**<https://internal.cloud-solutioning.ibm.com>**

**Exported output as configured:**

****

**What this example is**

We’re designing a **single‑site, dedicated** VMware Cloud Foundation (VCF) environment on IBM Cloud. In the patterns catalog, *single site* is achieved by starting from the standard two‑region reference and **removing Region B and its Direct Link and CIS** components (keep CIS only if you specifically need DDoS protection on the public edge). That’s why you’ll see only one region and one cluster in this build.

On the sizing side, the tool computes hosts and storage using the built‑in formulas for CPU, memory and vSAN (it accounts for buffers, hyper‑threading, RAID/FTT, and dedup when enabled). That is what drives the “Hardware configurations” you saw.

|  |  |
| --- | --- |
| A screenshot of a black screen  AI-generated content may be incorrect. | **1) Projects list – the tile for EX\_VMware\_SingleSite**  **Caption:**  *Your project at a glance.* This card simply shows that the project has **VMware** plus **Common Services (CS)** attached. Common Services is where you add cross‑cutting items such as Transit Gateway/Direct Link networking and Security & Compliance Center (SCC) protections, which are used by any offering in the project. |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **2) Project Overview – name, opportunity, offerings**  **Caption:** *Tell the tool what you’re sizing.*   * **Offerings:** VMware is selected; “Common Services” is also ticked so we can add shared networking and security later. * **Discount:** Left at **0%** because discounts require formal approval. Keeping it at 0% is best for first‑pass estimates. * **Why nothing else here?** We didn’t pick PowerVS, VPC, SAP, or Satellite/ROKS because this example teaches a *VMware‑only* design. If you add them, the project grows into a multi‑offering solution (the tool supports that, but it’s out of scope for this example). |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **3) VMware → Instance header – VCF as a Service, Single‑Tenant, Region**  **Caption:** *Choose the service flavor and location.*   * **VCF as a Service** (managed by IBM) is selected instead of “VCF (Self‑managed)”. For beginners this reduces operational burden—IBM runs the control stack and lifecycle, you size the workload clusters. * **Single Tenant / Dedicated:** Ensures hardware isolation (no shared hosts). That aligns with the “Dedicated” wording in the patterns. * **Region & DC:** **Dallas (DAL10)** chosen for proximity and availability. * **Regional High Availability:** **Off** because this is a **single‑site** pattern. If you need cross‑site continuity, turn this on and add Region B (that would track the two‑region reference). |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **4) VMware → Workload cluster basics (workload‑a)**  **Caption:** *Define one cluster to run the app.*   * **Workload Type:** **Production** * **Flavor:** **Standard** (a balanced profile) * **Storage Type:** **vSAN + optional NFS**. vSAN is our primary datastore for VM disks. NFS remains available if we later add shared file capacities (we left NFS sizing at 0 in this run; see screen 7). * **Requirement ID (WKLD‑001):** A helpful label that will appear in exports.   **Why one cluster?** This example is intentionally simple. A second cluster (e.g., “management” or “database only”) is a valid option if you need fault isolation, different host profiles, or licensing separation. |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **5) Workloads table – app VM counts and sizes**  **Caption:** *Tell the tool what you plan to run.* We entered three tiers:   * **web:** 25 VMs × 2 vCPU × 8 GB RAM × 100 GB * **app:** 18 VMs × 4 vCPU × 16 GB RAM × 200 GB * **db:** 10 VMs × 8 vCPU × 64 GB RAM × 1,000 GB   These numbers drive the host count and storage capacity. The tool translates them using its CPU/RAM/vSAN formulas (buffers, hyper‑threading, oversubscription, and vSAN overhead are applied automatically—see next screen). |

|  |  |
| --- | --- |
| A black screen with white text  AI-generated content may be incorrect. | **6) Hardware configurations – computed host profile & quantity**  **Caption:** *What hardware meets the workload?* The tool found a fit with:   * **7 hosts** of **Dual Intel Xeon Platinum 8260 (48 cores, 2.4 GHz)** * **768 GB RAM per host** * **vSAN usable per node:** 23 TB (based on the vSAN/RAID/FTT math and buffers) * **Estimated cost** shown for the cluster (compute + vSAN) for this host profile.   **Why this result?**   * The **CPU/vCPU** capacity is calculated from total cores minus CPU buffer, plus hyper‑threading, then multiplied by vCPU‑per‑core and by the number of nodes. * The **RAM** capacity applies a RAM buffer and any oversubscription factor you set. * The **vSAN usable TB** applies the selected **RAID/FTT** ratio and **headroom**; dedup/compression can be applied when enabled.   If your inputs can’t fit a single cluster, the dialog will tell you to split workloads (common when you push node limits or mix very different storage needs). |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **7) Cluster Overhead Configuration Details – the dials**  **Caption: *Safety margins and architecture choices.* Defaults here express good practice for production sizing:**   * **Include Buffer: On – protects against growth and bursts.** * **Example values (as seen on the page): CPU buffer 30%, RAM buffer 20%, Hyper‑threading 10%, vSAN overhead 10%, RAM oversubscription 1.3, Max nodes 64.** * **CPU model chosen: the same 8260 profile the tool matched.** * **Nodes quantity: 7 (the minimum to satisfy the workload with these buffers).** * **Usable Node Capacity (TB): 23 per node, matching the vSAN math.**   **Why keep buffers? Turning them off can under‑size the environment. The sizing engine is designed to use these factors when it computes total vCPU, memory and vSAN capacity.** |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **8) Cluster Storage – Attached NFS Storages**  **Caption:** *We didn’t add NFS in this run.* All NFS tiers are **0 TB**. That’s intentional: the app’s VM disks live on **vSAN**. **When would you add NFS?** Shared file workloads, repositories, or cost‑optimized archival volumes. You’d size the tier (0.25/2/4/10 IOPS/GB) in **TB** increments to match your performance/price target. |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **9) Licenses & Network (instance‑level)**  **Caption:** *What we left out—and why.*   * **OS Licenses:** All set to **0**. Many clients bring their own OS licenses, or you can add Linux/Windows license quantities here to include them in price. * **Private network endpoints** (1 Gbps / 10 Gbps) left at **0**. For connectivity, we used **Common Services → Network** to design the core attachment (Transit Gateway + Direct Link), which is the right place to capture shared network services for a project. Instance‑level private endpoints are optional and not required for this VMware example. |

**Add Common Services -> SCC**

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **10) Security & Compliance Center (SCC) – Workload Protection**  **Caption:** *Add protection and posture at project level.*   * **Association:** Bound to *VMware Service Instance 1* in **us‑south**. * **Cloud Posture (agentless):** **7** *compute instances* (one per ESXi host in the cluster). * **Host Protection (agent‑based):** **53** *VM agents* (25 web + 18 app + 10 db). SCC pricing is then computed for these resources. This is a simple, transparent mapping beginners can follow: **one posture unit per host**, **one agent per protected VM**. |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **11) Security & Compliance Center (SCC) – Workload Protection setup**  **Caption:** *Connect SCC to your VMware instance and tell it what to protect.*  **What you’re seeing**   * **Associated with:** *VMware Service Instance 1* — this ties SCC to the exact VMware environment you’re building so counts and pricing line up. * **Name:** scc-protect-1 — a friendly label you’ll see again in summaries. * **Region:** *us‑south* — SCC runs as a regional service; us‑south is a common default. * **Cloud Posture (Agentless):** **7** total compute instances — one posture check per ESXi host. * **Host Protection (Agent‑based):** **53** virtual machines — one agent per VM you want to protect. * **Kubernetes (Agent‑based):** **0** — we’re not sizing containers in this example.   **Why these settings**   * The counts reflect the cluster we sized earlier: **7 hosts** and **53 VMs** (25 web + 18 app + 10 db). Keeping SCC in a single region matches the single‑site pattern.   **What we didn’t include (on purpose)**   * No OpenShift/Kubernetes worker nodes (set to 0) because this build focuses on VMs, not containers. * No extra SCC add‑ons; we’re covering posture (agentless) for hosts and agent‑based protection for VMs to keep the example straightforward for beginners. |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **12) SCC – Totals panel**  **Caption:** *A quick check that your protection counts match your environment.*  **What you’re seeing**   * **Kubernetes workers:** **0** * **VMs protected:** **53** * **Total resources:** **60** (7 hosts + 53 VMs)   **Why these settings**   * This page is a “sanity check.” If you later add more VMs or a second cluster, this total should rise accordingly. For a first pass, matching the earlier compute plan keeps cost and protection consistent.   **Options you could add later**   * If you deploy containers, increase **Kubernetes worker nodes** to include them in SCC. * If you only want a subset of VMs protected (e.g., just production), lower the VM count to exactly what you want to license and protect. |

**Add Services -> Network**

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **13) Common Services → Network – Add Transit Gateway (initial form)**  **Caption:** *Start the network hub your environment will attach to.*  **What you’re seeing**   * The **Transit Gateway** tile is selected under **Add Services → Network**. * Fields for **Association**, **Name**, **Location**, **Routing Type** are empty. * A starter **Connections** row (named *Connection 1*) with placeholders for:   + **Connection Type** (e.g., Direct Link or VPN)   + **Number of Connections**   + **Egress Data (GB) per month** (estimate for outbound traffic)   **Why we’re using Transit Gateway**   * Think of Transit Gateway as a **hub** that can connect your VMware environment to your on‑prem network and to other IBM Cloud networks as you grow. * Capturing this under **Common Services** keeps shared networking in one place for the whole project (not just VMware).   **What we didn’t add yet**   * We didn’t select a connection type here because this screen is just the starting point; we’ll fill the details next. * We didn’t add VPC attachments or VPNs because this example focuses on a **private, high‑capacity on‑premises link** (Direct Link). |

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | **14) Common Services → Network – Transit Gateway (completed)**  **Caption:** *Define where the hub lives and how you connect to it.*  **What you’re seeing**   * **Associated with:** *VMware Service Instance 1* — so this TGW serves the VMware environment you sized. * **Name:** NET-EX\_VMware\_SingleSite * **Location:** *Dallas* — matches the VMware region to keep things simple. * **Routing Type:** *Global* — lets this gateway route across regions if you add them later. * **Connections table:**   + **Connection 1 → Direct Link**   + **Number of Connections:** **2** — two circuits for resilience.   + **Egress Data (GB) per month:** **10** — a small placeholder for outbound usage.   **Why these settings**   * **Direct Link** gives a **private, predictable path** between your data center and IBM Cloud — ideal for production VMware. * **Two** links protect against a single circuit failure. * **Global routing** keeps future options open without forcing any multi‑region today.   **Options we didn’t include (and why)**   * **Local routing** — cheaper but only local scope; we chose **Global** to avoid redesign if you later add another region or service. * **VPN** — great for labs or small sites, but for a production VMware estate, Direct Link’s bandwidth and reliability are a better fit. * **CIS (Cloud Internet Services)** — not added because this example doesn’t build a public edge; per your pattern notes, keep CIS only if you need DDoS/WAF at the public perimeter. |

|  |
| --- |
| A screenshot of a computer  AI-generated content may be incorrect. |
| **15) Solution & Pricing → Solution (VMware tab)**  **Caption:** *The tool rolls up what it will build for VMware.*  **What you’re seeing**   * **Instance details:** name, region (Dallas / DAL10), and **Single Tenant / Dedicated**. * **Workload cluster:** workload-a with the **7‑node** plan we sized. * **Capacity summary:** shows how the **requirements** (your VM counts) compare to the **solution capacity** (what 7 nodes provide) for CPU, RAM, and storage. * **Other Sections:** lists the **Host Profile** (dual‑socket, 48 cores, 768 GB RAM, 23 TB vSAN per node) and placeholders for **Licenses** and **Network** we left at zero in this example.   **Why this matters**   * This view confirms the math: a **7‑node** cluster satisfies your inputs with the recommended buffers/overheads applied earlier (no hidden knobs to surprise beginners).   **What we didn’t include**   * **OS licenses** (Windows/Red Hat) — kept at 0 because many customers bring their own; add them here only if you want them priced in this estimate. * **Instance‑level network endpoints** — we captured networking under **Common Services → Network** so it’s reusable across offerings, and clearer for teaching. |

|  |
| --- |
| A screenshot of a computer  AI-generated content may be incorrect. |
| **16) Solution & Pricing → Solution (Common Services → Security roll‑up)**  **Caption:** *SCC items summarized so they appear on exports and quotes.*  **What you’re seeing**   * **SCC Workload Protection** block for scc-protect-1 (region us‑south). * Counts restated as **7 Multi‑Cloud Compute Instances** (agentless posture on ESXi) and **53 VM Nodes** (agent‑based protection). * **Kubernetes workers:** **0** — consistent with the VMware‑only example.   **Why this matters**   * This section ensures SCC quantities show up alongside VMware in the final solution output, so sizing and pricing stay in sync. It’s exactly the one‑unit‑per‑host and one‑agent‑per‑VM mapping beginners can follow.   **What we didn’t include**   * No container agents or extra SCC features here to keep the tutorial focused. If you add containers later, just increase **worker nodes** to match. |

|  |
| --- |
| A screenshot of a computer  AI-generated content may be incorrect. |
| **17) Solution & Pricing → Pricing (VMware tab)**  **Caption:** *A first‑pass monthly estimate for the VMware stack.*  **What you’re seeing**   * **Pricing table:** “**VMware as a Service**” with a monthly total (shown as **$47,012** in the capture). * **Other Sections → Host Profile:** unit price (about **$6,716**/host/month) × **7 hosts** = the same total shown above. * **Licenses / Network:** **$0.00** in this example because we left those items out intentionally. * The yellow banner reminds you this is a **non‑binding initial estimate**; final pricing is validated in the IBM Cloud Portal.   **Why this is useful**   * It shows the *entire* VMware platform cost in one place for the chosen host profile and node count. Later, you can add licenses, public edge, or more networking and immediately see the impact.   **What we didn’t include (yet)**   * **Network (Direct Link/TGW) pricing** and **SCC pricing** appear under their respective Common Services sections when you export the full solution; we kept this tab focused on the VMware stack for clarity. * **Discounts** — left at **0%** in early sizing (discounts require approvals and vary by deal). |

**Quick recap of the teaching points**

* A **single‑site** VMware design stays simple: one region, one cluster, one Transit Gateway, and one SCC policy.
* **Counts drive everything**: 7 hosts and 53 VMs flow through VMware capacity, SCC protection, and solution/pricing.
* **Common Services** (Network + Security) keep shared items in one place and reusable across offerings as your project grows.
* We **intentionally left out** Kubernetes, OS licenses, and a public edge to keep the walkthrough focused; you can add them later using the same screens when needed.