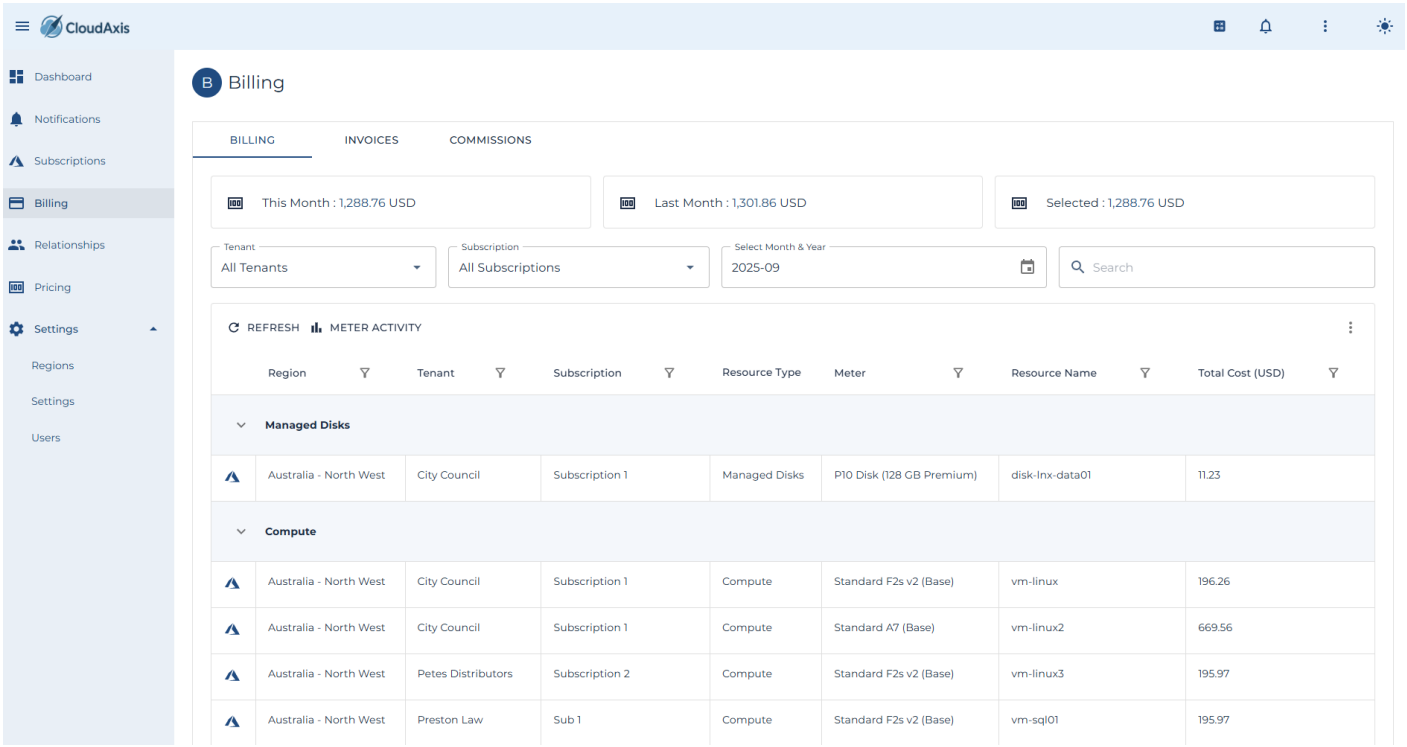
Introduction

Overview

The **CMS for Azure Stack Hub is** a management and billing platform for Azure Stack Hub that connects technical service delivery with commercial operations. It is designed for service providers, enterprises, and partners who need to operate Azure Stack Hub as a structured, multi-tenant cloud service.



At its core, the CMS provides a layered model that reflects how services are delivered and monetized:

- **Platform Administrators** operate and configure the system. They publish pricing, manage global settings, and oversee usage processing across regions.
- **Distributors** represent the top commercial tier beneath the platform. They manage multiple partners and have visibility into aggregated usage, billing, and commissions.
- **Partners** act as the customer-facing organizations. They onboard tenants, manage subscriptions, and handle tenant-level billing and support.
- **Tenants** are the consuming organizations. They manage their own subscriptions and users, while billing and usage flow upward to the partner and distributor.
- **Subscriptions** are the unit of consumption within a tenant. They tie workloads to plans and quotas, generate usage records, and form the basis for billing and invoicing.

This hierarchy ensures that both technical and commercial responsibilities are aligned. Usage flows upward from subscriptions to tenants, partners, and distributors, while pricing, commissions, and governance flow downward from the platform.

Key capabilities include:

- **Unified Multi-Tier Management** – Administrators, distributors, partners, and tenants are modeled consistently in the system.
- **Automated Billing** – Usage data is collected, rated, and reconciled into invoices at the subscription and tenant level.
- **Flexible Pricing and Discounts** – Flat and tiered rates, credits, and overrides can be applied at multiple scopes.
- **Commission Frameworks** – Built-in logic for calculating and attributing commissions to partners and distributors.
- **API-Driven Operations** – All functions are exposed through a standards-based REST API for automation and integration.
- **Branding and Delegation** – Each level of the hierarchy can be branded appropriately, with role-based access control ensuring least-privilege delegation.

The CMS provides the foundation to operate Azure Stack Hub as a commercial service, supporting both external service providers and internal enterprise IT. By unifying consumption data, pricing models, billing, and commissions under a single platform, it enables organizations to deliver cloud services with transparency, accountability, and commercial accuracy.

Supported Scenarios

The CMS is designed to support a wide range of operational and business models for organizations using Azure Stack Hub. It combines a multi-tier commercial hierarchy with flexible deployment patterns, ensuring that both service providers and enterprises can align technical delivery with business outcomes.

Platform Administrators

Operate and govern the CMS at the highest level. Typical activities include publishing pricing and plans, managing usage processing, configuring branding and authentication, and ensuring compliance across regions.

Distributors

Represent the top commercial tier beneath the platform. They manage multiple partners, view aggregated billing and commission data, and apply distributor-level commission structures.

Partners

Act as the customer-facing entities for tenants. They onboard tenants, assign subscriptions, and manage billing at the tenant level. Partner-specific pricing and commission rates can be applied where required.

Tenants

Represent the consuming organizations (end customers or internal business units). Tenants manage their own subscriptions and users, while billing and usage data flow upward to their partner and distributor.

Subscriptions

Provide the unit of consumption within a tenant. Subscriptions map workloads to specific plans and quotas, generate usage records, and serve as the basis for billing and invoicing.

Industry Use Cases

- **Service Providers**
  Deliver Azure Stack Hub as a multi-tenant service platform. Use the CMS to onboard tenants, meter consumption, generate invoices, and ensure commissions flow accurately to partners and distributors.
- **Enterprise IT**
  Adopt the CMS as an internal service management layer. Departments become tenants, projects or environments are modeled as subscriptions, and chargeback or showback reporting ensures cost transparency.

- **Hybrid Cloud Operators**
  Consolidate usage across Azure Stack Hub and external cloud platforms. Apply consistent billing and governance policies across all regions, even in partially disconnected environments.
- **Regulated Environments**
  Deploy the CMS where compliance and auditability are essential. Role-based access control (RBAC), usage records, and audit trails support regulatory frameworks.

Out of Scope

The CMS does not replace the core resource providers in Azure Stack Hub (Compute, Storage, Network) or native Azure monitoring tools. Instead, it complements them by adding commercial, operational, and governance layers on top of existing capabilities.

System Requirements

The CMS is delivered as a .NET-based application backed by a MySQL database. It is packaged for deployment in containerized or virtualized environments and can be operated either alongside Azure Stack Hub infrastructure or on external platforms.

Software Requirements

- **Database**: MySQL 8.0 or later
- **Application Runtime**: .NET 8
- **Frontend**: Blazor Server (runs as part of the API service or in a separate container)
- **Optional Components**: Docker or Kubernetes for container orchestration

Hardware Requirements

- **Production Deployment**
  - Minimum 4 vCPUs
  - 16 GB RAM
  - 100 GB storage (expandable based on retention of usage and billing data)
- **Development or Evaluation Deployment**
  - Minimum 2 vCPUs
  - 8 GB RAM
  - 50 GB storage

Network Requirements

- Outbound HTTPS connectivity to Azure Stack Hub administrative endpoints
- Inbound HTTPS (TCP 443) for API and portal access
- Optional VPN or ExpressRoute for hybrid or private connectivity scenarios

Identity Requirements

- Integration with Microsoft Entra ID or another supported identity provider
- Federated authentication supported through token validation endpoints
- Role-based access control (RBAC) enforced throughout the portal and API

High Availability and Recovery

The CMS can be deployed in a high-availability configuration using container clustering or database replication. Backup and recovery procedures should be established to protect the MySQL database and configuration state. For detailed operational guidance, see *Backups & DR* in the Operations chapter.