InterSystems
Developer Community

↳WEBINAR
15 DE JUNIO
16:00 MADRID (CEST)

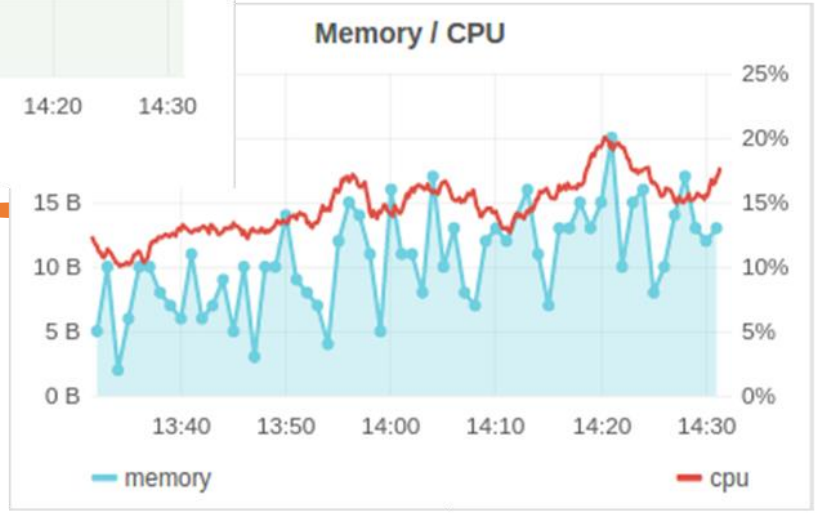
# SAM

## Monitorizando InterSystems IRIS con Grafana y Prometheus

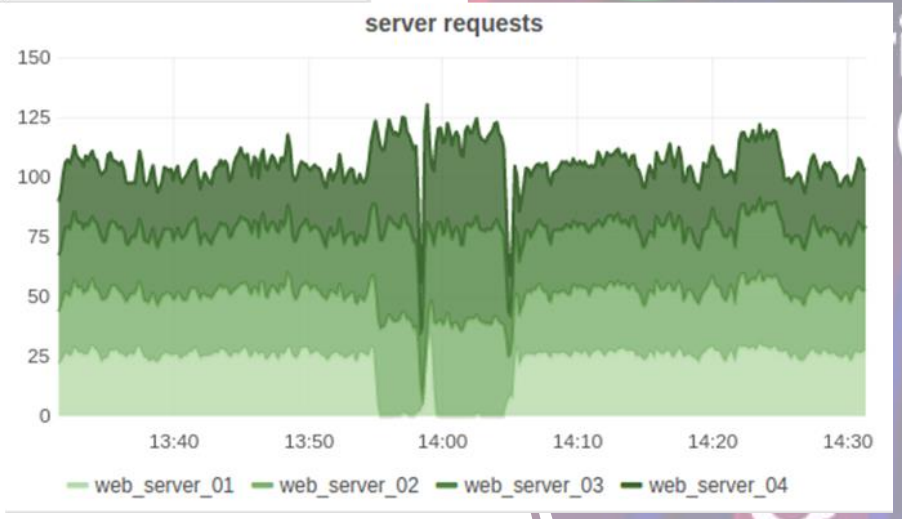
https://github.com/es-comunidad-intersystems/webinar-sam.git

Pierre-Yves Duquesnoy
Sales Engineer, InterSystems Iberia


logins
80
60
40
20
0
13:40
13:50
14:00
14:10
14:20
14:30
logins
logins (-1 hour)

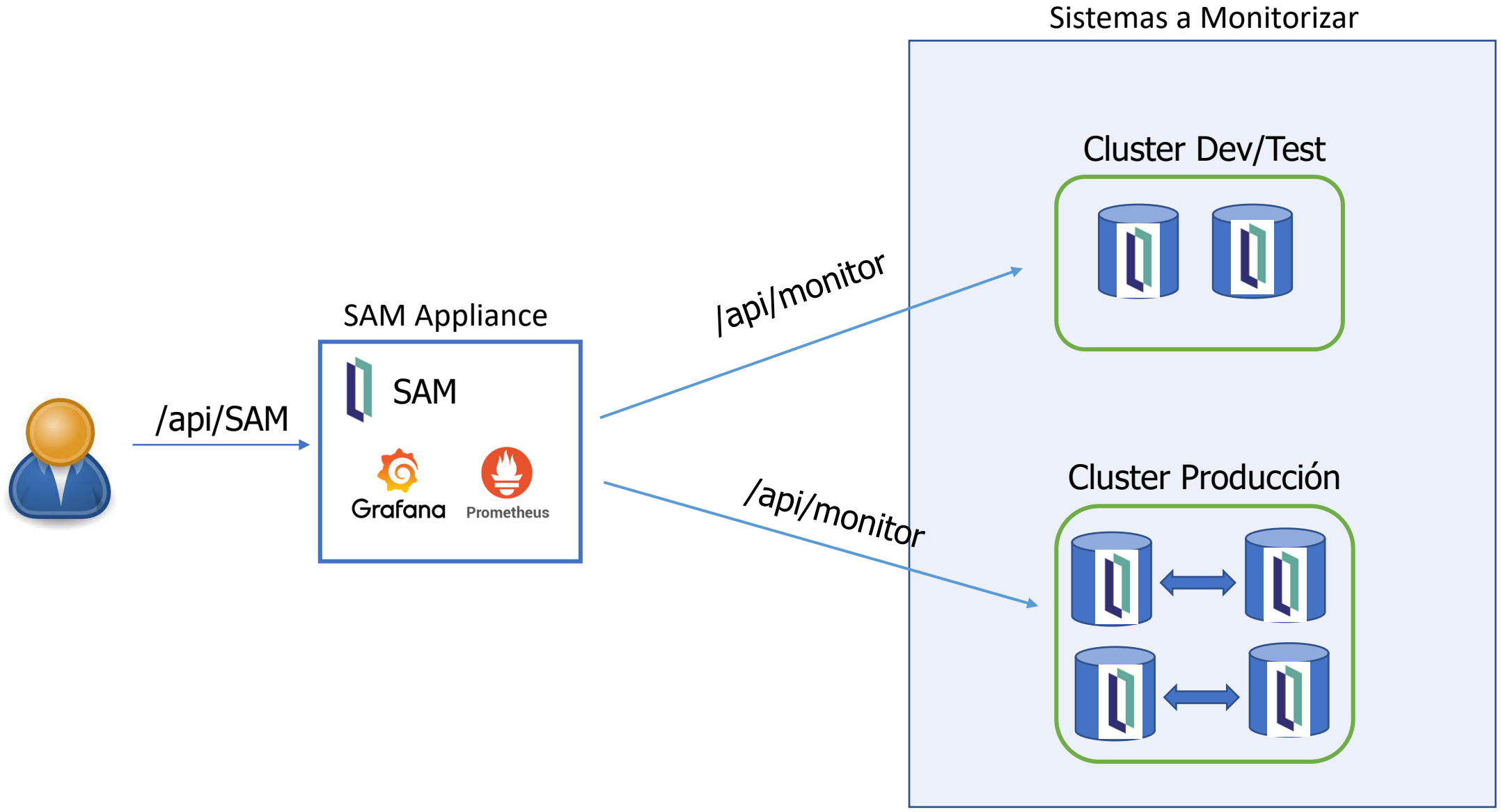

Memory / CPU
25%
20%
15%
10%
5%
0%
15 B
10 B
5 B
0 B
13:40
13:50
14:00
14:10
14:20
14:30
memory
cpu


Grafana
Prometheus


server requests
150
125
100
75
50
25
0
13:40
13:50
14:00
14:10
14:20
14:30
web_server_01
web_server_02
web_server_03
web_server_04


SAM
izando InterSystems IR
Grafana y Prometheus
PIERRE-YVES DUQUESNOY
Senior Sales Engineer
de InterSystems Iberia

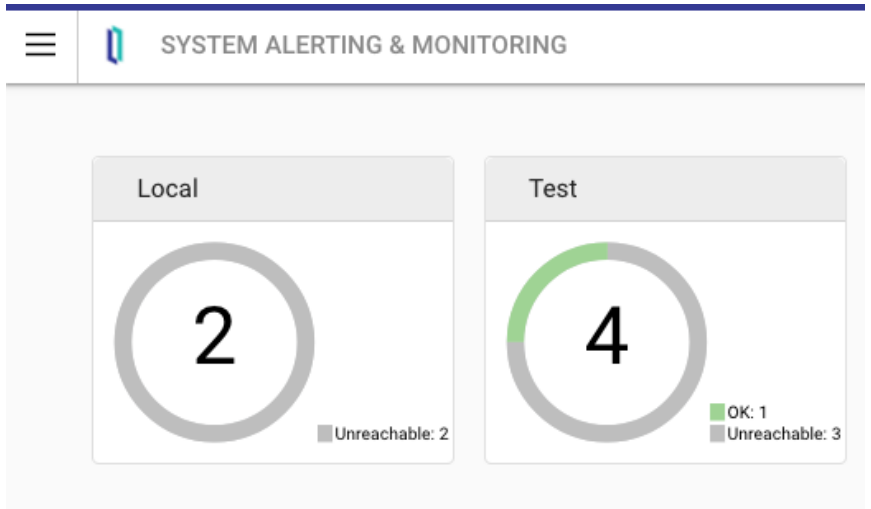
# Vista General



# System Alerting & Monitoring (SAM)

## ¿Que es?

- una herramienta unificada de monitorización de clusters




## ¿Beneficios?

- Un único panel para todas las instancias de IRIS
- Agrupación de las instancias en Clusters
- Implementación simplificada
- Versión “Community” que se puede extender a Enterprise
- Metrics pre-construidas de IRIS: nada que instalar en las instancias a monitorizar (IRIS 2020.1+)

## ¿Como?

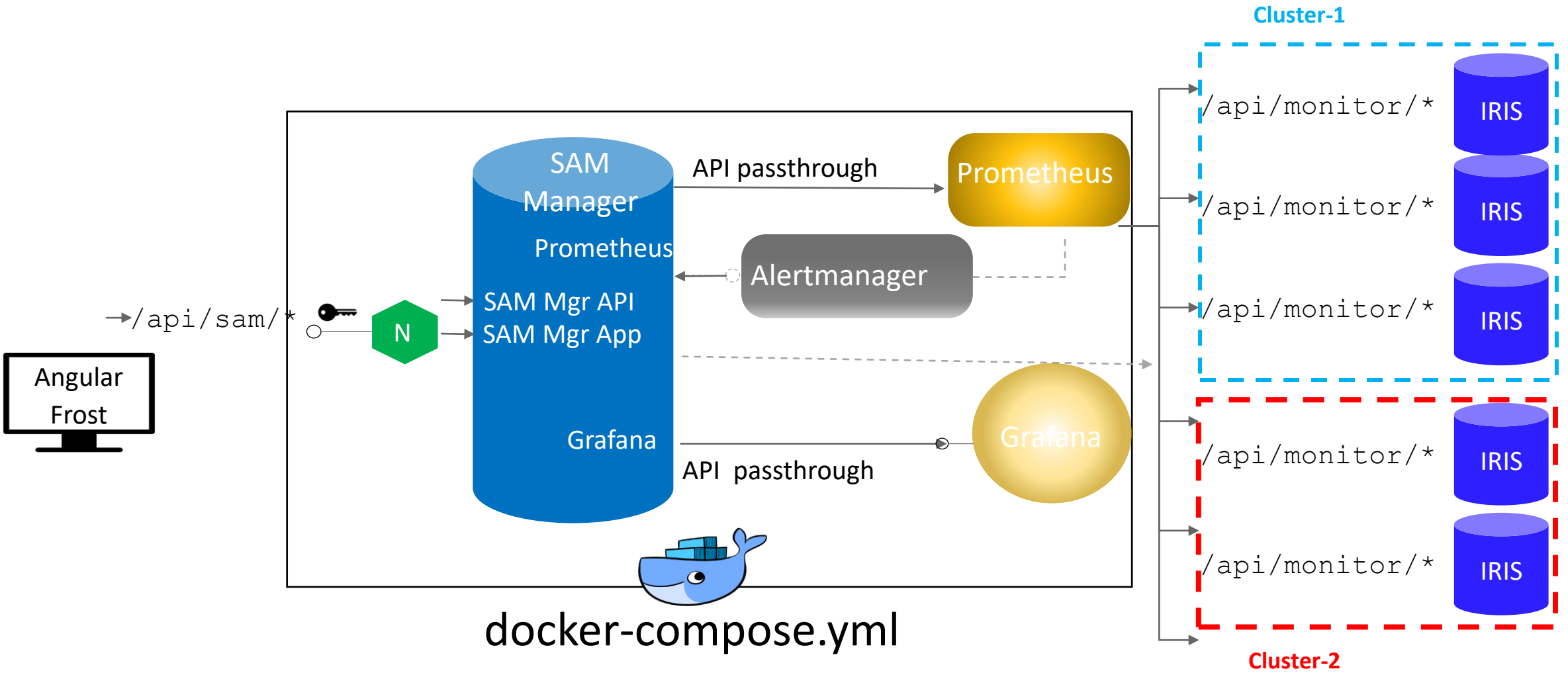
- Usando software Open Source
  - Prometheus – un proyecto CNCF
  - Grafana – El visualizado de metricas más utilizado
  - Agrupados en un docker-compose

# Que es System Alerting and Monitoring?

Una solución abierta para monitorizar InterSytem IRIS, proporcionando

- Un Interfaz Web
  - Definición de Clusters y Instancias
  - Monitorizacion del estado de las instancias
  - Visión de alertas
  - Definición de reglas de metricas
  - Visualización de las metricas
- Apoyado en componentes open source bien mantenidos y documentados
  - Alertmanager v0.20.0
  - Grafana v6.7.1
  - Nginx 1.17.9
  - Prometheus v2.17.1
  - SAM Manager 1.0
  - Extensible

# SAM – Detalles internos



# SAM en InterSystems IRIS

- Incluido y habilitado en IRIS 2020.1+
  - Más funcionalidades en 2021+
- Construido encima de las herramientas existentes
  - System Monitor
    - Habilidato por defecto, Estado del Sistema “OK”,”WARNING”,”CRITICAL” +System Metrics (CPU, Lock Table…)
- Exportador para Prometheus
  - REST API
    - /api/monitor/metrics
    - /api/monitor/alerts

# InterSystems IRIS: System Monitor

- Mantiene un único valor de salud del sistema
  - $System.Monitor.State()
  - Habilitado de fabrica
  - Indicadores de estado y uso de recursos (lck-table-%full, CPU-warning, ECP, etc.)
- Genera notificaciones y alertas
  - System Alerts (severity 2) Warnings (severity 1), OK (severity 0)
    - Reglas fijas para evaluar los valors y identificar desviaciones
    - Escribe en messages.log
  - Log Monitor (^MONITOR), Habilitado de fabrica
    - messages.log
    - alerts.log
  - SYS.Monitor.SystemSensors
    - SYS.Monitor.DashboardSensors
      - SYS.Monitor.SAM.Sensors – Recoje sensores disponibles sin almacenarlos; Interogado por SAM Client
        - GetSensors() cada 30 segundos

# InterSystems IRIS: Metricas de Servidor

- /api/monitor/metrics
  - Simple Clave/Valor
    - Iris_cpu_usage, iris_glo_ref_per_sec, iris_db_latency …
  - Permite definición de Metricas de aplicación Adicionales

# Metrica a Medida

- SubClase de %SYS.Monitor.SAM.Abstract
- Agrupar las métricas en un “PRODUCT”
- Implementar GetSensors()
  - SetSensor(Sensor , Value , Item As %String = "")
    - Do ..SetSensor(“TubesReceived”,+$Get(^User.TubesReceivedD))
- Añadir Clase a la configuración

```
%SYS>Do ##class(SYS.Monitor.SAM.Config).AddApplicationClass(ClassName,Namespace)
```

- **Añadir Permisos** a Aplicación Web /api/monitor
  - Ejecutar código y acceder a datos en “Namespace”
- Probar con http://ServerIP:Port/csp/monitor/metrics

# SAM Appliance (cliente de Monitorización)

- Contenedores Docker arrancados con docker-compose
- SAM Manager
  - Instancia IRIS, con la Applicación SAM principal
  - Mantiene Historial de Datos
- Nginx
  - Servidor Web, control de acesssos
- Prometheus
  - BBDD TimeSeries, recogida de datos, visualización, integración
- Grafana
  - Visualization
- Alert Manager
  - Deduplication, Enrutamiento de alertas

# SAM: Licenciamiento

## SAM Community Edition

- 5 usuarios concurrentes de monitorización
- 8 cores, 10 GB de BBDD
- Suficiente para monitorizar 40 instancias de InterSystems IRIS

## • SAM Enterprise

- Usuarios y Cores según Licencia
- Permite InterSystems Mirroring (para Alta Disponibilidad del propio SAM)

# SAM: Instalación

- Descargar SAM
  - WRC->Componentes: sam-<version>.tar.gz
  - Github: https://github.com/intersystems-community/sam
- Decomprimir con:
  - `tar zpxvf sam-<version>.tar.gz`
- Arrancar: `./start.sh`
  - Verificar permisos RW en ./config y subdirectorios.
  - Nota: modificar Docker-compose para Docker 20.10.14+ [ command: --check-caps false]
- Acceder:
  - `http://localhost:8080/api/sam/app/index.csp`
- Parar: `./stop.sh`

# SAM: primeros pasos

- Cluster: Grupo de Instancias a gestionar juntas
- Desde el portal: definir el cluster y añadir Instancias IRIS
  - Instancias IRIS 2020.1+ pre-configuradas para ser accesibles en la APIs: /api/monitor
    - Verificar que acceso sin Authenticar esta permitido
- Añadir graficas al Dashboard de Grafana
- Ajustar la configuración básica
  - Numero de días (1 to 30) de datos históricos a almacenar

# SAM: Reglas de alertas de Prometheus

- Para generar Alertas según condiciones
  - Prometheus Query Language (PromQL)
  - https://prometheus.io/docs/prometheus/latest/querying/basics/
  - Sintaxis:

```
metric_name{cluster="cluster_name",label(s)}>value
```

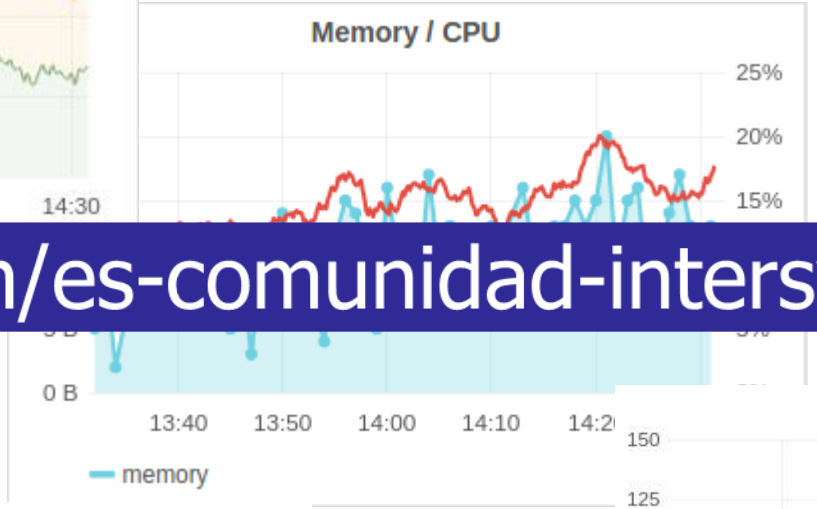
  - Permite Operadores logicos y numericos

```
iris_cpu_usage{cluster="test"}>90

(iris_db_size_mb{cluster="test",id="USER"}/iris_db_max_size_mb{cluster="test",id="USER"})*100>90

iris_ecp_conn{cluster="production"}<1 or iris_ecp_conn{cluster="production"}>20

iris_system_alerts_new{cluster="test"}>=1 and iris_system_monitor_health_state{cluster="test"}!=0
```

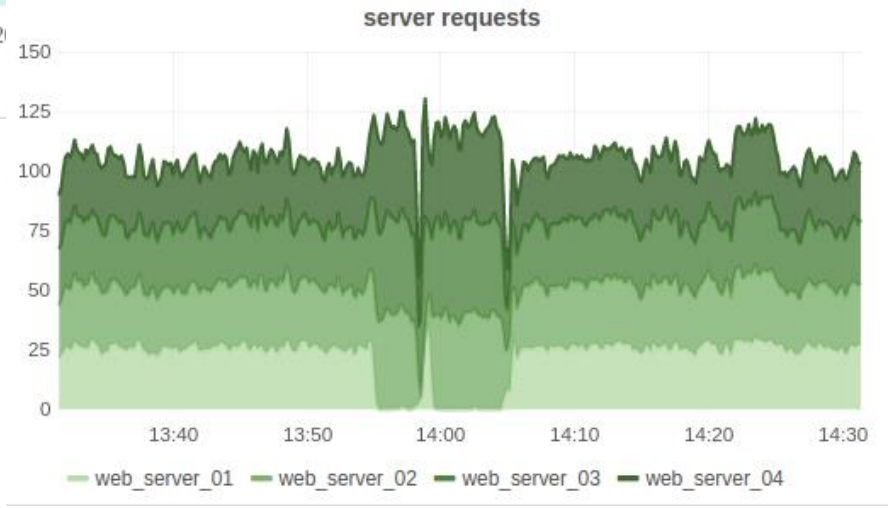
# SAM: Gestionar Alertas en SAM

- Permite Customizar las acciones a realizar en caso de Alerta en SAM
- SubClase de %SAM.AbstractAlertHandler
  - Método HandleAlerts()
- Permite enviar correo desde SAM en caso de alerta
- Desarrollo del AbtractHandler.
  - Realizar el desarrollo fuera de SAM e importar en SAM via portal

# SAM: Demo


https://github.com/es-comunidad-intersystems/webinar-sam.git




Grafana

Prometheus