Vnomic

Automated Engineered SAP Landscape as a Service

Vnomic Declarative Deployment and Governance Automation Introduction and Basic Operation

Vnomic Proprietary and Confidential

VNTOBAS104

Contents

Introduction

The Vnomic Declarative Deployment and Governance Platform enables fully automated AI-Model Driven Delivery and Governance of business application deployments on hyperscaler's. This paper introduces the approach taken by Vnomic to address the operational challenges of deploying and managing sophisticated applications on today's newest programmable infrastructures.

Application Deployment and Governance Challenges

The growing sophistication and deployment complexity of next-generation application landscapes means that traditional approaches to application deployment and governance no longer meet business expectations. In a competitive world, customers must deploy services quickly, then rapidly and safely add or remove resources from already running applications to meet the demands of users, based on service level requirements, time scheduling and cost factors, while adhering to the security and governance requirements.

Complex, large scale, and dynamic application landscapes require a new approach to automation, which can't be achieved with today's tools that depend on risky manual processes. The next generation of automation must abstract application requirements and lifecycle semantics into a set of abstractions, which afford the fundamental operations used to manage applications throughout their lifecycles. This Application Centric perspective raises the application to the highest level of importance views and places infrastructure as subservient to the application. In an Application Centric world, the value of the infrastructures is determined by how well its capabilities can be used to fulfill the requirements of the application.

New Challenges with Programmable Infrastructure

Programmable infrastructure (PI) has transformed infrastructure management from a logistically and human intensive physical activity into a software configuration/programming activity. Clouds have reduced the traditional unbounded diversity of data centers into a well-defined set of resources represented as a small number of logical abstractions. The new challenge is to map the application requirements into the semantics and APIs of each cloud. For example, in order for a virtual server to access storage the programmable compute resources have to be configured interact with the programmable network resources to reach the programmable storage resources on the storage ports with the appropriate bandwidth and fail over-semantics. This requires semantic alignment across all elements of the Programmable Infrastructure. Whether using graphical consoles, scripting/automation tools, or direct APIs, operations personnel are responsible for learning the semantic details of all Programmable Infrastructure, defining the specific configuration or programming steps, and understanding implications and verifying correctness of the changes made.

Vnomic Approach

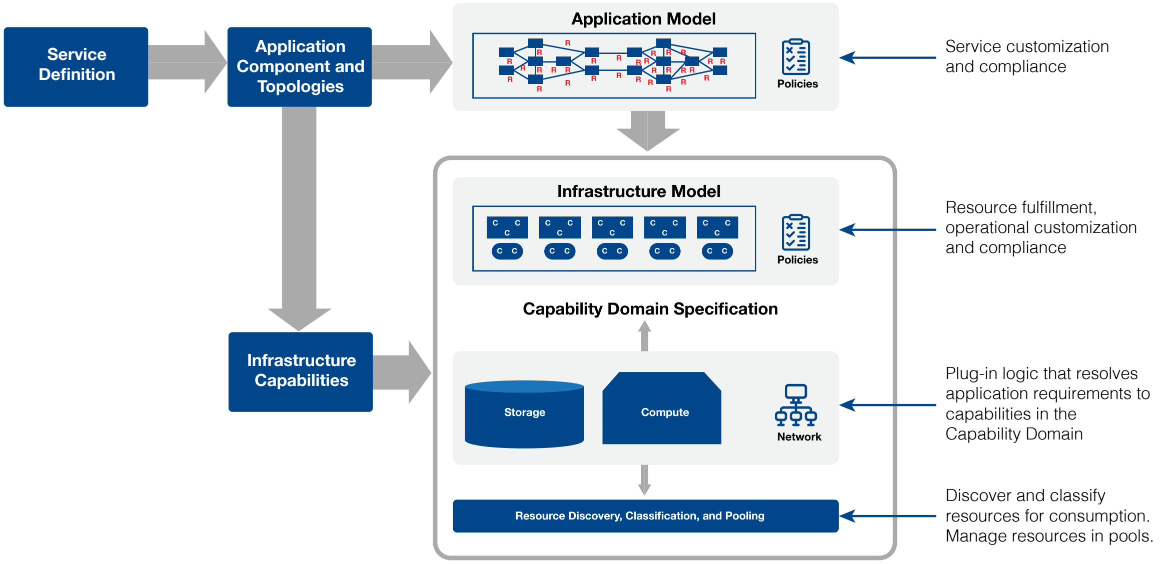
Vnomic takes an Application Centric approach where application semantics are completely and concisely expressed using Application Models in terms of application components, their lifecycles, their dependencies and inter-relationships and infrastructure requirements, eliminating the need for ongoing expert intervention for the most common and complex application lifecycle management tasks.

Vnomic provides a new level of automation capable of deployment and governance of sophisticated application landscapes portably across diverse cloud and private data center infrastructures. Vnomic enables a new operations paradigm where, using a set of application level abstractions, the automation solution takes responsibility for understanding the application and automatically computing and executing the necessary actions to achieve a desired state operating state.

This section describes the key attributes of the Vnomic automation technology.

Application Centric

As already mentioned, first and foremost, Vnomic takes an Application Centric approach such that the application intent and requirements drive the entire automation process. The Application Centric perspective demands that application semantics must be understood by the automation platform. Application requirements are completely and concisely described such that they can be automatically fulfilled with completely and concisely described infrastructure capabilities.


Service Definition
Application Component and Topologies
Application Model
Policies
Infrastructure Model
Policies
Capability Domain Specification
Infrastructure Capabilities
Storage
Compute
Network
Resource Discovery, Classification, and Pooling

Service customization and compliance

Resource fulfillment, operational customization and compliance

Plug-in logic that resolves application requirements to capabilities in the Capability Domain

Discover and classify resources for consumption. Manage resources in pools.

Application Modeling

Vnomic has pioneered a set of general modeling techniques capable of expressing application semantics in terms of their structure, behavior, interdependencies; infrastructure requirements (compute, storage, and network resource needs); scalability and availability; service level objectives; security and compliance. In this document, this set of application centric semantics representations will be referred to as Application Models.

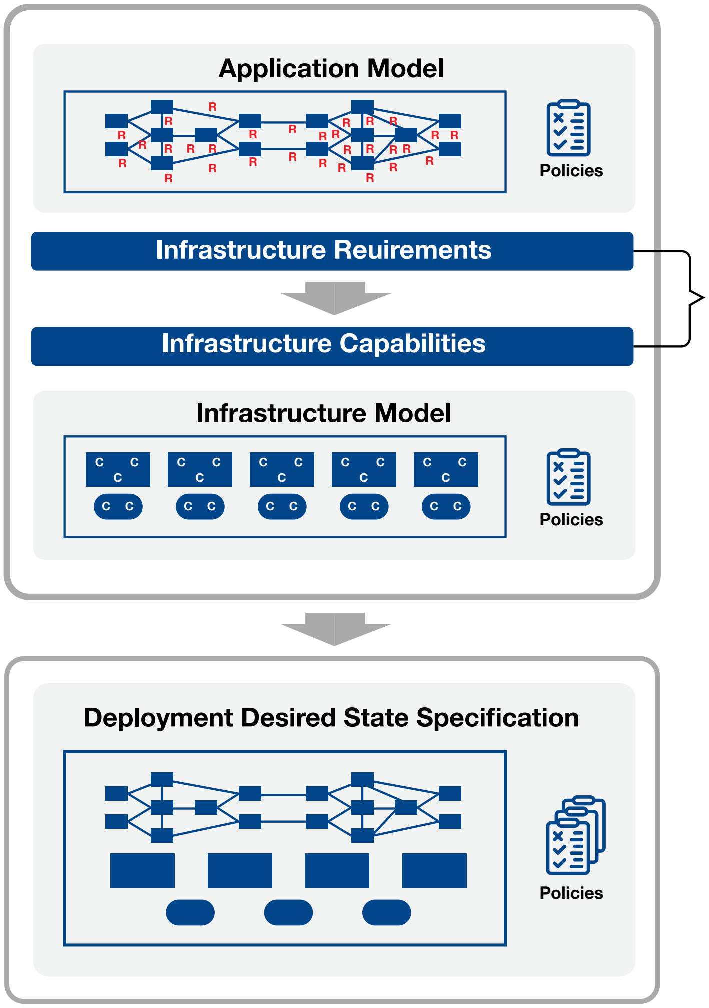
These Application Models represent the entire application topology and the services offered by the application as a single logical abstraction with well-defined lifecycle operations and transformations which enable automation of the most common operations such as initial provisioning, update, topology change, migration, decommissioning, de-provisioning.

Infrastructure Modeling

Given an Application Model, Vnomic must determine how to fulfill the Requirements of the application with the available infrastructure. Vnomic does this by expressing the semantics of clouds and infrastructures as Capabilities, i.e. the resources, features or behaviors they can provide to application components. In this document, this set of infrastructure semantics will be referred to as Infrastructure Models. The sole purpose of Infrastructure Models is to describe how a managed infrastructure can fulfill application Requirements enabling Vnomic to automatically select the resources needed to meet application Requirements, eliminating the need for users to manually perform or supervise the selection of the appropriate kinds and amounts of infrastructure resources.

Application Requirements to Infrastructure Capabilities Matching

Using the Infrastructure Requirements contained in Application Models and the Infrastructure Capabilities contained in Infrastructure Models, the process of matching Requirements to Capabilities can be generalized and fully automated. This allows an infrastructure Capability Provider to fulfill requirements in more than one way. Additionally, different sets of requirements can be fulfilled in different ways according to context such as calling an API, invoking a tool, or consuming an already available resource. Furthermore, this avoids the traditional least common denominator problems found with standardized APIs, which limit the expressible semantics that can be communicated over the API. This makes it possible for infrastructure vendors to fulfill requirements in new ways as they develop new features. Moreover, since there is complete visibility into which capabilities are used to fulfill which requirements, customers see and appreciate the differences between infrastructures from the viewpoint of their application's specific needs.


Application Model
R
Policies
Infrastructure Reuirements
Infrastructure Capabilities
Infrastructure Model
C
Policies
Deployment Desired State Specification
Policies

Application components and their respective Relationships, Requirements, and Capabilities. Policies control component behavior in specific contexts

Application infrastructure Requiments are matched to infrastructure capabilities for fulfillment

Application components and their respective Relationships, and Requirements, and Capabilities. Policies control component behavior in specific contexts.

The Desired State Specification describes the structure and state deployment from which the Vnomic Desired State Controller checks compliance and remediates as needed.

Models Capture, Preserve, and Document Best Practices

One of the primary challenges of automation is having a sufficient understanding of a component in order to know how to manipulate it. Vnomic has pioneered modeling techniques that provide sufficient expressiveness to describe a broad set of application component and infrastructure semantics. By using semantic models, expert knowledge and best practices are captured when the models are created and leveraged in every automation execution. Because semantic models declaratively represent application and infrastructure entities, their inter-relationships, dependencies, requirements, capabilities and behaviors, models serve as documentation regarding the components of the system, its structure, and its valid states. Semantic models use formal type and constraint systems to ensure only correct instantiations of application components and topologies occur. Modeled components "snap" together in design tools and are inherently reusable.

Traditional non-modeling approaches rely on executable logic, such as scripts or workflows, where semantics and context are hardcoded and not easily understood, reused, or extensible, and changes must be manually performed by experts and re-tested for each new usage scenario. Other traditional approaches, such as those which use manifests, don't scale beyond the specific semantics implied in the manifests and are not composable or reusable outside of the context they were originally developed for. The root of these issues is the lack of an explicit meta-model and taxonomy of types, which is the core of modeling approaches.

Policies Enable Constraint-based Control and Customization

Vnomic uses policies to provide control specifications for its Desired State Controller and component customization to avoid having to create large numbers of very specialized application models.

Policies are sets of constraints that can be applied to one or more modeled entities. Sets of policies can be associated with applications models to:

- Specify specific configuration values
- Select specific modes of operation
- Enforce specific environment constraints

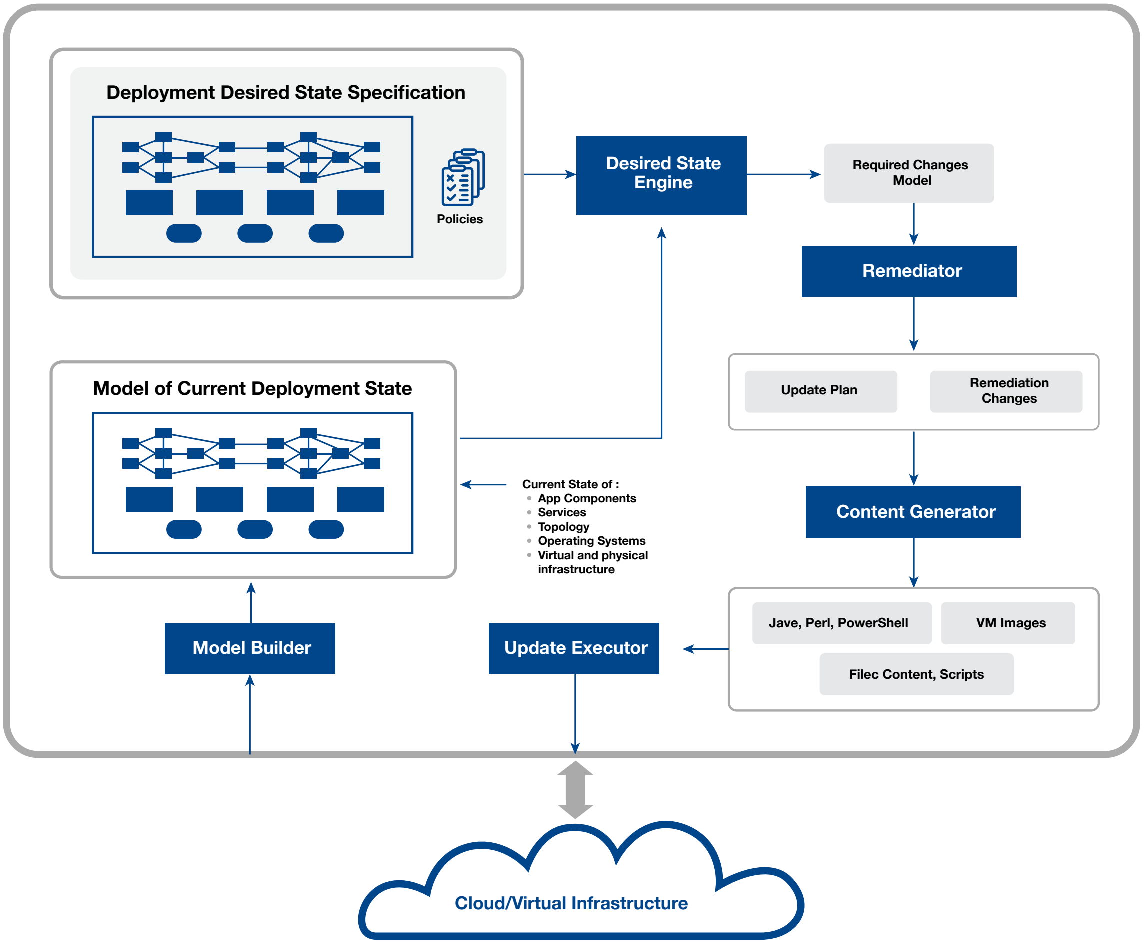
The Vnomic Desired State Controller (VDSC) contains a policy processing component which evaluates all active policies and computes deltas to bring the system back into compliance. Additionally, the use of policies allows automatic detection of semantic conflicts across the system.

Policy Defined, Declarative Desired State

Modeling is extremely useful for abstracting and expressing semantics.

However, there is still the significant challenge of how to use this information to physically automate the necessary application lifecycle operations. Vnomic uses a Declarative Desired State (DDS) paradigm to manage the state of managed software and infrastructure. DDS uses a Desired State Specification (DSS) automatically computed by transforming the Application Model requirements into a set of provisioning, deployment and configuration policies. The policies specify what is desired and not how to achieve the desire. The Vnomic Desired State Controller (VSC) then uses these policies in its control loop to check them against the current state of the software and infrastructure, compute the delta between the current state and desired state, and using the application and infrastructure models and their respective plug-ins, compute a remediation consisting of an Update Plan and the specific changes which must be made to the software and infrastructure to move it to the desired state.

A key benefit of the DDS approach is that it can work throughout the application lifecycle, not just for initial deployment. This includes, but is not limited to, common actions like updating component versions, changing the size or scale of the topology, migrating from one infrastructure to another.



Layered on Top of Existing Automation Tools and APIs

Due to the abstraction afforded by modeling infrastructure semantics as Capabilities, Vnomic Capability Providers (CPs) can draw on any existing tools or APIs to fulfill Requirements.

CPs simply have to encode an appropriate set of update commands that will be routed to and executed by the tool or API during execution of the Update Plan. This allows existing tools to be leveraged with little or no change and completely new APIs or tools used to automate other software and infrastructure elements.

Deployment across Heterogeneous Infrastructures

Due to the abstraction afforded by modeling infrastructure semantics as Capabilities, Vnomic is able to deploy an application across different Clouds.

Vnomic uses the concept of Capability Domain to describe a set of infrastructure resources such as software defined compute, storage, network, and external services that can host application deployments. This enables users to create diverse infrastructure pools with different semantics and/or for different purposes.

Application Portability and Infrastructure Independence

By design, application models express their requirements of the infrastructure but not how to concretely fulfill them. For example, application models will specify that software component A and component B should communicate but they will provide no concrete specification of how this requirement should be fulfilled. The VDSC will invoke the appropriate Capability Providers in the target Capability Domain to realize the fulfillment. Thus, as long as the Capability Domain can meet the requirements of the application, the application may be hosted on the infrastructure comprising it. It is this aspect that makes them infrastructure independent and thus portable across clouds infrastructures.

In general, as long as the infrastructure has the capabilities to meet the application requirements, it can host the application. Of course, the infrastructure must be modeled and included in a Capability Domain. This allows large sets of applications to be supported on new or additional infrastructure simply by implementing the necessary Capability Providers.

Layered Infrastructure Resource Management

The VDSC can work with different kinds of infrastructure including bare metal, virtualized/ cloud, and software containers. This is a fundamental requirement since applications may require their components to be deployed on any of these. As a result, the VDSC must be capable of provisioning each of these as well as any other resources required to fulfill them. For example, in order to provision a VM it may be necessary to instantiate a hypervisor on a bare metal server. Similarly, to provision a software container, it may be necessary to provision a virtual machine. And in order to meet the resource requirements of each of these it is necessary to track the capacity and demand on each of these resources to know if a given instance can be used or another should be created.

Integration with other Data Center Services

By modeling cloud infrastructure capabilities and providing Capability Provider plug-ins to fulfill specific application requirements, existing cloud services can be used to fulfill requirements or provide additional capabilities which applications can require explicitly. For example, application support requirements such as monitoring, backup, and disaster recovery can be included in application models as part of the concisely defined set of applications requirements instead of being left as an additional manual configuration operation, eliminating error prone manual steps.

Pre-validated Application Automation for Programmable Infrastructures

Software and infrastructure vendors aim to ensure their products are deployed and configured correctly and efficiently. Vnomic works directly with software and infrastructure vendors and expert consultants to build robust models which include detailed knowledge and best practices. The models are validated in dedicated labs for end-to-end system integration and certification with prescribed combinations of software and infrastructure ensuring the automation results in correct functionality, performance and security. This results in sets of application models and infrastructure designs that users consume to build robust and supportable SaaS implementations.

This means users don't have to take responsibility for building the automation and are assured the automation will provide the best performance and security. Users can then focus on selecting the right application topologies and infrastructures to support them. Users are able to augment the orchestration to achieve their specific customizations and integration requirements while preserving the integrity of the automation.

Vnomic Architecture

This section provides an overview of the Vnomic Declarative Deployment and Governance Platform architecture.

Automation Stack

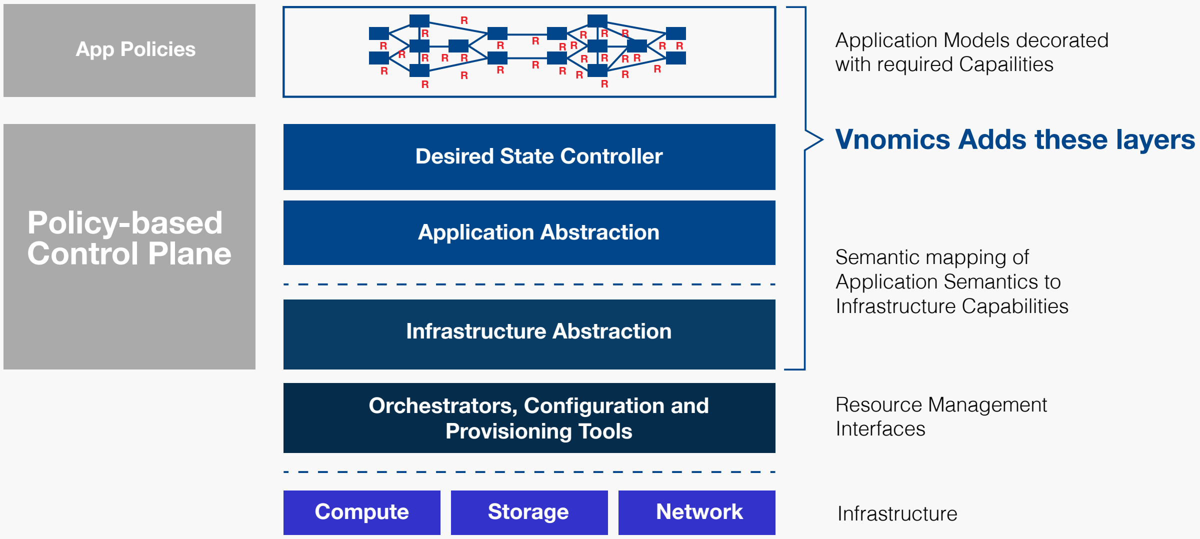
The automation stack consists of a series of abstraction layers along with a policy control plane. All abstraction layers share a single meta-model.

At the core of the stack is the Vnomic Desired State Controller (VDSC). The VDSC provides application lifecycle support using Application Models as input and a set of command and control operations for overall automation.

The Application Abstraction Layer provides a set of applications entities, relationships, requirements, and policies that represent the application semantics supported by the system. Application models use this ontology to express their semantics. This ontology contains 100s of concepts like web server, data base, file system, package, artifact, etc. which represent applications.

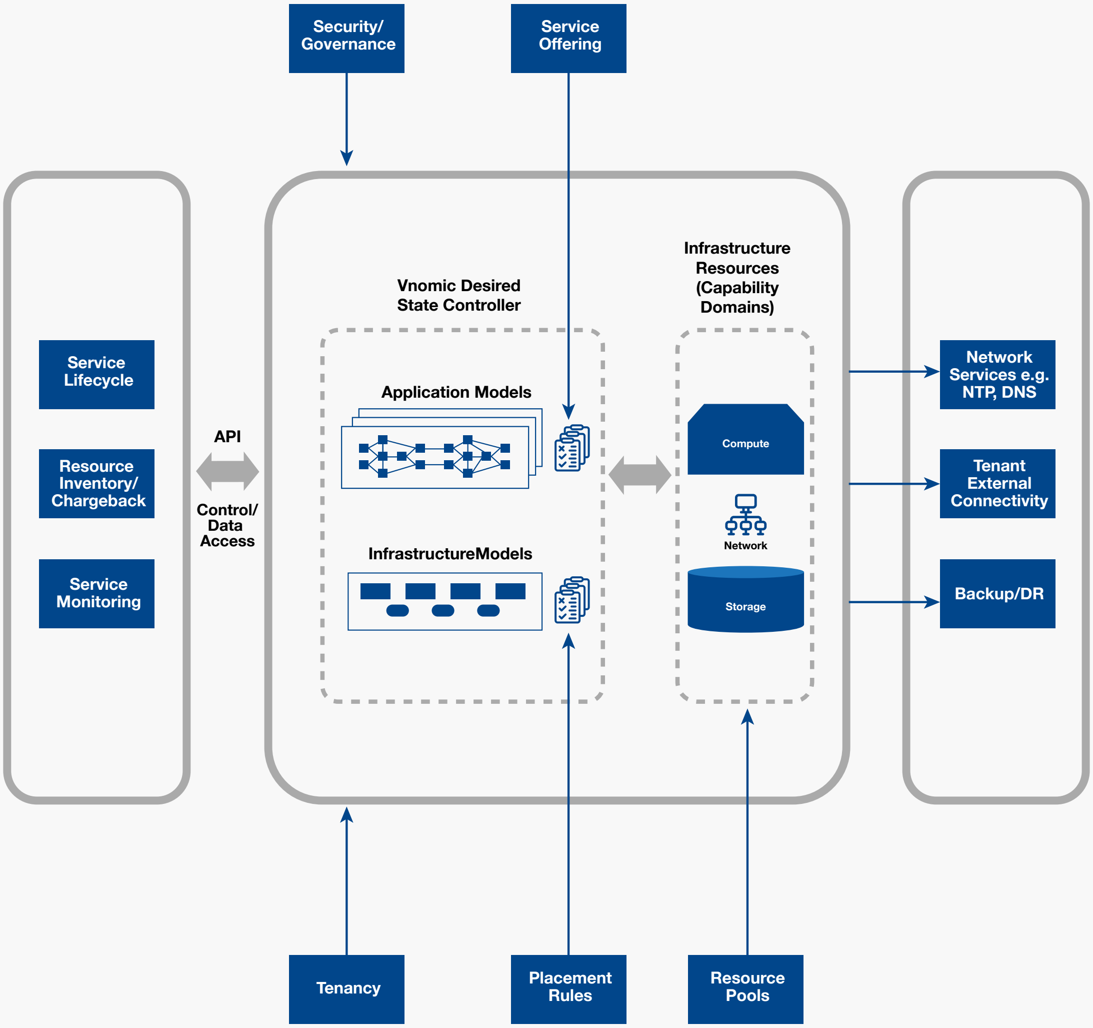
The Infrastructure Abstraction Layer provides a set of infrastructure entities, relationships, capabilities and policies that represent the infrastructure semantics supported by the system. Infrastructure models use this ontology to express their semantics. This ontology contains 100s of concepts like containers, networks, endpoints, volumes, VLANs, etc. which represent infrastructure. The infrastructure Abstraction Layer also hosts the Capability Domains and their Capability Providers for the automated fulfillment of application requirements.

The Policy Control Plane allows policies to be applied to any level of the stack. Policies can be pre-packages in application or infrastructure models or added by operators to control automation behavior.


App Policies
Policy-based Control Plane
Desired State Controller
Application Abstraction
Infrastructure Abstraction
Orchestrators, Configuration and Provisioning Tools
Compute
Storage
Network
Application Models decorated with required Capailities
Vnomics Adds these layers
Semantic mapping of Application Semantics to Infrastructure Capabilities
Resource Management Interfaces
Infrastructure

Logical Integration

The following diagram shows the primary integration touch points in a typical Vnomic deployment. At the center is the VDSC and surrounding it are the metadata or information flows that can be integrated. Vnomic provides reasonable defaults for everything. The specific configuration is usually a function of the automation use cases needed and the constraints and capabilities of the hosting environment.



Vnomic

Automated Engineered SAP Landscape as a Service