**MCPX‑KendoBridge — Scale‑Out & Capacity Runbook (SSE‑Aware, Session‑Sticky)**

**Document:** runbooks/scale\_out.docx  
**Version:** 2.0.0 (Kendo Migration)  
**Last Updated:** 2025‑09‑27  
**Owner:** SRE Lead (Responsible) — DoSE (Accountable) — DocFactory (Author)  
**Applies to:** API (.NET 8, Streamable‑HTTP + SSE), Admin Portal (KendoReact Fluent v12, read‑only)

**Purpose.** Provide a **repeatable, auditable** approach to scale MCPX‑KendoBridge safely under load while preserving **SSE** quality and **session stickiness** by Mcp‑Session‑Id. The runbook aligns to Technijian’s **GitHub‑first** SDLC, four environments (**Alpha → Beta → RTM → Prod**), and Evidence Pack retention (≥ 1 year). It enforces the project’s non‑functional budgets: JSON latency p50 ≤ 300 ms / p95 ≤ 800 ms; **SSE TTFB p95 ≤ 200 ms**; restart‑to‑ready ≤ 30 s; ≥ 200 concurrent sessions/replica.

**DB & Secrets Compliance (always in effect).**  
**Add‑only** schema; **Stored‑procedure‑only** DAL; **No‑Hard‑Coding** of dynamic values (child cmd/args/cwd, request timeout, heartbeat cadence, Origin allow‑list, feature flags). All dynamic values are **DB‑sourced** (non‑secret) via sp\_Config\_\*, sp\_Feature\_IsEnabled, sp\_Lookup\_Get. **Secrets** (SQL connection strings, **Telerik license**) are never stored in code/DB/logs and live only in **GitHub Environments**.

**1) Scope & Preconditions**

* **Scope:** Horizontal scale‑out/in of the **API** deployment and associated ingress/load‑balancing, with **SSE pass‑through** and **sticky routing** by Mcp‑Session‑Id.
* **Preconditions:**
  1. Ingress **does not buffer** text/event-stream;
  2. Sticky routing by header (or cookie) is enabled;
  3. Readiness/health probes configured;
  4. **PDB** protects against mass eviction;
  5. **HPA** targets CPU (primary) and can include session metrics.

**2) Scale Triggers & Targets**

**When to scale out (one or more sustained for ≥ 5–10 min):**

* CPU ≥ **70%** across replicas, or **session\_count**/replica near **200** (design limit).
* **SSE TTFB p95 > 180 ms** trending upward toward 200 ms.
* Heartbeat gap p95 creeping above **1.5×** Network:SseKeepAliveSeconds.
* Growing **child\_restart\_count** coupled with higher concurrency per pod.

**Scale targets (guidance):**

* Add replicas incrementally (e.g., +1 or +2), observe **TTFB**/**heartbeat** recovery, then reassess.
* Maintain **min 2 replicas** in Beta/Prod for resilience; use PDB to keep ≥1 available during maintenance.

**3) Ingress & Sticky Sessions (required settings)**

**3.1 NGINX Ingress — SSE pass‑through + header‑hash stickiness**

metadata:

annotations:

nginx.ingress.kubernetes.io/proxy-read-timeout: "3600"

nginx.ingress.kubernetes.io/proxy-send-timeout: "3600"

nginx.ingress.kubernetes.io/proxy-buffering: "off" # critical for SSE

nginx.ingress.kubernetes.io/upstream-hash-by: "$http\_mcp\_session\_id" # sticky by header

nginx.ingress.kubernetes.io/configuration-snippet: |

proxy\_set\_header Connection "";

chunked\_transfer\_encoding off;

**Note:** Clients must send Mcp-Session-Id on streamed calls and subscriptions; the API echoes/assigns the header per FR‑003. Sticky routing keeps the same session on the same pod during its lifetime.

**3.2 Envoy (example) — disable stream idle timeout for SSE**

http\_connection\_manager:

stream\_idle\_timeout: 0s

**4) Pod‑Level Draining & Probes (SSE‑aware)**

* **Readiness probe** flips **false before termination** to quiesce new connections.
* **Liveness probe** remains **true** during drain to avoid hard kills.
* **PreStop hook**: allow final heartbeats/frames to flush.
* **terminationGracePeriodSeconds**: set to cover at least one Network:SseKeepAliveSeconds cycle × 2.

**Deployment fragment (illustrative):**

spec:

terminationGracePeriodSeconds: 60

containers:

- name: api

lifecycle:

preStop:

exec:

command: ["sh","-c","/app/graceful-drain.sh || sleep 10"]

readinessProbe:

httpGet: { path: /api/ready, port: 8080 }

periodSeconds: 5

livenessProbe:

httpGet: { path: /api/healthz, port: 8080 }

periodSeconds: 10

**5) HPA & PDB (templates)**

**5.1 HorizontalPodAutoscaler (autoscaling/v2)**

apiVersion: autoscaling/v2

kind: HorizontalPodAutoscaler

metadata:

name: mcp-proxy

spec:

scaleTargetRef:

apiVersion: apps/v1

kind: Deployment

name: mcp-proxy

minReplicas: 2

maxReplicas: 10

metrics:

- type: Resource

resource:

name: cpu

target:

type: Utilization

averageUtilization: 70

# Optional: custom metric for sessions/pod

# - type: Pods

# pods:

# metric:

# name: session\_count

# target:

# type: AverageValue

# averageValue: "180"

**5.2 PodDisruptionBudget (keep ≥1 available)**

apiVersion: policy/v1

kind: PodDisruptionBudget

metadata:

name: mcp-proxy-pdb

spec:

minAvailable: 1

selector:

matchLabels:

app: mcp-proxy

HPA scale‑in can cause **SSE truncation** if readiness/drain is misconfigured. Always validate with a **streaming probe** before and after scaling operations.

**6) Operating Procedures**

**6.1 Manual scale‑out (immediate relief)**

kubectl -n <ns> scale deploy mcp-proxy --replicas=<n>

kubectl -n <ns> rollout status deploy/mcp-proxy --timeout=5m

**Validate:** /ready green; **SSE TTFB p95 ≤ 200 ms**; heartbeat gap near configured cadence; JSON latency within budgets.

**6.2 Controlled scale‑in (SSE‑aware drain)**

1. **Set replica goal**: reduce by 1 each step.
2. **Confirm readiness flips** to **false** on terminating pod, then wait for **active streams** to drain (watch app logs/metrics).
3. **Ensure PDB** prevents multiple pods down simultaneously.
4. **Validate**: run streaming probe; confirm no truncation and budgets remain met.

**6.3 Canary scaling (optional, Prod)**

* Increase replicas by **+1**; route a fraction of **new sessions** to the new pod; watch **TTFB** and **error codes** for 10–15 min before further scaling.

**7) Verification & Evidence (attach to Release)**

* **Before/after**: /ready, /healthz, /config/effective (non‑secret) snapshots.
* **Perf**: JSON p50/p95; **SSE TTFB p50/p95**; heartbeat cadence.
* **Capacity**: sessions/replica, CPU/memory headroom, child restart counts.
* **Ingress settings**: prove proxy\_buffering off (NGINX) or equivalent.
* Upload to **Evidence Pack** and retain **≥ 1 year**.

**8) Troubleshooting (quick checks)**

| **Symptom** | **Likely Cause** | **What to check / do** |
| --- | --- | --- |
| **TTFB p95 > 200 ms** | Under‑provisioned replicas; ingress buffering | Scale out; verify proxy\_buffering off; confirm CPU ceiling |
| Heartbeats irregular/gapped | Ingress timeouts too low; overloaded pod | Increase read/idle timeouts; add replicas; verify Network:SseKeepAliveSeconds |
| Many bad\_gateway\_child\_unavailable | Child crashing, misconfig command/args | Check Mcp:Child\* in /config/effective; inspect child\_restart\_count |
| Sessions move between pods | Sticky routing off/mis‑keyed | Confirm upstream-hash-by: "$http\_mcp\_session\_id"; verify clients send Mcp-Session-Id |
| Readiness flapping during scale | DB SP hiccups; child spawn probe slow | Inspect readiness logs; ensure restart‑to‑ready ≤ 30 s; scale gradually |

**9) Capacity Planning (rules of thumb)**

* Start with **2 replicas** and raise in steps of **+1 or +2** while monitoring **SSE TTFB** and **session\_count**.
* Maintain **headroom**: aim for ≤ 70% CPU and ≤ 75% of the **200 sessions/replica** design point under normal peaks.
* For RTM, keep capacity sufficient to validate against **Prod DB (read‑only)** without impacting Prod.

**10) RACI (Scale‑Out)**

| **Activity** | **A** | **R** | **C** | **I** |
| --- | --- | --- | --- | --- |
| Scale policy & thresholds | DoSE | SRE Lead | Dev Lead, QA Lead | SecLead |
| HPA/PDB configuration | DoSE | SRE | Dev Lead | QA |
| Ingress sticky/stream settings | DoSE | SRE | Dev Lead | QA |
| Verification & evidence | DoSE | SRE/QA | DocFactory | All |

**11) References**

* **Monitoring & SLOs** — TTFB, heartbeat, availability, 24‑h checks.
* **Deploy/Rollback Runbooks** — promotion, **graceful SSE drain**, image flip.
* **Compliance** — secrets policy, DB/SP rules, No‑Hard‑Coding.
* **OpenAPI** — transport headers, error envelope, SSE examples.

**12) Assumptions**

1. Ingress respects **text/event-stream** and configured timeouts; no buffering.
2. Sticky routing uses the Mcp‑Session‑Id header (or a derived affinity cookie) for consistent hashing.
3. **RTM** validates against **Prod DB (read‑only)** and never performs writes.

**13) Next Steps**

* Add **session\_count** custom metric to HPA (optional) and rehearse **scale‑in drain** quarterly.
* Bake **streaming probes** (TTFB & heartbeat) into deployment verification.
* Include an **ingress conformance** check in CI/CD to assert required SSE annotations.

**Footer (optional for Word header/footer):**  
*MCPX‑KendoBridge • Scale‑Out & Capacity Runbook • v2.0.0 • 2025‑09‑27 • Confidential — Technijian Internal*