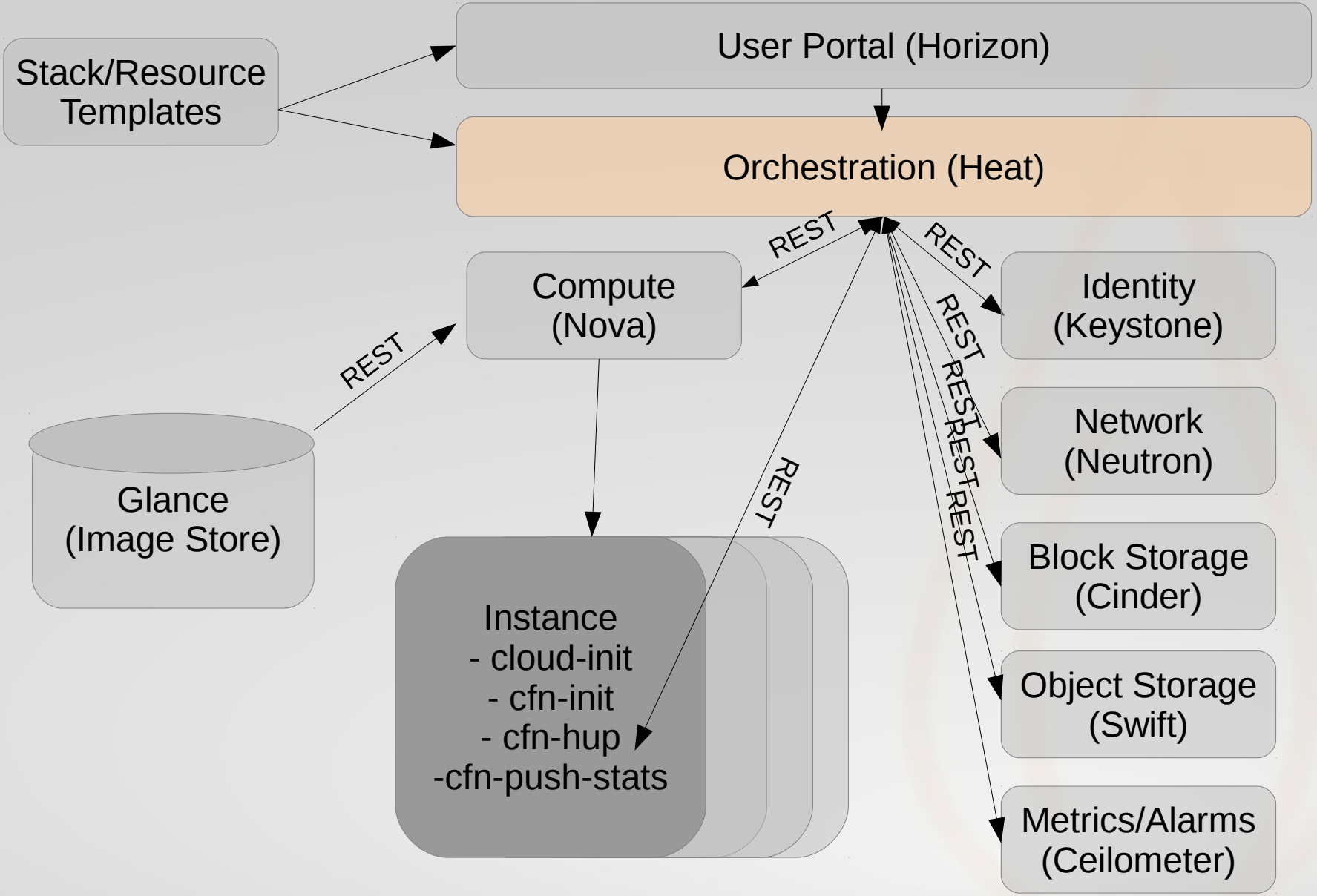
Heat - Orchestration for OpenStack

Overview, Intro to Provider Resources

Steven Hardy (shardy@redhat.com)

3rd October 2013

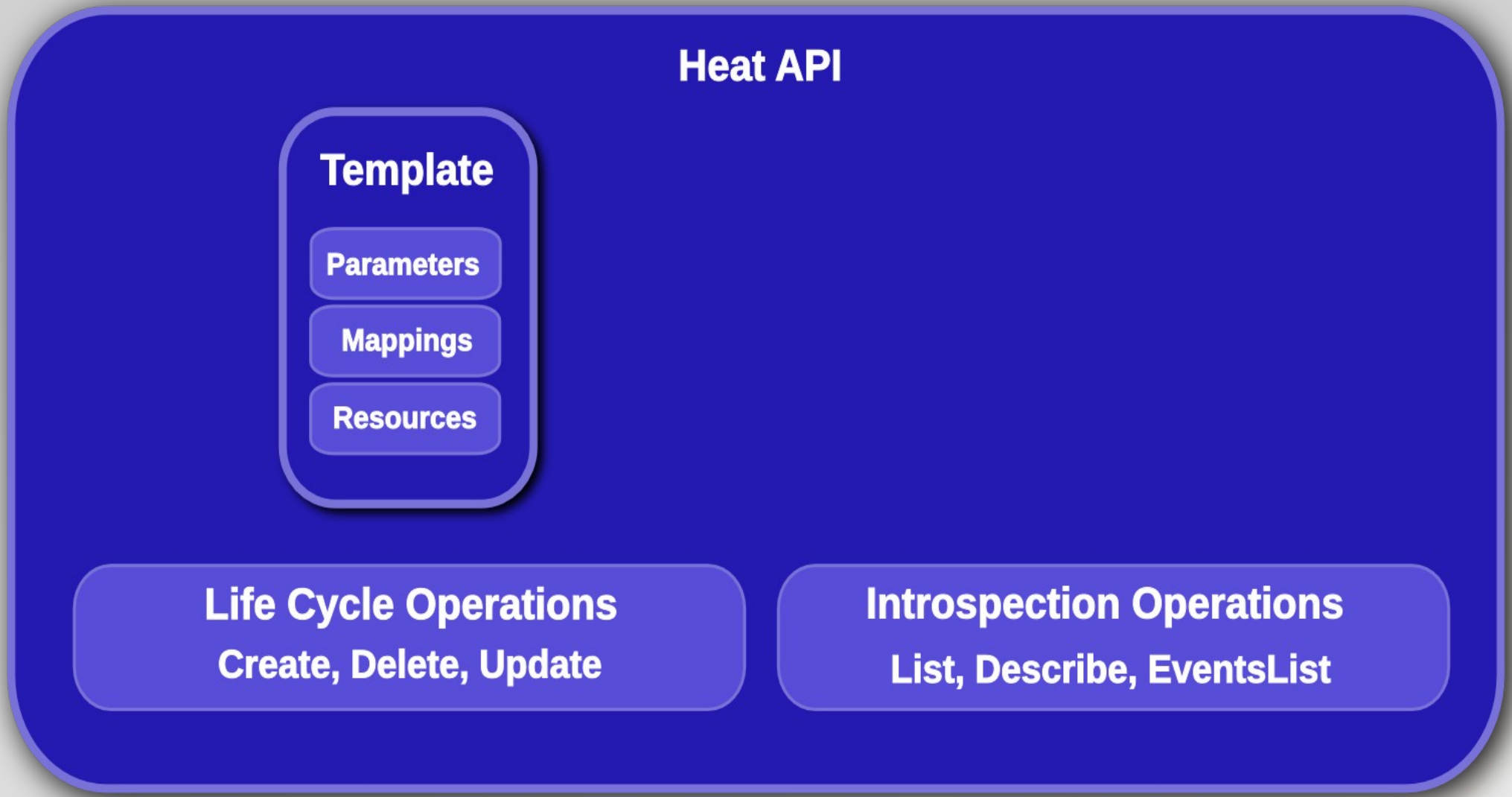
Heat Overview



Havana Features

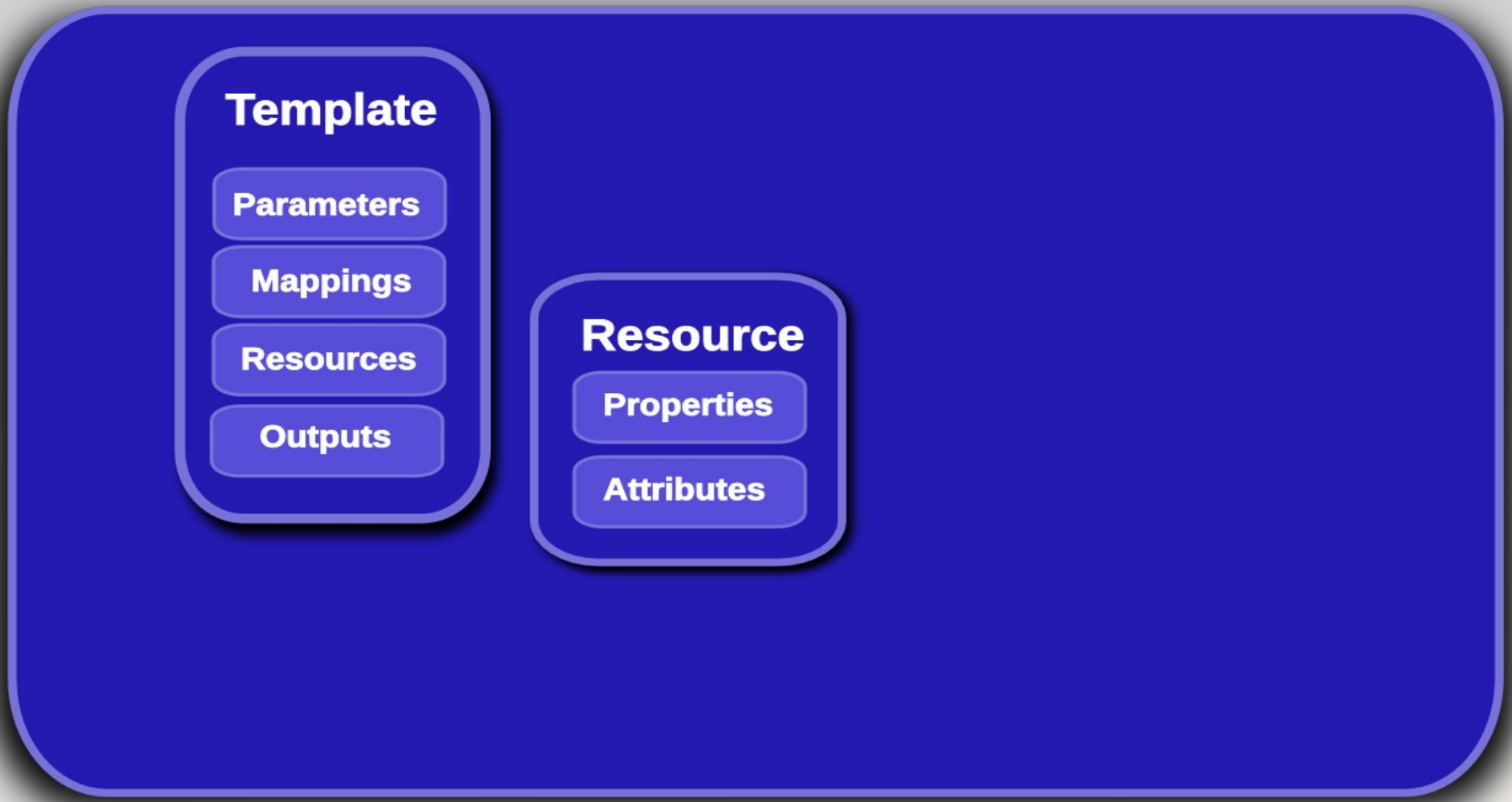
- Concurrent resource operations
- Much improved networking/Neutron support
- Initial support for native template language (HOT)
- “Provider/Environments” abstractions (template-defined resources)
- Ceilometer integration for metrics/monitoring/alarms
- UpdateStack improvements
- Initial integration with keystone trusts functionality
- Many more native resource types
- Horizon (UI) integration
- Stack “actions” (suspend/resume)

Heat Template Overview

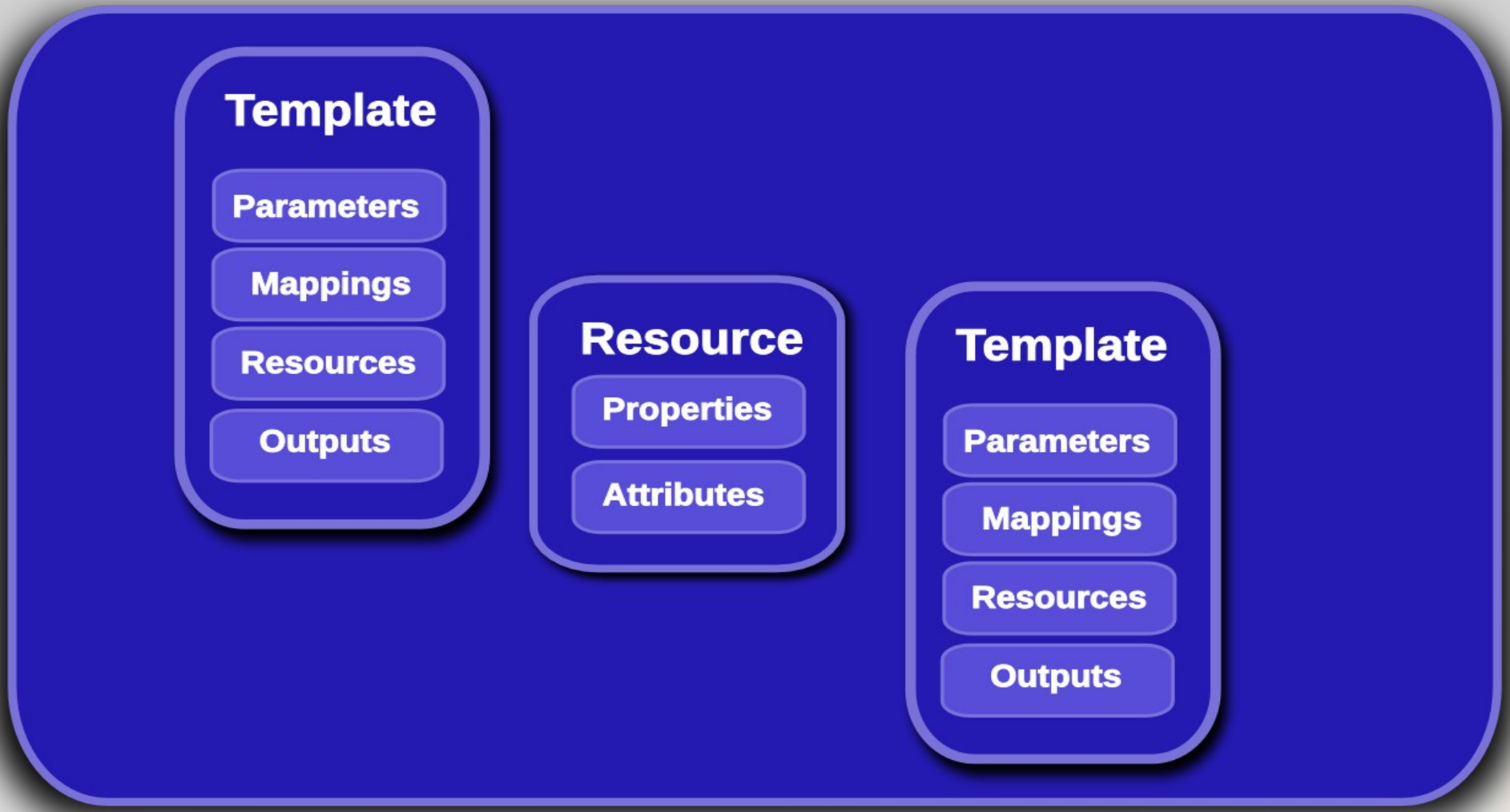


Heat Template Overview

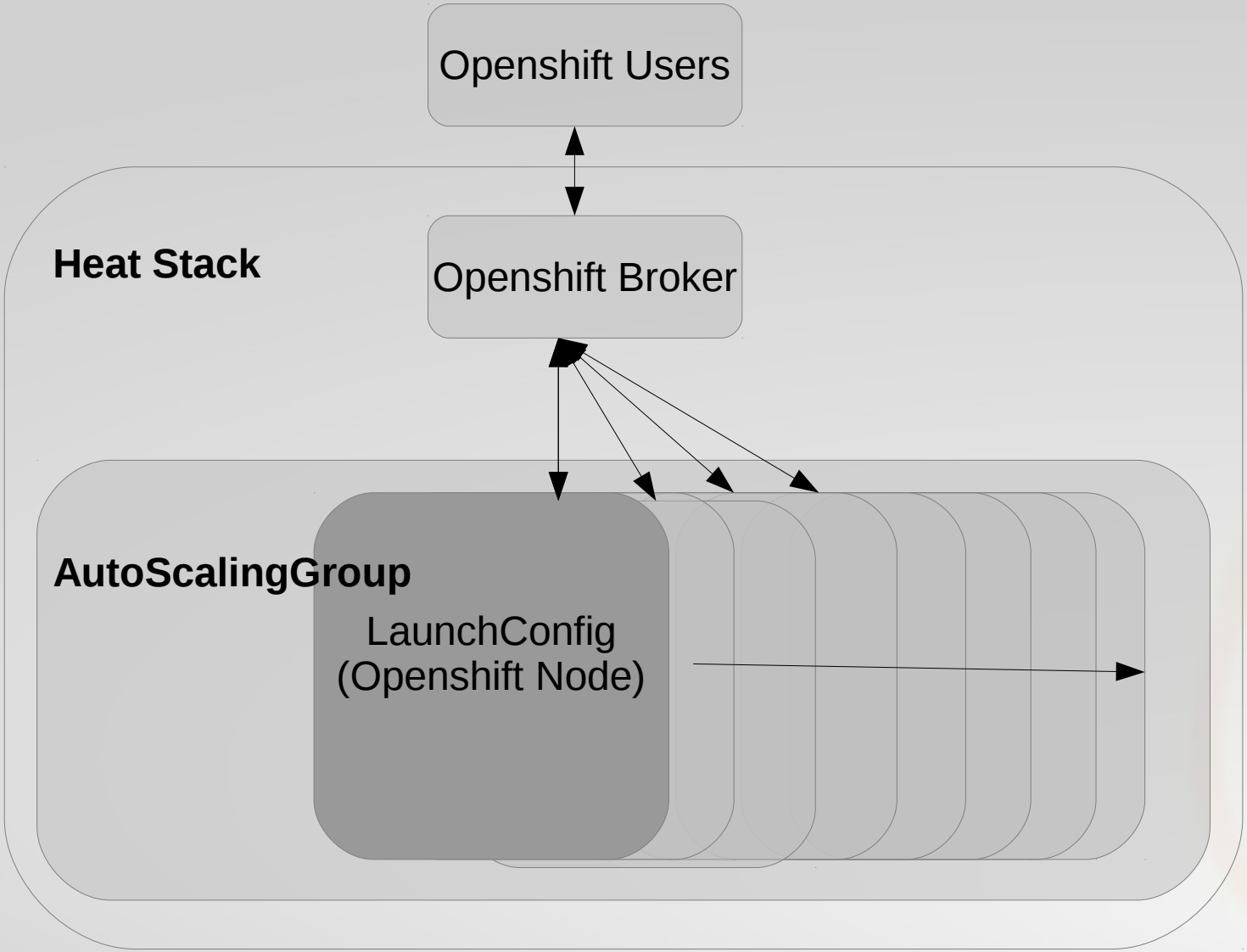
Heat Template Overview



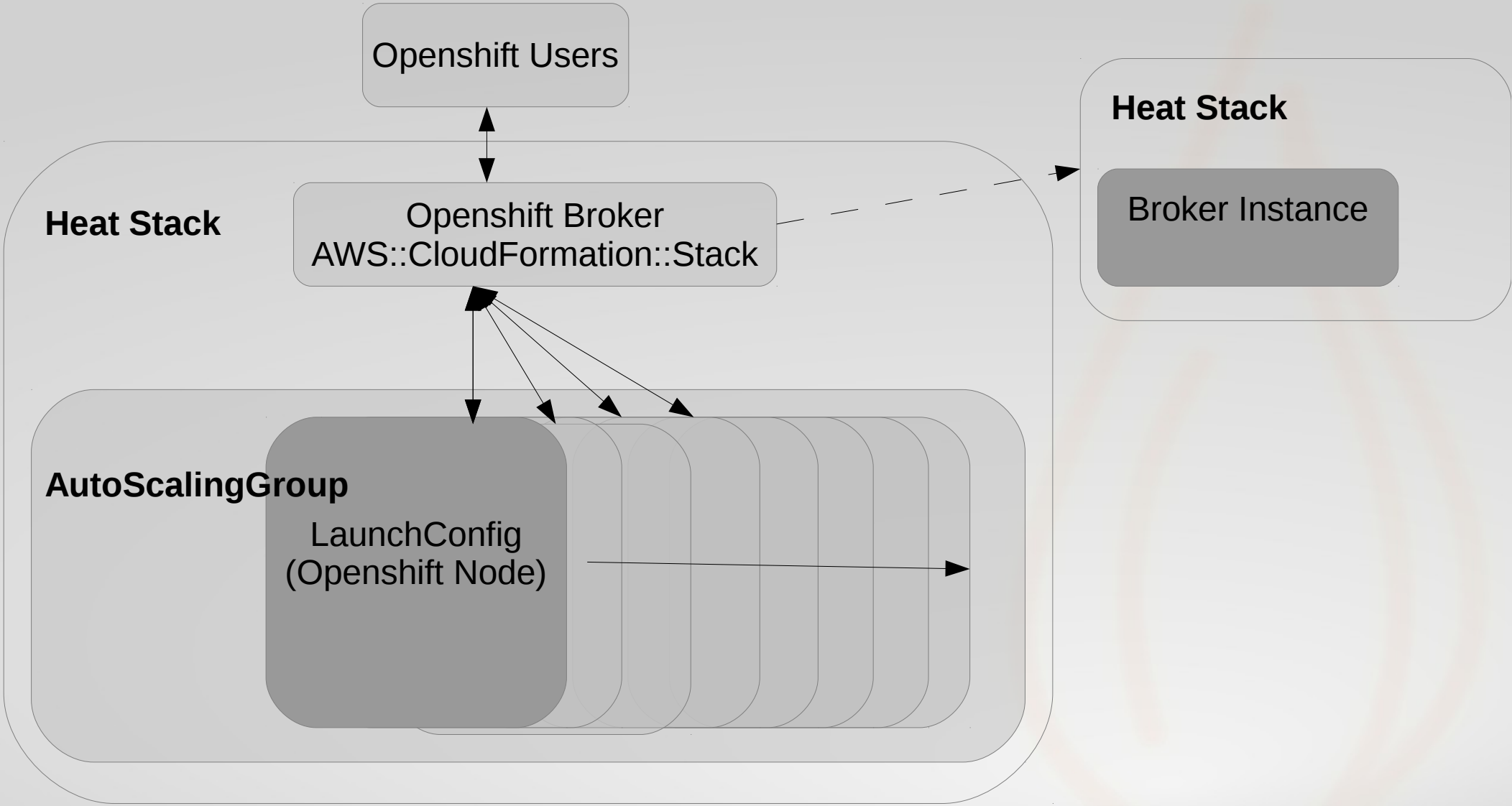
Heat Nested Stack Templates



Openshift Example



Openshift Example

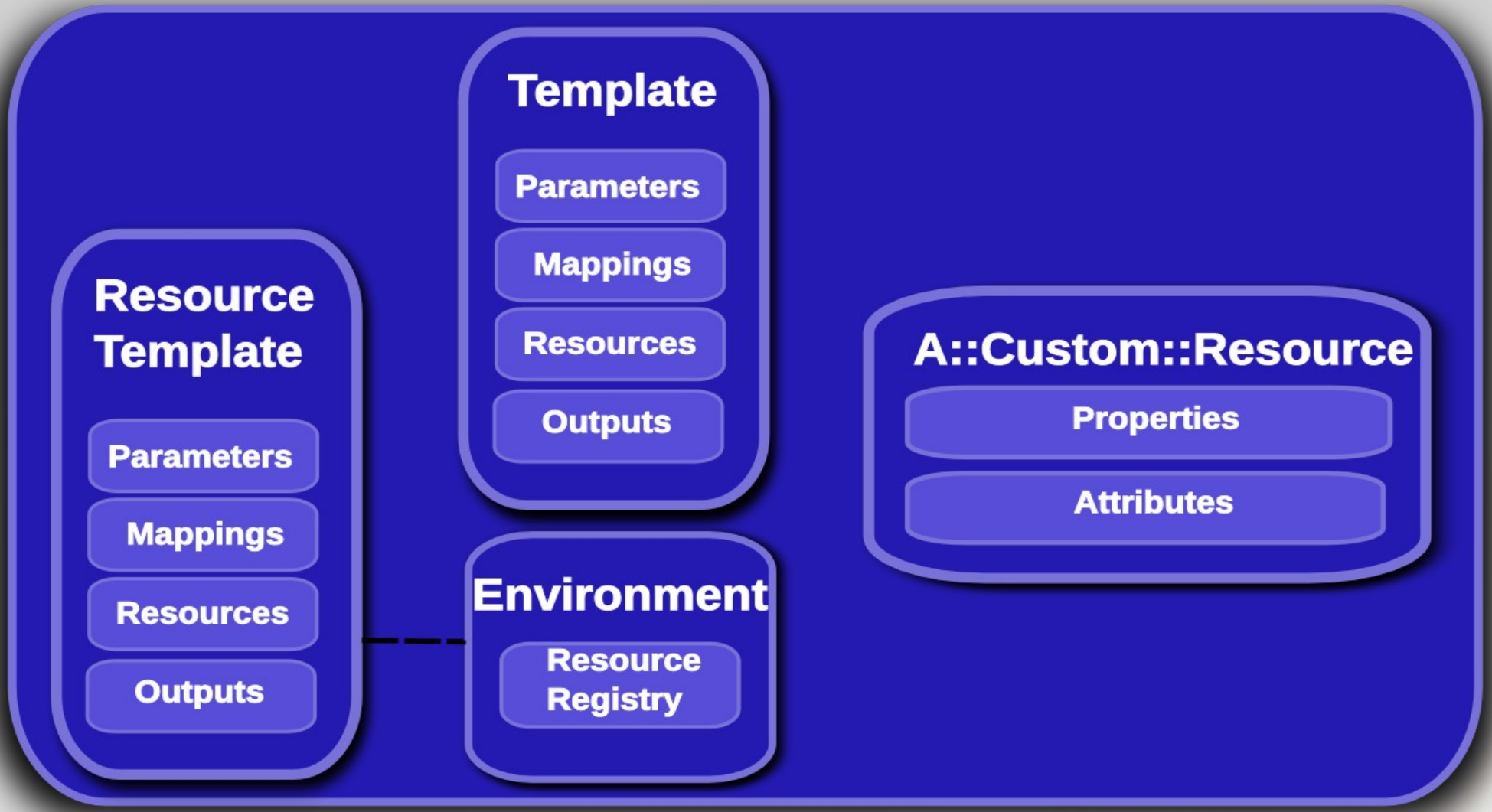


Heat Nested Stack Templates

```
"Resources" : {
  "OpenshiftBroker": {
    "Type": "AWS::CloudFormation::Stack",
    "Properties": {
      "TemplateURL": "https://somewhere/something.template",
      “Parameters”: <map containing stack parameters>
    }
  },
```

- Available in Grizzly Heat
- AWS compatible interface to nested stacks
- Allows user to compose layered/reusable deployments
- Hard-coded URL's is inconvenient

Provider Resources



Provider Resources

- Heat native interface to nested stacks
- No hard-coded URLs (in the template)
- Staging workflow/testing much simplified
- Allows deployer **and user** to define custom resources
  - /etc/heat/environment.d
  - /etc/heat/templates
  - Users heat stack-create -environment-file=foo.yaml
    - Users can override default deployer resources!

resource_registry:
"My::Custom::Server": file:///foo/bar.yaml

Heat Resource Types

AWS::AutoScaling::AutoScalingGroup
AWS::AutoScaling::LaunchConfiguration
AWS::AutoScaling::ScalingPolicy
AWS::CloudFormation::Stack
AWS::CloudFormation::WaitCondition
AWS::CloudFormation::WaitConditionHandle
AWS::EC2::EIP
AWS::EC2::EIPAssociation
AWS::EC2::Instance
AWS::EC2::InternetGateway
AWS::EC2::NetworkInterface
AWS::EC2::RouteTable
AWS::EC2::SecurityGroup
AWS::EC2::Subnet
AWS::EC2::SubnetRouteTableAssocation
AWS::EC2::Volume
AWS::EC2::VolumeAttachment
AWS::EC2::VPC
AWS::EC2::VPCGatewayAttachment
AWS::ElasticLoadBalancing::LoadBalancer
AWS::IAM::AccessKey
AWS::IAM::User
AWS::RDS::DBInstance
AWS::S3::Bucket

OS::Ceilometer::Alarm
OS::Cinder::Volume
OS::Cinder::VolumeAttachment
OS::Heat::AccessPolicy
OS::Heat::CWLiteAlarm
OS::Heat::HARestarter
OS::Heat::InstanceGroup
OS::Neutron::Firewall
OS::Neutron::FirewallPolicy
OS::Neutron::FirewallRule
OS::Neutron::FloatingIP
OS::Neutron::FloatingIPAssociation
OS::Neutron::HealthMonitor
OS::Neutron::IKEPolicy
OS::Neutron::IPsecPolicy
OS::Neutron::IPsecSiteConnection
OS::Neutron::LoadBalancer
OS::Neutron::Net
OS::Neutron::Pool
OS::Neutron::Port
OS::Neutron::Router
OS::Neutron::RouterGateway
OS::Neutron::RouterInterface
OS::Neutron::Subnet
OS::Neutron::VPNService
OS::Nova::Server
OS::Swift::Container

Icehouse Roadmap

- Developing HOT DSL
- Engine scale-out
- Rolling Updates (UpdatePolicy, Metadata update)
- Native in-instance tools
- Template-function plugins
- Software configuration resources (Puppet/Chef/...)
- More native resource types
- Stack snapshot/restore
- Per-resource suspend/resume
- Better cloud-init integration
- ...

Links, any Questions?

- http://docs.openstack.org/developer/heat/
- http://openstack.redhat.com/Docs
- http://github.com/openstack/heat
- https://launchpad.net/heat
- http://wiki.openstack.org/wiki/Heat
- http://hardysteven.blogspot.co.uk
- https://github.com/hardys/presentations