Description	69
3.62 nova.compute.api.SecurityGroupAPI Class Reference	69
3.62.1 Detailed Description	70
3.62.2 Member Function Documentation	70
3.62.2.1 add_to_instance	70
3.62.2.2 ensure_default	71
3.62.2.3 is_associated_with_server	71
3.62.2.4 remove_from_instance	71
3.62.2.5 rule_exists	71
3.62.2.6 trigger_members_refresh	71
3.62.2.7 trigger_rules_refresh	71
3.62.2.8 validate_property	71
3.63 nova.db.sqlalchemy.models.SecurityGroupIngressRule Class Reference	71
3.63.1 Detailed Description	72
3.63.2 Member Data Documentation	72
3.63.2.1 grantee_group	72
3.63.2.2 parent_group	72
3.64 nova.db.sqlalchemy.models.SecurityGroupInstanceAssociation Class Reference	73
3.65 nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityPropertiesTemplate Class Reference	73
3.66 nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityPropertyTemplate Class Reference	73
3.67 nova.api.openstack.compute.servers.ServerAdminPassTemplate Class Reference	74
3.68 nova.api.openstack.compute.servers.ServersTemplate Class Reference	74
3.69 nova.api.openstack.compute.servers.ServerTemplate Class Reference	75
3.70 nova.db.sqlalchemy.models.Service Class Reference	75
3.70.1 Detailed Description	75
3.71 nova.db.sqlalchemy.models.SMBBackendConf Class Reference	76
3.71.1 Detailed Description	76
3.72 nova.db.sqlalchemy.models.SMFlavors Class Reference	76
3.72.1 Detailed Description	77
3.73 nova.db.sqlalchemy.models.SMVolume Class Reference	77
3.73.1 Member Data Documentation	77
3.73.1.1 backend_id	77
3.74 nova.db.sqlalchemy.models.Snapshot Class Reference	77
3.74.1 Detailed Description	78
3.75 nova.db.sqlalchemy.models.SnapshotIdMapping Class Reference	78
3.75.1 Detailed Description	79
3.76 nova.db.sqlalchemy.models.TaskLog Class Reference	79

3.76.1 Detailed Description . . . . .	79
3.77 nova.db.sqlalchemy.models.VirtualInterface Class Reference . . . . .	79
3.77.1 Detailed Description . . . . .	80
3.78 nova.db.sqlalchemy.models.Volume Class Reference . . . . .	80
3.78.1 Detailed Description . . . . .	81
3.79 nova.db.sqlalchemy.models.VolumeldMapping Class Reference . . . . .	81
3.79.1 Detailed Description . . . . .	81
3.80 nova.db.sqlalchemy.models.VolumeMetadata Class Reference . . . . .	82
3.80.1 Detailed Description . . . . .	82
3.80.2 Member Data Documentation . . . . .	82
3.80.2.1 volume . . . . .	82
3.81 nova.db.sqlalchemy.models.VolumeTypeExtraSpecs Class Reference . . . . .	82
3.81.1 Detailed Description . . . . .	83
3.81.2 Member Data Documentation . . . . .	83
3.81.2.1 volume_type . . . . .	83
3.81.2.2 volume_type_id . . . . .	83
3.82 nova.db.sqlalchemy.models.VolumeTypes Class Reference . . . . .	83
3.82.1 Detailed Description . . . . .	84
3.82.2 Member Data Documentation . . . . .	84
3.82.2.1 volumes . . . . .	84
3.83 nova.db.sqlalchemy.models.WhiteLists Class Reference . . . . .	84
3.83.1 Detailed Description . . . . .	84

## Index

84

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

#### BASE

nova.db.sqlalchemy.models.AgentBuild . . . . .	8
nova.db.sqlalchemy.models.Aggregate . . . . .	9
nova.db.sqlalchemy.models.AggregateHost . . . . .	11
nova.db.sqlalchemy.models.AggregateMetadata . . . . .	11
nova.db.sqlalchemy.models.BandwidthUsage . . . . .	19
nova.db.sqlalchemy.models.BlockDeviceMapping . . . . .	20
nova.db.sqlalchemy.models.Certificate . . . . .	21
nova.db.sqlalchemy.models.ComputeNode . . . . .	33
nova.db.sqlalchemy.models.ComputeNodeStat . . . . .	34
nova.db.sqlalchemy.models.Console . . . . .	35
nova.db.sqlalchemy.models.ConsolePool . . . . .	36
nova.db.sqlalchemy.models.DNSDomain . . . . .	39
nova.db.sqlalchemy.models.FixedIp . . . . .	40
nova.db.sqlalchemy.models.FloatingIp . . . . .	41
nova.db.sqlalchemy.models.Instance . . . . .	45
nova.db.sqlalchemy.models.InstanceFault . . . . .	46
nova.db.sqlalchemy.models.InstanceIdMapping . . . . .	47
nova.db.sqlalchemy.models.InstanceInfoCache . . . . .	47
nova.db.sqlalchemy.models.InstanceMetadata . . . . .	48
nova.db.sqlalchemy.models.InstanceSystemMetadata . . . . .	49
nova.db.sqlalchemy.models.InstanceTypeExtraSpecs . . . . .	51
nova.db.sqlalchemy.models.InstanceTypeProjects . . . . .	51
nova.db.sqlalchemy.models.InstanceTypes . . . . .	52
nova.db.sqlalchemy.models.IscsiTarget . . . . .	53
nova.db.sqlalchemy.models.KeyPair . . . . .	54
nova.db.sqlalchemy.models.Migration . . . . .	56
nova.db.sqlalchemy.models.Network . . . . .	57
nova.db.sqlalchemy.models.ProviderFirewallRule . . . . .	61
nova.db.sqlalchemy.models.Quota . . . . .	61
nova.db.sqlalchemy.models.QuotaClass . . . . .	62
nova.db.sqlalchemy.models.QuotaUsage . . . . .	62
nova.db.sqlalchemy.models.Requirement . . . . .	63
nova.db.sqlalchemy.models.Reservation . . . . .	66
nova.db.sqlalchemy.models.S3Image . . . . .	67
nova.db.sqlalchemy.models.SecurityGroup . . . . .	68
nova.db.sqlalchemy.models.SecurityGroupIngressRule . . . . .	71
nova.db.sqlalchemy.models.SecurityGroupInstanceAssociation . . . . .	73

nova.db.sqlalchemy.models.Service . . . . .	75
nova.db.sqlalchemy.models.SMBBackendConf . . . . .	76
nova.db.sqlalchemy.models.SMFlavors . . . . .	76
nova.db.sqlalchemy.models.SMVolume . . . . .	77
nova.db.sqlalchemy.models.Snapshot . . . . .	77
nova.db.sqlalchemy.models.SnapshotIdMapping . . . . .	78
nova.db.sqlalchemy.models.TaskLog . . . . .	79
nova.db.sqlalchemy.models.VirtualInterface . . . . .	79
nova.db.sqlalchemy.models.Volume . . . . .	80
nova.db.sqlalchemy.models.VolumeIdMapping . . . . .	81
nova.db.sqlalchemy.models.VolumeMetadata . . . . .	82
nova.db.sqlalchemy.models.VolumeTypeExtraSpecs . . . . .	82
nova.db.sqlalchemy.models.VolumeTypes . . . . .	83
nova.db.sqlalchemy.models.WhiteLists . . . . .	84
Base	
nova.compute.api.AggregateAPI . . . . .	9
nova.compute.api.API . . . . .	12
nova.compute.api.HostAPI . . . . .	42
nova.compute.api.KeypairAPI . . . . .	55
nova.compute.api.SecurityGroupAPI . . . . .	69
BaseHostFilter	
nova.scheduler.filters.acaas_filter.ACaaSFilter . . . . .	7
Controller	
nova.api.openstack.compute.servers.Controller . . . . .	37
ExtensionDescriptor	
nova.api.openstack.compute.contrib.hostsecurityproperties.Hostsecurityproperties . . . . .	43
nova.api.openstack.compute.contrib.requirements.Requirements . . . . .	64
MetadataXMLDeserializer	
nova.api.openstack.compute.servers.CommonDeserializer . . . . .	21
nova.api.openstack.compute.servers.ActionDeserializer . . . . .	7
nova.api.openstack.compute.servers.CreateDeserializer . . . . .	39
NovaException	
nova.db.api.NoMoreNetworks . . . . .	58
nova.db.api.NoMoreTargets . . . . .	59
object	
nova.api.openstack.compute.contrib.hostsecurityproperties.HostSecurityPropertiesController . . . . .	43
nova.api.openstack.compute.contrib.requirements.RequirementsController . . . . .	64
nova.db.sqlalchemy.api.Constraint . . . . .	37
nova.db.sqlalchemy.api.EqualityCondition . . . . .	40
nova.db.sqlalchemy.api.InequalityCondition . . . . .	44
nova.db.sqlalchemy.models.NovaBase . . . . .	59
nova.db.sqlalchemy.models.AgentBuild . . . . .	8
nova.db.sqlalchemy.models.Aggregate . . . . .	9
nova.db.sqlalchemy.models.AggregateHost . . . . .	11
nova.db.sqlalchemy.models.AggregateMetadata . . . . .	11
nova.db.sqlalchemy.models.BandwidthUsage . . . . .	19
nova.db.sqlalchemy.models.BlockDeviceMapping . . . . .	20
nova.db.sqlalchemy.models.Certificate . . . . .	21
nova.db.sqlalchemy.models.ComputeNode . . . . .	33
nova.db.sqlalchemy.models.ComputeNodeStat . . . . .	34
nova.db.sqlalchemy.models.Console . . . . .	35
nova.db.sqlalchemy.models.ConsolePool . . . . .	36
nova.db.sqlalchemy.models.DNSDomain . . . . .	39
nova.db.sqlalchemy.models.FixedIp . . . . .	40
nova.db.sqlalchemy.models.FloatingIp . . . . .	41
nova.db.sqlalchemy.models.Instance . . . . .	45
nova.db.sqlalchemy.models.InstanceFault . . . . .	46
nova.db.sqlalchemy.models.InstanceIdMapping . . . . .	47

nova.db.sqlalchemy.models.InstanceInfoCache . . . . .	47
nova.db.sqlalchemy.models.InstanceMetadata . . . . .	48
nova.db.sqlalchemy.models.InstanceSystemMetadata . . . . .	49
nova.db.sqlalchemy.models.InstanceTypeExtraSpecs . . . . .	51
nova.db.sqlalchemy.models.InstanceTypeProjects . . . . .	51
nova.db.sqlalchemy.models.InstanceTypes . . . . .	52
nova.db.sqlalchemy.models.IscsiTarget . . . . .	53
nova.db.sqlalchemy.models.KeyPair . . . . .	54
nova.db.sqlalchemy.models.Migration . . . . .	56
nova.db.sqlalchemy.models.Network . . . . .	57
nova.db.sqlalchemy.models.ProviderFirewallRule . . . . .	61
nova.db.sqlalchemy.models.Quota . . . . .	61
nova.db.sqlalchemy.models.QuotaClass . . . . .	62
nova.db.sqlalchemy.models.QuotaUsage . . . . .	62
nova.db.sqlalchemy.models.Requirement . . . . .	63
nova.db.sqlalchemy.models.Reservation . . . . .	66
nova.db.sqlalchemy.models.S3Image . . . . .	67
nova.db.sqlalchemy.models.SecurityGroup . . . . .	68
nova.db.sqlalchemy.models.SecurityGroupIngressRule . . . . .	71
nova.db.sqlalchemy.models.SecurityGroupInstanceAssociation . . . . .	73
nova.db.sqlalchemy.models.Service . . . . .	75
nova.db.sqlalchemy.models.SMBackendConf . . . . .	76
nova.db.sqlalchemy.models.SMFlavors . . . . .	76
nova.db.sqlalchemy.models.SMVolume . . . . .	77
nova.db.sqlalchemy.models.Snapshot . . . . .	77
nova.db.sqlalchemy.models.SnapshotIdMapping . . . . .	78
nova.db.sqlalchemy.models.TaskLog . . . . .	79
nova.db.sqlalchemy.models.VirtualInterface . . . . .	79
nova.db.sqlalchemy.models.Volume . . . . .	80
nova.db.sqlalchemy.models.VolumeldMapping . . . . .	81
nova.db.sqlalchemy.models.VolumeMetadata . . . . .	82
nova.db.sqlalchemy.models.VolumeTypeExtraSpecs . . . . .	82
nova.db.sqlalchemy.models.VolumeTypes . . . . .	83
nova.db.sqlalchemy.models.WhiteLists . . . . .	84
RpcProxy	
nova.compute.rpcapi.ComputeAPI . . . . .	22
nova.compute.rpcapi.SecurityGroupAPI . . . . .	69
SchedulerDependentManager	
nova.compute.manager.ComputeManager . . . . .	25
SchedulerManager	
nova.scheduler.manager_integrity.SchedulerManager . . . . .	67
TemplateBuilder	
nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityPropertiesTemplate . . . . .	73
nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityPropertyTemplate . . . . .	73
nova.api.openstack.compute.contrib.requirements.RequirementsTemplate . . . . .	65
nova.api.openstack.compute.contrib.requirements.RequirementTemplate . . . . .	66
nova.api.openstack.compute.servers.MinimalServersTemplate . . . . .	57
nova.api.openstack.compute.servers.ServerAdminPassTemplate . . . . .	74
nova.api.openstack.compute.servers.ServersTemplate . . . . .	74
nova.api.openstack.compute.servers.ServerTemplate . . . . .	75





## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">nova.scheduler.filters.acaas_filter.ACaaSFilter</a>	7
<a href="#">nova.api.openstack.compute.servers.ActionDeserializer</a>	7
<a href="#">nova.db.sqlalchemy.models.AgentBuild</a>	8
<a href="#">nova.db.sqlalchemy.models.Aggregate</a>	9
<a href="#">nova.compute.api.AggregateAPI</a>	9
<a href="#">nova.db.sqlalchemy.models.AggregateHost</a>	11
<a href="#">nova.db.sqlalchemy.models.AggregateMetadata</a>	11
<a href="#">nova.compute.api.API</a>	12
<a href="#">nova.db.sqlalchemy.models.BandwidthUsage</a>	19
<a href="#">nova.db.sqlalchemy.models.BlockDeviceMapping</a>	20
<a href="#">nova.db.sqlalchemy.models.Certificate</a>	21
<a href="#">nova.api.openstack.compute.servers.CommonDeserializer</a>	21
<a href="#">nova.compute.rpcapi.ComputeAPI</a>	22
<a href="#">nova.compute.manager.ComputeManager</a>	25
<a href="#">nova.db.sqlalchemy.models.ComputeNode</a>	33
<a href="#">nova.db.sqlalchemy.models.ComputeNodeStat</a>	34
<a href="#">nova.db.sqlalchemy.models.Console</a>	35
<a href="#">nova.db.sqlalchemy.models.ConsolePool</a>	36
<a href="#">nova.db.sqlalchemy.api.Constraint</a>	37
<a href="#">nova.api.openstack.compute.servers.Controller</a>	37
<a href="#">nova.api.openstack.compute.servers.CreateDeserializer</a>	39
<a href="#">nova.db.sqlalchemy.models.DNSDomain</a>	39
<a href="#">nova.db.sqlalchemy.api.EqualityCondition</a>	40
<a href="#">nova.db.sqlalchemy.models.FixedIp</a>	40
<a href="#">nova.db.sqlalchemy.models.FloatingIp</a>	41
<a href="#">nova.compute.api.HostAPI</a>	42
<a href="#">nova.api.openstack.compute.contrib.hostsecurityproperties.Hostsecurityproperties</a>	43
<a href="#">nova.api.openstack.compute.contrib.hostsecurityproperties.HostSecurityPropertiesController</a>	43
<a href="#">nova.db.sqlalchemy.api.InequalityCondition</a>	44
<a href="#">nova.db.sqlalchemy.models.Instance</a>	45
<a href="#">nova.db.sqlalchemy.models.InstanceFault</a>	46
<a href="#">nova.db.sqlalchemy.models.InstanceIdMapping</a>	47
<a href="#">nova.db.sqlalchemy.models.InstanceInfoCache</a>	47
<a href="#">nova.db.sqlalchemy.models.InstanceMetadata</a>	48
<a href="#">nova.db.sqlalchemy.models.InstanceSystemMetadata</a>	49
<a href="#">nova.db.sqlalchemy.models.InstanceTypeExtraSpecs</a>	51
<a href="#">nova.db.sqlalchemy.models.InstanceTypeProjects</a>	51
<a href="#">nova.db.sqlalchemy.models.InstanceTypes</a>	52

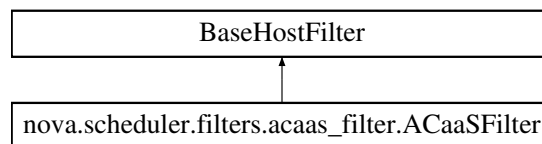
<a href="#">nova.db.sqlalchemy.models.IscsiTarget</a>	53
<a href="#">nova.db.sqlalchemy.models.KeyPair</a>	54
<a href="#">nova.compute.api.KeypairAPI</a>	55
<a href="#">nova.db.sqlalchemy.models.Migration</a>	56
<a href="#">nova.api.openstack.compute.servers.MinimalServersTemplate</a>	57
<a href="#">nova.db.sqlalchemy.models.Network</a>	57
<a href="#">nova.db.api.NoMoreNetworks</a>	58
<a href="#">nova.db.api.NoMoreTargets</a>	59
<a href="#">nova.db.sqlalchemy.models.NovaBase</a>	59
<a href="#">nova.db.sqlalchemy.models.ProviderFirewallRule</a>	61
<a href="#">nova.db.sqlalchemy.models.Quota</a>	61
<a href="#">nova.db.sqlalchemy.models.QuotaClass</a>	62
<a href="#">nova.db.sqlalchemy.models.QuotaUsage</a>	62
<a href="#">nova.db.sqlalchemy.models.Requirement</a>	63
<a href="#">nova.api.openstack.compute.contrib.requirements.Requirements</a>	64
<a href="#">nova.api.openstack.compute.contrib.requirements.RequirementsController</a>	64
<a href="#">nova.api.openstack.compute.contrib.requirements.RequirementsTemplate</a>	65
<a href="#">nova.api.openstack.compute.contrib.requirements.RequirementTemplate</a>	66
<a href="#">nova.db.sqlalchemy.models.Reservation</a>	66
<a href="#">nova.db.sqlalchemy.models.S3Image</a>	67
<a href="#">nova.scheduler.manager_integrity.SchedulerManager</a>	67
<a href="#">nova.db.sqlalchemy.models.SecurityGroup</a>	68
<a href="#">nova.compute.rpcapi.SecurityGroupAPI</a>	69
<a href="#">nova.compute.api.SecurityGroupAPI</a>	69
<a href="#">nova.db.sqlalchemy.models.SecurityGroupIngressRule</a>	71
<a href="#">nova.db.sqlalchemy.models.SecurityGroupInstanceAssociation</a>	73
<a href="#">nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityPropertiesTemplate</a>	73
<a href="#">nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityPropertyTemplate</a>	73
<a href="#">nova.api.openstack.compute.servers.ServerAdminPassTemplate</a>	74
<a href="#">nova.api.openstack.compute.servers.ServersTemplate</a>	74
<a href="#">nova.api.openstack.compute.servers.ServerTemplate</a>	75
<a href="#">nova.db.sqlalchemy.models.Service</a>	75
<a href="#">nova.db.sqlalchemy.models.SMBBackendConf</a>	76
<a href="#">nova.db.sqlalchemy.models.SMFlavors</a>	76
<a href="#">nova.db.sqlalchemy.models.SMVolume</a>	77
<a href="#">nova.db.sqlalchemy.models.Snapshot</a>	77
<a href="#">nova.db.sqlalchemy.models.SnapshotIdMapping</a>	78
<a href="#">nova.db.sqlalchemy.models.TaskLog</a>	79
<a href="#">nova.db.sqlalchemy.models.VirtualInterface</a>	79
<a href="#">nova.db.sqlalchemy.models.Volume</a>	80
<a href="#">nova.db.sqlalchemy.models.VolumeldMapping</a>	81
<a href="#">nova.db.sqlalchemy.models.VolumeMetadata</a>	82
<a href="#">nova.db.sqlalchemy.models.VolumeTypeExtraSpecs</a>	82
<a href="#">nova.db.sqlalchemy.models.VolumeTypes</a>	83
<a href="#">nova.db.sqlalchemy.models.WhiteLists</a>	84

## Chapter 3

# Class Documentation

### 3.1 nova.scheduler.filters.acaas\_filter.ACaaSFilter Class Reference

Inheritance diagram for nova.scheduler.filters.acaas\_filter.ACaaSFilter:



#### Public Member Functions

- def `host_passes`

#### 3.1.1 Detailed Description

Filter to support Access Control as a Service scheduling

#### 3.1.2 Member Function Documentation

##### 3.1.2.1 def nova.scheduler.filters.acaas\_filter.ACaaSFilter.host\_passes ( self, host\_state, filter\_properties )

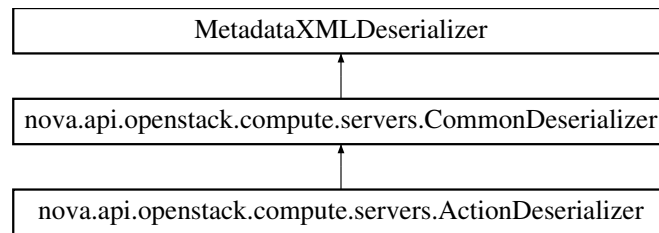
Filter the full host list (from the ZoneManager)

The documentation for this class was generated from the following file:

- nova/scheduler/filters/acaas\_filter.py

### 3.2 nova.api.openstack.compute.servers.ActionDeserializer Class Reference

Inheritance diagram for nova.api.openstack.compute.servers.ActionDeserializer:



## Public Member Functions

- def **default**

## Additional Inherited Members

### 3.2.1 Detailed Description

Deserializer to handle xml-formatted server action requests.

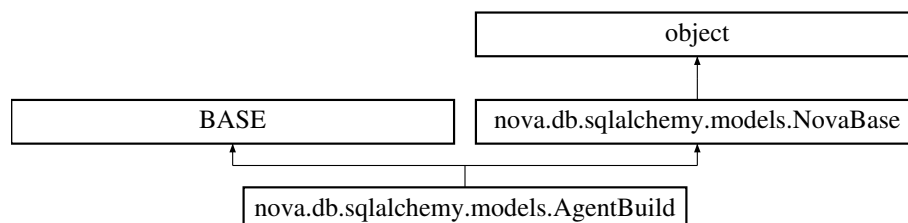
Handles standard server attributes as well as optional metadata and personality attributes

The documentation for this class was generated from the following file:

- `nova/api/openstack/compute/servers.py`

## 3.3 nova.db.sqlalchemy.models.AgentBuild Class Reference

Inheritance diagram for `nova.db.sqlalchemy.models.AgentBuild`:



## Static Public Attributes

- tuple **id** = `Column(Integer, primary_key=True)`
- tuple **hypervisor** = `Column(String(255))`
- tuple **os** = `Column(String(255))`
- tuple **architecture** = `Column(String(255))`
- tuple **version** = `Column(String(255))`
- tuple **url** = `Column(String(255))`
- tuple **md5hash** = `Column(String(255))`

## Additional Inherited Members

### 3.3.1 Detailed Description

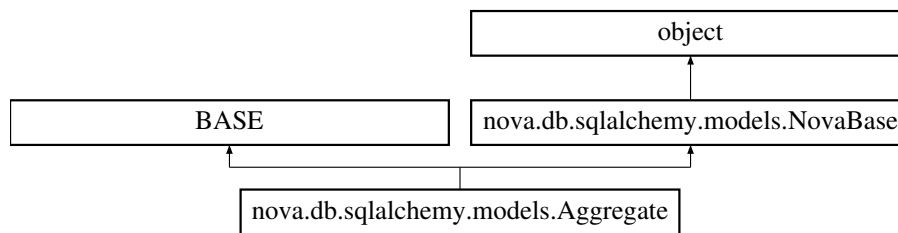
Represents an agent build.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.4 nova.db.sqlalchemy.models.Aggregate Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Aggregate:



### Public Member Functions

- def **hosts**
- def **metadetails**

### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, autoincrement=True)
- tuple **name** = Column(String(255))
- tuple **availability\_zone** = Column(String(255), nullable=False)

### Additional Inherited Members

#### 3.4.1 Detailed Description

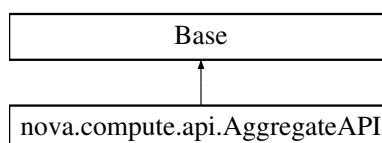
Represents a cluster of hosts that exists in this zone.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.5 nova.compute.api.AggregateAPI Class Reference

Inheritance diagram for nova.compute.api.AggregateAPI:



## Public Member Functions

- def `__init__`
- def `create_aggregate`
- def `get_aggregate`
- def `get_aggregate_list`
- def `update_aggregate`
- def `update_aggregate_metadata`
- def `delete_aggregate`
- def `add_host_to_aggregate`
- def `remove_host_from_aggregate`

## Public Attributes

- `compute_rpcapi`

### 3.5.1 Detailed Description

Sub-set of the Compute Manager API for managing host aggregates.

### 3.5.2 Member Function Documentation

**3.5.2.1** `def nova.compute.api.AggregateAPI.add_host_to_aggregate ( self, context, aggregate_id, host )`

Adds the host to an aggregate.

**3.5.2.2** `def nova.compute.api.AggregateAPI.create_aggregate ( self, context, aggregate_name, availability_zone )`

Creates the model for the aggregate.

**3.5.2.3** `def nova.compute.api.AggregateAPI.delete_aggregate ( self, context, aggregate_id )`

Deletes the aggregate.

**3.5.2.4** `def nova.compute.api.AggregateAPI.get_aggregate ( self, context, aggregate_id )`

Get an aggregate by id.

**3.5.2.5** `def nova.compute.api.AggregateAPI.get_aggregate_list ( self, context )`

Get all the aggregates.

**3.5.2.6** `def nova.compute.api.AggregateAPI.remove_host_from_aggregate ( self, context, aggregate_id, host )`

Removes host from the aggregate.

### 3.5.2.7 `def nova.compute.api.AggregateAPI.update_aggregate( self, context, aggregate_id, values )`

Update the properties of an aggregate.

### 3.5.2.8 `def nova.compute.api.AggregateAPI.update_aggregate_metadata( self, context, aggregate_id, metadata )`

Updates the aggregate metadata.

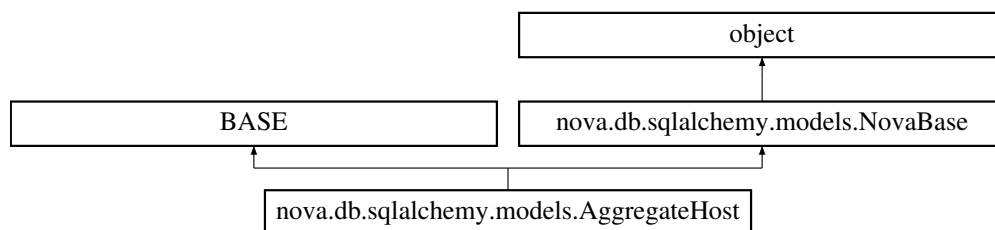
If a key is set to None, it gets removed from the aggregate metadata.

The documentation for this class was generated from the following file:

- nova/compute/api.py

## 3.6 nova.db.sqlalchemy.models.AggregateHost Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.AggregateHost:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, autoincrement=True)
- tuple **host** = Column(String(255), unique=False)
- tuple **aggregate\_id** = Column(Integer, ForeignKey('aggregates.id'), nullable=False)

### Additional Inherited Members

#### 3.6.1 Detailed Description

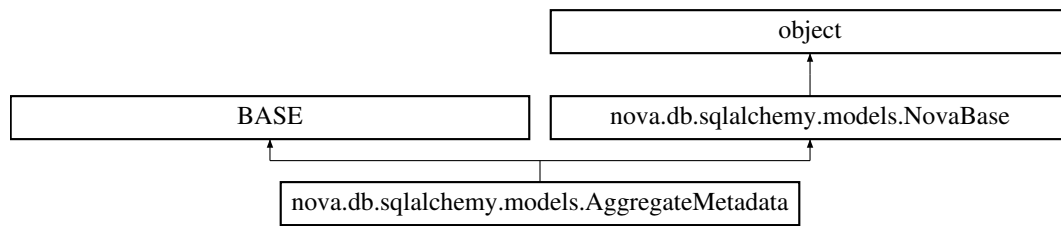
Represents a host that is member of an aggregate.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.7 nova.db.sqlalchemy.models.AggregateMetadata Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.AggregateMetadata:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **key** = Column(String(255), nullable=False)
- tuple **value** = Column(String(255), nullable=False)
- tuple **aggregate\_id** = Column(Integer, ForeignKey('aggregates.id'), nullable=False)

### Additional Inherited Members

#### 3.7.1 Detailed Description

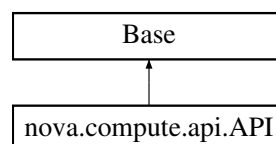
Represents a metadata key/value pair for an aggregate.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.8 nova.compute.api.API Class Reference

Inheritance diagram for nova.compute.api.API:



### Public Member Functions

- def **\_\_init\_\_**
- def [create\\_db\\_entry\\_for\\_new\\_instance](#)
- def [create](#)
- def [trigger\\_provider\\_fw\\_rules\\_refresh](#)
- def [update](#)
- def [soft\\_delete](#)
- def [delete](#)
- def [restore](#)
- def [force\\_delete](#)
- def [stop](#)
- def [start](#)
- def [get\\_active\\_by\\_window](#)
- def [get\\_instance\\_type](#)
- def [get](#)



- def [get\\_all](#)
- def [backup](#)
- def [snapshot](#)
- def [snapshot\\_volume\\_backed](#)
- def [reboot](#)
- def [rebuild](#)
- def [revert\\_resize](#)
- def [confirm\\_resize](#)
- def [resize](#)
- def [add\\_fixed\\_ip](#)
- def [remove\\_fixed\\_ip](#)
- def [pause](#)
- def [unpause](#)
- def [get\\_diagnostics](#)
- def [suspend](#)
- def [resume](#)
- def [rescue](#)
- def [unrescue](#)
- def [set\\_admin\\_password](#)
- def [inject\\_file](#)
- def [get\\_vnc\\_console](#)
- def [get\\_console\\_output](#)
- def [lock](#)
- def [unlock](#)
- def [get\\_lock](#)
- def [reset\\_network](#)
- def [inject\\_network\\_info](#)
- def [attach\\_volume](#)
- def [detach\\_volume](#)
- def [get\\_instance\\_metadata](#)
- def [delete\\_instance\\_metadata](#)
- def [update\\_instance\\_metadata](#)
- def [get\\_instance\\_faults](#)
- def [get\\_instance\\_bdms](#)
- def **is\_volume\_backed\_instance**
- def [live\\_migrate](#)

### Public Attributes

- **image\_service**
- **network\_api**
- **volume\_api**
- **security\_group\_api**
- **sgh**
- **consoleauth\_rpcapi**
- **scheduler\_rpcapi**
- **compute\_rpcapi**

### Static Public Attributes

- list **task\_state** = [None]

### 3.8.1 Detailed Description

API for interacting with the compute manager.

### 3.8.2 Member Function Documentation

#### 3.8.2.1 `def nova.compute.api.API.add_fixed_ip( self, context, instance, network_id )`

Add fixed\_ip from specified network to given instance.

#### 3.8.2.2 `def nova.compute.api.API.attach_volume( self, context, instance, volume_id, device=None )`

Attach an existing volume to an existing instance.

#### 3.8.2.3 `def nova.compute.api.API.backup( self, context, instance, name, backup_type, rotation, extra_properties=None )`

Backup the given instance

```
:param instance: nova.db.sqlalchemy.models.Instance
:param name: name of the backup or snapshot
    name = backup_type # daily backups are called 'daily'
:param rotation: int representing how many backups to keep around;
    None if rotation shouldn't be used (as in the case of snapshots)
:param extra_properties: dict of extra image properties to include
```

#### 3.8.2.4 `def nova.compute.api.API.confirm_resize( self, context, instance )`

Confirms a migration/resize and deletes the 'old' instance.

#### 3.8.2.5 `def nova.compute.api.API.create( self, context, instance_type, image_href, kernel_id=None, ramdisk_id=None, min_count=None, max_count=None, display_name=None, display_description=None, key_name=None, key_data=None, security_group=None, availability_zone=None, user_data=None, metadata=None, injected_files=None, admin_password=None, block_device_mapping=None, access_ip_v4=None, access_ip_v6=None, requested_networks=None, config_drive=None, auto_disk_config=None, scheduler_hints=None, req_id=None )`

Provision instances, sending instance information to the scheduler. The scheduler will determine where the instance(s) go and will handle creating the DB entries.

Returns a tuple of (instances, reservation\_id)

#### 3.8.2.6 `def nova.compute.api.API.create_db_entry_for_new_instance( self, context, instance_type, image, base_options, security_group, block_device_mapping )`

Create an entry in the DB for this new instance, including any related table updates (such as security group, etc).

This is called by the scheduler after a location for the instance has been determined.

**3.8.2.7 def nova.compute.api.API.delete ( self, context, instance )**

Terminate an instance.

**3.8.2.8 def nova.compute.api.API.delete\_instance\_metadata ( self, context, instance, key )**

Delete the given metadata item from an instance.

**3.8.2.9 def nova.compute.api.API.detach\_volume ( self, context, volume\_id )**

Detach a volume from an instance.

**3.8.2.10 def nova.compute.api.API.force\_delete ( self, context, instance )**

Force delete a previously deleted (but not reclaimed) instance.

**3.8.2.11 def nova.compute.api.API.get ( self, context, instance\_id )**

Get a single instance with the given instance\_id.

**3.8.2.12 def nova.compute.api.API.get\_active\_by\_window ( self, context, begin, end=None, project\_id=None )**

Get instances that were continuously active over a window.

**3.8.2.13 def nova.compute.api.API.get\_all ( self, context, search\_opts=None, sort\_key='created\_at', sort\_dir='desc', limit=None, marker=None )**

Get all instances filtered by one of the given parameters.

If there is no filter and the context is an admin, it will retrieve all instances in the system.

Deleted instances will be returned by default, unless there is a search option that says otherwise.

The results will be returned sorted in the order specified by the 'sort\_dir' parameter using the key specified in the 'sort\_key' parameter.

**3.8.2.14 def nova.compute.api.API.get\_console\_output ( self, context, instance, tail\_length=None )**

Get console output for an instance.

**3.8.2.15 def nova.compute.api.API.get\_diagnostics ( self, context, instance )**

Retrieve diagnostics for the given instance.

**3.8.2.16** `def nova.compute.api.API.get_instance_bdm ( self, context, instance )`

Get all bdm tables for specified instance.

**3.8.2.17** `def nova.compute.api.API.get_instance_faults ( self, context, instances )`

Get all faults for a list of instance uuids.

**3.8.2.18** `def nova.compute.api.API.get_instance_metadata ( self, context, instance )`

Get all metadata associated with an instance.

**3.8.2.19** `def nova.compute.api.API.get_instance_type ( self, context, instance_type_id )`

Get an instance type by instance type id.

**3.8.2.20** `def nova.compute.api.API.get_lock ( self, context, instance )`

Return the boolean state of given instance's lock.

**3.8.2.21** `def nova.compute.api.API.get_vnc_console ( self, context, instance, console_type )`

Get a url to an instance Console.

**3.8.2.22** `def nova.compute.api.API.inject_file ( self, context, instance, path, file_contents )`

Write a file to the given instance.

**3.8.2.23** `def nova.compute.api.API.inject_network_info ( self, context, instance )`

Inject network info for the instance.

**3.8.2.24** `def nova.compute.api.API.live_migrate ( self, context, instance, block_migration, disk_over_commit, host )`

Migrate a server lively to a new host.

**3.8.2.25** `def nova.compute.api.API.lock ( self, context, instance )`

Lock the given instance.

**3.8.2.26** `def nova.compute.api.API.pause ( self, context, instance )`

Pause the given instance.

**3.8.2.27** `def nova.compute.api.API.reboot( self, context, instance, reboot_type )`

Reboot the given instance.

**3.8.2.28** `def nova.compute.api.API.rebuild( self, context, instance, image_href, admin_password, kwargs )`

Rebuild the given instance with the provided attributes.

**3.8.2.29** `def nova.compute.api.API.remove_fixed_ip( self, context, instance, address )`

Remove fixed\_ip from specified network to given instance.

**3.8.2.30** `def nova.compute.api.API.rescue( self, context, instance, rescue_password=None )`

Rescue the given instance.

**3.8.2.31** `def nova.compute.api.API.reset_network( self, context, instance )`

Reset networking on the instance.

**3.8.2.32** `def nova.compute.api.API.resize( self, context, instance, flavor_id=None, kwargs )`

Resize (ie, migrate) a running instance.

If flavor\_id is None, the process is considered a migration, keeping the original flavor\_id. If flavor\_id is not None, the instance should be migrated to a new host and resized to the new flavor\_id.

**3.8.2.33** `def nova.compute.api.API.restore( self, context, instance )`

Restore a previously deleted (but not reclaimed) instance.

**3.8.2.34** `def nova.compute.api.API.resume( self, context, instance )`

Resume the given instance.

**3.8.2.35** `def nova.compute.api.API.revert_resize( self, context, instance )`

Reverts a resize, deleting the 'new' instance in the process.

**3.8.2.36** `def nova.compute.api.API.set_admin_password( self, context, instance, password=None )`

Set the root/admin password for the given instance.

**3.8.2.37** `def nova.compute.api.API.snapshot ( self, context, instance, name, extra_properties=None )`

Snapshot the given instance.

```
:param instance: nova.db.sqlalchemy.models.Instance
:param name: name of the backup or snapshot
:param extra_properties: dict of extra image properties to include

:returns: A dict containing image metadata
```

**3.8.2.38** `def nova.compute.api.API.snapshot_volume_backed ( self, context, instance, image_meta, name, extra_properties=None )`

Snapshot the given volume-backed instance.

```
:param instance: nova.db.sqlalchemy.models.Instance
:param image_meta: metadata for the new image
:param name: name of the backup or snapshot
:param extra_properties: dict of extra image properties to include

:returns: the new image metadata
```

**3.8.2.39** `def nova.compute.api.API.soft_delete ( self, context, instance )`

Terminate an instance.

**3.8.2.40** `def nova.compute.api.API.start ( self, context, instance )`

Start an instance.

**3.8.2.41** `def nova.compute.api.API.stop ( self, context, instance, do_cast=True )`

Stop an instance.

**3.8.2.42** `def nova.compute.api.API.suspend ( self, context, instance )`

Suspend the given instance.

**3.8.2.43** `def nova.compute.api.API.trigger_provider_fw_rules_refresh ( self, context )`

Called when a rule is added/removed from a provider firewall

**3.8.2.44** `def nova.compute.api.API.unlock ( self, context, instance )`

Unlock the given instance.

**3.8.2.45** `def nova.compute.api.API.unpause ( self, context, instance )`

Unpause the given instance.

**3.8.2.46** `def nova.compute.api.API.unrescue ( self, context, instance )`

Unrescue the given instance.

**3.8.2.47** `def nova.compute.api.API.update ( self, context, instance, kwargs )`

Updates the instance in the datastore.

```
:param context: The security context
:param instance: The instance to update
:param kwargs: All additional keyword args are treated
               as data fields of the instance to be
               updated

:returns: None
```

**3.8.2.48** `def nova.compute.api.API.update_instance_metadata ( self, context, instance, metadata, delete = False )`

Updates or creates instance metadata.

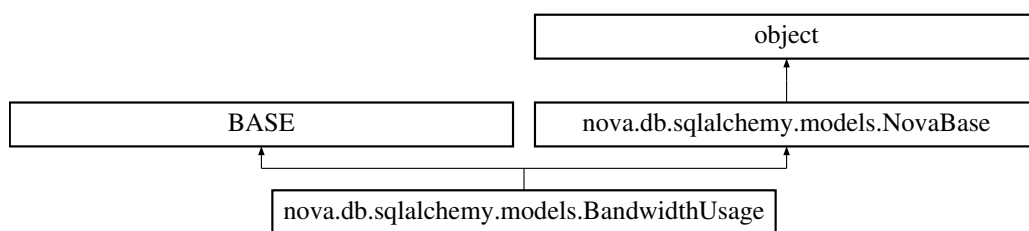
If delete is True, metadata items that are not specified in the 'metadata' argument will be deleted.

The documentation for this class was generated from the following file:

- nova/compute/api.py

**3.9 nova.db.sqlalchemy.models.BandwidthUsage Class Reference**

Inheritance diagram for nova.db.sqlalchemy.models.BandwidthUsage:

**Static Public Attributes**

- tuple **id** = Column(Integer, primary\_key=True, nullable=False)
- tuple **uuid** = Column(String(36), nullable=False)
- tuple **mac** = Column(String(255), nullable=False)
- tuple **start\_period** = Column(DateTime, nullable=False)
- tuple **last\_refreshed** = Column(DateTime)
- tuple **bw\_in** = Column(BigInteger)
- tuple **bw\_out** = Column(BigInteger)

## Additional Inherited Members

### 3.9.1 Detailed Description

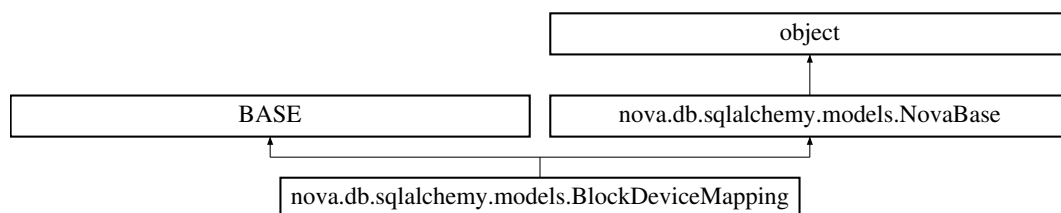
Cache for instance bandwidth usage data pulled from the hypervisor

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.10 nova.db.sqlalchemy.models.BlockDeviceMapping Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.BlockDeviceMapping:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, autoincrement=True)
- tuple **instance\_uuid**
- tuple **instance**
- tuple **device\_name** = Column(String(255), nullable=False)
- tuple **delete\_on\_termination** = Column(Boolean, default=False)
- tuple **virtual\_name** = Column(String(255), nullable=True)
- tuple **snapshot\_id** = Column(String(36))
- tuple **volume\_id** = Column(String(36), nullable=True)
- tuple **volume\_size** = Column(Integer, nullable=True)
- tuple **no\_device** = Column(Boolean, nullable=True)
- tuple **connection\_info** = Column(Text, nullable=True)

## Additional Inherited Members

### 3.10.1 Detailed Description

Represents block device mapping that is defined by EC2

### 3.10.2 Member Data Documentation

#### 3.10.2.1 tuple nova.db.sqlalchemy.models.BlockDeviceMapping.instance [static]

Initial value:

```

1 = relationship(Instance,
2                 backref=backref('block_device_mapping'),
3                 foreign_keys=instance_uuid,
4                 primaryjoin='and_(BlockDeviceMapping.'
5                                 'instance_uuid==\'
6                                 \'Instance.uuid,\''
7                                 \'BlockDeviceMapping.deleted==\'
8                                 \'False)')

```



## 3.10.2.2 tuple nova.db.sqlalchemy.models.BlockDeviceMapping.instance\_uuid [static]

## Initial value:

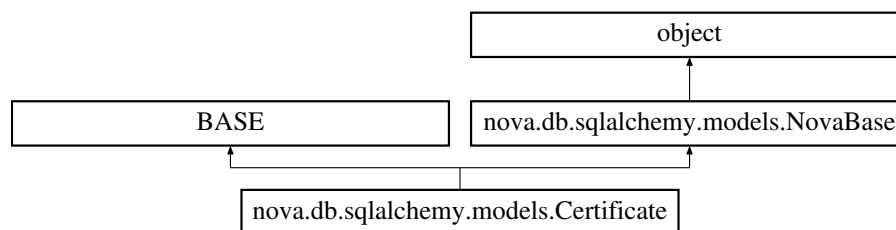
```
1 = Column(Integer, ForeignKey('instances.uuid'),
2         nullable=False)
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.11 nova.db.sqlalchemy.models.Certificate Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Certificate:



## Static Public Attributes

- tuple `id` = `Column(Integer, primary_key=True)`
- tuple `user_id` = `Column(String(255))`
- tuple `project_id` = `Column(String(255))`
- tuple `file_name` = `Column(String(255))`

## Additional Inherited Members

## 3.11.1 Detailed Description

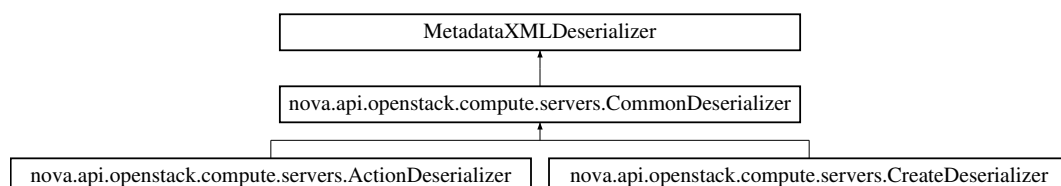
Represents a x509 certificate

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.12 nova.api.openstack.compute.servers.CommonDeserializer Class Reference

Inheritance diagram for nova.api.openstack.compute.servers.CommonDeserializer:



## Static Public Attributes

- tuple **metadata\_deserializer** = common.MetadataXMLDeserializer()

### 3.12.1 Detailed Description

Common deserializer to handle xml-formatted server create requests.

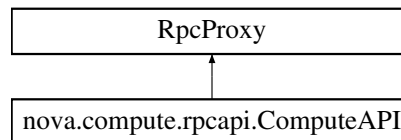
Handles standard server attributes as well as optional metadata and personality attributes

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/servers.py

## 3.13 nova.compute.rpcapi.ComputeAPI Class Reference

Inheritance diagram for nova.compute.rpcapi.ComputeAPI:



## Public Member Functions

- def **\_\_init\_\_**
- def [add\\_aggregate\\_host](#)
- def **update\_security\_properties**
- def **get\_security\_properties**
- def **remove\_security\_properties**
- def **add\_fixed\_ip\_to\_instance**
- def **attach\_volume**
- def **change\_instance\_metadata**
- def **check\_can\_live\_migrate\_destination**
- def **check\_can\_live\_migrate\_source**
- def **confirm\_resize**
- def **detach\_volume**
- def **finish\_resize**
- def **finish\_revert\_resize**
- def **get\_console\_output**
- def **get\_console\_pool\_info**
- def **get\_console\_topic**
- def **get\_diagnostics**
- def **get\_vnc\_console**
- def **validate\_console\_port**
- def [host\\_maintenance\\_mode](#)
- def **host\_power\_action**
- def **inject\_file**
- def **inject\_network\_info**
- def **live\_migration**
- def **pause\_instance**

- def **post\_live\_migration\_at\_destination**
- def **power\_off\_instance**
- def **power\_on\_instance**
- def **pre\_live\_migration**
- def **prep\_resize**
- def **reboot\_instance**
- def **rebuild\_instance**
- def **refresh\_provider\_fw\_rules**
- def [remove\\_aggregate\\_host](#)
- def **remove\_fixed\_ip\_from\_instance**
- def **remove\_volume\_connection**
- def **rescue\_instance**
- def **reset\_network**
- def **resize\_instance**
- def **resume\_instance**
- def **revert\_resize**
- def **rollback\_live\_migration\_at\_destination**
- def **run\_instance**
- def **set\_admin\_password**
- def **set\_host\_enabled**
- def **get\_host\_uptime**
- def **reserve\_block\_device\_name**
- def **snapshot\_instance**
- def **start\_instance**
- def **stop\_instance**
- def **suspend\_instance**
- def **terminate\_instance**
- def **unpause\_instance**
- def **unrescue\_instance**

### Static Public Attributes

- string **BASE\_RPC\_API\_VERSION** = '2.0'

#### 3.13.1 Detailed Description

Client side of the compute rpc API.

API version history:

- 1.0 - Initial version.
- 1.1 - Adds `get_host_uptime()`
- 1.2 - Adds `check_can_live_migrate_[destination|source]`
- 1.3 - Adds `change_instance_metadata()`
- 1.4 - Remove `instance_uuid`, add `instance` argument to `reboot_instance()`
- 1.5 - Remove `instance_uuid`, add `instance` argument to `pause_instance()`, `unpause_instance()`
- 1.6 - Remove `instance_uuid`, add `instance` argument to `suspend_instance()`
- 1.7 - Remove `instance_uuid`, add `instance` argument to `get_console_output()`
- 1.8 - Remove `instance_uuid`, add `instance` argument to `add_fixed_ip_to_instance()`
- 1.9 - Remove `instance_uuid`, add `instance` argument to `attach_volume()`
- 1.10 - Remove `instance_id`, add `instance` argument to `check_can_live_migrate_destination()`
- 1.11 - Remove `instance_id`, add `instance` argument to `check_can_live_migrate_source()`
- 1.12 - Remove `instance_uuid`, add `instance` argument to `confirm_resize()`
- 1.13 - Remove `instance_uuid`, add `instance` argument to `detach_volume()`
- 1.14 - Remove `instance_uuid`, add `instance` argument to `finish_resize()`

- 1.15 - Remove instance\_uuid, add instance argument to finish\_revert\_resize()
- 1.16 - Remove instance\_uuid, add instance argument to get\_diagnostics()
- 1.17 - Remove instance\_uuid, add instance argument to get\_vnc\_console()
- 1.18 - Remove instance\_uuid, add instance argument to inject\_file()
- 1.19 - Remove instance\_uuid, add instance argument to inject\_network\_info()
- 1.20 - Remove instance\_id, add instance argument to post\_live\_migration\_at\_destination()
- 1.21 - Remove instance\_uuid, add instance argument to power\_off\_instance() and stop\_instance()
- 1.22 - Remove instance\_uuid, add instance argument to power\_on\_instance() and start\_instance()
- 1.23 - Remove instance\_id, add instance argument to pre\_live\_migration()
- 1.24 - Remove instance\_uuid, add instance argument to rebuild\_instance()
- 1.25 - Remove instance\_uuid, add instance argument to remove\_fixed\_ip\_from\_instance()
- 1.26 - Remove instance\_id, add instance argument to remove\_volume\_connection()
- 1.27 - Remove instance\_uuid, add instance argument to rescue\_instance()
- 1.28 - Remove instance\_uuid, add instance argument to reset\_network()
- 1.29 - Remove instance\_uuid, add instance argument to resize\_instance()
- 1.30 - Remove instance\_uuid, add instance argument to resume\_instance()
- 1.31 - Remove instance\_uuid, add instance argument to revert\_resize()
- 1.32 - Remove instance\_id, add instance argument to rollback\_live\_migration\_at\_destination()
- 1.33 - Remove instance\_uuid, add instance argument to set\_admin\_password()
- 1.34 - Remove instance\_uuid, add instance argument to snapshot\_instance()
- 1.35 - Remove instance\_uuid, add instance argument to unrescue\_instance()
- 1.36 - Remove instance\_uuid, add instance argument to change\_instance\_metadata()
- 1.37 - Remove instance\_uuid, add instance argument to terminate\_instance()
- 1.38 - Changes to prep\_resize():
  - remove instance\_uuid, add instance
  - remove instance\_type\_id, add instance\_type
  - remove topic, it was unused
- 1.39 - Remove instance\_uuid, add instance argument to run\_instance()
- 1.40 - Remove instance\_id, add instance argument to live\_migration()
- 1.41 - Adds refresh\_instance\_security\_rules()
- 1.42 - Add reservations arg to prep\_resize(), resize\_instance(), finish\_resize(), confirm\_resize(), revert\_resize() and finish\_revert\_resize()
- 1.43 - Add migrate\_data to live\_migration()
- 1.44 - Adds reserve\_block\_device\_name()
  
- 2.0 - Remove 1.x backwards compat
- 2.1 - Adds orig\_sys\_metadata to rebuild\_instance()
- 2.2 - Adds slave\_info parameter to add\_aggregate\_host() and remove\_aggregate\_host()

### 3.13.2 Member Function Documentation

#### 3.13.2.1 `def nova.compute.rpcapi.ComputeAPI.add_aggregate_host( self, ctxt, aggregate_id, host_param, host, slave_info=None )`

Add aggregate host.

```
:param ctxt: request context
:param aggregate_id:
:param host_param: This value is placed in the message to be the 'host'
                  parameter for the remote method.
:param host: This is the host to send the message to.
```

### 3.13.2.2 `def nova.compute.rpcapi.ComputeAPI.host_maintenance_mode( self, ctxt, host_param, mode, host )`

Set host maintenance mode

```
:param ctxt: request context
:param host_param: This value is placed in the message to be the 'host'
                  parameter for the remote method.
:param mode:
:param host: This is the host to send the message to.
```

### 3.13.2.3 `def nova.compute.rpcapi.ComputeAPI.remove_aggregate_host( self, ctxt, aggregate_id, host_param, host, slave_info = None )`

Remove aggregate host.

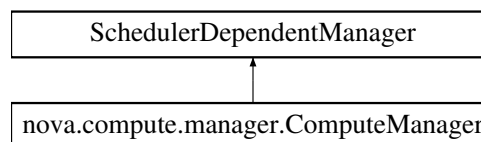
```
:param ctxt: request context
:param aggregate_id:
:param host_param: This value is placed in the message to be the 'host'
                  parameter for the remote method.
:param host: This is the host to send the message to.
```

The documentation for this class was generated from the following file:

- nova/compute/rpcapi.py

## 3.14 nova.compute.manager.ComputeManager Class Reference

Inheritance diagram for nova.compute.manager.ComputeManager:



### Public Member Functions

- `def __init__`
- `def init_host`
- `def get_console_topic`
- `def get_console_pool_info`
- `def refresh_security_group_rules`
- `def refresh_security_group_members`
- `def refresh_instance_security_rules`
- `def refresh_provider_fw_rules`
- `def run_instance`
- `def terminate_instance`
- `def stop_instance`
- `def start_instance`
- `def power_off_instance`
- `def power_on_instance`
- `def rebuild_instance`
- `def reboot_instance`
- `def snapshot_instance`
- `def set_admin_password`

- def [inject\\_file](#)
- def [rescue\\_instance](#)
- def [unrescue\\_instance](#)
- def [change\\_instance\\_metadata](#)
- def [confirm\\_resize](#)
- def [revert\\_resize](#)
- def [finish\\_revert\\_resize](#)
- def [prep\\_resize](#)
- def [resize\\_instance](#)
- def [finish\\_resize](#)
- def [add\\_fixed\\_ip\\_to\\_instance](#)
- def [remove\\_fixed\\_ip\\_from\\_instance](#)
- def [pause\\_instance](#)
- def [unpause\\_instance](#)
- def [host\\_power\\_action](#)
- def [host\\_maintenance\\_mode](#)
- def [set\\_host\\_enabled](#)
- def [get\\_host\\_uptime](#)
- def [get\\_diagnostics](#)
- def [suspend\\_instance](#)
- def [resume\\_instance](#)
- def [reset\\_network](#)
- def [inject\\_network\\_info](#)
- def [get\\_console\\_output](#)
- def [get\\_vnc\\_console](#)
- def **validate\_console\_port**
- def **reserve\_block\_device\_name**
- def [attach\\_volume](#)
- def [detach\\_volume](#)
- def [remove\\_volume\\_connection](#)
- def [check\\_can\\_live\\_migrate\\_destination](#)
- def [check\\_can\\_live\\_migrate\\_source](#)
- def [pre\\_live\\_migration](#)
- def [live\\_migration](#)
- def [post\\_live\\_migration\\_at\\_destination](#)
- def [rollback\\_live\\_migration\\_at\\_destination](#)
- def [update\\_available\\_resource](#)
- def **get\_compute\_node\_ref**
- def **remove\_security\_properties**
- def **get\_security\_properties**
- def **update\_security\_properties**
- def [add\\_aggregate\\_host](#)
- def [remove\\_aggregate\\_host](#)

## Public Attributes

- **driver**
- **network\_api**
- **volume\_api**
- **network\_manager**
- **compute\_api**
- **compute\_rpcapi**
- **scheduler\_rpcapi**
- **consoleauth\_rpcapi**
- **resource\_tracker**

## Static Public Attributes

- string **RPC\_API\_VERSION** = '2.2'
- **ticks\_between\_runs** = FLAGS.running\_deleted\_instance\_poll\_interval)

### 3.14.1 Detailed Description

Manages the running instances from creation to destruction.

### 3.14.2 Constructor & Destructor Documentation

#### 3.14.2.1 `def nova.compute.manager.ComputeManager.__init__( self, compute_driver=None, args, kwargs )`

Load configuration options and connect to the hypervisor.

### 3.14.3 Member Function Documentation

#### 3.14.3.1 `def nova.compute.manager.ComputeManager.add_aggregate_host( self, context, aggregate_id, host, slave_info = None )`

Notify hypervisor of change (for hypervisor pools).

#### 3.14.3.2 `def nova.compute.manager.ComputeManager.add_fixed_ip_to_instance( self, context, network_id, instance )`

Calls network\_api to add new fixed\_ip to instance then injects the new network info and resets instance networking.

#### 3.14.3.3 `def nova.compute.manager.ComputeManager.attach_volume( self, context, volume_id, mountpoint, instance )`

Attach a volume to an instance.

#### 3.14.3.4 `def nova.compute.manager.ComputeManager.change_instance_metadata( self, context, diff, instance )`

Update the metadata published to the instance.

#### 3.14.3.5 `def nova.compute.manager.ComputeManager.check_can_live_migrate_destination( self, ctxt, instance, block_migration=False, disk_over_commit=False )`

Check if it is possible to execute live migration.

This runs checks on the destination host, and then calls back to the source host to check the results.

```
:param context: security context
:param instance: dict of instance data
:param block_migration: if true, prepare for block migration
:param disk_over_commit: if true, allow disk over commit
```

Returns a mapping of values required in case of block migration and None otherwise.

**3.14.3.6** `def nova.compute.manager.ComputeManager.check_can_live_migrate_source( self, ctxt, instance, dest_check_data )`

Check if it is possible to execute live migration.

This checks if the live migration can succeed, based on the results from `check_can_live_migrate_destination`.

```
:param context: security context
:param instance: dict of instance data
:param dest_check_data: result of check_can_live_migrate_destination
```

**3.14.3.7** `def nova.compute.manager.ComputeManager.confirm_resize( self, context, migration_id, instance, reservations = None )`

Destroys the source instance.

**3.14.3.8** `def nova.compute.manager.ComputeManager.detach_volume( self, context, volume_id, instance )`

Detach a volume from an instance.

**3.14.3.9** `def nova.compute.manager.ComputeManager.finish_resize( self, context, migration_id, disk_info, image, instance, reservations = None )`

Completes the migration process.

Sets up the newly transferred disk and turns on the instance at its new host machine.

**3.14.3.10** `def nova.compute.manager.ComputeManager.finish_revert_resize( self, context, migration_id, instance, reservations = None )`

Finishes the second half of reverting a resize.

Power back on the source instance and revert the resized attributes in the database.

**3.14.3.11** `def nova.compute.manager.ComputeManager.get_console_output( self, context, instance, tail_length = None )`

Send the console output for the given instance.

**3.14.3.12** `def nova.compute.manager.ComputeManager.get_console_topic( self, context )`

Retrieves the console host for a project on this host.

Currently this is just set in the flags for each compute host.

**3.14.3.13** `def nova.compute.manager.ComputeManager.get_diagnostics( self, context, instance )`

Retrieve diagnostics for an instance on this host.



**3.14.3.14** `def nova.compute.manager.ComputeManager.get_host_uptime ( self, context, host )`

Returns the result of calling "uptime" on the target host.

**3.14.3.15** `def nova.compute.manager.ComputeManager.get_vnc_console ( self, context, console_type, instance )`

Return connection information for a vnc console.

**3.14.3.16** `def nova.compute.manager.ComputeManager.host_maintenance_mode ( self, context, host, mode )`

Start/Stop host maintenance window. On start, it triggers guest VMs evacuation.

**3.14.3.17** `def nova.compute.manager.ComputeManager.host_power_action ( self, context, host=None, action=None )`

Reboots, shuts down or powers up the host.

**3.14.3.18** `def nova.compute.manager.ComputeManager.init_host ( self )`

Initialization for a standalone compute service.

**3.14.3.19** `def nova.compute.manager.ComputeManager.inject_file ( self, context, path, file_contents, instance )`

Write a file to the specified path in an instance on this host.

**3.14.3.20** `def nova.compute.manager.ComputeManager.inject_network_info ( self, context, instance )`

Inject network info, but don't return the info.

**3.14.3.21** `def nova.compute.manager.ComputeManager.live_migration ( self, context, dest, instance, block_migration = False, migrate_data=None )`

Executing live migration.

```
:param context: security context
:param instance: instance dict
:param dest: destination host
:param block_migration: if true, prepare for block migration
:param migrate_data: implementation specific params
```

**3.14.3.22** `def nova.compute.manager.ComputeManager.pause_instance ( self, context, instance )`

Pause an instance on this host.

**3.14.3.23** `def nova.compute.manager.ComputeManager.post_live_migration_at_destination ( self, context, instance, block_migration = False )`

Post operations for live migration .

```
:param context: security context
:param instance: Instance dict
:param block_migration: if true, prepare for block migration
```

**3.14.3.24** `def nova.compute.manager.ComputeManager.power_off_instance ( self, context, instance, final_state = vm_states.SOFT_DELETED )`

Power off an instance on this host.

**3.14.3.25** `def nova.compute.manager.ComputeManager.power_on_instance ( self, context, instance )`

Power on an instance on this host.

**3.14.3.26** `def nova.compute.manager.ComputeManager.pre_live_migration ( self, context, instance, block_migration = False, disk = None )`

Preparations for live migration at dest host.

```
:param context: security context
:param instance: dict of instance data
:param block_migration: if true, prepare for block migration
```

**3.14.3.27** `def nova.compute.manager.ComputeManager.prep_resize ( self, context, image, instance, instance_type, reservations = None )`

Initiates the process of moving a running instance to another host.

Possibly changes the RAM and disk size in the process.

**3.14.3.28** `def nova.compute.manager.ComputeManager.reboot_instance ( self, context, instance, reboot_type = "SOFT" )`

Reboot an instance on this host.

**3.14.3.29** `def nova.compute.manager.ComputeManager.rebuild_instance ( self, context, instance, orig_image_ref, image_ref, injected_files, new_pass, orig_sys_metadata = None )`

Destroy and re-make this instance.

A 'rebuild' effectively purges all existing data from the system and remakes the VM with given 'metadata' and 'personalities'.

```
:param context: `nova.RequestContext` object
:param instance: Instance dict
:param orig_image_ref: Original image_ref before rebuild
:param image_ref: New image_ref for rebuild
:param injected_files: Files to inject
:param new_pass: password to set on rebuilt instance
:param orig_sys_metadata: instance system metadata from pre-rebuild
```

**3.14.3.30** `def nova.compute.manager.ComputeManager.refresh_instance_security_rules ( self, context, instance )`

Tell the virtualization driver to refresh security rules for an instance.

Passes straight through to the virtualization driver.

**3.14.3.31** `def nova.compute.manager.ComputeManager.refresh_provider_fw_rules ( self, context )`

This call passes straight through to the virtualization driver.

**3.14.3.32** `def nova.compute.manager.ComputeManager.refresh_security_group_members ( self, context, security_group_id )`

Tell the virtualization driver to refresh security group members.

Passes straight through to the virtualization driver.

**3.14.3.33** `def nova.compute.manager.ComputeManager.refresh_security_group_rules ( self, context, security_group_id )`

Tell the virtualization driver to refresh security group rules.

Passes straight through to the virtualization driver.

**3.14.3.34** `def nova.compute.manager.ComputeManager.remove_aggregate_host ( self, context, aggregate_id, host, slave_info=None )`

Removes a host from a physical hypervisor pool.

**3.14.3.35** `def nova.compute.manager.ComputeManager.remove_fixed_ip_from_instance ( self, context, address, instance )`

Calls `network_api` to remove existing `fixed_ip` from instance by injecting the altered network info and resetting instance networking.

**3.14.3.36** `def nova.compute.manager.ComputeManager.remove_volume_connection ( self, context, volume_id, instance )`

Remove a volume connection using the volume api

**3.14.3.37** `def nova.compute.manager.ComputeManager.rescue_instance ( self, context, instance, rescue_password=None )`

Rescue an instance on this host.

:param rescue\_password: password to set on rescue instance

**3.14.3.38** `def nova.compute.manager.ComputeManager.reset_network ( self, context, instance )`

Reset networking on the given instance.

**3.14.3.39** `def nova.compute.manager.ComputeManager.resize_instance ( self, context, instance, migration_id, image, reservations = None )`

Starts the migration of a running instance to another host.

**3.14.3.40** `def nova.compute.manager.ComputeManager.resume_instance ( self, context, instance )`

Resume the given suspended instance.

**3.14.3.41** `def nova.compute.manager.ComputeManager.revert_resize ( self, context, instance, migration_id, reservations = None )`

Destroys the new instance on the destination machine.

Reverts the model changes, and powers on the old instance on the source machine.

**3.14.3.42** `def nova.compute.manager.ComputeManager.rollback_live_migration_at_destination ( self, context, instance )`

Cleaning up image directory that is created pre\_live\_migration.

```
:param context: security context
:param instance: an Instance dict sent over rpc
```

**3.14.3.43** `def nova.compute.manager.ComputeManager.set_admin_password ( self, context, instance, new_pass = None )`

Set the root/admin password for an instance on this host.

This is generally only called by API password resets after an image has been built.

**3.14.3.44** `def nova.compute.manager.ComputeManager.set_host_enabled ( self, context, host = None, enabled = None )`

Sets the specified host's ability to accept new instances.

**3.14.3.45** `def nova.compute.manager.ComputeManager.snapshot_instance ( self, context, image_id, instance, image_type = 'snapshot', backup_type = None, rotation = None )`

Snapshot an instance on this host.

```
:param context: security context
:param instance: an Instance dict
:param image_id: glance.db.sqlalchemy.models.Image.Id
:param image_type: snapshot | backup
:param backup_type: daily | weekly
:param rotation: int representing how many backups to keep around;
                  None if rotation shouldn't be used (as in the case of snapshots)
```

**3.14.3.46** `def nova.compute.manager.ComputeManager.start_instance ( self, context, instance )`

Starting an instance on this host.

Alias for power\_on\_instance for compatibility

**3.14.3.47** `def nova.compute.manager.ComputeManager.stop_instance ( self, context, instance )`

Stopping an instance on this host.

Alias for power\_off\_instance for compatibility

**3.14.3.48** `def nova.compute.manager.ComputeManager.suspend_instance ( self, context, instance )`

Suspend the given instance.

**3.14.3.49** `def nova.compute.manager.ComputeManager.terminate_instance ( self, context, instance )`

Terminate an instance on this host.

**3.14.3.50** `def nova.compute.manager.ComputeManager.unpause_instance ( self, context, instance )`

Unpause a paused instance on this host.

**3.14.3.51** `def nova.compute.manager.ComputeManager.unrescue_instance ( self, context, instance )`

Rescue an instance on this host.

**3.14.3.52** `def nova.compute.manager.ComputeManager.update_available_resource ( self, context )`

See driver.get\_available\_resource()

Periodic process that keeps that the compute host's understanding of resource availability and usage in sync with the underlying hypervisor.

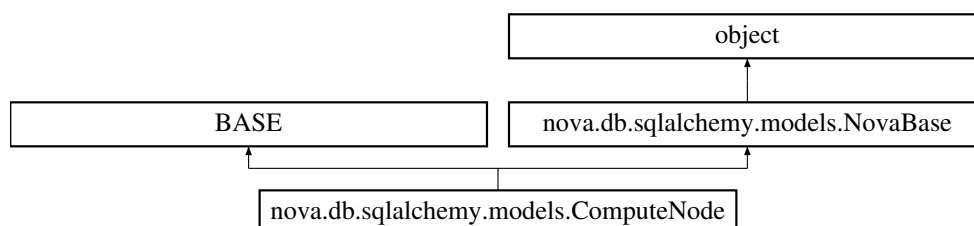
:param context: security context

The documentation for this class was generated from the following file:

- nova/compute/manager.py

## 3.15 nova.db.sqlalchemy.models.ComputeNode Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.ComputeNode:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **service\_id** = Column(Integer, ForeignKey('services.id'), nullable=True)
- tuple **service**
- tuple **vcpus** = Column(Integer)
- tuple **memory\_mb** = Column(Integer)
- tuple **local\_gb** = Column(Integer)
- tuple **vcpus\_used** = Column(Integer)
- tuple **memory\_mb\_used** = Column(Integer)
- tuple **local\_gb\_used** = Column(Integer)
- tuple **hypervisor\_type** = Column(Text)
- tuple **hypervisor\_version** = Column(Integer)
- tuple **hypervisor\_hostname** = Column(String(255))
- tuple **free\_ram\_mb** = Column(Integer)
- tuple **free\_disk\_gb** = Column(Integer)
- tuple **current\_workload** = Column(Integer)
- tuple **running\_vms** = Column(Integer)
- tuple **cpu\_info** = Column(Text, nullable=True)
- tuple **disk\_available\_least** = Column(Integer)
- tuple **security\_properties** = Column(Text, nullable=True)

## Additional Inherited Members

### 3.15.1 Detailed Description

Represents a running compute service on a host.

### 3.15.2 Member Data Documentation

#### 3.15.2.1 tuple nova.db.sqlalchemy.models.ComputeNode.service [static]

##### Initial value:

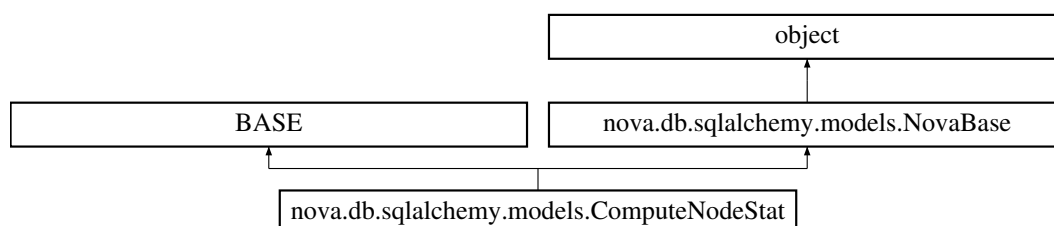
```
1 = relationship(Service,
2                     backref=backref('compute_node'),
3                     foreign_keys=service_id,
4                     primaryjoin='and_('
5                                 'ComputeNode.service_id == Service.id,'
6                                 'ComputeNode.deleted == False)')
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.16 nova.db.sqlalchemy.models.ComputeNodeStat Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.ComputeNodeStat:



## Public Member Functions

- def `__str__`

## Static Public Attributes

- tuple `id` = Column(Integer, primary\_key=True)
- tuple `key` = Column(String(511))
- tuple `value` = Column(String(255))
- tuple `compute_node_id` = Column(Integer, ForeignKey('compute\_nodes.id'))
- tuple `primary_join`
- tuple `stats`

## Additional Inherited Members

### 3.16.1 Detailed Description

Stats related to the current workload of a compute host that are intended to aid in making scheduler decisions.

### 3.16.2 Member Data Documentation

#### 3.16.2.1 tuple nova.db.sqlalchemy.models.ComputeNodeStat.primary\_join [static]

Initial value:

```
1 = ('and_(ComputeNodeStat.compute_node_id == '  
2     'ComputeNode.id, ComputeNodeStat.deleted == False)')
```

#### 3.16.2.2 tuple nova.db.sqlalchemy.models.ComputeNodeStat.stats [static]

Initial value:

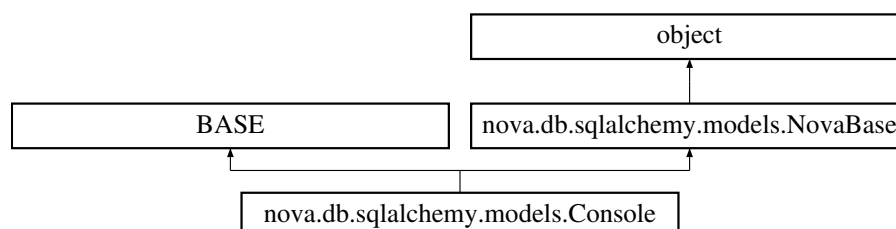
```
1 = relationship("ComputeNode", backref="stats",  
2     primaryjoin=primary_join)
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.17 nova.db.sqlalchemy.models.Console Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Console:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **instance\_name** = Column(String(255))
- tuple **instance\_uuid** = Column(String(36))
- tuple **password** = Column(String(255))
- tuple **port** = Column(Integer, nullable=True)
- tuple **pool\_id** = Column(Integer, ForeignKey('console\_pools.id'))
- tuple **pool** = relationship(ConsolePool, backref=backref('consoles'))

## Additional Inherited Members

### 3.17.1 Detailed Description

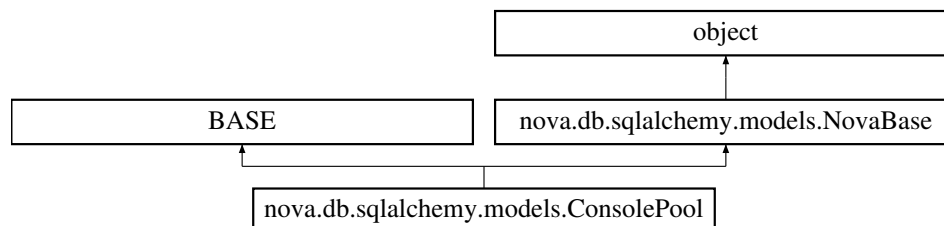
Represents a console session for an instance.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.18 nova.db.sqlalchemy.models.ConsolePool Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.ConsolePool:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **address** = Column(String(255))
- tuple **username** = Column(String(255))
- tuple **password** = Column(String(255))
- tuple **console\_type** = Column(String(255))
- tuple **public\_hostname** = Column(String(255))
- tuple **host** = Column(String(255))
- tuple **compute\_host** = Column(String(255))

## Additional Inherited Members

### 3.18.1 Detailed Description

Represents pool of consoles on the same physical node.

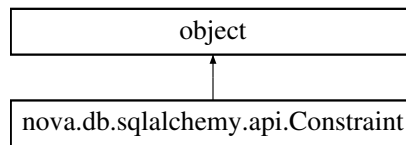
The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py



## 3.19 nova.db.sqlalchemy.api.Constraint Class Reference

Inheritance diagram for nova.db.sqlalchemy.api.Constraint:



### Public Member Functions

- def **\_\_init\_\_**
- def **apply**

### Public Attributes

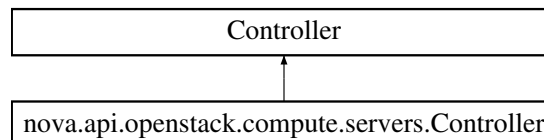
- **conditions**

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/api.py

## 3.20 nova.api.openstack.compute.servers.Controller Class Reference

Inheritance diagram for nova.api.openstack.compute.servers.Controller:



### Public Member Functions

- def **\_\_init\_\_**
- def [index](#)
- def [detail](#)
- def [show](#)
- def [create](#)
- def [update](#)
- def [delete](#)

### Public Attributes

- **compute\_api**
- **ext\_mgr**
- **quantum\_attempted**
- **have\_quantum**

## Static Public Attributes

- tuple **B64\_REGEX**

### 3.20.1 Detailed Description

The Server API base controller class for the OpenStack API.

### 3.20.2 Member Function Documentation

#### 3.20.2.1 `def nova.api.openstack.compute.servers.Controller.create ( self, req, body )`

Creates a new server for a given user.

#### 3.20.2.2 `def nova.api.openstack.compute.servers.Controller.delete ( self, req, id )`

Destroys a server.

#### 3.20.2.3 `def nova.api.openstack.compute.servers.Controller.detail ( self, req )`

Returns a list of server details for a given user.

#### 3.20.2.4 `def nova.api.openstack.compute.servers.Controller.index ( self, req )`

Returns a list of server names and ids for a given user.

#### 3.20.2.5 `def nova.api.openstack.compute.servers.Controller.show ( self, req, id )`

Returns server details by server id.

#### 3.20.2.6 `def nova.api.openstack.compute.servers.Controller.update ( self, req, id, body )`

Update server then pass on to version-specific controller.

### 3.20.3 Member Data Documentation

#### 3.20.3.1 `tuple nova.api.openstack.compute.servers.Controller.B64_REGEX [static]`

**Initial value:**

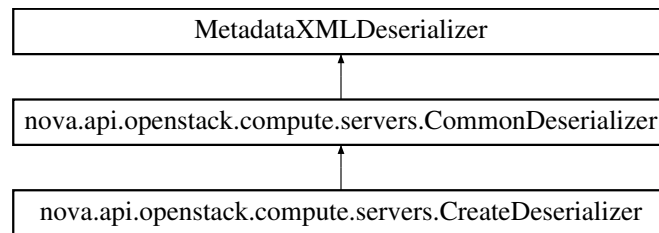
```
1 = re.compile('^(?:[A-Za-z0-9+\/]{4})*'
2     ' (?:[A-Za-z0-9+\/]{2}==|'
3     ' | [A-Za-z0-9+\/]{3}=) ?$')
```

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/servers.py

## 3.21 nova.api.openstack.compute.servers.CreateDeserializer Class Reference

Inheritance diagram for nova.api.openstack.compute.servers.CreateDeserializer:



### Public Member Functions

- def [default](#)

### Additional Inherited Members

#### 3.21.1 Detailed Description

Deserializer to handle xml-formatted server create requests.

Handles standard server attributes as well as optional metadata and personality attributes

#### 3.21.2 Member Function Documentation

##### 3.21.2.1 def nova.api.openstack.compute.servers.CreateDeserializer.default ( self, string )

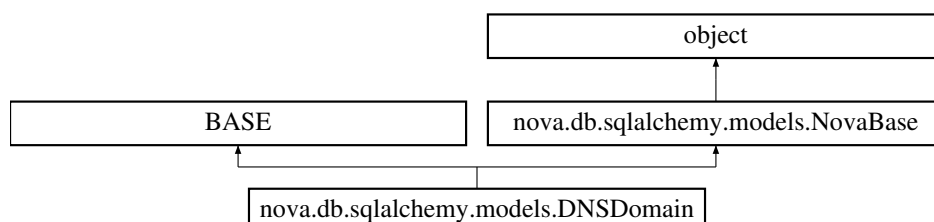
Deserialize an xml-formatted server create request.

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/servers.py

## 3.22 nova.db.sqlalchemy.models.DNSDomain Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.DNSDomain:



### Static Public Attributes

- tuple **domain** = Column(String(512), primary\_key=True)
- tuple **scope** = Column(String(255))

- tuple **availability\_zone** = Column(String(255))
- tuple **project\_id** = Column(String(255))

## Additional Inherited Members

### 3.22.1 Detailed Description

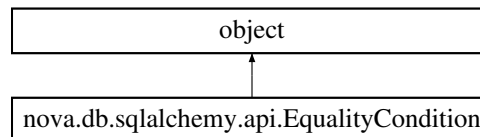
Represents a DNS domain with availability zone or project info.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.23 nova.db.sqlalchemy.api.EqualityCondition Class Reference

Inheritance diagram for nova.db.sqlalchemy.api.EqualityCondition:



## Public Member Functions

- def **\_\_init\_\_**
- def **clauses**

## Public Attributes

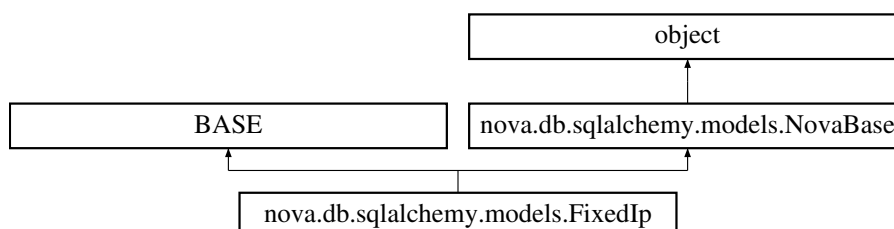
- **values**

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/api.py

## 3.24 nova.db.sqlalchemy.models.FixedIp Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.FixedIp:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **address** = Column(String(255))
- tuple **network\_id** = Column(Integer, nullable=True)
- tuple **virtual\_interface\_id** = Column(Integer, nullable=True)
- tuple **instance\_uuid** = Column(String(36), nullable=True)
- tuple **allocated** = Column(Boolean, default=False)
- tuple **leased** = Column(Boolean, default=False)
- tuple **reserved** = Column(Boolean, default=False)
- tuple **host** = Column(String(255))

## Additional Inherited Members

### 3.24.1 Detailed Description

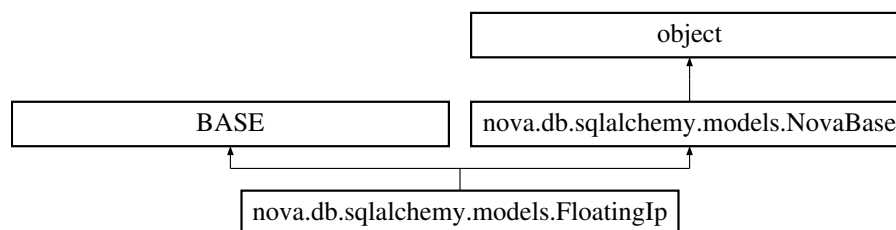
Represents a fixed ip for an instance.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.25 nova.db.sqlalchemy.models.FloatingIp Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.FloatingIp:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **address** = Column(String(255))
- tuple **fixed\_ip\_id** = Column(Integer, nullable=True)
- tuple **project\_id** = Column(String(255))
- tuple **host** = Column(String(255))
- tuple **auto\_assigned** = Column(Boolean, default=False, nullable=False)
- tuple **pool** = Column(String(255))
- tuple **interface** = Column(String(255))

## Additional Inherited Members

### 3.25.1 Detailed Description

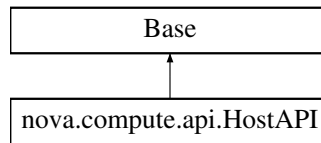
Represents a floating ip that dynamically forwards to a fixed ip.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.26 nova.compute.api.HostAPI Class Reference

Inheritance diagram for nova.compute.api.HostAPI:



### Public Member Functions

- `def __init__`
- `def set_host_enabled`
- `def get_host_uptime`
- `def host_power_action`
- `def set_host_maintenance`

### Public Attributes

- `compute_rpcapi`

### 3.26.1 Member Function Documentation

#### 3.26.1.1 `def nova.compute.api.HostAPI.get_host_uptime ( self, context, host )`

Returns the result of calling "uptime" on the target host.

#### 3.26.1.2 `def nova.compute.api.HostAPI.host_power_action ( self, context, host, action )`

Reboots, shuts down or powers up the host.

#### 3.26.1.3 `def nova.compute.api.HostAPI.set_host_enabled ( self, context, host, enabled )`

Sets the specified host's ability to accept new instances.

#### 3.26.1.4 `def nova.compute.api.HostAPI.set_host_maintenance ( self, context, host, mode )`

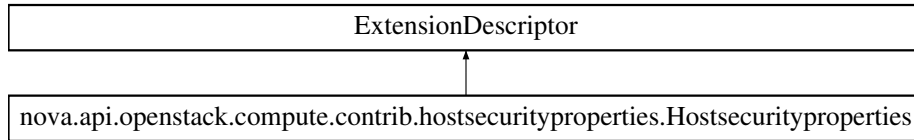
Start/Stop host maintenance window. On start, it triggers guest VMs evacuation.

The documentation for this class was generated from the following file:

- `nova/compute/api.py`

## 3.27 nova.api.openstack.compute.contrib.hostsecurityproperties.Hostsecurityproperties Class Reference

Inheritance diagram for nova.api.openstack.compute.contrib.hostsecurityproperties.Hostsecurityproperties:



### Public Member Functions

- def **get\_resources**

### Static Public Attributes

- string **name** = "HostSecurityProperties"
- string **alias** = "os-host-security-properties"
- tuple **namespace**
- string **updated** = "2012-10-12T00:00:00+00:00"

### 3.27.1 Detailed Description

Host Security Properties

### 3.27.2 Member Data Documentation

3.27.2.1 tuple nova.api.openstack.compute.contrib.hostsecurityproperties.Hostsecurityproperties.namespace [static]

**Initial value:**

```

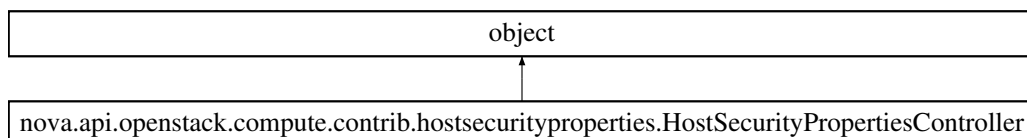
1 = ("http://docs.openstack.org/compute/ext/"
2   "host_security_properties/api/v1.1")
  
```

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/contrib/hostsecurityproperties.py

## 3.28 nova.api.openstack.compute.contrib.hostsecurityproperties.HostSecurityProperties-Controller Class Reference

Inheritance diagram for nova.api.openstack.compute.contrib.hostsecurityproperties.HostSecurityProperties-Controller:



## Public Member Functions

- def [index](#)
- def **create\_or\_update**
- def **create**
- def **update**
- def [show](#)
- def [delete](#)

### 3.28.1 Detailed Description

The Host Properties API controller for the OpenStack API

### 3.28.2 Member Function Documentation

**3.28.2.1** `def nova.api.openstack.compute.contrib.hostsecurityproperties.HostSecurityPropertiesController.delete ( self, req, host_id, id )`

Deletes an existing security property

**3.28.2.2** `def nova.api.openstack.compute.contrib.hostsecurityproperties.HostSecurityPropertiesController.index ( self, req, host_id )`

Returns the list of security properties for a given host

**3.28.2.3** `def nova.api.openstack.compute.contrib.hostsecurityproperties.HostSecurityPropertiesController.show ( self, req, host_id, id )`

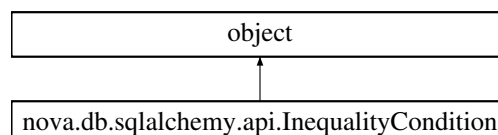
Return a single security property item

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/contrib/hostsecurityproperties.py

## 3.29 nova.db.sqlalchemy.api.InequalityCondition Class Reference

Inheritance diagram for nova.db.sqlalchemy.api.InequalityCondition:



## Public Member Functions

- def **\_\_init\_\_**
- def **clauses**



## Public Attributes

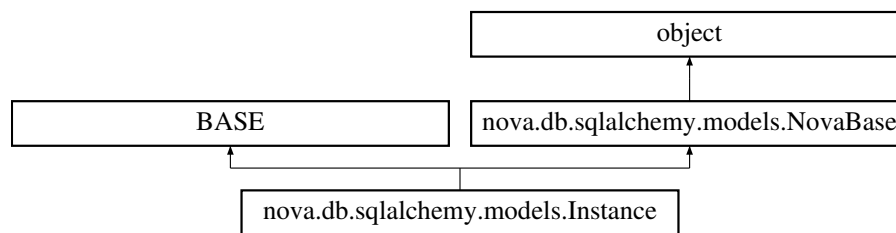
- **values**

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/api.py

## 3.30 nova.db.sqlalchemy.models.Instance Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Instance:



## Public Member Functions

- def **name**

## Static Public Attributes

- list **injected\_files** = []
- tuple **id** = Column(Integer, primary\_key=True, autoincrement=True)
- tuple **user\_id** = Column(String(255))
- tuple **project\_id** = Column(String(255))
- tuple **image\_ref** = Column(String(255))
- tuple **kernel\_id** = Column(String(255))
- tuple **ramdisk\_id** = Column(String(255))
- tuple **server\_name** = Column(String(255))
- tuple **launch\_index** = Column(Integer)
- tuple **key\_name** = Column(String(255))
- tuple **key\_data** = Column(Text)
- tuple **power\_state** = Column(Integer)
- tuple **vm\_state** = Column(String(255))
- tuple **task\_state** = Column(String(255))
- tuple **memory\_mb** = Column(Integer)
- tuple **vcpus** = Column(Integer)
- tuple **root\_gb** = Column(Integer)
- tuple **ephemeral\_gb** = Column(Integer)
- tuple **hostname** = Column(String(255))
- tuple **host** = Column(String(255))
- tuple **instance\_type\_id** = Column(Integer)
- tuple **user\_data** = Column(Text)
- tuple **reservation\_id** = Column(String(255))
- tuple **scheduled\_at** = Column(DateTime)
- tuple **launched\_at** = Column(DateTime)
- tuple **terminated\_at** = Column(DateTime)

- tuple **availability\_zone** = Column(String(255))
- tuple **display\_name** = Column(String(255))
- tuple **display\_description** = Column(String(255))
- tuple **launched\_on** = Column(Text)
- tuple **locked** = Column(Boolean)
- tuple **os\_type** = Column(String(255))
- tuple **architecture** = Column(String(255))
- tuple **vm\_mode** = Column(String(255))
- tuple **uuid** = Column(String(36))
- tuple **root\_device\_name** = Column(String(255))
- tuple **default\_ephemeral\_device** = Column(String(255), nullable=True)
- tuple **default\_swap\_device** = Column(String(255), nullable=True)
- tuple **config\_drive** = Column(String(255))
- tuple **access\_ip\_v4** = Column(String(255))
- tuple **access\_ip\_v6** = Column(String(255))
- tuple **auto\_disk\_config** = Column(Boolean())
- tuple **progress** = Column(Integer)
- tuple **shutdown\_terminate** = Column(Boolean(), default=False, nullable=False)
- tuple **disable\_terminate** = Column(Boolean(), default=False, nullable=False)

## Additional Inherited Members

### 3.30.1 Detailed Description

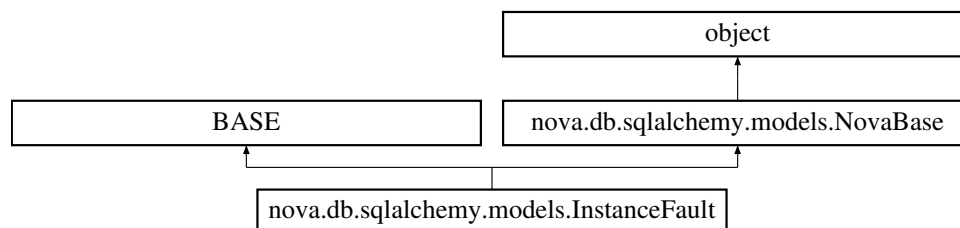
Represents a guest VM.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.31 nova.db.sqlalchemy.models.InstanceFault Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.InstanceFault:



## Static Public Attributes

- tuple **id** = Column(Integer(), primary\_key=True, autoincrement=True)
- tuple **instance\_uuid**
- tuple **code** = Column(Integer(), nullable=False)
- tuple **message** = Column(String(255))
- tuple **details** = Column(Text)

## Additional Inherited Members

### 3.31.1 Member Data Documentation

#### 3.31.1.1 tuple nova.db.sqlalchemy.models.InstanceFault.instance\_uuid [static]

**Initial value:**

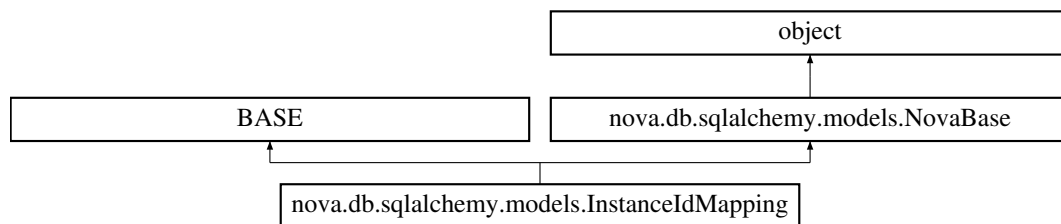
```
1 = Column(String(36),
2         ForeignKey('instances.uuid'),
3         nullable=False)
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.32 nova.db.sqlalchemy.models.InstanceIdMapping Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.InstanceIdMapping:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, nullable=False, autoincrement=True)
- tuple **uuid** = Column(String(36), nullable=False)

## Additional Inherited Members

### 3.32.1 Detailed Description

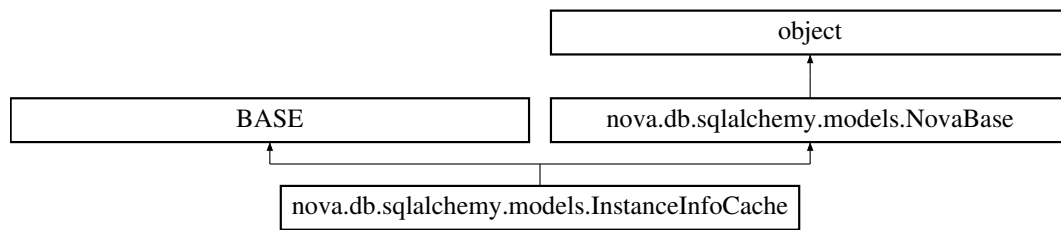
Comptability layer for the EC2 instance service

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.33 nova.db.sqlalchemy.models.InstanceInfoCache Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.InstanceInfoCache:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, autoincrement=True)
- tuple **network\_info** = Column(Text)
- tuple **instance\_uuid**
- tuple **instance**

## Additional Inherited Members

### 3.33.1 Detailed Description

Represents a cache of information about an instance

### 3.33.2 Member Data Documentation

#### 3.33.2.1 tuple nova.db.sqlalchemy.models.InstanceInfoCache.instance [static]

##### Initial value:

```

1 = relationship(Instance,
2                 backref=backref('info_cache', uselist=False),
3                 foreign_keys=instance_uuid,
4                 primaryjoin=instance_uuid == Instance.uuid)

```

#### 3.33.2.2 tuple nova.db.sqlalchemy.models.InstanceInfoCache.instance\_uuid [static]

##### Initial value:

```

1 = Column(String(36), ForeignKey('instances.uuid'),
2           nullable=False, unique=True)

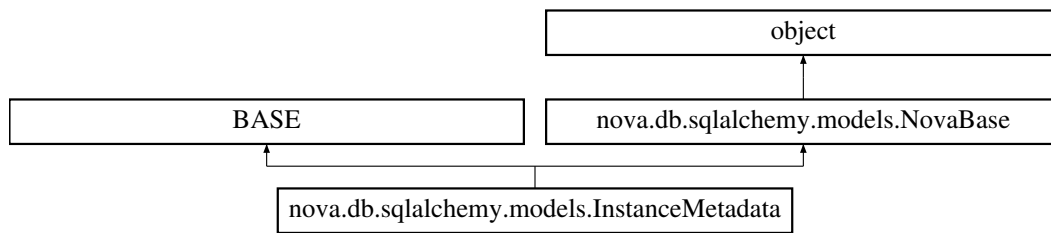
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.34 nova.db.sqlalchemy.models.InstanceMetadata Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.InstanceMetadata:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **key** = Column(String(255))
- tuple **value** = Column(String(255))
- tuple **instance\_uuid**
- tuple **instance**

### Additional Inherited Members

#### 3.34.1 Detailed Description

Represents a user-provided metadata key/value pair for an instance

#### 3.34.2 Member Data Documentation

##### 3.34.2.1 tuple nova.db.sqlalchemy.models.InstanceMetadata.instance [static]

**Initial value:**

```

1 = relationship(Instance, backref="metadata",
2                 foreign_keys=instance_uuid,
3                 primaryjoin='and_('
4                 'InstanceMetadata.instance_uuid == '
5                 'Instance.uuid,'
6                 'InstanceMetadata.deleted == False)')
```

##### 3.34.2.2 tuple nova.db.sqlalchemy.models.InstanceMetadata.instance\_uuid [static]

**Initial value:**

```

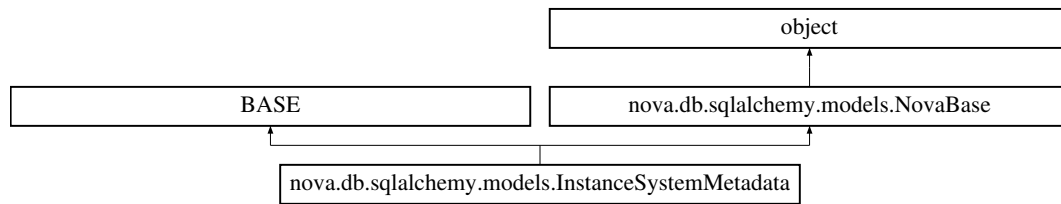
1 = Column(String(36), ForeignKey('instances.uuid'),
2           nullable=False)
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.35 nova.db.sqlalchemy.models.InstanceSystemMetadata Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.InstanceSystemMetadata:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **key** = Column(String(255))
- tuple **value** = Column(String(255))
- tuple **instance\_uuid**
- tuple **primary\_join**
- tuple **instance**

## Additional Inherited Members

### 3.35.1 Detailed Description

Represents a system-owned metadata key/value pair for an instance

### 3.35.2 Member Data Documentation

3.35.2.1 tuple `nova.db.sqlalchemy.models.InstanceSystemMetadata.instance` [static]

**Initial value:**

```

1 = relationship(Instance, backref="system_metadata",
2                 foreign_keys=instance_uuid,
3                 primaryjoin=primary_join)

```

3.35.2.2 tuple `nova.db.sqlalchemy.models.InstanceSystemMetadata.instance_uuid` [static]

**Initial value:**

```

1 = Column(String(36),
2           ForeignKey('instances.uuid'),
3           nullable=False)

```

3.35.2.3 tuple `nova.db.sqlalchemy.models.InstanceSystemMetadata.primary_join` [static]

**Initial value:**

```

1 = ('and_(InstanceSystemMetadata.instance_uuid == '
2       'Instance.uuid, InstanceSystemMetadata.deleted == False)')

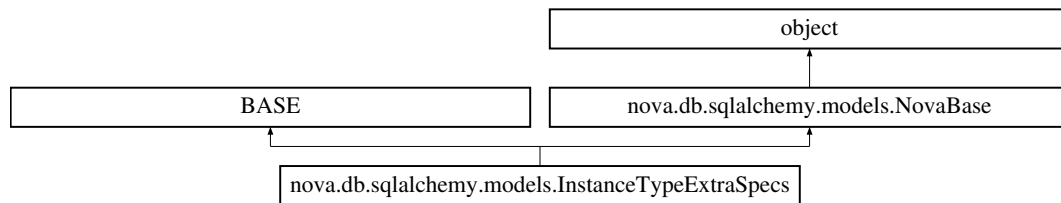
```

The documentation for this class was generated from the following file:

- `nova/db/sqlalchemy/models.py`

### 3.36 nova.db.sqlalchemy.models.InstanceTypeExtraSpecs Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.InstanceTypeExtraSpecs:



#### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **key** = Column(String(255))
- tuple **value** = Column(String(255))
- tuple **instance\_type\_id**
- tuple **instance\_type**

#### Additional Inherited Members

##### 3.36.1 Detailed Description

Represents additional specs as key/value pairs for an instance\_type

##### 3.36.2 Member Data Documentation

###### 3.36.2.1 tuple nova.db.sqlalchemy.models.InstanceTypeExtraSpecs.instance\_type [static]

**Initial value:**

```

1 = relationship(InstanceTypes, backref="extra_specs",
2     foreign_keys=instance_type_id,
3     primaryjoin='and_(
4         InstanceTypeExtraSpecs.instance_type_id == InstanceTypes.id,
5         InstanceTypeExtraSpecs.deleted == False)')
```

###### 3.36.2.2 tuple nova.db.sqlalchemy.models.InstanceTypeExtraSpecs.instance\_type\_id [static]

**Initial value:**

```

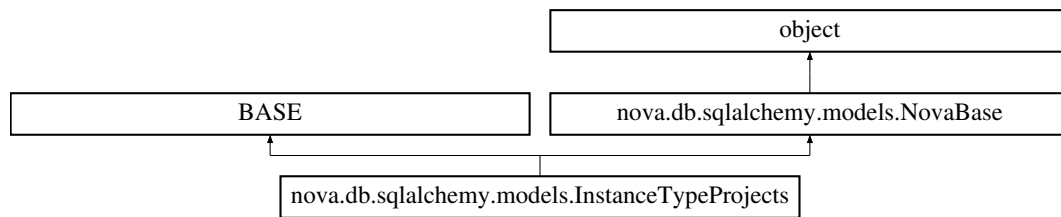
1 = Column(Integer, ForeignKey('instance_types.id'),
2     nullable=False)
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

### 3.37 nova.db.sqlalchemy.models.InstanceTypeProjects Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.InstanceTypeProjects:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **instance\_type\_id**
- tuple **project\_id** = Column(String(255))
- tuple **instance\_type**

### Additional Inherited Members

#### 3.37.1 Detailed Description

Represent projects associated instance\_types

#### 3.37.2 Member Data Documentation

##### 3.37.2.1 tuple nova.db.sqlalchemy.models.InstanceTypeProjects.instance\_type [static]

**Initial value:**

```

1 = relationship(InstanceTypes, backref="projects",
2               foreign_keys=instance_type_id,
3               primaryjoin='and_('
4               'InstanceTypeProjects.instance_type_id == InstanceTypes.id,'
5               'InstanceTypeProjects.deleted == False)')
```

##### 3.37.2.2 tuple nova.db.sqlalchemy.models.InstanceTypeProjects.instance\_type\_id [static]

**Initial value:**

```

1 = Column(Integer, ForeignKey('instance_types.id'),
2           nullable=False)
```

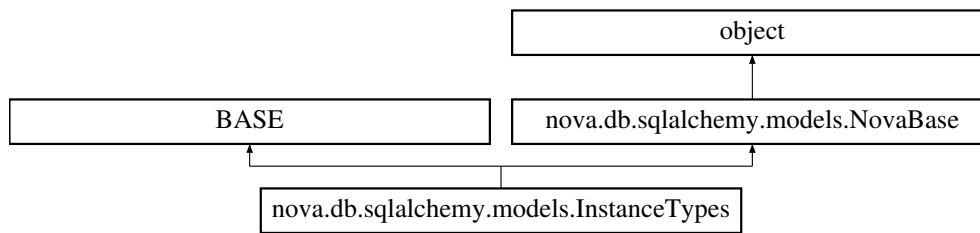
The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.38 nova.db.sqlalchemy.models.InstanceTypes Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.InstanceTypes:





### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **name** = Column(String(255))
- tuple **memory\_mb** = Column(Integer)
- tuple **vcpus** = Column(Integer)
- tuple **root\_gb** = Column(Integer)
- tuple **ephemeral\_gb** = Column(Integer)
- tuple **flavorid** = Column(String(255))
- tuple **swap** = Column(Integer, nullable=False, default=0)
- tuple **rxtx\_factor** = Column(Float, nullable=False, default=1)
- tuple **vcpu\_weight** = Column(Integer, nullable=True)
- tuple **disabled** = Column(Boolean, default=False)
- tuple **is\_public** = Column(Boolean, default=True)
- tuple **instances**

### Additional Inherited Members

#### 3.38.1 Detailed Description

Represent possible instance\_types or flavor of VM offered

#### 3.38.2 Member Data Documentation

##### 3.38.2.1 tuple nova.db.sqlalchemy.models.InstanceTypes.instances [static]

**Initial value:**

```

1 = relationship(Instance,
2                 backref=backref('instance_type', uselist=False),
3                 foreign_keys=id,
4                 primaryjoin='and_('
5                     'Instance.instance_type_id == '
6                     'InstanceTypes.id)')

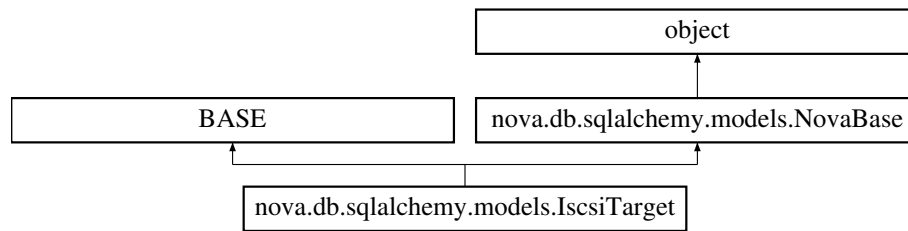
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.39 nova.db.sqlalchemy.models.IscsiTarget Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.IscsiTarget:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **target\_num** = Column(Integer)
- tuple **host** = Column(String(255))
- tuple **volume\_id** = Column(String(36), ForeignKey('volumes.id'), nullable=True)
- tuple **volume**

### Additional Inherited Members

#### 3.39.1 Detailed Description

Represents an iscsi target for a given host

#### 3.39.2 Member Data Documentation

##### 3.39.2.1 tuple nova.db.sqlalchemy.models.IscsiTarget.volume [static]

**Initial value:**

```

1 = relationship(Volume,
2                 backref=backref('iscsi_target', uselist=False),
3                 foreign_keys=volume_id,
4                 primaryjoin='and_(IscsiTarget.volume_id==Volume.id,
5                                   'IscsiTarget.deleted==False)')

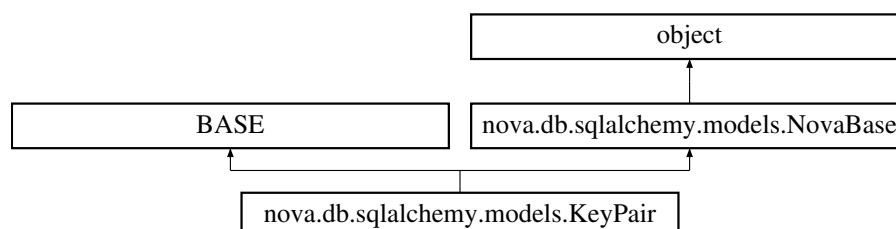
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.40 nova.db.sqlalchemy.models.KeyPair Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.KeyPair:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **name** = Column(String(255))
- tuple **user\_id** = Column(String(255))
- tuple **fingerprint** = Column(String(255))
- tuple **public\_key** = Column(Text)

## Additional Inherited Members

### 3.40.1 Detailed Description

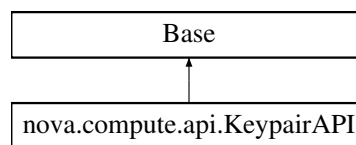
Represents a public key pair for ssh.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.41 nova.compute.api.KeypairAPI Class Reference

Inheritance diagram for nova.compute.api.KeypairAPI:



## Public Member Functions

- def **\_\_init\_\_**
- def [import\\_key\\_pair](#)
- def [create\\_key\\_pair](#)
- def [delete\\_key\\_pair](#)
- def [get\\_key\\_pairs](#)
- def [get\\_key\\_pair](#)

### 3.41.1 Detailed Description

Sub-set of the Compute Manager API for managing key pairs.

### 3.41.2 Member Function Documentation

**3.41.2.1** def nova.compute.api.KeypairAPI.create\_key\_pair ( self, context, user\_id, key\_name )

Create a new key pair.

**3.41.2.2** def nova.compute.api.KeypairAPI.delete\_key\_pair ( self, context, user\_id, key\_name )

Delete a keypair by name.

**3.41.2.3** `def nova.compute.api.KeypairAPI.get_key_pair ( self, context, user_id, key_name )`

Get a keypair by name.

**3.41.2.4** `def nova.compute.api.KeypairAPI.get_key_pairs ( self, context, user_id )`

List key pairs.

**3.41.2.5** `def nova.compute.api.KeypairAPI.import_key_pair ( self, context, user_id, key_name, public_key )`

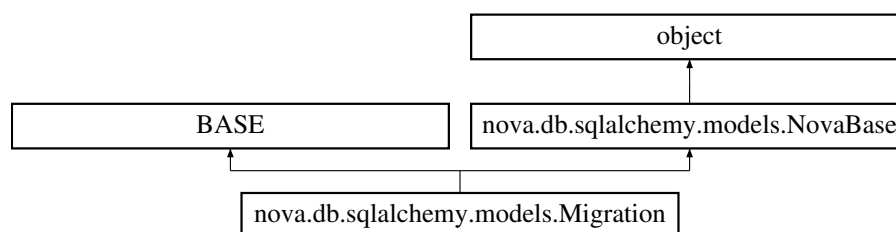
Import a key pair using an existing public key.

The documentation for this class was generated from the following file:

- nova/compute/api.py

## 3.42 nova.db.sqlalchemy.models.Migration Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Migration:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, nullable=False)
- tuple **source\_compute** = Column(String(255))
- tuple **dest\_compute** = Column(String(255))
- tuple **dest\_host** = Column(String(255))
- tuple **old\_instance\_type\_id** = Column(Integer())
- tuple **new\_instance\_type\_id** = Column(Integer())
- tuple **instance\_uuid**
- tuple **status** = Column(String(255))

### Additional Inherited Members

#### 3.42.1 Detailed Description

Represents a running host-to-host migration.

### 3.42.2 Member Data Documentation

#### 3.42.2.1 tuple nova.db.sqlalchemy.models.Migration.instance\_uuid [static]

**Initial value:**

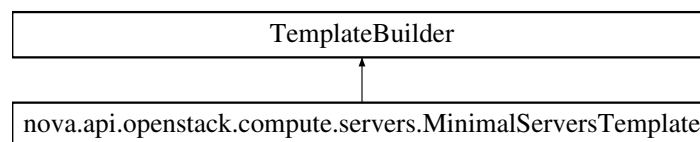
```
1 = Column(String(255), ForeignKey('instances.uuid'),
2         nullable=True)
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.43 nova.api.openstack.compute.servers.MinimalServersTemplate Class Reference

Inheritance diagram for nova.api.openstack.compute.servers.MinimalServersTemplate:



### Public Member Functions

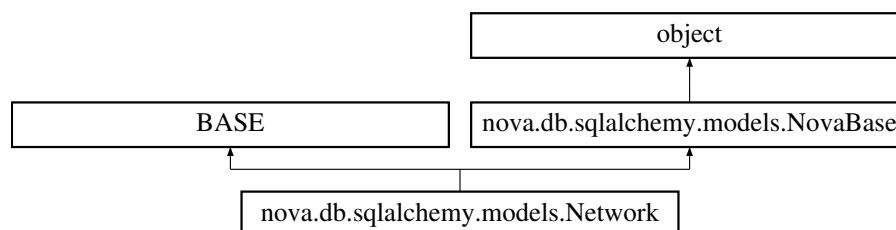
- def **construct**

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/servers.py

## 3.44 nova.db.sqlalchemy.models.Network Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Network:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **label** = Column(String(255))
- tuple **injected** = Column(Boolean, default=False)
- tuple **cidr** = Column(String(255), unique=True)
- tuple **cidr\_v6** = Column(String(255), unique=True)
- tuple **multi\_host** = Column(Boolean, default=False)

- tuple **gateway\_v6** = Column(String(255))
- tuple **netmask\_v6** = Column(String(255))
- tuple **netmask** = Column(String(255))
- tuple **bridge** = Column(String(255))
- tuple **bridge\_interface** = Column(String(255))
- tuple **gateway** = Column(String(255))
- tuple **broadcast** = Column(String(255))
- tuple **dns1** = Column(String(255))
- tuple **dns2** = Column(String(255))
- tuple **vlan** = Column(Integer)
- tuple **vpn\_public\_address** = Column(String(255))
- tuple **vpn\_public\_port** = Column(Integer)
- tuple **vpn\_private\_address** = Column(String(255))
- tuple **dhcp\_start** = Column(String(255))
- tuple **rxtx\_base** = Column(Integer)
- tuple **project\_id** = Column(String(255))
- tuple **priority** = Column(Integer)
- tuple **host** = Column(String(255))
- tuple **uuid** = Column(String(36))

## Additional Inherited Members

### 3.44.1 Detailed Description

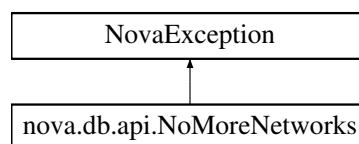
Represents a network.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.45 nova.db.api.NoMoreNetworks Class Reference

Inheritance diagram for nova.db.api.NoMoreNetworks:



### 3.45.1 Detailed Description

No more available networks.

The documentation for this class was generated from the following file:

- nova/db/api.py



## Public Member Functions

- def [save](#)
- def [delete](#)
- def [\\_\\_setitem\\_\\_](#)
- def [\\_\\_getitem\\_\\_](#)
- def [get](#)
- def [\\_\\_iter\\_\\_](#)
- def [next](#)
- def [update](#)
- def [iteritems](#)

## Public Attributes

- **deleted**
- **deleted\_at**

## Static Public Attributes

- tuple **created\_at** = Column(DateTime, default=timeutils.utcnow)
- tuple **updated\_at** = Column(DateTime, onupdate=timeutils.utcnow)
- tuple **deleted\_at** = Column(DateTime)
- tuple **deleted** = Column(Boolean, default=False)
- **metadata** = None

### 3.47.1 Detailed Description

Base class for Nova Models.

### 3.47.2 Member Function Documentation

**3.47.2.1** def nova.db.sqlalchemy.models.NovaBase.delete ( *self*, *session* = None )

Delete this object.

**3.47.2.2** def nova.db.sqlalchemy.models.NovaBase.iteritems ( *self* )

Make the model object behave like a dict.

Includes attributes from joins.

**3.47.2.3** def nova.db.sqlalchemy.models.NovaBase.save ( *self*, *session* = None )

Save this object.

**3.47.2.4** def nova.db.sqlalchemy.models.NovaBase.update ( *self*, *values* )

Make the model object behave like a dict

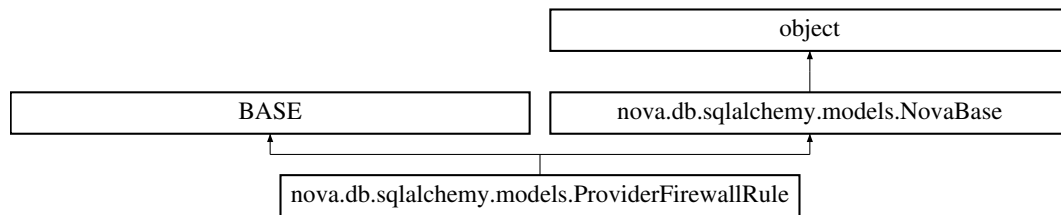
The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py



### 3.48 nova.db.sqlalchemy.models.ProviderFirewallRule Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.ProviderFirewallRule:



#### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **protocol** = Column(String(5))
- tuple **from\_port** = Column(Integer)
- tuple **to\_port** = Column(Integer)
- tuple **cidr** = Column(String(255))

#### Additional Inherited Members

##### 3.48.1 Detailed Description

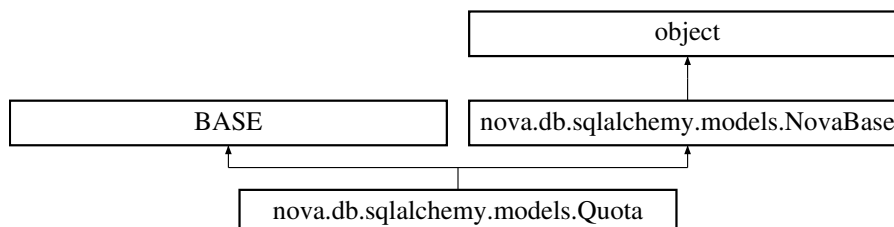
Represents a rule in a security group.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

### 3.49 nova.db.sqlalchemy.models.Quota Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Quota:



#### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **project\_id** = Column(String(255), index=True)
- tuple **resource** = Column(String(255))
- tuple **hard\_limit** = Column(Integer, nullable=True)

## Additional Inherited Members

### 3.49.1 Detailed Description

Represents a single quota override for a project.

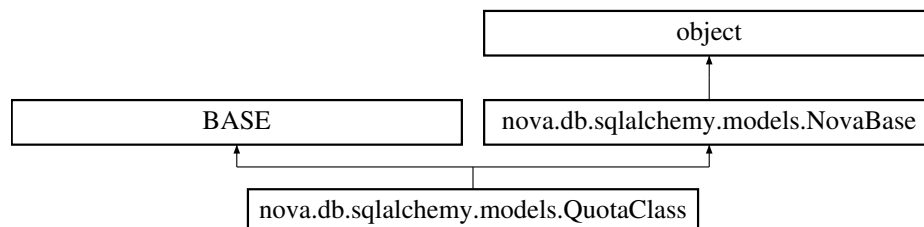
If there is no row for a given project id and resource, then the default for the quota class is used. If there is no row for a given quota class and resource, then the default for the deployment is used. If the row is present but the hard limit is Null, then the resource is unlimited.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.50 nova.db.sqlalchemy.models.QuotaClass Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.QuotaClass:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **class\_name** = Column(String(255), index=True)
- tuple **resource** = Column(String(255))
- tuple **hard\_limit** = Column(Integer, nullable=True)

## Additional Inherited Members

### 3.50.1 Detailed Description

Represents a single quota override for a quota class.

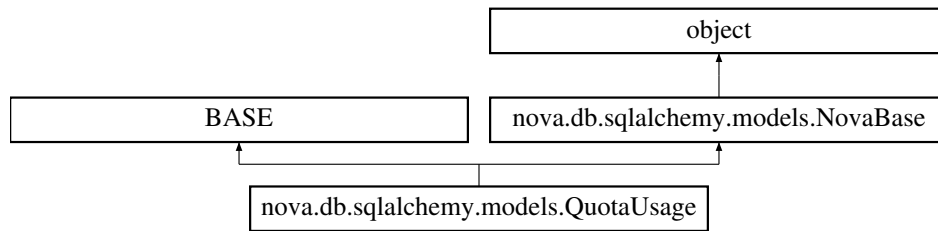
If there is no row for a given quota class and resource, then the default for the deployment is used. If the row is present but the hard limit is Null, then the resource is unlimited.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.51 nova.db.sqlalchemy.models.QuotaUsage Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.QuotaUsage:



### Public Member Functions

- def **total**

### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **project\_id** = Column(String(255), index=True)
- tuple **resource** = Column(String(255))
- tuple **in\_use** = Column(Integer)
- tuple **reserved** = Column(Integer)
- tuple **until\_refresh** = Column(Integer, nullable=True)

### Additional Inherited Members

#### 3.51.1 Detailed Description

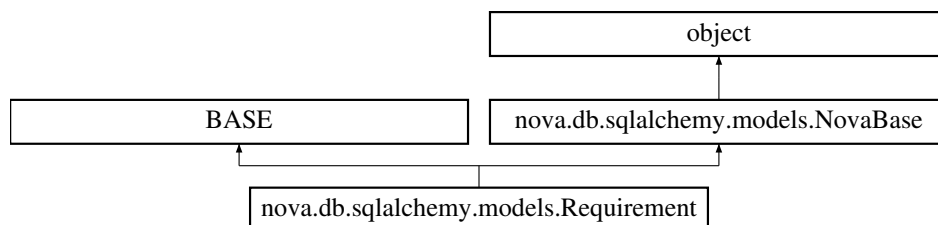
Represents the current usage for a given resource.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.52 nova.db.sqlalchemy.models.Requirement Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Requirement:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, autoincrement=True)
- tuple **requirements** = Column(Text)
- tuple **req\_desc** = Column(Text)
- tuple **req\_options** = Column(Text)

## Additional Inherited Members

### 3.52.1 Detailed Description

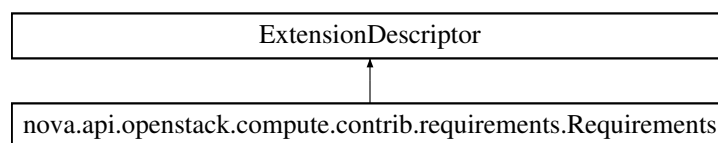
Represents an ACaaS requirement

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.53 nova.api.openstack.compute.contrib.requirements.Requirements Class Reference

Inheritance diagram for nova.api.openstack.compute.contrib.requirements.Requirements:



### Public Member Functions

- def **get\_resources**

### Static Public Attributes

- string **name** = "Requirements"
- string **alias** = "os-requirements"
- tuple **namespace** = ("http://docs.openstack.org/compute/ext/requirements/api/v1.1")
- string **updated** = "2012-10-18T00:00:00+00:00"

### 3.53.1 Detailed Description

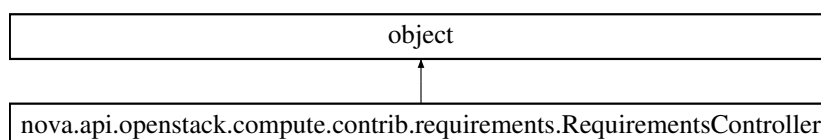
Requirements

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/contrib/requirements.py

## 3.54 nova.api.openstack.compute.contrib.requirements.RequirementsController Class Reference

Inheritance diagram for nova.api.openstack.compute.contrib.requirements.RequirementsController:



## Public Member Functions

- def [index](#)
- def **create\_or\_update**
- def **create**
- def **update**
- def [show](#)
- def [delete](#)

### 3.54.1 Detailed Description

The Requirements API controller for the OpenStack API

### 3.54.2 Member Function Documentation

**3.54.2.1** `def nova.api.openstack.compute.contrib.requirements.RequirementsController.delete ( self, req, id )`

Deletes an existing requirement

**3.54.2.2** `def nova.api.openstack.compute.contrib.requirements.RequirementsController.index ( self, req )`

Returns the list of defined requirements

**3.54.2.3** `def nova.api.openstack.compute.contrib.requirements.RequirementsController.show ( self, req, id )`

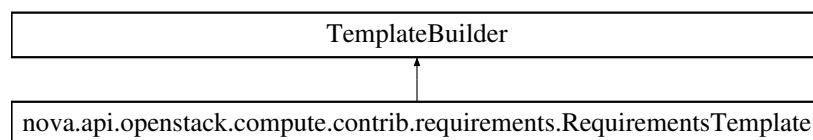
Return a single requirement item

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/contrib/requirements.py

## 3.55 nova.api.openstack.compute.contrib.requirements.RequirementsTemplate Class Reference

Inheritance diagram for nova.api.openstack.compute.contrib.requirements.RequirementsTemplate:



## Public Member Functions

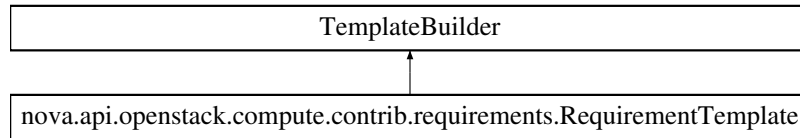
- def **construct**

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/contrib/requirements.py

### 3.56 nova.api.openstack.compute.contrib.requirements.RequirementTemplate Class Reference

Inheritance diagram for nova.api.openstack.compute.contrib.requirements.RequirementTemplate:



#### Public Member Functions

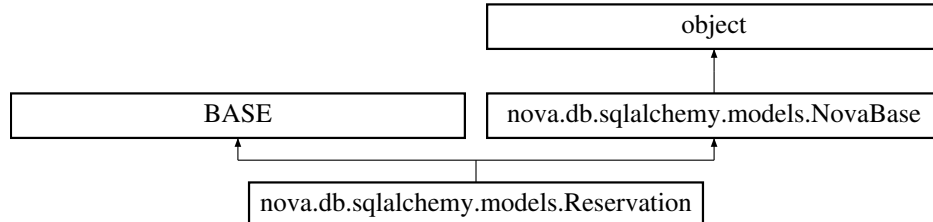
- def **construct**

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/contrib/requirements.py

### 3.57 nova.db.sqlalchemy.models.Reservation Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Reservation:



#### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **uuid** = Column(String(36), nullable=False)
- tuple **usage\_id** = Column(Integer, ForeignKey('quota\_usages.id'), nullable=False)
- tuple **project\_id** = Column(String(255), index=True)
- tuple **resource** = Column(String(255))
- tuple **delta** = Column(Integer)
- tuple **expire** = Column(DateTime, nullable=False)
- tuple **usage**

#### Additional Inherited Members

##### 3.57.1 Detailed Description

Represents a resource reservation for quotas.

### 3.57.2 Member Data Documentation

#### 3.57.2.1 tuple nova.db.sqlalchemy.models.Reservation.usage [static]

**Initial value:**

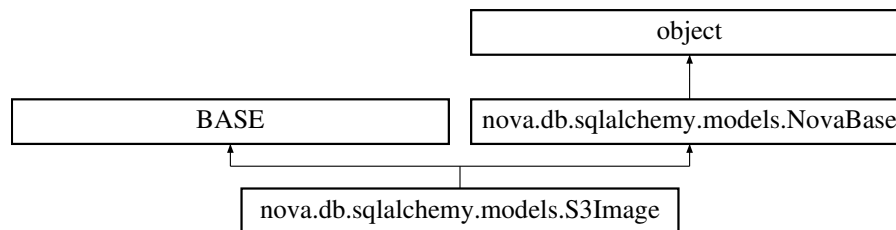
```
1 = relationship(
2     "QuotaUsage",
3     foreign_keys=usage_id,
4     primaryjoin='and_(Reservation.usage_id == QuotaUsage.id,
5                    'QuotaUsage.deleted == False)')
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.58 nova.db.sqlalchemy.models.S3Image Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.S3Image:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, nullable=False, autoincrement=True)
- tuple **uuid** = Column(String(36), nullable=False)

### Additional Inherited Members

#### 3.58.1 Detailed Description

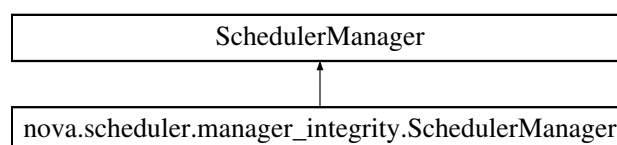
Compatibility layer for the S3 image service talking to Glance

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.59 nova.scheduler.manager\_integrity.SchedulerManager Class Reference

Inheritance diagram for nova.scheduler.manager\_integrity.SchedulerManager:



## Public Member Functions

- `def __init__`
- `def periodic_tasks`

### 3.59.1 Detailed Description

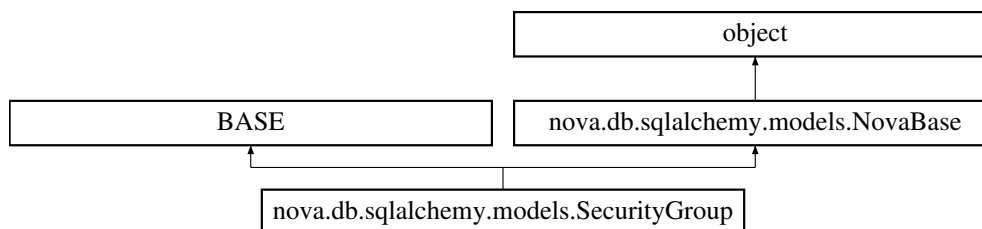
inject a periodic\_task into schedule for building integrity cache

The documentation for this class was generated from the following file:

- `nova/scheduler/manager_integrity.py`

## 3.60 nova.db.sqlalchemy.models.SecurityGroup Class Reference

Inheritance diagram for `nova.db.sqlalchemy.models.SecurityGroup`:



## Static Public Attributes

- tuple **id** = `Column(Integer, primary_key=True)`
- tuple **name** = `Column(String(255))`
- tuple **description** = `Column(String(255))`
- tuple **user\_id** = `Column(String(255))`
- tuple **project\_id** = `Column(String(255))`
- tuple **instances**

## Additional Inherited Members

### 3.60.1 Detailed Description

Represents a security group.

### 3.60.2 Member Data Documentation

#### 3.60.2.1 tuple `nova.db.sqlalchemy.models.SecurityGroup.instances` [static]

Initial value:

```

1 = relationship(Instance,
2                 secondary="security_group_instance_association",
3                 primaryjoin='and_('
4                 'SecurityGroup.id == ',
5                 'SecurityGroupInstanceAssociation.security_group_id,',
6                 'SecurityGroupInstanceAssociation.deleted == False,',
7                 'SecurityGroup.deleted == False)',
8                 secondaryjoin='and_('

```



```

9         'SecurityGroupInstanceAssociation.instance_uuid == Instance.uuid,'
10         # (anthony) the condition below shouldn't be necessary now that the
11         # association is being marked as deleted. However, removing this
12         # may cause existing deployments to choke, so I'm leaving it
13         'Instance.deleted == False)',
14         backref='security_groups')

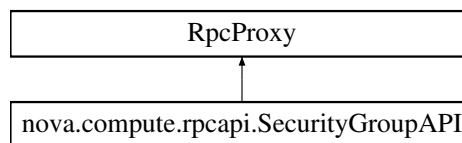
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.61 nova.compute.rpcapi.SecurityGroupAPI Class Reference

Inheritance diagram for nova.compute.rpcapi.SecurityGroupAPI:



### Public Member Functions

- def **\_\_init\_\_**
- def **refresh\_security\_group\_rules**
- def **refresh\_security\_group\_members**
- def **refresh\_instance\_security\_rules**

### Static Public Attributes

- string **BASE\_RPC\_API\_VERSION** = '2.0'

#### 3.61.1 Detailed Description

Client side of the security group rpc API.

API version history:

```

1.0 - Initial version.
1.41 - Adds refresh_instance_security_rules()
2.0 - Remove 1.x backwards compat

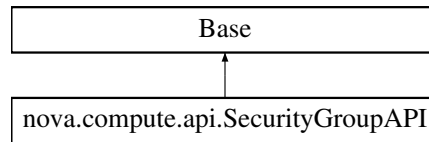
```

The documentation for this class was generated from the following file:

- nova/compute/rpcapi.py

## 3.62 nova.compute.api.SecurityGroupAPI Class Reference

Inheritance diagram for nova.compute.api.SecurityGroupAPI:



## Public Member Functions

- def **\_\_init\_\_**
- def [validate\\_property](#)
- def [ensure\\_default](#)
- def **create**
- def **get**
- def **list**
- def **destroy**
- def [is\\_associated\\_with\\_server](#)
- def [add\\_to\\_instance](#)
- def [remove\\_from\\_instance](#)
- def **trigger\_handler**
- def [trigger\\_rules\\_refresh](#)
- def [trigger\\_members\\_refresh](#)
- def **parse\_cidr**
- def [rule\\_exists](#)
- def **get\_rule**
- def **add\_rules**
- def **remove\_rules**

## Static Public Member Functions

- def **new\_group\_ingress\_rule**
- def **new\_cidr\_ingress\_rule**
- def **raise\_invalid\_property**
- def **raise\_group\_already\_exists**
- def **raise\_invalid\_group**
- def **raise\_invalid\_cidr**
- def **raise\_over\_quota**
- def **raise\_not\_found**

## Public Attributes

- **security\_group\_rpcapi**
- **sgh**

### 3.62.1 Detailed Description

Sub-set of the Compute API related to managing security groups and security group rules

### 3.62.2 Member Function Documentation

#### 3.62.2.1 def nova.compute.api.SecurityGroupAPI.add\_to\_instance ( *self*, *context*, *instance*, *security\_group\_name* )

Add security group to the instance

**3.62.2.2 def nova.compute.api.SecurityGroupAPI.ensure\_default ( self, context )**

Ensure that a context has a security group.

Creates a security group for the security context if it does not already exist.

:param context: the security context

**3.62.2.3 def nova.compute.api.SecurityGroupAPI.is\_associated\_with\_server ( self, security\_group, instance\_uuid )**

Check if the security group is already associated with the instance. If Yes, return True.

**3.62.2.4 def nova.compute.api.SecurityGroupAPI.remove\_from\_instance ( self, context, instance, security\_group\_name )**

Remove the security group associated with the instance

**3.62.2.5 def nova.compute.api.SecurityGroupAPI.rule\_exists ( self, security\_group, values )**

Indicates whether the specified rule values are already defined in the given security group.

**3.62.2.6 def nova.compute.api.SecurityGroupAPI.trigger\_members\_refresh ( self, context, group\_ids )**

Called when a security group gains a new or loses a member.

Sends an update request to each compute node for each instance for which this is relevant.

**3.62.2.7 def nova.compute.api.SecurityGroupAPI.trigger\_rules\_refresh ( self, context, id )**

Called when a rule is added to or removed from a security\_group.

**3.62.2.8 def nova.compute.api.SecurityGroupAPI.validate\_property ( self, value, property, allowed )**

Validate given security group property.

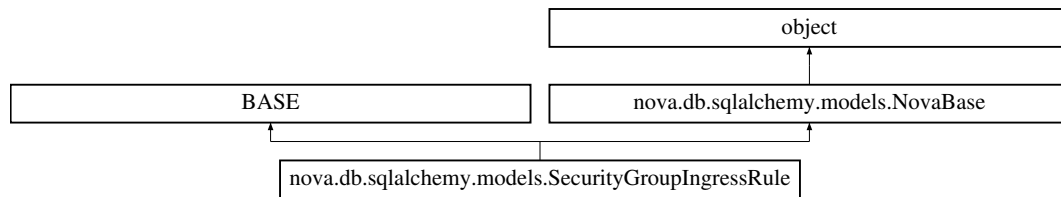
```
:param value:      the value to validate, as a string or unicode
:param property:   the property, either 'name' or 'description'
:param allowed:    the range of characters allowed
```

The documentation for this class was generated from the following file:

- nova/compute/api.py

**3.63 nova.db.sqlalchemy.models.SecurityGroupIngressRule Class Reference**

Inheritance diagram for nova.db.sqlalchemy.models.SecurityGroupIngressRule:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **parent\_group\_id** = Column(Integer, ForeignKey('security\_groups.id'))
- tuple **parent\_group**
- tuple **protocol** = Column(String(5))
- tuple **from\_port** = Column(Integer)
- tuple **to\_port** = Column(Integer)
- tuple **cidr** = Column(String(255))
- tuple **group\_id** = Column(Integer, ForeignKey('security\_groups.id'))
- tuple **grantee\_group**

## Additional Inherited Members

### 3.63.1 Detailed Description

Represents a rule in a security group.

### 3.63.2 Member Data Documentation

#### 3.63.2.1 tuple nova.db.sqlalchemy.models.SecurityGroupIngressRule.grantee\_group [static]

**Initial value:**

```

1 = relationship("SecurityGroup",
2               foreign_keys=group_id,
3               primaryjoin='and_('
4               'SecurityGroupIngressRule.group_id == SecurityGroup.id,'
5               'SecurityGroupIngressRule.deleted == False)')
```

#### 3.63.2.2 tuple nova.db.sqlalchemy.models.SecurityGroupIngressRule.parent\_group [static]

**Initial value:**

```

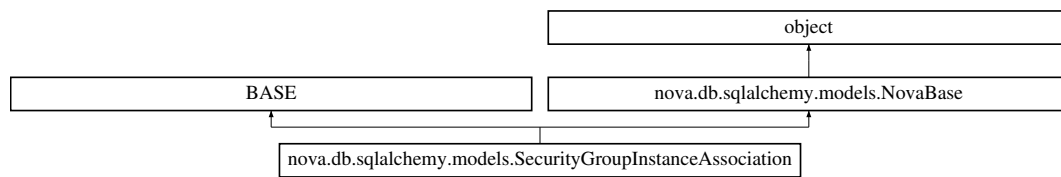
1 = relationship("SecurityGroup", backref="rules",
2               foreign_keys=parent_group_id,
3               primaryjoin='and_('
4               'SecurityGroupIngressRule.parent_group_id == SecurityGroup.id,'
5               'SecurityGroupIngressRule.deleted == False)')
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

### 3.64 nova.db.sqlalchemy.models.SecurityGroupInstanceAssociation Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.SecurityGroupInstanceAssociation:



#### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **security\_group\_id** = Column(Integer, ForeignKey('security\_groups.id'))
- tuple **instance\_uuid** = Column(String(36), ForeignKey('instances.uuid'))

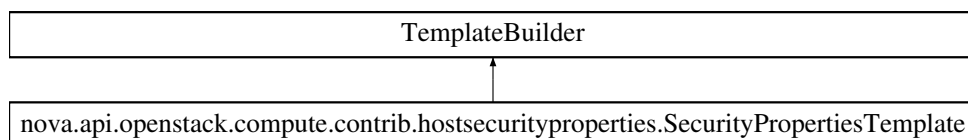
#### Additional Inherited Members

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

### 3.65 nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityProperties-Template Class Reference

Inheritance diagram for nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityPropertiesTemplate:



#### Public Member Functions

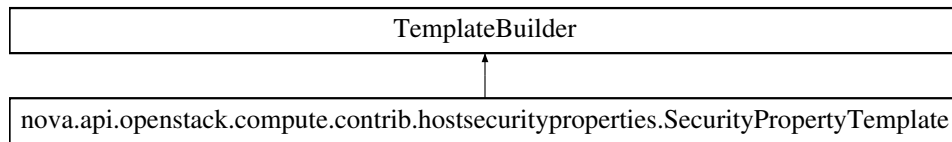
- def **construct**

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/contrib/hostsecurityproperties.py

### 3.66 nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityProperty-Template Class Reference

Inheritance diagram for nova.api.openstack.compute.contrib.hostsecurityproperties.SecurityPropertyTemplate:



### Public Member Functions

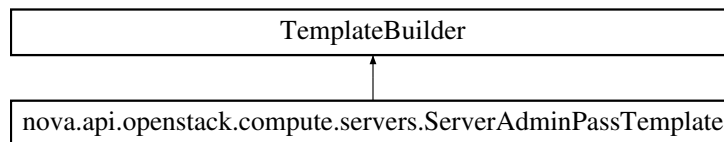
- def **construct**

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/contrib/hostsecurityproperties.py

## 3.67 nova.api.openstack.compute.servers.ServerAdminPassTemplate Class Reference

Inheritance diagram for nova.api.openstack.compute.servers.ServerAdminPassTemplate:



### Public Member Functions

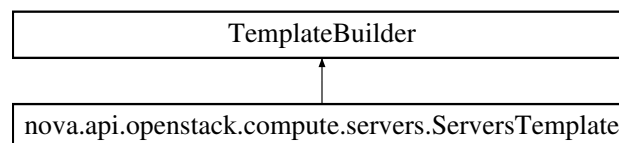
- def **construct**

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/servers.py

## 3.68 nova.api.openstack.compute.servers.ServersTemplate Class Reference

Inheritance diagram for nova.api.openstack.compute.servers.ServersTemplate:



### Public Member Functions

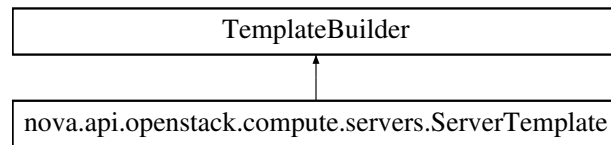
- def **construct**

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/servers.py

### 3.69 nova.api.openstack.compute.servers.ServerTemplate Class Reference

Inheritance diagram for nova.api.openstack.compute.servers.ServerTemplate:



#### Public Member Functions

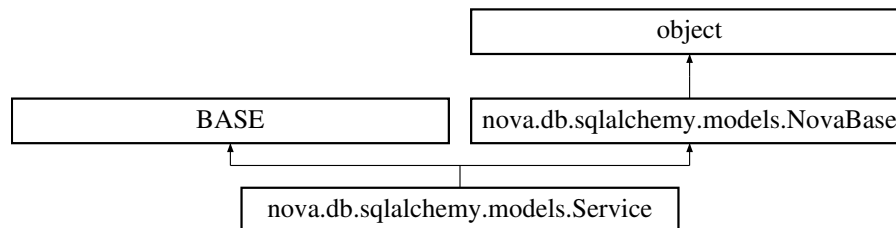
- def **construct**

The documentation for this class was generated from the following file:

- nova/api/openstack/compute/servers.py

### 3.70 nova.db.sqlalchemy.models.Service Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Service:



#### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **host** = Column(String(255))
- tuple **binary** = Column(String(255))
- tuple **topic** = Column(String(255))
- tuple **report\_count** = Column(Integer, nullable=False, default=0)
- tuple **disabled** = Column(Boolean, default=False)
- tuple **availability\_zone** = Column(String(255), default='nova')

#### Additional Inherited Members

##### 3.70.1 Detailed Description

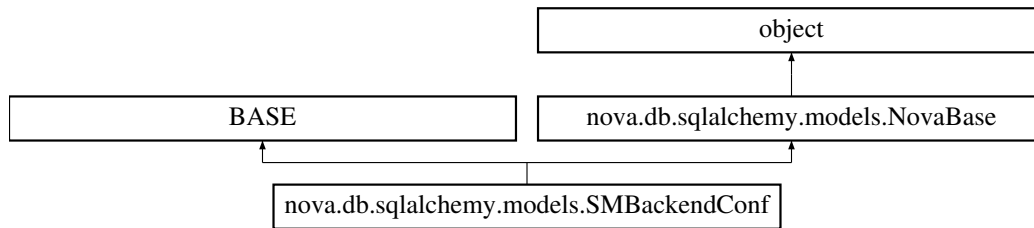
Represents a running service on a host.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

### 3.71 nova.db.sqlalchemy.models.SMBackendConf Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.SMBackendConf:



#### Static Public Attributes

- tuple **id** = Column(Integer(), primary\_key=True)
- tuple **flavor\_id** = Column(Integer, ForeignKey('sm\_flavors.id'), nullable=False)
- tuple **sr\_uuid** = Column(String(255))
- tuple **sr\_type** = Column(String(255))
- tuple **config\_params** = Column(String(2047))

#### Additional Inherited Members

##### 3.71.1 Detailed Description

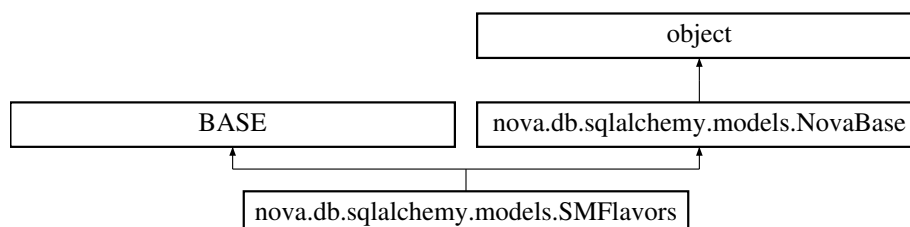
Represents the connection to the backend for SM.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

### 3.72 nova.db.sqlalchemy.models.SMFlavors Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.SMFlavors:



#### Static Public Attributes

- tuple **id** = Column(Integer(), primary\_key=True)
- tuple **label** = Column(String(255))
- tuple **description** = Column(String(255))



## Additional Inherited Members

### 3.72.1 Detailed Description

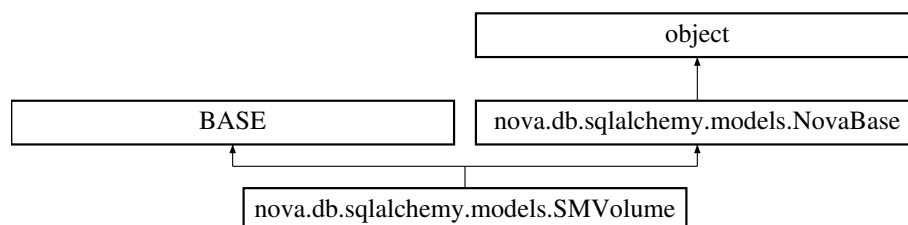
Represents a flavor for SM volumes.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.73 nova.db.sqlalchemy.models.SMVolume Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.SMVolume:



## Static Public Attributes

- tuple **id** = Column(String(36), ForeignKey(Volume.id), primary\_key=True)
- tuple **backend\_id**
- tuple **vdi\_uuid** = Column(String(255))

## Additional Inherited Members

### 3.73.1 Member Data Documentation

3.73.1.1 tuple nova.db.sqlalchemy.models.SMVolume.backend\_id [static]

**Initial value:**

```

1 = Column(Integer, ForeignKey('sm_backend_config.id'),
2         nullable=False)

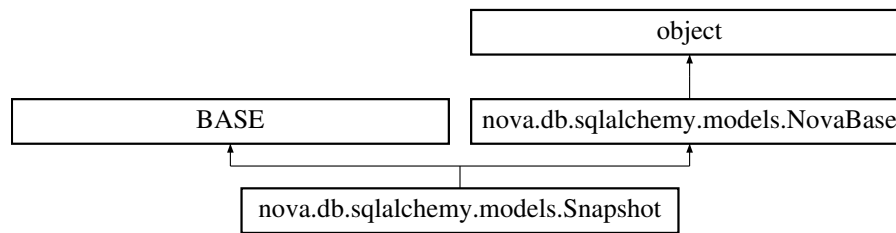
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.74 nova.db.sqlalchemy.models.Snapshot Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Snapshot:



## Public Member Functions

- def **name**
- def **volume\_name**

## Static Public Attributes

- tuple **id** = Column(String(36), primary\_key=True)
- tuple **user\_id** = Column(String(255))
- tuple **project\_id** = Column(String(255))
- tuple **volume\_id** = Column(String(36))
- tuple **status** = Column(String(255))
- tuple **progress** = Column(String(255))
- tuple **volume\_size** = Column(Integer)
- tuple **display\_name** = Column(String(255))
- tuple **display\_description** = Column(String(255))

## Additional Inherited Members

### 3.74.1 Detailed Description

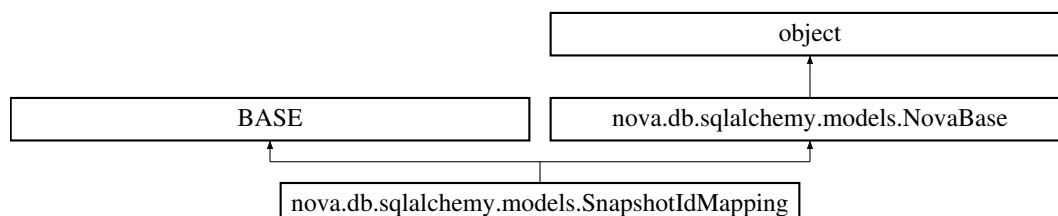
Represents a block storage device that can be attached to a VM.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.75 nova.db.sqlalchemy.models.SnapshotIdMapping Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.SnapshotIdMapping:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, nullable=False, autoincrement=True)
- tuple **uuid** = Column(String(36), nullable=False)

## Additional Inherited Members

### 3.75.1 Detailed Description

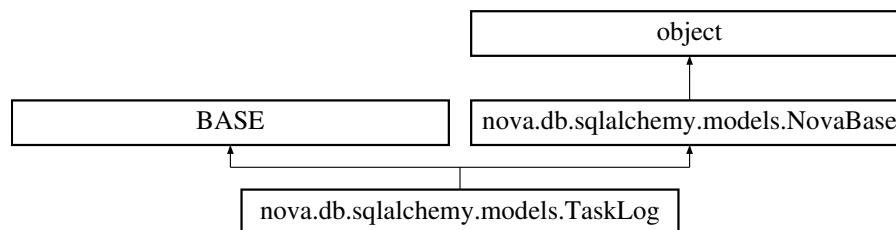
Compatibility layer for the EC2 snapshot service

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.76 nova.db.sqlalchemy.models.TaskLog Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.TaskLog:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, nullable=False, autoincrement=True)
- tuple **task\_name** = Column(String(255), nullable=False)
- tuple **state** = Column(String(255), nullable=False)
- tuple **host** = Column(String(255))
- tuple **period\_beginning** = Column(String(255), default=timeutils.utcnow)
- tuple **period\_ending** = Column(String(255), default=timeutils.utcnow)
- tuple **message** = Column(String(255), nullable=False)
- tuple **task\_items** = Column(Integer(), default=0)
- tuple **errors** = Column(Integer(), default=0)

## Additional Inherited Members

### 3.76.1 Detailed Description

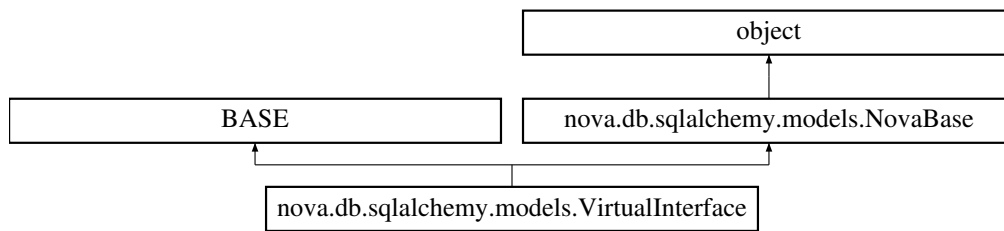
Audit log for background periodic tasks

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.77 nova.db.sqlalchemy.models.VirtualInterface Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.VirtualInterface:



### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **address** = Column(String(255), unique=True)
- tuple **network\_id** = Column(Integer, nullable=False)
- tuple **instance\_uuid** = Column(String(36), nullable=False)
- tuple **uuid** = Column(String(36))

### Additional Inherited Members

#### 3.77.1 Detailed Description

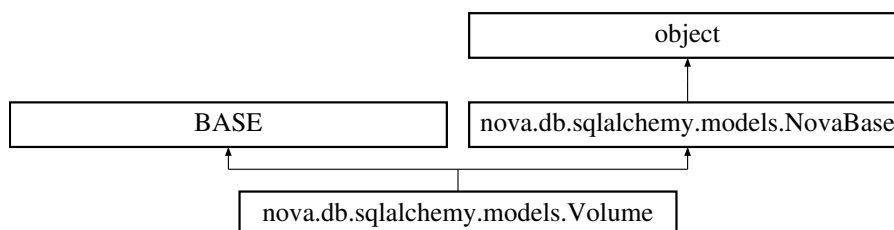
Represents a virtual interface on an instance.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.78 nova.db.sqlalchemy.models.Volume Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.Volume:



### Public Member Functions

- def **name**

### Static Public Attributes

- tuple **id** = Column(String(36), primary\_key=True)
- tuple **ec2\_id** = Column(Integer)
- tuple **user\_id** = Column(String(255))
- tuple **project\_id** = Column(String(255))
- tuple **snapshot\_id** = Column(String(36))
- tuple **host** = Column(String(255))

- tuple **size** = Column(Integer)
- tuple **availability\_zone** = Column(String(255))
- tuple **instance\_uuid** = Column(String(36))
- tuple **mountpoint** = Column(String(255))
- tuple **attach\_time** = Column(DateTime)
- tuple **status** = Column(String(255))
- tuple **attach\_status** = Column(String(255))
- tuple **scheduled\_at** = Column(DateTime)
- tuple **launched\_at** = Column(DateTime)
- tuple **terminated\_at** = Column(DateTime)
- tuple **display\_name** = Column(String(255))
- tuple **display\_description** = Column(String(255))
- tuple **provider\_location** = Column(String(255))
- tuple **provider\_auth** = Column(String(255))
- tuple **volume\_type\_id** = Column(Integer)

## Additional Inherited Members

### 3.78.1 Detailed Description

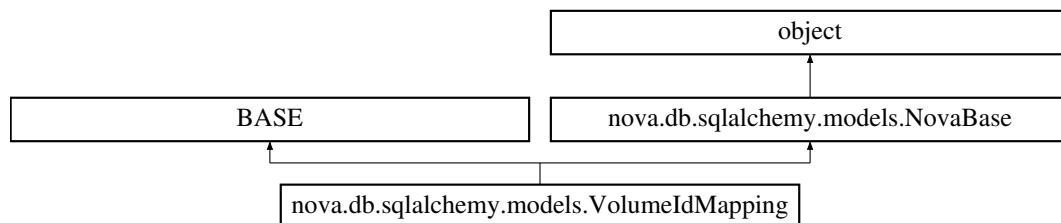
Represents a block storage device that can be attached to a VM.

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.79 nova.db.sqlalchemy.models.VolumeIdMapping Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.VolumeIdMapping:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True, nullable=False, autoincrement=True)
- tuple **uuid** = Column(String(36), nullable=False)

## Additional Inherited Members

### 3.79.1 Detailed Description

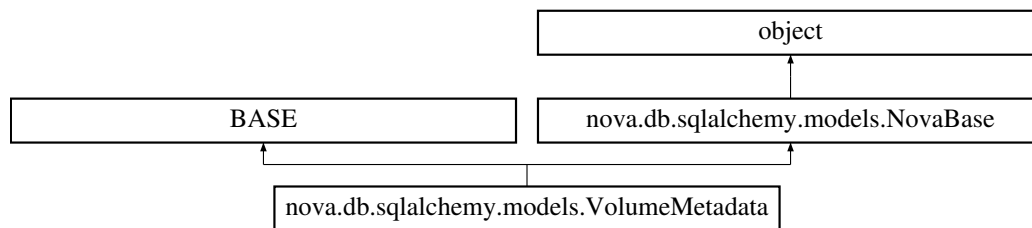
Compatibility layer for the EC2 volume service

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

### 3.80 nova.db.sqlalchemy.models.VolumeMetadata Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.VolumeMetadata:



#### Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **key** = Column(String(255))
- tuple **value** = Column(String(255))
- tuple **volume\_id** = Column(String(36), ForeignKey('volumes.id'), nullable=False)
- tuple **volume**

#### Additional Inherited Members

##### 3.80.1 Detailed Description

Represents a metadata key/value pair for a volume

##### 3.80.2 Member Data Documentation

###### 3.80.2.1 tuple nova.db.sqlalchemy.models.VolumeMetadata.volume [static]

Initial value:

```

1 = relationship(Volume, backref="volume_metadata",
2                 foreign_keys=volume_id,
3                 primaryjoin='and_('
4                     'VolumeMetadata.volume_id == Volume.id,'
5                     'VolumeMetadata.deleted == False)')

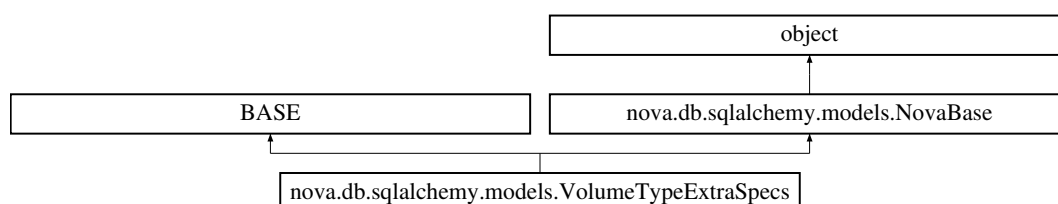
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

### 3.81 nova.db.sqlalchemy.models.VolumeTypeExtraSpecs Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.VolumeTypeExtraSpecs:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **key** = Column(String(255))
- tuple **value** = Column(String(255))
- tuple **volume\_type\_id**
- tuple **volume\_type**

## Additional Inherited Members

### 3.81.1 Detailed Description

Represents additional specs as key/value pairs for a volume\_type

### 3.81.2 Member Data Documentation

3.81.2.1 tuple nova.db.sqlalchemy.models.VolumeTypeExtraSpecs.volume\_type [static]

Initial value:

```
1 = relationship(VolumeTypes, backref="extra_specs",
2               foreign_keys=volume_type_id,
3               primaryjoin='and_(
4               'VolumeTypeExtraSpecs.volume_type_id == VolumeTypes.id,'
5               'VolumeTypeExtraSpecs.deleted == False)')
```

3.81.2.2 tuple nova.db.sqlalchemy.models.VolumeTypeExtraSpecs.volume\_type\_id [static]

Initial value:

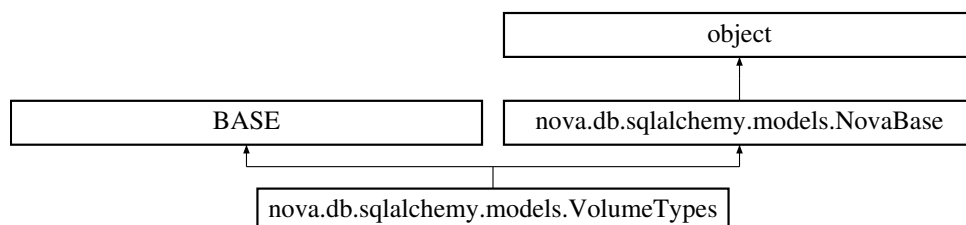
```
1 = Column(Integer, ForeignKey('volume_types.id'),
2           nullable=False)
```

The documentation for this class was generated from the following file:

- nova/db/sqlalchemy/models.py

## 3.82 nova.db.sqlalchemy.models.VolumeTypes Class Reference

Inheritance diagram for nova.db.sqlalchemy.models.VolumeTypes:



## Static Public Attributes

- tuple **id** = Column(Integer, primary\_key=True)
- tuple **name** = Column(String(255))
- tuple **volumes**

## Additional Inherited Members

### 3.82.1 Detailed Description

Represent possible `volume_types` of volumes offered

### 3.82.2 Member Data Documentation

#### 3.82.2.1 `tuple nova.db.sqlalchemy.models.VolumeTypes.volumes` [static]

**Initial value:**

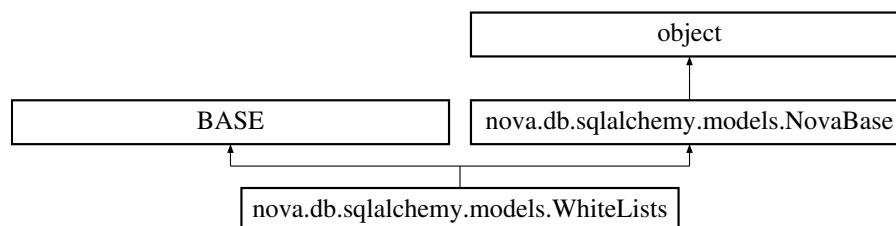
```
1 = relationship(Volume,
2                 backref=backref('volume_type', uselist=False),
3                 foreign_keys=id,
4                 primaryjoin='and_('
5                     'Volume.volume_type_id == VolumeTypes.id, '
6                     'VolumeTypes.deleted == False)')
```

The documentation for this class was generated from the following file:

- `nova/db/sqlalchemy/models.py`

## 3.83 `nova.db.sqlalchemy.models.WhiteLists` Class Reference

Inheritance diagram for `nova.db.sqlalchemy.models.WhiteLists`:



### Static Public Attributes

- tuple **id** = `Column(Integer, primary_key=True, autoincrement=True)`
- tuple **whitelists** = `Column(Text)`

## Additional Inherited Members

### 3.83.1 Detailed Description

Represents an ACaaS white list

The documentation for this class was generated from the following file:

- `nova/db/sqlalchemy/models.py`



# Index

- `__init__`
  - `nova::compute::manager::ComputeManager`, 27
- `add_aggregate_host`
  - `nova::compute::manager::ComputeManager`, 27
  - `nova::compute::rpcapi::ComputeAPI`, 24
- `add_fixed_ip`
  - `nova::compute::api::API`, 14
- `add_fixed_ip_to_instance`
  - `nova::compute::manager::ComputeManager`, 27
- `add_host_to_aggregate`
  - `nova::compute::api::AggregateAPI`, 10
- `add_to_instance`
  - `nova::compute::api::SecurityGroupAPI`, 70
- `attach_volume`
  - `nova::compute::api::API`, 14
  - `nova::compute::manager::ComputeManager`, 27
- `B64_REGEX`
  - `nova::api::openstack::compute::servers::Controller`, 38
- `backend_id`
  - `nova::db::sqlalchemy::models::SMVolume`, 77
- `backup`
  - `nova::compute::api::API`, 14
- `change_instance_metadata`
  - `nova::compute::manager::ComputeManager`, 27
- `check_can_live_migrate_destination`
  - `nova::compute::manager::ComputeManager`, 27
- `check_can_live_migrate_source`
  - `nova::compute::manager::ComputeManager`, 27
- `confirm_resize`
  - `nova::compute::api::API`, 14
  - `nova::compute::manager::ComputeManager`, 28
- `create`
  - `nova::api::openstack::compute::servers::Controller`, 38
  - `nova::compute::api::API`, 14
- `create_aggregate`
  - `nova::compute::api::AggregateAPI`, 10
- `create_db_entry_for_new_instance`
  - `nova::compute::api::API`, 14
- `create_key_pair`
  - `nova::compute::api::KeypairAPI`, 55
- `default`
  - `nova::api::openstack::compute::servers::Create-Deserializer`, 39
- `delete`
  - `nova::api::openstack::compute::contrib::hostsecurityproperties-::HostSecurityPropertiesController`, 44
  - `nova::api::openstack::compute::contrib::requirements-::RequirementsController`, 65
  - `nova::api::openstack::compute::servers::Controller`, 38
  - `nova::compute::api::API`, 14
  - `nova::db::sqlalchemy::models::NovaBase`, 60
- `delete_aggregate`
  - `nova::compute::api::AggregateAPI`, 10
- `delete_instance_metadata`
  - `nova::compute::api::API`, 15
- `delete_key_pair`
  - `nova::compute::api::KeypairAPI`, 55
- `detach_volume`
  - `nova::compute::api::API`, 15
  - `nova::compute::manager::ComputeManager`, 28
- `detail`
  - `nova::api::openstack::compute::servers::Controller`, 38
- `ensure_default`
  - `nova::compute::api::SecurityGroupAPI`, 70
- `finish_resize`
  - `nova::compute::manager::ComputeManager`, 28
- `finish_revert_resize`
  - `nova::compute::manager::ComputeManager`, 28
- `force_delete`
  - `nova::compute::api::API`, 15
- `get`
  - `nova::compute::api::API`, 15
- `get_active_by_window`
  - `nova::compute::api::API`, 15
- `get_aggregate`
  - `nova::compute::api::AggregateAPI`, 10
- `get_aggregate_list`
  - `nova::compute::api::AggregateAPI`, 10
- `get_all`
  - `nova::compute::api::API`, 15
- `get_console_output`
  - `nova::compute::api::API`, 15
  - `nova::compute::manager::ComputeManager`, 28
- `get_console_topic`
  - `nova::compute::manager::ComputeManager`, 28
- `get_diagnostics`
  - `nova::compute::api::API`, 15
  - `nova::compute::manager::ComputeManager`, 28
- `get_host_uptime`

- nova::compute::api::HostAPI, 42
  - nova::compute::manager::ComputeManager, 28
- get\_instance\_bdms
  - nova::compute::api::API, 15
- get\_instance\_faults
  - nova::compute::api::API, 16
- get\_instance\_metadata
  - nova::compute::api::API, 16
- get\_instance\_type
  - nova::compute::api::API, 16
- get\_key\_pair
  - nova::compute::api::KeypairAPI, 55
- get\_key\_pairs
  - nova::compute::api::KeypairAPI, 56
- get\_lock
  - nova::compute::api::API, 16
- get\_vnc\_console
  - nova::compute::api::API, 16
  - nova::compute::manager::ComputeManager, 29
- grantee\_group
  - nova::db::sqlalchemy::models::SecurityGroup-IngressRule, 72
- host\_maintenance\_mode
  - nova::compute::manager::ComputeManager, 29
  - nova::compute::rpcapi::ComputeAPI, 24
- host\_passes
  - nova::scheduler::filters::acaas\_filter::ACaaSFilter, 7
- host\_power\_action
  - nova::compute::api::HostAPI, 42
  - nova::compute::manager::ComputeManager, 29
- import\_key\_pair
  - nova::compute::api::KeypairAPI, 56
- index
  - nova::api::openstack::compute::contrib::hostsecurityproperties::HostSecurityPropertiesController, 44
  - nova::api::openstack::compute::contrib::requirements::RequirementsController, 65
  - nova::api::openstack::compute::servers::Controller, 38
- init\_host
  - nova::compute::manager::ComputeManager, 29
- inject\_file
  - nova::compute::api::API, 16
  - nova::compute::manager::ComputeManager, 29
- inject\_network\_info
  - nova::compute::api::API, 16
  - nova::compute::manager::ComputeManager, 29
- instance
  - nova::db::sqlalchemy::models::BlockDevice-Mapping, 20
  - nova::db::sqlalchemy::models::InstanceInfoCache, 48
  - nova::db::sqlalchemy::models::InstanceMetadata, 49
  - nova::db::sqlalchemy::models::InstanceSystem-Metadata, 50
- instance\_type
  - nova::db::sqlalchemy::models::InstanceTypeExtra-Specs, 51
  - nova::db::sqlalchemy::models::InstanceType-Projects, 52
- instance\_type\_id
  - nova::db::sqlalchemy::models::InstanceTypeExtra-Specs, 51
  - nova::db::sqlalchemy::models::InstanceType-Projects, 52
- instance\_uuid
  - nova::db::sqlalchemy::models::BlockDevice-Mapping, 20
  - nova::db::sqlalchemy::models::InstanceFault, 47
  - nova::db::sqlalchemy::models::InstanceInfoCache, 48
  - nova::db::sqlalchemy::models::InstanceMetadata, 49
  - nova::db::sqlalchemy::models::InstanceSystem-Metadata, 50
  - nova::db::sqlalchemy::models::Migration, 57
- instances
  - nova::db::sqlalchemy::models::InstanceTypes, 53
  - nova::db::sqlalchemy::models::SecurityGroup, 68
- is\_associated\_with\_server
  - nova::compute::api::SecurityGroupAPI, 71
- iteritems
  - nova::db::sqlalchemy::models::NovaBase, 60
- live\_migrate
  - nova::compute::api::API, 16
- live\_migration
  - nova::compute::manager::ComputeManager, 29
- lock
  - nova::compute::api::API, 16
- properties
  - nova::api::openstack::compute::contrib::hostsecurityproperties::Hostsecurityproperties, 43
  - nova.api.openstack.compute.contrib.hostsecurityproperties.-HostSecurityPropertiesController, 43
  - nova.api.openstack.compute.contrib.hostsecurityproperties.-Hostsecurityproperties, 43
  - nova.api.openstack.compute.contrib.hostsecurityproperties.-SecurityPropertiesTemplate, 73
  - nova.api.openstack.compute.contrib.hostsecurityproperties.-SecurityPropertyTemplate, 73
  - nova.api.openstack.compute.contrib.requirements.-RequirementTemplate, 66
  - nova.api.openstack.compute.contrib.requirements.-Requirements, 64
  - nova.api.openstack.compute.contrib.requirements.-RequirementsController, 64
  - nova.api.openstack.compute.contrib.requirements.-RequirementsTemplate, 65
  - nova.api.openstack.compute.servers.ActionDeserializer, 7
  - nova.api.openstack.compute.servers.CommonDeserializer, 21

- nova.api.openstack.compute.servers.Controller, 37
- nova.api.openstack.compute.servers.CreateDeserializer, 39
- nova.api.openstack.compute.servers.MinimalServersTemplate, 57
- nova.api.openstack.compute.servers.ServerAdminPassTemplate, 74
- nova.api.openstack.compute.servers.ServerTemplate, 75
- nova.api.openstack.compute.servers.ServersTemplate, 74
- nova.compute.api.API, 12
- nova.compute.api.AggregateAPI, 9
- nova.compute.api.HostAPI, 42
- nova.compute.api.KeypairAPI, 55
- nova.compute.api.SecurityGroupAPI, 69
- nova.compute.manager.ComputeManager, 25
- nova.compute.rpcapi.ComputeAPI, 22
- nova.compute.rpcapi.SecurityGroupAPI, 69
- nova.db.api.NoMoreNetworks, 58
- nova.db.api.NoMoreTargets, 59
- nova.db.sqlalchemy.api.Constraint, 37
- nova.db.sqlalchemy.api.EqualityCondition, 40
- nova.db.sqlalchemy.api.InequalityCondition, 44
- nova.db.sqlalchemy.models.AgentBuild, 8
- nova.db.sqlalchemy.models.Aggregate, 9
- nova.db.sqlalchemy.models.AggregateHost, 11
- nova.db.sqlalchemy.models.AggregateMetadata, 11
- nova.db.sqlalchemy.models.BandwidthUsage, 19
- nova.db.sqlalchemy.models.BlockDeviceMapping, 20
- nova.db.sqlalchemy.models.Certificate, 21
- nova.db.sqlalchemy.models.ComputeNode, 33
- nova.db.sqlalchemy.models.ComputeNodeStat, 34
- nova.db.sqlalchemy.models.Console, 35
- nova.db.sqlalchemy.models.ConsolePool, 36
- nova.db.sqlalchemy.models.DNSDomain, 39
- nova.db.sqlalchemy.models.FixedIp, 40
- nova.db.sqlalchemy.models.FloatingIp, 41
- nova.db.sqlalchemy.models.Instance, 45
- nova.db.sqlalchemy.models.InstanceFault, 46
- nova.db.sqlalchemy.models.InstanceIdMapping, 47
- nova.db.sqlalchemy.models.InstanceInfoCache, 47
- nova.db.sqlalchemy.models.InstanceMetadata, 48
- nova.db.sqlalchemy.models.InstanceSystemMetadata, 49
- nova.db.sqlalchemy.models.InstanceTypeExtraSpecs, 51
- nova.db.sqlalchemy.models.InstanceTypeProjects, 51
- nova.db.sqlalchemy.models.InstanceTypes, 52
- nova.db.sqlalchemy.models.IscsiTarget, 53
- nova.db.sqlalchemy.models.KeyPair, 54
- nova.db.sqlalchemy.models.Migration, 56
- nova.db.sqlalchemy.models.Network, 57
- nova.db.sqlalchemy.models.NovaBase, 59
- nova.db.sqlalchemy.models.ProviderFirewallRule, 61
- nova.db.sqlalchemy.models.Quota, 61
- nova.db.sqlalchemy.models.QuotaClass, 62
- nova.db.sqlalchemy.models.QuotaUsage, 62
- nova.db.sqlalchemy.models.Requirement, 63
- nova.db.sqlalchemy.models.Reservation, 66
- nova.db.sqlalchemy.models.S3Image, 67
- nova.db.sqlalchemy.models.SMBBackendConf, 76
- nova.db.sqlalchemy.models.SMFlavors, 76
- nova.db.sqlalchemy.models.SMVolume, 77
- nova.db.sqlalchemy.models.SecurityGroup, 68
- nova.db.sqlalchemy.models.SecurityGroupIngressRule, 71
- nova.db.sqlalchemy.models.SecurityGroupInstanceAssociation, 73
- nova.db.sqlalchemy.models.Service, 75
- nova.db.sqlalchemy.models.Snapshot, 77
- nova.db.sqlalchemy.models.SnapshotIdMapping, 78
- nova.db.sqlalchemy.models.TaskLog, 79
- nova.db.sqlalchemy.models.VirtualInterface, 79
- nova.db.sqlalchemy.models.Volume, 80
- nova.db.sqlalchemy.models.VolumeldMapping, 81
- nova.db.sqlalchemy.models.VolumeMetadata, 82
- nova.db.sqlalchemy.models.VolumeTypeExtraSpecs, 82
- nova.db.sqlalchemy.models.VolumeTypes, 83
- nova.db.sqlalchemy.models.WhiteLists, 84
- nova.scheduler.filters.acaas\_filter.ACaaSFilter, 7
- nova.scheduler.manager\_integrity.SchedulerManager, 67
- nova::api::openstack::compute::contrib::hostsecurityproperties::HostSecurityPropertiesController
  - delete, 44
  - index, 44
  - show, 44
- nova::api::openstack::compute::contrib::hostsecurityproperties::Hostsecurityproperties
  - namespace, 43
- nova::api::openstack::compute::contrib::requirements::RequirementsController
  - delete, 65
  - index, 65
  - show, 65
- nova::api::openstack::compute::servers::Controller
  - B64\_REGEX, 38
  - create, 38
  - delete, 38
  - detail, 38
  - index, 38
  - show, 38
  - update, 38
- nova::api::openstack::compute::servers::CreateDeserializer
  - default, 39
- nova::compute::api::API
  - add\_fixed\_ip, 14
  - attach\_volume, 14
  - backup, 14
  - confirm\_resize, 14
  - create, 14
  - create\_db\_entry\_for\_new\_instance, 14
  - delete, 14
  - delete\_instance\_metadata, 15
  - detach\_volume, 15

- force\_delete, [15](#)
- get, [15](#)
- get\_active\_by\_window, [15](#)
- get\_all, [15](#)
- get\_console\_output, [15](#)
- get\_diagnostics, [15](#)
- get\_instance\_bdms, [15](#)
- get\_instance\_faults, [16](#)
- get\_instance\_metadata, [16](#)
- get\_instance\_type, [16](#)
- get\_lock, [16](#)
- get\_vnc\_console, [16](#)
- inject\_file, [16](#)
- inject\_network\_info, [16](#)
- live\_migrate, [16](#)
- lock, [16](#)
- pause, [16](#)
- reboot, [16](#)
- rebuild, [17](#)
- remove\_fixed\_ip, [17](#)
- rescue, [17](#)
- reset\_network, [17](#)
- resize, [17](#)
- restore, [17](#)
- resume, [17](#)
- revert\_resize, [17](#)
- set\_admin\_password, [17](#)
- snapshot, [17](#)
- snapshot\_volume\_backed, [18](#)
- soft\_delete, [18](#)
- start, [18](#)
- stop, [18](#)
- suspend, [18](#)
- trigger\_provider\_fw\_rules\_refresh, [18](#)
- unlock, [18](#)
- unpause, [18](#)
- unrescue, [18](#)
- update, [19](#)
- update\_instance\_metadata, [19](#)
- nova::compute::api::AggregateAPI
  - add\_host\_to\_aggregate, [10](#)
  - create\_aggregate, [10](#)
  - delete\_aggregate, [10](#)
  - get\_aggregate, [10](#)
  - get\_aggregate\_list, [10](#)
  - remove\_host\_from\_aggregate, [10](#)
  - update\_aggregate, [10](#)
  - update\_aggregate\_metadata, [11](#)
- nova::compute::api::HostAPI
  - get\_host\_uptime, [42](#)
  - host\_power\_action, [42](#)
  - set\_host\_enabled, [42](#)
  - set\_host\_maintenance, [42](#)
- nova::compute::api::KeypairAPI
  - create\_key\_pair, [55](#)
  - delete\_key\_pair, [55](#)
  - get\_key\_pair, [55](#)
  - get\_key\_pairs, [56](#)
- import\_key\_pair, [56](#)
- nova::compute::api::SecurityGroupAPI
  - add\_to\_instance, [70](#)
  - ensure\_default, [70](#)
  - is\_associated\_with\_server, [71](#)
  - remove\_from\_instance, [71](#)
  - rule\_exists, [71](#)
  - trigger\_members\_refresh, [71](#)
  - trigger\_rules\_refresh, [71](#)
  - validate\_property, [71](#)
- nova::compute::manager::ComputeManager
  - \_\_init\_\_, [27](#)
  - add\_aggregate\_host, [27](#)
  - add\_fixed\_ip\_to\_instance, [27](#)
  - attach\_volume, [27](#)
  - change\_instance\_metadata, [27](#)
  - check\_can\_live\_migrate\_destination, [27](#)
  - check\_can\_live\_migrate\_source, [27](#)
  - confirm\_resize, [28](#)
  - detach\_volume, [28](#)
  - finish\_resize, [28](#)
  - finish\_revert\_resize, [28](#)
  - get\_console\_output, [28](#)
  - get\_console\_topic, [28](#)
  - get\_diagnostics, [28](#)
  - get\_host\_uptime, [28](#)
  - get\_vnc\_console, [29](#)
  - host\_maintenance\_mode, [29](#)
  - host\_power\_action, [29](#)
  - init\_host, [29](#)
  - inject\_file, [29](#)
  - inject\_network\_info, [29](#)
  - live\_migration, [29](#)
  - pause\_instance, [29](#)
  - post\_live\_migration\_at\_destination, [29](#)
  - power\_off\_instance, [30](#)
  - power\_on\_instance, [30](#)
  - pre\_live\_migration, [30](#)
  - prep\_resize, [30](#)
  - reboot\_instance, [30](#)
  - rebuild\_instance, [30](#)
  - refresh\_instance\_security\_rules, [30](#)
  - refresh\_provider\_fw\_rules, [31](#)
  - refresh\_security\_group\_members, [31](#)
  - refresh\_security\_group\_rules, [31](#)
  - remove\_aggregate\_host, [31](#)
  - remove\_fixed\_ip\_from\_instance, [31](#)
  - remove\_volume\_connection, [31](#)
  - rescue\_instance, [31](#)
  - reset\_network, [31](#)
  - resize\_instance, [31](#)
  - resume\_instance, [32](#)
  - revert\_resize, [32](#)
  - rollback\_live\_migration\_at\_destination, [32](#)
  - set\_admin\_password, [32](#)
  - set\_host\_enabled, [32](#)
  - snapshot\_instance, [32](#)
  - start\_instance, [32](#)

- stop\_instance, 32
- suspend\_instance, 33
- terminate\_instance, 33
- unpause\_instance, 33
- unrescue\_instance, 33
- update\_available\_resource, 33
- nova::compute::rpcapi::ComputeAPI
  - add\_aggregate\_host, 24
  - host\_maintenance\_mode, 24
  - remove\_aggregate\_host, 25
- nova::db::sqlalchemy::models::BlockDeviceMapping
  - instance, 20
  - instance\_uuid, 20
- nova::db::sqlalchemy::models::ComputeNode
  - service, 34
- nova::db::sqlalchemy::models::ComputeNodeStat
  - primary\_join, 35
  - stats, 35
- nova::db::sqlalchemy::models::InstanceFault
  - instance\_uuid, 47
- nova::db::sqlalchemy::models::InstanceInfoCache
  - instance, 48
  - instance\_uuid, 48
- nova::db::sqlalchemy::models::InstanceMetadata
  - instance, 49
  - instance\_uuid, 49
- nova::db::sqlalchemy::models::InstanceSystemMetadata
  - instance, 50
  - instance\_uuid, 50
  - primary\_join, 50
- nova::db::sqlalchemy::models::InstanceTypeExtraSpecs
  - instance\_type, 51
  - instance\_type\_id, 51
- nova::db::sqlalchemy::models::InstanceTypeProjects
  - instance\_type, 52
  - instance\_type\_id, 52
- nova::db::sqlalchemy::models::InstanceTypes
  - instances, 53
- nova::db::sqlalchemy::models::IscsiTarget
  - volume, 54
- nova::db::sqlalchemy::models::Migration
  - instance\_uuid, 57
- nova::db::sqlalchemy::models::NovaBase
  - delete, 60
  - iteritems, 60
  - save, 60
  - update, 60
- nova::db::sqlalchemy::models::Reservation
  - usage, 67
- nova::db::sqlalchemy::models::SMVolume
  - backend\_id, 77
- nova::db::sqlalchemy::models::SecurityGroup
  - instances, 68
- nova::db::sqlalchemy::models::SecurityGroupIngress-  
Rule
  - grantee\_group, 72
  - parent\_group, 72
- nova::db::sqlalchemy::models::VolumeMetadata
  - volume, 82
- nova::db::sqlalchemy::models::VolumeTypeExtraSpecs
  - volume\_type, 83
  - volume\_type\_id, 83
- nova::db::sqlalchemy::models::VolumeTypes
  - volumes, 84
- nova::scheduler::filters::acaas\_filter::ACaaSFilter
  - host\_passes, 7
- parent\_group
  - nova::db::sqlalchemy::models::SecurityGroup-  
IngressRule, 72
- pause
  - nova::compute::api::API, 16
- pause\_instance
  - nova::compute::manager::ComputeManager, 29
- post\_live\_migration\_at\_destination
  - nova::compute::manager::ComputeManager, 29
- power\_off\_instance
  - nova::compute::manager::ComputeManager, 30
- power\_on\_instance
  - nova::compute::manager::ComputeManager, 30
- pre\_live\_migration
  - nova::compute::manager::ComputeManager, 30
- prep\_resize
  - nova::compute::manager::ComputeManager, 30
- primary\_join
  - nova::db::sqlalchemy::models::ComputeNodeStat,  
35
  - nova::db::sqlalchemy::models::InstanceSystem-  
Metadata, 50
- reboot
  - nova::compute::api::API, 16
- reboot\_instance
  - nova::compute::manager::ComputeManager, 30
- rebuild
  - nova::compute::api::API, 17
- rebuild\_instance
  - nova::compute::manager::ComputeManager, 30
- refresh\_instance\_security\_rules
  - nova::compute::manager::ComputeManager, 30
- refresh\_provider\_fw\_rules
  - nova::compute::manager::ComputeManager, 31
- refresh\_security\_group\_members
  - nova::compute::manager::ComputeManager, 31
- refresh\_security\_group\_rules
  - nova::compute::manager::ComputeManager, 31
- remove\_aggregate\_host
  - nova::compute::manager::ComputeManager, 31
  - nova::compute::rpcapi::ComputeAPI, 25
- remove\_fixed\_ip
  - nova::compute::api::API, 17
- remove\_fixed\_ip\_from\_instance
  - nova::compute::manager::ComputeManager, 31
- remove\_from\_instance
  - nova::compute::api::SecurityGroupAPI, 71
- remove\_host\_from\_aggregate
  - nova::compute::api::AggregateAPI, 10

- remove\_volume\_connection
  - nova::compute::manager::ComputeManager, 31
- rescue
  - nova::compute::api::API, 17
- rescue\_instance
  - nova::compute::manager::ComputeManager, 31
- reset\_network
  - nova::compute::api::API, 17
  - nova::compute::manager::ComputeManager, 31
- resize
  - nova::compute::api::API, 17
- resize\_instance
  - nova::compute::manager::ComputeManager, 31
- restore
  - nova::compute::api::API, 17
- resume
  - nova::compute::api::API, 17
- resume\_instance
  - nova::compute::manager::ComputeManager, 32
- revert\_resize
  - nova::compute::api::API, 17
  - nova::compute::manager::ComputeManager, 32
- rollback\_live\_migration\_at\_destination
  - nova::compute::manager::ComputeManager, 32
- rule\_exists
  - nova::compute::api::SecurityGroupAPI, 71
- save
  - nova::db::sqlalchemy::models::NovaBase, 60
- service
  - nova::db::sqlalchemy::models::ComputeNode, 34
- set\_admin\_password
  - nova::compute::api::API, 17
  - nova::compute::manager::ComputeManager, 32
- set\_host\_enabled
  - nova::compute::api::HostAPI, 42
  - nova::compute::manager::ComputeManager, 32
- set\_host\_maintenance
  - nova::compute::api::HostAPI, 42
- show
  - nova::api::openstack::compute::contrib::hostsecurityproperties::HostSecurityPropertiesController, 44
  - nova::api::openstack::compute::contrib::requirements::RequirementsController, 65
  - nova::api::openstack::compute::servers::Controller, 38
- snapshot
  - nova::compute::api::API, 17
- snapshot\_instance
  - nova::compute::manager::ComputeManager, 32
- snapshot\_volume\_backed
  - nova::compute::api::API, 18
- soft\_delete
  - nova::compute::api::API, 18
- start
  - nova::compute::api::API, 18
- start\_instance
  - nova::compute::manager::ComputeManager, 32
- stats
  - nova::db::sqlalchemy::models::ComputeNodeStat, 35
- stop
  - nova::compute::api::API, 18
- stop\_instance
  - nova::compute::manager::ComputeManager, 32
- suspend
  - nova::compute::api::API, 18
- suspend\_instance
  - nova::compute::manager::ComputeManager, 33
- terminate\_instance
  - nova::compute::manager::ComputeManager, 33
- trigger\_members\_refresh
  - nova::compute::api::SecurityGroupAPI, 71
- trigger\_provider\_fw\_rules\_refresh
  - nova::compute::api::API, 18
- trigger\_rules\_refresh
  - nova::compute::api::SecurityGroupAPI, 71
- unlock
  - nova::compute::api::API, 18
- unpause
  - nova::compute::api::API, 18
- unpause\_instance
  - nova::compute::manager::ComputeManager, 33
- unrescue
  - nova::compute::api::API, 18
- unrescue\_instance
  - nova::compute::manager::ComputeManager, 33
- update
  - nova::api::openstack::compute::servers::Controller, 38
  - nova::compute::api::API, 19
  - nova::db::sqlalchemy::models::NovaBase, 60
- update\_aggregate
  - nova::compute::api::AggregateAPI, 10
- update\_aggregate\_metadata
  - nova::compute::api::AggregateAPI, 11
- update\_available\_resource
  - nova::compute::manager::ComputeManager, 33
- update\_instance\_metadata
  - nova::compute::api::API, 19
- usage
  - nova::db::sqlalchemy::models::Reservation, 67
- validate\_property
  - nova::compute::api::SecurityGroupAPI, 71
- volume
  - nova::db::sqlalchemy::models::IscsiTarget, 54
  - nova::db::sqlalchemy::models::VolumeMetadata, 82
- volume\_type
  - nova::db::sqlalchemy::models::VolumeTypeExtraSpecs, 83
- volume\_type\_id
  - nova::db::sqlalchemy::models::VolumeTypeExtraSpecs, 83
- volumes

nova::db::sqlalchemy::models::VolumeTypes, [84](#)