



TI EN MAPAS

+300 MAPAS MENTALES
DE TECNOLOGÍA DE LA INFORMACIÓN

APRENDE Y REPASA TI DE FORMA VISUAL

TI EN MAPAS

+300 MAPAS MENTALES
DE TECNOLOGÍA DE LA INFORMACIÓN

ESTUDIO VISUAL

IDEAL PARA CONCURSOS

CONOCIMIENTO ORGANIZADO

REDES DE COMPUTADORAS

- Conceptos básicos
 - Qué es una red
 - Tipos de red
 - Topologías
 - Modelos de referencia
- Dispositivos
 - Router
 - Switch
 - Access Point
 - Firewall
- Protocolos
 - TCP/IP
 - Direcciones IP
 - Puertos
 - Protocolos de aplicación
- Seguridad
 - Políticas de seguridad
 - Cifrado
 - Control de acceso
 - Buenas prácticas

SISTEMAS OPERATIVOS

- Conceptos
 - Qué es un SO
 - Funciones
 - Tipos de SO
 - Arquitectura
- Administración
 - Usuarios y permisos
 - Gestión de procesos
 - Gestión de memoria
 - Sistema de archivos
- Windows
 - Comandos principales
 - Herramientas de administración
 - Políticas de grupo
 - Seguridad
- Linux
 - Comandos básicos
 - Estructura de directorios
 - Permisos
 - Shell y scripting

Modelado

- Conceptos
 - Qué es una BD
 - Tipos de BD
 - Modelo de datos
 - Lenguaje SQL
- Modelo relacional
 - Entidades y relaciones
 - Normalización
 - Diagrama ER
- SQL
 - Consultas básicas
 - JOINs
 - Subconsultas
 - Funciones
- Administración
 - Respaldos
 - Seguridad
 - Rendimiento
 - Buenas prácticas

+300 MAPAS MENTALES
EN ALTA CALIDAD

IDEAL PARA CONCURSOS DE TI
Y ESTUDIOS EN GENERAL

TEMAS ORGANIZADOS
Y ACTUALIZADOS

ACCESO INMEDIATO
EN TODOS TUS DISPOSITIVOS

GUÍA MAESTRA DE ESTUDIO

+345 mapas mentales • 15 áreas de TI • rutas, práctica y revisión

Cómo aprovechar TI EN MAPAS

TI EN MAPAS fue pensado como una biblioteca visual de fundamentos. Su objetivo es ayudarte a entender relaciones, repasar conceptos y detectar rápidamente qué temas necesitas profundizar.

No intentes memorizar 345 mapas. Usa el material como un sistema de navegación: estudia un tema, explícalo con tus palabras y después aplícalo en un ejercicio, laboratorio, comando, consulta SQL o pequeño proyecto.

La regla del producto es simple: VER → EXPLICAR → APLICAR → REPASAR.

Método VEAR

PASO	ACCIÓN
VER	Lee el mapa completo y observa cómo se conectan los conceptos.
EXPLICAR	Cierra el mapa y explica el tema como si enseñaras a otra persona.
APLICAR	Haz un ejemplo real: comando, ejercicio, código, consulta o diagnóstico.
REPASAR	Vuelve al mapa 24 h, 7 días y 30 días después.

Qué incluye

- 345 mapas mentales en PNG organizados por 15 módulos
- 5 rutas de estudio según objetivo
- Plan de 30 días
- Glosario de conceptos
- Checklist para entrevistas técnicas
- Índice completo para localizar cualquier mapa

Módulo 01 — Fundamentos de TI

Construir vocabulario base y entender cómo se relacionan datos, sistemas y servicios.

Mapas incluidos

Tecnología de la Información	Dato vs. información
Sistema de información	Hardware vs. software
Bits, bytes y unidades	Sistema binario
Hexadecimal	Codificación ASCII y Unicode
Arquitectura cliente-servidor	Computación distribuida
Procesamiento por lotes	Procesamiento en tiempo real
Disponibilidad y confiabilidad	Escalabilidad
Rendimiento y latencia	Interoperabilidad
Licencias de software	Software libre y código abierto
SaaS, PaaS e IaaS	Ciclo de vida de la información

Práctica recomendada

1. Explica la diferencia entre dato e información.
2. Convierte unidades básicas: bits, bytes, KB, MB y GB.
3. Dibuja un ejemplo simple de arquitectura cliente-servidor.

Autoevaluación rápida

- ¿Qué diferencia existe entre hardware y software?
- ¿Qué significa escalabilidad?
- ¿Qué aporta Unicode?
- ¿Qué diferencia hay entre SaaS, PaaS e IaaS?
- ¿Qué significa disponibilidad?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 02 — Hardware y Arquitectura

Entender los componentes físicos y qué afecta el rendimiento de un equipo.

Mapas incluidos

CPU	Núcleos e hilos
Frecuencia de CPU	Memoria RAM
Memoria caché	ROM y firmware
Placa madre	Chipset
Bus de datos	GPU
HDD	SSD SATA
SSD NVMe	RAID
Fuente de alimentación	BIOS y UEFI
Periféricos de entrada	Periféricos de salida
Puertos USB	HDMI y DisplayPort
Arquitectura x86 y ARM	Virtualización por hardware

Práctica recomendada

4. Compara un HDD, SSD SATA y SSD NVMe.
5. Identifica CPU, RAM, almacenamiento y puertos de tu equipo.
6. Explica cuándo RAID ayuda y cuándo no reemplaza un backup.

Autoevaluación rápida

- ¿Qué hace la CPU?
- ¿Para qué sirve la memoria caché?
- ¿Qué diferencia hay entre RAM y almacenamiento?
- ¿Qué es UEFI?
- ¿Qué función cumple una GPU?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 03 — Sistemas Operativos

Comprender cómo el SO gestiona procesos, memoria, archivos, usuarios y servicios.

Mapas incluidos

Funciones de un sistema operativo	Kernel
Procesos	Hilos
Planificación de CPU	Estados de un proceso
Memoria virtual	Paginación
Sistema de archivos	Permisos de archivos
Usuarios y grupos	Windows: conceptos básicos
Linux: conceptos básicos	macOS: conceptos básicos
CLI vs. GUI	Shell
Bash	PowerShell
Variables de entorno	Servicios y demonios
Gestión de paquetes	Logs del sistema
Arranque del sistema	Contenedores vs. máquinas virtuales

Práctica recomendada

7. Abre el monitor de procesos de tu sistema y observa CPU/RAM.
8. Crea una carpeta y practica permisos básicos.
9. Localiza logs del sistema y explica para qué sirven.

Autoevaluación rápida

- ¿Qué es un kernel?
- ¿Qué diferencia hay entre proceso e hilo?
- ¿Qué es memoria virtual?
- ¿Qué es un sistema de archivos?
- ¿Qué diferencia hay entre CLI y GUI?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 04 — Redes de Computadoras

Seguir el recorrido de un dato desde un dispositivo hasta otro y reconocer protocolos.

Mapas incluidos

Modelo OSI	Modelo TCP/IP
Dirección IPv4	Dirección IPv6
Máscara de subred	CIDR
Subnetting	Gateway predeterminado
MAC address	ARP
DNS	DHCP
NAT	TCP
UDP	HTTP
HTTPS	FTP y SFTP
SSH	SMTP
POP3 e IMAP	ICMP
Puertos de red	Router
Switch	Access Point
VLAN	VPN
Wi-Fi 2.4 vs. 5 GHz	Topologías de red
LAN, MAN y WAN	Balanceo de carga

Práctica recomendada

10. Consulta tu IP, gateway y DNS.
11. Prueba ping, traceroute/tracert y nslookup.
12. Diseña una red doméstica con router, switch y access point.

Autoevaluación rápida

- ¿Qué diferencia hay entre TCP y UDP?
- ¿Qué hace DNS?
- ¿Qué hace DHCP?
- ¿Qué es una VLAN?
- ¿Para qué sirve NAT?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 05 — Ciberseguridad

Comprender riesgo, identidad, cifrado, controles y amenazas frecuentes.

Mapas incluidos

Confidencialidad, integridad y disponibilidad	Amenaza, vulnerabilidad y riesgo
Autenticación vs. autorización	MFA
Contraseñas seguras	Hashing
Cifrado simétrico	Cifrado asimétrico
PKI y certificados digitales	TLS
Firewall	IDS e IPS
Antivirus y EDR	Phishing
Ingeniería social	Malware
Ransomware	SQL Injection
XSS	CSRF
Principio de mínimo privilegio	Zero Trust
Backups 3-2-1	Gestión de parches
SIEM	Respuesta a incidentes
Seguridad en Wi-Fi	OWASP Top 10

Práctica recomendada

- Activa MFA en una cuenta de prueba.
- Clasifica 5 ejemplos entre amenaza, vulnerabilidad y riesgo.
- Revisa una URL sospechosa y enumera señales de phishing.

Autoevaluación rápida

- ¿Qué significa CIA?
- ¿Qué diferencia hay entre hashing y cifrado?
- ¿Qué es mínimo privilegio?
- ¿Qué es Zero Trust?
- ¿Qué protege TLS?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 06 — Bases de Datos y SQL

Modelar datos y ejecutar consultas básicas con criterio de integridad.

Mapas incluidos

Base de datos relacional	Tabla, fila y columna
Clave primaria	Clave foránea
Normalización	1FN, 2FN y 3FN
Índices	Transacciones
ACID	SELECT
WHERE	ORDER BY
GROUP BY	HAVING
JOIN	INNER JOIN
LEFT JOIN	INSERT
UPDATE	DELETE
Funciones agregadas	Subconsultas
Vistas	Procedimientos almacenados
NoSQL	

Práctica recomendada

16. Diseña dos tablas relacionadas con clave primaria y foránea.
17. Escribe SELECT con WHERE y ORDER BY.
18. Practica INNER JOIN y LEFT JOIN.

Autoevaluación rápida

- ¿Qué es una clave primaria?
- ¿Qué significa ACID?
- ¿Para qué sirve un índice?
- ¿Qué hace GROUP BY?
- ¿Qué diferencia hay entre SQL y NoSQL?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 07 — Programacion y Algoritmos

Aprender lógica, estructuras de datos y conceptos esenciales de programación.

Mapas incluidos

Algoritmo	Pseudocódigo
Variables	Tipos de datos
Operadores	Condicionales
Bucles	Funciones
Parámetros y retorno	Scope
Arrays	Listas
Pilas	Colas
Diccionarios y mapas	Conjuntos
Recursión	Búsqueda lineal
Búsqueda binaria	Ordenamiento
Complejidad Big O	POO
Clases y objetos	Herencia
Polimorfismo	Manejo de excepciones

Práctica recomendada

19. Escribe pseudocódigo para validar una contraseña.
20. Implementa búsqueda lineal en el lenguaje que prefieras.
21. Compara $O(n)$ y $O(\log n)$ con un ejemplo.

Autoevaluación rápida

- ¿Qué es un algoritmo?
- ¿Qué diferencia hay entre variable y tipo de dato?
- ¿Qué es scope?
- ¿Qué significa recursión?
- ¿Qué pilares se asocian a POO?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 08 — Desarrollo Web y APIs

Entender cómo navegador, servidor, HTTP, APIs y datos trabajan juntos.

Mapas incluidos

Cómo funciona la Web	HTML
CSS	JavaScript
DOM	Frontend vs. backend
Cliente y servidor	URL
Métodos HTTP	Códigos de estado HTTP
Cookies	Sesiones
Local Storage	API
REST	JSON
XML	Autenticación con token
JWT	CORS
Webhooks	Arquitectura MVC

Práctica recomendada

22. Inspecciona una petición HTTP en las herramientas del navegador.
23. Crea una página HTML mínima con CSS.
24. Consume una API pública y observa JSON.

Autoevaluación rápida

- ¿Qué diferencia hay entre frontend y backend?
- ¿Qué es REST?
- ¿Qué es CORS?
- ¿Qué diferencia hay entre cookie y sesión?
- ¿Qué es un webhook?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 09 — Git DevOps y CI/CD

Usar control de versiones y comprender automatización de build, test y deploy.

Mapas incluidos

Control de versiones	Git
Repositorio local y remoto	Commit
Branch	Merge
Rebase	Pull Request
Git clone, pull y push	Resolución de conflictos
GitHub, GitLab y Bitbucket	DevOps
CI/CD	Pipeline
Build	Testing automatizado
Deploy	Infraestructura como código
Observabilidad	Rollback

Práctica recomendada

25. Crea un repositorio y realiza tres commits.
26. Crea una branch, modifica un archivo y haz merge.
27. Dibuja un pipeline con build, test y deploy.

Autoevaluación rápida

- ¿Qué representa un commit?
- ¿Qué diferencia hay entre merge y rebase?
- ¿Qué es CI/CD?
- ¿Qué es rollback?
- ¿Para qué sirve un Pull Request?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 10 — Cloud y Virtualizacion

Comprender modelos cloud, máquinas virtuales, contenedores y escalabilidad.

Mapas incluidos

Cloud Computing	IaaS
PaaS	SaaS
Nube pública, privada e híbrida	Regiones y zonas de disponibilidad
Elasticidad	Alta disponibilidad
Escalado vertical y horizontal	Serverless
Máquinas virtuales	Hipervisor
Contenedores	Docker
Imágenes y contenedores	Dockerfile
Docker Compose	Kubernetes
Pods	Servicios en Kubernetes
Almacenamiento en la nube	CDN
Object Storage	Disaster Recovery

Práctica recomendada

28. Compara IaaS, PaaS y SaaS con ejemplos.
29. Dibuja una arquitectura web con alta disponibilidad.
30. Explica diferencias entre VM y contenedor.

Autoevaluación rápida

- ¿Qué es elasticidad?
- ¿Qué diferencia hay entre escalado vertical y horizontal?
- ¿Qué es serverless?
- ¿Qué hace Kubernetes?
- ¿Qué es una CDN?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 11 — Ingeniería de Software y Testing

Mejorar calidad, diseño, pruebas y mantenibilidad del software.

Mapas incluidos

Ciclo de vida del software	Requisitos funcionales y no funcionales
Casos de uso	Historias de usuario
Arquitectura de software	Patrones de diseño
SOLID	Clean Code
Refactoring	Deuda técnica
Pruebas unitarias	Pruebas de integración
Pruebas de sistema	Pruebas de aceptación
TDD	BDD
Mocking	QA vs. Testing
Versionado semántico	Documentación técnica
UML	Mantenimiento de software

Práctica recomendada

31. Escribe una historia de usuario con criterio de aceptación.
32. Diseña una prueba unitaria para una función simple.
33. Identifica un ejemplo de deuda técnica.

Autoevaluación rápida

- ¿Qué es refactoring?
- ¿Qué son requisitos no funcionales?
- ¿Qué diferencia hay entre prueba unitaria e integración?
- ¿Qué es TDD?
- ¿Qué significa SOLID?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 12 — Datos IA y BI

Entender cómo se transforman datos en métricas, análisis y modelos.

Mapas incluidos

Data Analytics	Business Intelligence
ETL	ELT
Data Warehouse	Data Lake
Data Mart	Big Data
Las 5V de Big Data	OLTP vs. OLAP
KPIs	Dashboard
Machine Learning	Aprendizaje supervisado
Aprendizaje no supervisado	Clasificación
Regresión	Clustering
Overfitting y underfitting	Redes neuronales
IA generativa	Prompt engineering

Práctica recomendada

34. Define un KPI para una tienda online.
35. Dibuja un flujo ETL simple.
36. Separa un ejemplo de clasificación y uno de regresión.

Autoevaluación rápida

- ¿Qué diferencia hay entre Data Lake y Data Warehouse?
- ¿Qué es OLAP?
- ¿Qué es overfitting?
- ¿Qué diferencia hay entre supervisado y no supervisado?
- ¿Qué es IA generativa?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 13 — Gestion Agil ITIL y Gobernanza

Organizar trabajo, servicios, prioridades y responsabilidades con marcos conocidos.

Mapas incluidos

Gestión de proyectos	Alcance, tiempo y costo
Riesgos de proyecto	Stakeholders
Agile	Scrum
Roles de Scrum	Eventos de Scrum
Artefactos de Scrum	Kanban
WIP	Lean
MVP	OKR
KPI	ITIL 4
Gestión de incidentes	Gestión de problemas
Gestión de cambios	SLA
Gobernanza de TI	COBIT

Práctica recomendada

37. Crea un tablero Kanban con límite WIP.
38. Escribe una historia de usuario para un pequeño producto.
39. Define un SLA básico para soporte.

Autoevaluación rápida

- ¿Qué roles existen en Scrum?
- ¿Qué es WIP?
- ¿Qué diferencia hay entre incidente y problema?
- ¿Qué es un SLA?
- ¿Qué busca la gobernanza de TI?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 14 — Soporte y Troubleshooting

Resolver incidentes de forma metódica, documentada y reproducible.

Mapas incluidos

Método de troubleshooting	Diagnóstico por capas
Problemas de conectividad	Problemas de DNS
Problemas de DHCP	PC lento
Alto uso de CPU	Falta de memoria RAM
Disco lleno	Errores de arranque
Pantalla azul	Problemas de impresora
Permisos insuficientes	Aplicación no abre
Actualizaciones fallidas	Recuperación de archivos
Backup y restauración	Documentación de incidentes

Práctica recomendada

40. Simula un problema de DNS y registra pasos de diagnóstico.
41. Crea una plantilla de ticket con síntomas, pruebas y solución.
42. Diseña un árbol simple para “sin Internet”.

Autoevaluación rápida

- ¿Por qué conviene cambiar una variable a la vez?
- ¿Qué información debe contener un ticket?
- ¿Cómo separar síntoma de causa?
- ¿Qué revisar ante un PC lento?
- ¿Por qué probar la restauración de backups?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Módulo 15 — Carrera Certificaciones y Entrevistas

Convertir el estudio en una ruta profesional con evidencias y práctica.

Mapas incluidos

Ruta de entrada a TI	Help Desk
Soporte técnico	Administrador de sistemas
Administrador de redes	Analista de ciberseguridad
Desarrollador frontend	Desarrollador backend
Analista de datos	DevOps Engineer
Cloud Engineer	Portafolio de TI
CV para TI	LinkedIn para TI
Entrevista técnica	Método STAR
Certificaciones iniciales	Plan de estudio de 90 días

Práctica recomendada

43. Crea un CV de una página enfocado en TI.
44. Publica un pequeño proyecto en GitHub.
45. Practica tres respuestas con método STAR.

Autoevaluación rápida

- ¿Qué debe mostrar un portafolio?
- ¿Qué es el método STAR?
- ¿Cómo elegir una certificación inicial?
- ¿Qué conviene hacer si no sabes una respuesta técnica?
- ¿Cómo convertir estudio en evidencia práctica?

Regla para avanzar: si puedes explicar 4 de las 5 preguntas sin mirar, avanza; si no, repasa los mapas marcados.

Rutas de estudio

Desde cero: Fundamentos → Hardware → Sistemas Operativos → Redes → Seguridad → Bases de Datos → Programación → Web.

Soporte / Help Desk: Fundamentos → Hardware → Sistemas Operativos → Redes → Seguridad → Soporte → Carrera.

Desarrollo: Fundamentos → Programación → Bases de Datos → Web/APIs → Git/DevOps → Ingeniería de Software → Cloud.

Ciberseguridad: Fundamentos → Sistemas Operativos → Redes → Ciberseguridad → Cloud → Soporte → Gobernanza.

Datos e IA: Fundamentos → Bases de Datos → Programación → Datos/IA/BI → Cloud → Ingeniería de Software.

Plan de 30 días

Días 1-3: Fundamentos

Días 4-6: Hardware

Días 7-10: Sistemas Operativos

Días 11-14: Redes

Días 15-18: Ciberseguridad

Días 19-21: Bases de Datos

Días 22-24: Programación

Días 25-26: Web/APIs

Día 27: Git/DevOps

Día 28: Cloud

Día 29: Ingeniería + Datos/IA

Día 30: Revisión + entrevista

Ritual diario

- 5 mapas nuevos
- 5 mapas de revisión
- 3 preguntas hechas por ti mismo
- 1 ejemplo práctico
- 1 tema marcado para revisar en 7 días

Checklist de entrevista técnica

Antes

- Investigar empresa, producto y stack
- Revisar fundamentos de la vacante
- Preparar 3 proyectos o experiencias para explicar
- Practicar presentación de 60 segundos

Durante

- Aclarar requisitos antes de responder
- Pensar en voz alta
- Explicar trade-offs
- Si no sabes, explicar cómo lo investigarías
- Usar ejemplos reales

Preguntas base

- TCP vs. UDP
- Qué ocurre al escribir una URL
- DNS y DHCP
- Autenticación vs. autorización
- API REST
- ACID
- Git
- VM vs. contenedor
- Diagnóstico de conectividad
- Priorización de incidentes

TI EN MAPAS no promete empleo, certificación ni aprobación automática. El material organiza y acelera el estudio; el dominio real depende de práctica, proyectos y experiencia.

Glosario de Tecnología de la Información

Consulta rápida de términos y conceptos clave usados en los mapas. Úsalo para resolver dudas sin salir del material.

— 1FN, 2FN y 3FN

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

— Access Point

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— ACID

Propiedades que ayudan a mantener transacciones confiables en bases de datos.

— Actualizaciones fallidas

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Administrador de redes

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Administrador de sistemas

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Agile

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— Alcance, tiempo y costo

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— Algoritmo

Secuencia finita y ordenada de pasos para resolver un problema.

— Almacenamiento en la nube

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Alta disponibilidad

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Alto uso de CPU

Componente o tecnología de diagnóstico y resolución de incidentes con impacto directo en capacidad, rendimiento o conectividad.

— Amenaza, vulnerabilidad y riesgo

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— Analista de ciberseguridad

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Analista de datos

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Antivirus y EDR

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— API

Interfaz que permite que sistemas o componentes se comuniquen de forma definida.

— Aplicación no abre

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Aprendizaje no supervisado

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Aprendizaje supervisado

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— ARP

Concepto de comunicación dentro de comunicación entre dispositivos que define cómo se intercambia o interpreta información.

— Arquitectura cliente-servidor

Marco o concepto de conceptos base de tecnología que ayuda a organizar decisiones, procesos y buenas prácticas.

— Arquitectura de software

Marco o concepto de calidad y diseño de software que ayuda a organizar decisiones, procesos y buenas prácticas.

— Arquitectura MVC

Marco o concepto de funcionamiento de aplicaciones web que ayuda a organizar decisiones, procesos y buenas prácticas.

— Arquitectura x86 y ARM

Marco o concepto de componentes físicos y arquitectura que ayuda a organizar decisiones, procesos y buenas prácticas.

— Arranque del sistema

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Arrays

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Artefactos de Scrum

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— Autenticación con token

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

— Autenticación vs. autorización

Comparación entre conceptos relacionados de protección de sistemas y datos, destacando cuándo conviene cada uno.

- **Backend**
Parte de una aplicación que procesa lógica, datos y servicios en el servidor.
- **Backup y restauración**
Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.
- **Backups 3-2-1**
Estrategia de respaldo: 3 copias, 2 medios distintos y 1 copia fuera del sitio.
- **Balanceo de carga**
Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.
- **Base de datos relacional**
Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.
- **Bash**
Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.
- **BDD**
Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.
- **Big Data**
Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.
- **BIOS y UEFI**
Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.
- **Bits, bytes y unidades**
Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.
- **Branch**
Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.
- **Bucles**
Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.
- **Build**
Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.
- **Bus de datos**
Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.
- **Business Intelligence**
Componente o tecnología de uso de datos para análisis y automatización con impacto directo en capacidad, rendimiento o conectividad.
- **Búsqueda binaria**
Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Búsqueda lineal

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Cache

Almacenamiento temporal usado para acelerar accesos frecuentes.

— Casos de uso

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— CDN

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Certificaciones iniciales

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Chipset

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

— CI/CD

Prácticas que automatizan integración, pruebas y entrega/despliegue de software.

— Ciclo de vida de la información

Marco o concepto de conceptos base de tecnología que ayuda a organizar decisiones, procesos y buenas prácticas.

— Ciclo de vida del software

Marco o concepto de calidad y diseño de software que ayuda a organizar decisiones, procesos y buenas prácticas.

— CIDR

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— Cifrado asimétrico

Cifrado basado en un par de claves: pública y privada.

— Cifrado simétrico

Cifrado que usa la misma clave para cifrar y descifrar información.

— Clases y objetos

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Clasificación

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Clave foránea

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

— Clave primaria

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

- **Clean Code**
Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.
- **CLI vs. GUI**
Comparación entre conceptos relacionados de gestión de recursos del sistema, destacando cuándo conviene cada uno.
- **Cliente y servidor**
Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.
- **Cloud**
Modelo de consumo de recursos de computación a través de una red.
- **Cloud Computing**
Entrega bajo demanda de recursos de computación a través de una red.
- **Cloud Engineer**
Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.
- **Clustering**
Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.
- **COBIT**
Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.
- **Codificación ASCII y Unicode**
Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.
- **Colas**
Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.
- **Commit**
Registro de un conjunto de cambios en un sistema de control de versiones.
- **Complejidad Big O**
Notación que describe cómo crece el costo de un algoritmo según el tamaño de entrada.
- **Computación distribuida**
Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.
- **Condicionales**
Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.
- **Confidencialidad, integridad y disponibilidad**
Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.
- **Conjuntos**
Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Contenedores

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Contenedores vs. máquinas virtuales

Comparación entre conceptos relacionados de gestión de recursos del sistema, destacando cuándo conviene cada uno.

— Contraseñas seguras

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— Control de versiones

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— Cookies

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

— CORS

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

— CPU

Unidad que ejecuta instrucciones y coordina operaciones del sistema.

— CSRF

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— CSS

Lenguaje de estilos para controlar presentación y diseño visual en la Web.

— CV para TI

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Códigos de estado HTTP

Concepto de comunicación dentro de funcionamiento de aplicaciones web que define cómo se intercambia o interpreta información.

— Cómo funciona la Web

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

— Dashboard

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Data Analytics

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Data Lake

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Data Mart

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

- **Data Warehouse**
Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.
- **Dato vs. información**
Comparación entre conceptos relacionados de conceptos base de tecnología, destacando cuándo conviene cada uno.
- **DELETE**
Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.
- **Deploy**
Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.
- **Desarrollador backend**
Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.
- **Desarrollador frontend**
Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.
- **Deuda técnica**
Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.
- **DevOps**
Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.
- **DevOps Engineer**
Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.
- **DHCP**
Protocolo que asigna automáticamente configuración IP a los dispositivos.
- **Diagnóstico por capas**
Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.
- **Diccionarios y mapas**
Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.
- **Dirección IPv4**
Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.
- **Dirección IPv6**
Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.
- **Disaster Recovery**
Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.
- **Disco lleno**
Componente o tecnología de diagnóstico y resolución de incidentes con impacto directo en capacidad, rendimiento o conectividad.

— Disponibilidad y confiabilidad

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— DNS

Servicio que resuelve nombres de dominio y otros registros de red.

— Docker

Plataforma para empaquetar y ejecutar aplicaciones en contenedores aislados.

— Docker Compose

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Dockerfile

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Documentación de incidentes

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Documentación técnica

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— DOM

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

— Elasticidad

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— ELT

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Entrevista técnica

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Errores de arranque

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Escalabilidad

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— Escalado vertical y horizontal

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Estados de un proceso

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— ETL

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Eventos de Scrum

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— Falta de memoria RAM

Componente o tecnología de diagnóstico y resolución de incidentes con impacto directo en capacidad, rendimiento o conectividad.

— Firewall

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— Framework

Conjunto de herramientas y convenciones que facilita desarrollar software.

— Frecuencia de CPU

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

— Frontend

Parte de una aplicación con la que interactúa directamente el usuario.

— Frontend vs. backend

Comparación entre conceptos relacionados de funcionamiento de aplicaciones web, destacando cuándo conviene cada uno.

— FTP y SFTP

Concepto de comunicación dentro de comunicación entre dispositivos que define cómo se intercambia o interpreta información.

— Fuente de alimentación

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

— Funciones

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Funciones agregadas

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

— Funciones de un sistema operativo

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Gateway

Punto que conecta redes o sirve como salida hacia otra red.

— Gateway predeterminado

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— Gestión de cambios

Marco o concepto de gestión de trabajo y servicios que ayuda a organizar decisiones, procesos y buenas prácticas.

— Gestión de incidentes

Marco o concepto de gestión de trabajo y servicios que ayuda a organizar decisiones, procesos y buenas prácticas.

- **Gestión de paquetes**
Marco o concepto de gestión de recursos del sistema que ayuda a organizar decisiones, procesos y buenas prácticas.
- **Gestión de parches**
Marco o concepto de protección de sistemas y datos que ayuda a organizar decisiones, procesos y buenas prácticas.
- **Gestión de problemas**
Marco o concepto de gestión de trabajo y servicios que ayuda a organizar decisiones, procesos y buenas prácticas.
- **Gestión de proyectos**
Marco o concepto de gestión de trabajo y servicios que ayuda a organizar decisiones, procesos y buenas prácticas.
- **Git**
Sistema distribuido de control de versiones para rastrear cambios en código y colaborar.
- **Git clone, pull y push**
Elemento práctico de colaboración y automatización de entrega utilizado para ejecutar una operación concreta de forma controlada.
- **GitHub, GitLab y Bitbucket**
Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.
- **Gobernanza de TI**
Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.
- **GPU**
Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.
- **GROUP BY**
Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.
- **Hardware vs. software**
Comparación entre conceptos relacionados de conceptos base de tecnología, destacando cuándo conviene cada uno.
- **Hashing**
Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.
- **HAVING**
Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.
- **HDD**
Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.
- **HDMI y DisplayPort**
Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.
- **Help Desk**
Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Herencia

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Hexadecimal

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— Hilos

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Hipervisor

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Historias de usuario

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— HTML

Lenguaje de marcado que estructura el contenido de una página web.

— HTTP

Protocolo de aplicación usado para intercambio de recursos en la Web.

— HTTPS

HTTP protegido mediante TLS para aportar cifrado, integridad y autenticación.

— IA generativa

Tipo de IA capaz de generar nuevo contenido a partir de patrones aprendidos.

— IaaS

Modelo cloud que ofrece cómputo, red y almacenamiento virtualizados.

— ICMP

Concepto de comunicación dentro de comunicación entre dispositivos que define cómo se intercambia o interpreta información.

— IDS e IPS

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— Imágenes y contenedores

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Infraestructura como código

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— Ingeniería social

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— INNER JOIN

Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.

— INSERT

Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.

— Interoperabilidad

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— IP

Dirección lógica usada para identificar y enrutar dispositivos en una red.

— ITIL 4

Marco de buenas prácticas para gestión de servicios de TI orientado a valor.

— JavaScript

Lenguaje de programación usado ampliamente para crear comportamiento dinámico en aplicaciones web.

— JOIN

Operación SQL para combinar filas relacionadas entre tablas.

— JSON

Formato de texto ligero para representar e intercambiar datos.

— JWT

Formato compacto de token firmado usado para transportar claims entre partes.

— Kanban

Método visual para gestionar flujo de trabajo y limitar trabajo en progreso.

— Kernel

Núcleo del sistema operativo que gestiona recursos y acceso al hardware.

— KPI

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— KPIs

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Kubernetes

Orquestador de contenedores que automatiza despliegue, escalado y operación.

— LAN, MAN y WAN

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— Las 5V de Big Data

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Latencia

Tiempo que tarda una operación o comunicación en producir una respuesta.

— Lean

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— LEFT JOIN

Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.

— Licencias de software

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— LinkedIn para TI

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— Linux

conceptos básicos: Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Listas

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Local Storage

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

— Log

Registro de eventos generado por un sistema o aplicación.

— Logs del sistema

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— MAC address

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— Machine Learning

Área de IA que crea modelos capaces de aprender patrones a partir de datos.

— macOS

conceptos básicos: Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Malware

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— Manejo de excepciones

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Mantenimiento de software

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Memoria caché

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

— Memoria RAM

Memoria volátil usada por programas y sistema operativo mientras están en ejecución.

— Memoria virtual

Componente o tecnología de gestión de recursos del sistema con impacto directo en capacidad, rendimiento o conectividad.

— Merge

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

- **MFA**
Autenticación que exige dos o más factores independientes para verificar identidad.
- **Mocking**
Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.
- **Modelo OSI**
Modelo conceptual de 7 capas para entender cómo se comunican los sistemas en red.
- **Modelo TCP/IP**
Modelo práctico usado en Internet para organizar la comunicación de red.
- **MVP**
Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.
- **Máquinas virtuales**
Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.
- **Máscara de subred**
Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.
- **Método de troubleshooting**
Marco o concepto de diagnóstico y resolución de incidentes que ayuda a organizar decisiones, procesos y buenas prácticas.
- **Método STAR**
Marco o concepto de entrada y evolución profesional en TI que ayuda a organizar decisiones, procesos y buenas prácticas.
- **Métodos HTTP**
Marco o concepto de funcionamiento de aplicaciones web que ayuda a organizar decisiones, procesos y buenas prácticas.
- **NAT**
Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.
- **Normalización**
Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.
- **NoSQL**
Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.
- **Nube pública, privada e híbrida**
Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.
- **Núcleos e hilos**
Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.
- **Object Storage**
Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Observabilidad

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— OKR

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— OLTP vs. OLAP

Comparación entre conceptos relacionados de uso de datos para análisis y automatización, destacando cuándo conviene cada uno.

— Operadores

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Ordenamiento

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— ORDER BY

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

— Overfitting y underfitting

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— OWASP Top 10

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— PaaS

Modelo cloud que ofrece una plataforma gestionada para desplegar aplicaciones.

— Paginación

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Pantalla azul

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Parámetros y retorno

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Patrones de diseño

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— PC lento

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Periféricos de entrada

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

— Periféricos de salida

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

— **Permisos de archivos**

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— **Permisos insuficientes**

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— **Phishing**

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— **Pilas**

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— **Pipeline**

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— **PKI y certificados digitales**

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— **Placa madre**

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

— **Plan de estudio de 90 días**

Marco o concepto de entrada y evolución profesional en TI que ayuda a organizar decisiones, procesos y buenas prácticas.

— **Planificación de CPU**

Marco o concepto de gestión de recursos del sistema que ayuda a organizar decisiones, procesos y buenas prácticas.

— **Pods**

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— **Polimorfismo**

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— **POO**

Paradigma que organiza software alrededor de objetos con estado y comportamiento.

— **POP3 e IMAP**

Concepto de comunicación dentro de comunicación entre dispositivos que define cómo se intercambia o interpreta información.

— **Portafolio de TI**

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— **PowerShell**

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— **Principio de mínimo privilegio**

Marco o concepto de protección de sistemas y datos que ayuda a organizar decisiones, procesos y buenas prácticas.

— Problemas de conectividad

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Problemas de DHCP

Concepto de comunicación dentro de diagnóstico y resolución de incidentes que define cómo se intercambia o interpreta información.

— Problemas de DNS

Concepto de comunicación dentro de diagnóstico y resolución de incidentes que define cómo se intercambia o interpreta información.

— Problemas de impresora

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Procedimientos almacenados

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

— Procesamiento en tiempo real

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— Procesamiento por lotes

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— Procesos

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Prompt engineering

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Pruebas de aceptación

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Pruebas de integración

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Pruebas de sistema

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Pruebas unitarias

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Pseudocódigo

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Puertos de red

Componente o tecnología de comunicación entre dispositivos con impacto directo en capacidad, rendimiento o conectividad.

— Puertos USB

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

— Pull Request

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— QA vs. Testing

Comparación entre conceptos relacionados de calidad y diseño de software, destacando cuándo conviene cada uno.

— RAID

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

— RAM

Memoria volátil usada por procesos en ejecución.

— Ransomware

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— Rebase

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— Recuperación de archivos

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

— Recursión

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Redes neuronales

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Refactoring

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Regiones y zonas de disponibilidad

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Regresión

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

— Rendimiento y latencia

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— Repositorio

Ubicación donde se almacena y versiona código u otros artefactos.

— Repositorio local y remoto

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— Requisitos funcionales y no funcionales

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Resolución de conflictos

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— Respuesta a incidentes

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— REST

Estilo arquitectónico para APIs basado en recursos, HTTP y operaciones sin estado.

— Riesgos de proyecto

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— Roles de Scrum

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— Rollback

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— ROM y firmware

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

— Router

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— Ruta de entrada a TI

Marco o concepto de entrada y evolución profesional en TI que ayuda a organizar decisiones, procesos y buenas prácticas.

— SaaS

Modelo en el que una aplicación completa se consume como servicio.

— SaaS, PaaS e IaaS

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— Scope

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— Scrum

Marco ágil de trabajo iterativo basado en sprints, roles, eventos y artefactos.

— Seguridad en Wi-Fi

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— SELECT

Sentencia SQL usada para consultar datos de una o más tablas.

— Serverless

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Servicios en Kubernetes

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

— Servicios y demonios

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Sesiones

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

— Shell

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— SIEM

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

— Sistema binario

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— Sistema de archivos

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— Sistema de información

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— SLA

Acuerdo que define niveles esperados de servicio y métricas asociadas.

— SMTP

Concepto de comunicación dentro de comunicación entre dispositivos que define cómo se intercambia o interpreta información.

— Software libre y código abierto

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— SOLID

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Soporte técnico

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

— SQL

Lenguaje usado para consultar y manipular bases de datos relacionales.

— SQL Injection

Ataque que intenta alterar consultas SQL mediante entradas no tratadas correctamente.

— SSD NVMe

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

— SSD SATA

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

— SSH

Concepto de comunicación dentro de comunicación entre dispositivos que define cómo se intercambia o interpreta información.

— Stakeholders

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— Subconsultas

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

— Subnetting

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— Switch

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— Tabla, fila y columna

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

— TCP

Protocolo de transporte orientado a conexión, confiable y con control de entrega.

— TDD

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

— Tecnología de la Información

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

— Testing automatizado

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

— Tipos de datos

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

— TLS

Protocolo criptográfico usado para proteger comunicaciones en red.

— Topologías de red

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

— Transacciones

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

- **UDP**
Protocolo de transporte sin conexión, ligero y sin garantía de entrega.
- **UML**
Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.
- **UPDATE**
Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.
- **URL**
Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.
- **Usuarios y grupos**
Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.
- **Variables**
Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.
- **Variables de entorno**
Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.
- **Versionado semántico**
Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.
- **Virtualización por hardware**
Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.
- **Vistas**
Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.
- **VLAN**
Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.
- **VPN**
Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.
- **Webhooks**
Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.
- **WHERE**
Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.
- **Wi-Fi 2.4 vs. 5 GHz**
Comparación entre conceptos relacionados de comunicación entre dispositivos, destacando cuándo conviene cada uno.
- **Windows**
conceptos básicos: Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

— WIP

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

— XML

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

— XSS

Vulnerabilidad que permite ejecutar scripts no confiables en el navegador de otra persona.

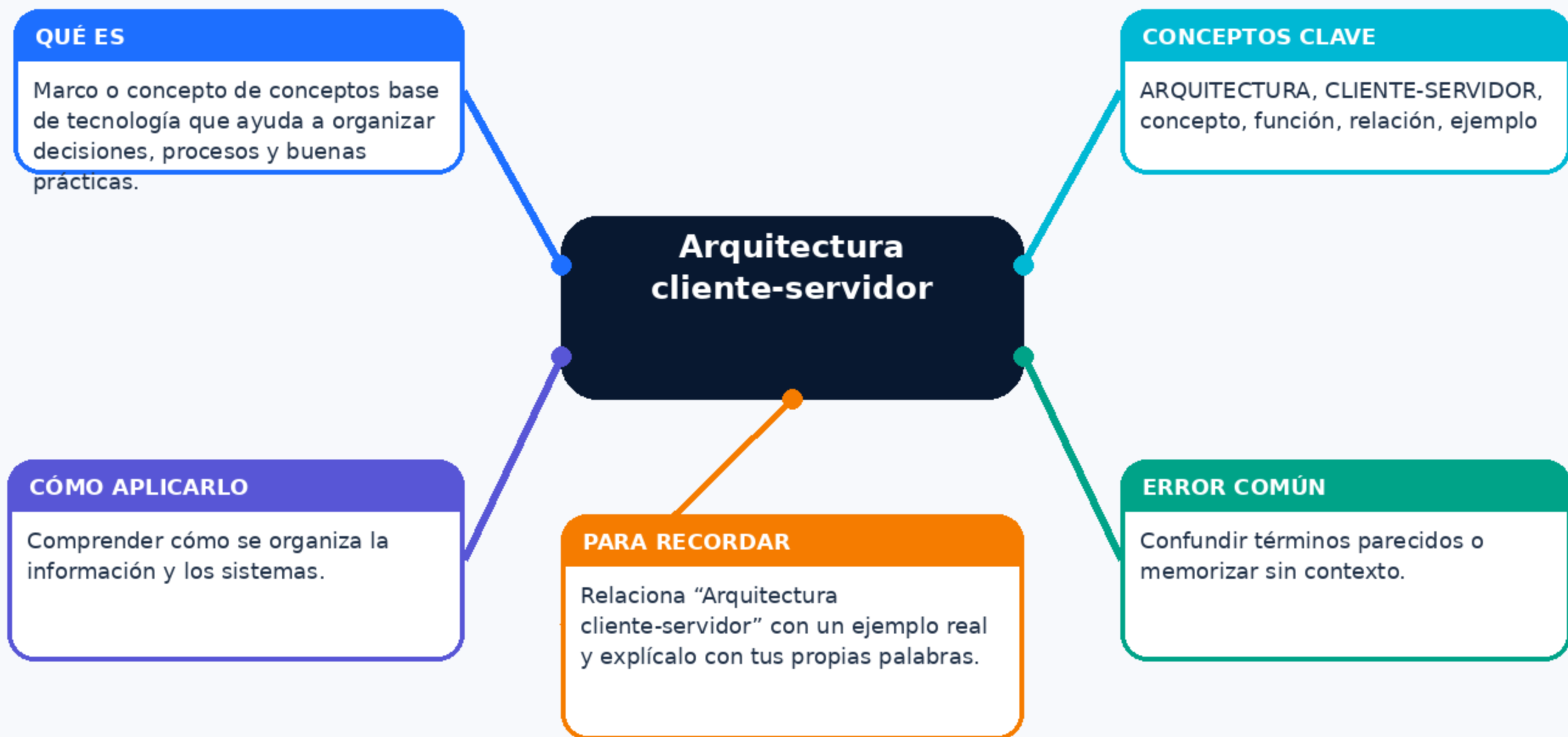
— Zero Trust

Enfoque de seguridad que evita confiar implícitamente y verifica cada acceso.

— Índices

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

Arquitectura cliente-servidor



Bits, bytes y unidades

QUÉ ES

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

BITS, BYTES, UNIDADES, concepto, función, relación

Bits, bytes y unidades

CÓMO APLICARLO

Comprender cómo se organiza la información y los sistemas.

PARA RECORDAR

Relaciona "Bits, bytes y unidades" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir términos parecidos o memorizar sin contexto.

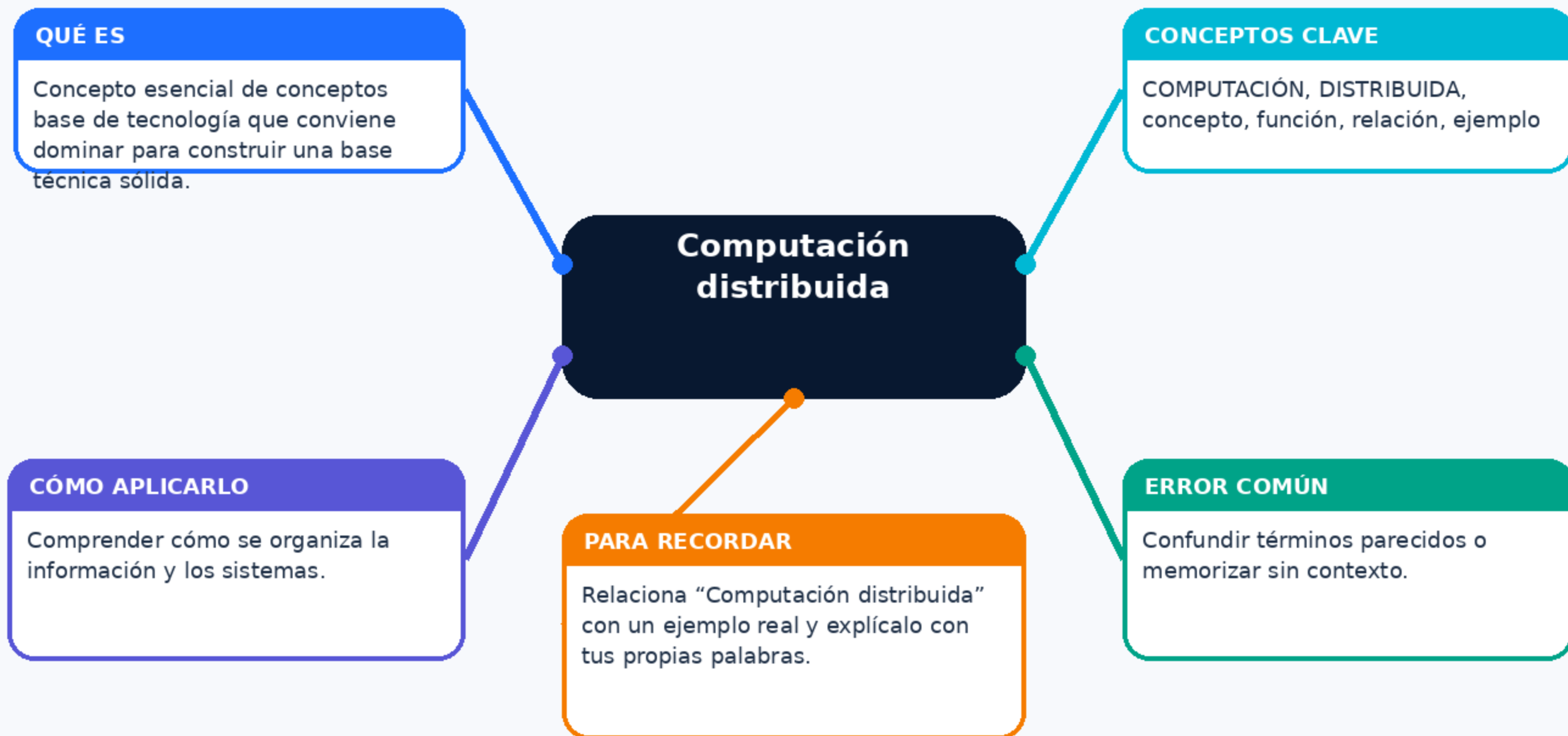
Ciclo de vida de la información



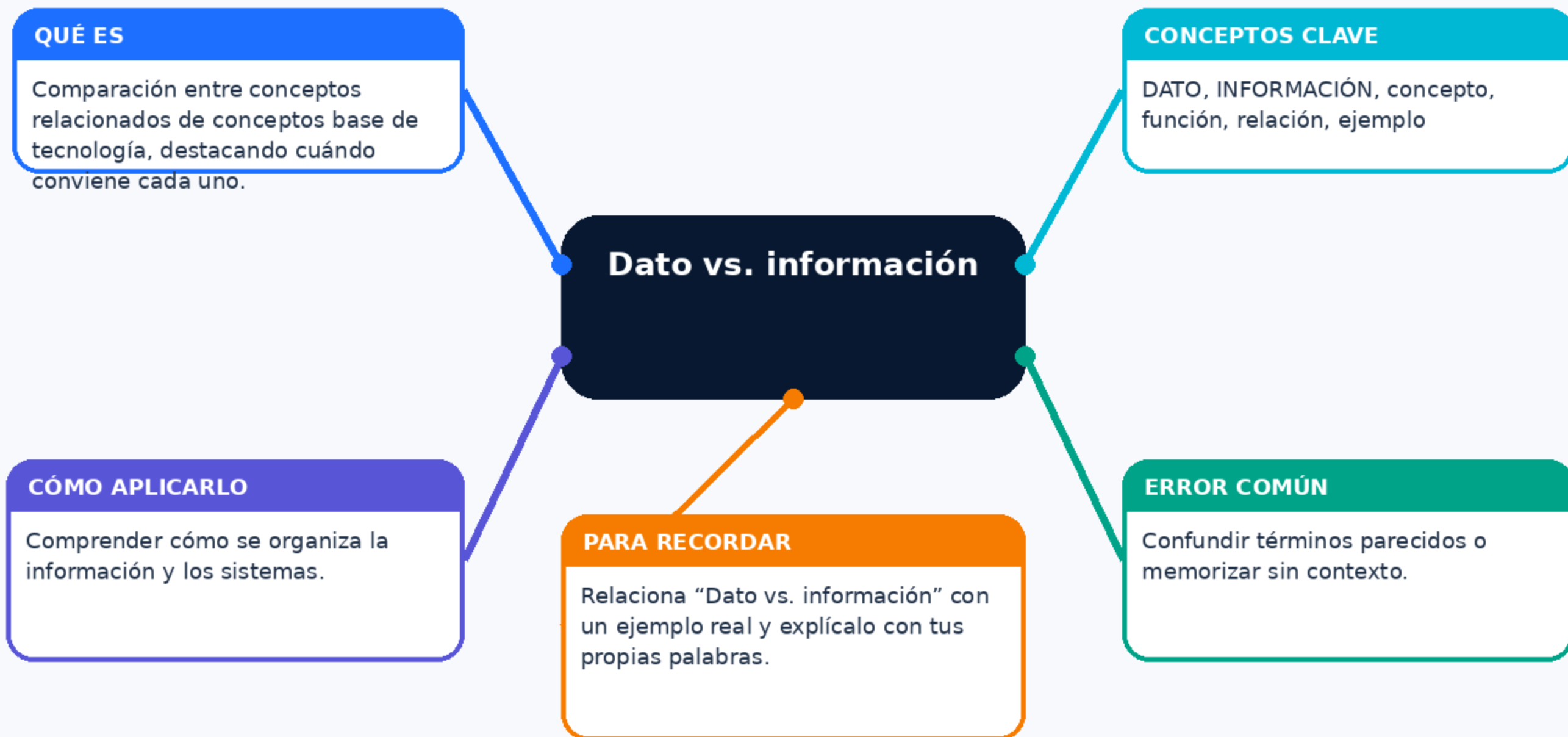
Codificación ASCII y Unicode



Computación distribuida



Dato vs. información



Disponibilidad y confiabilidad



Escalabilidad

QUÉ ES

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ESCALABILIDAD, concepto, función, relación, ejemplo, contexto

Escalabilidad

CÓMO APLICARLO

Comprender cómo se organiza la información y los sistemas.

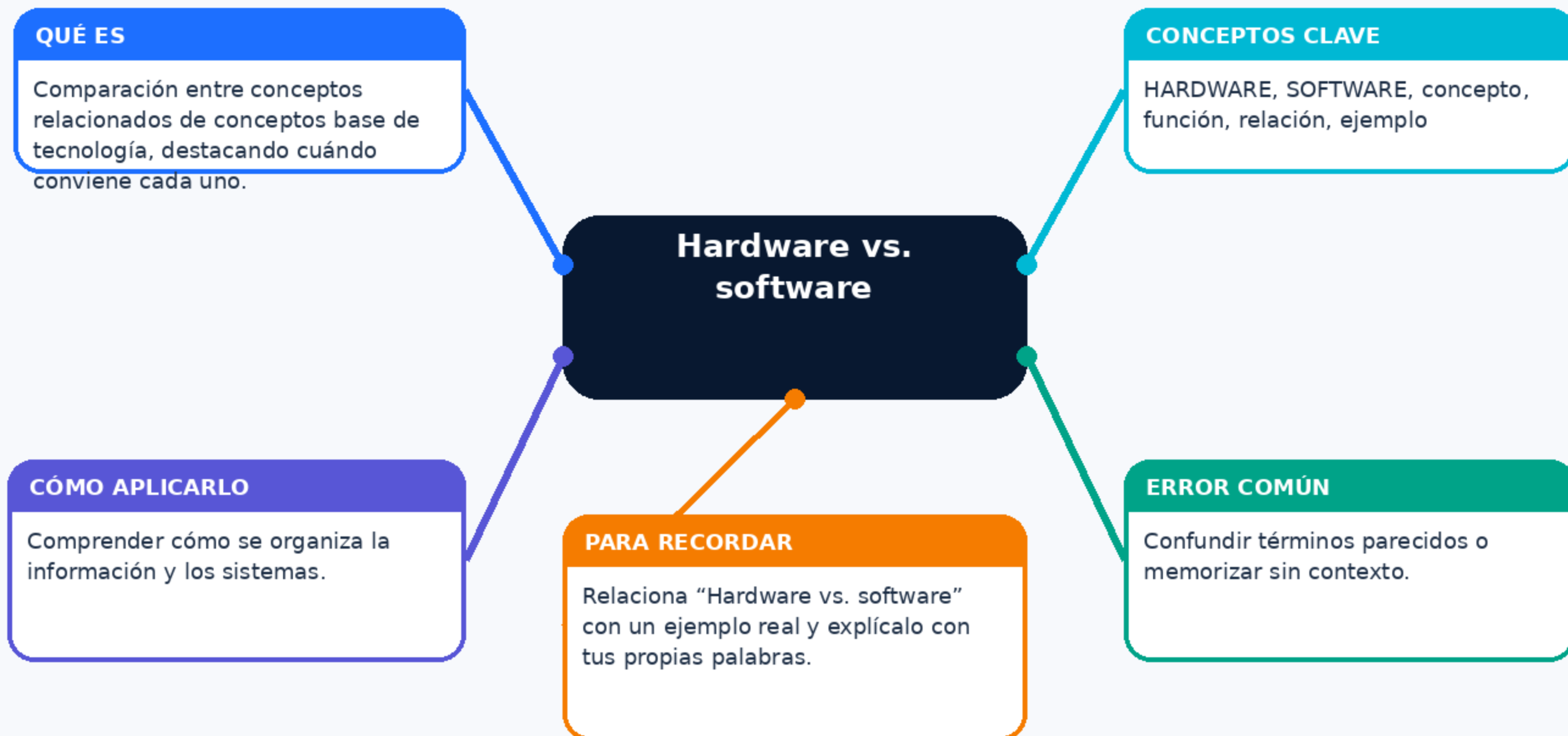
PARA RECORDAR

Relaciona "Escalabilidad" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir términos parecidos o memorizar sin contexto.

Hardware vs. software



Hexadecimal

QUÉ ES

Concepto esencial de conceptos base de tecnología que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

HEXADECIMAL, concepto, función, relación, ejemplo, contexto

Hexadecimal

CÓMO APLICARLO

Comprender cómo se organiza la información y los sistemas.

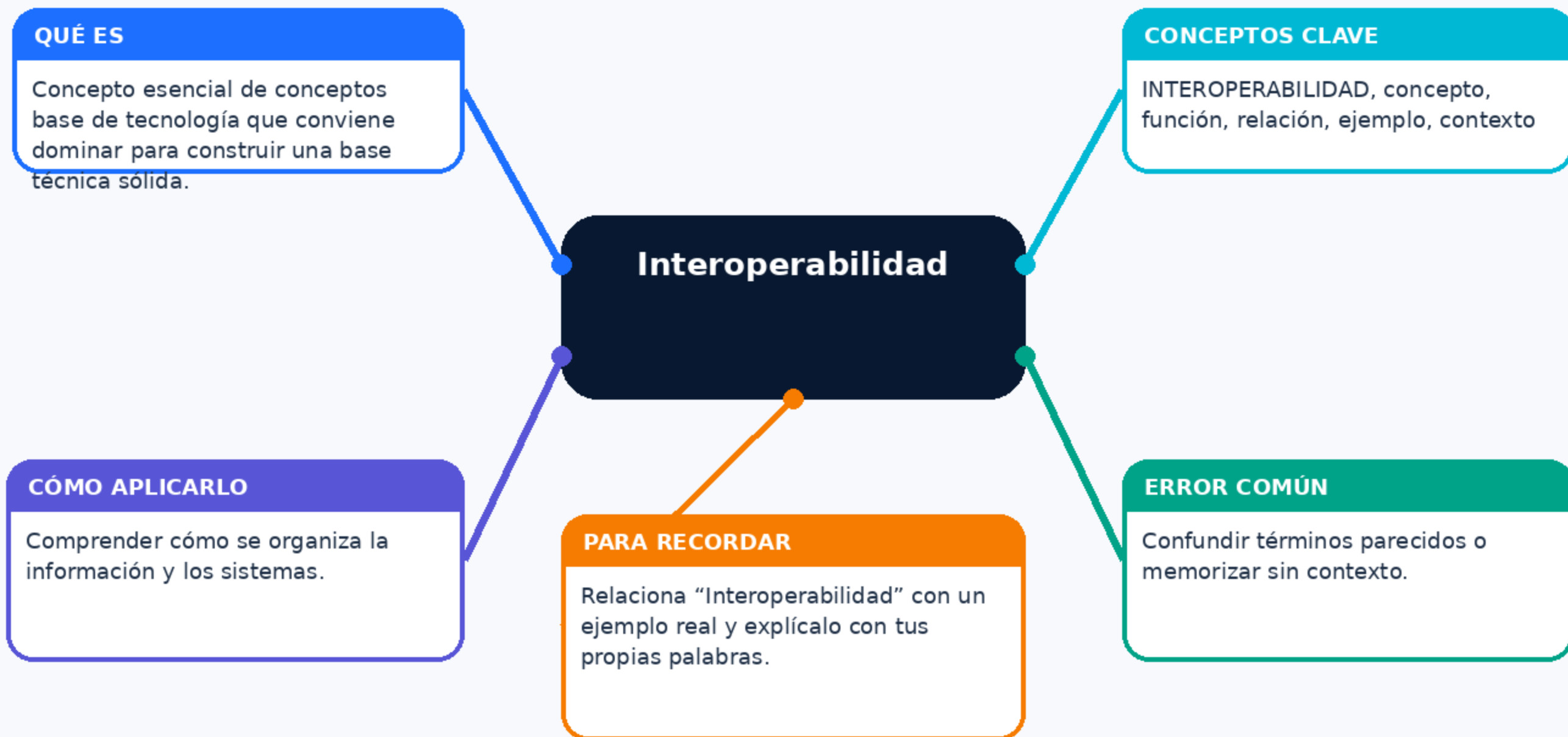
PARA RECORDAR

Relaciona "Hexadecimal" con un ejemplo real y explícalo con tus propias palabras.

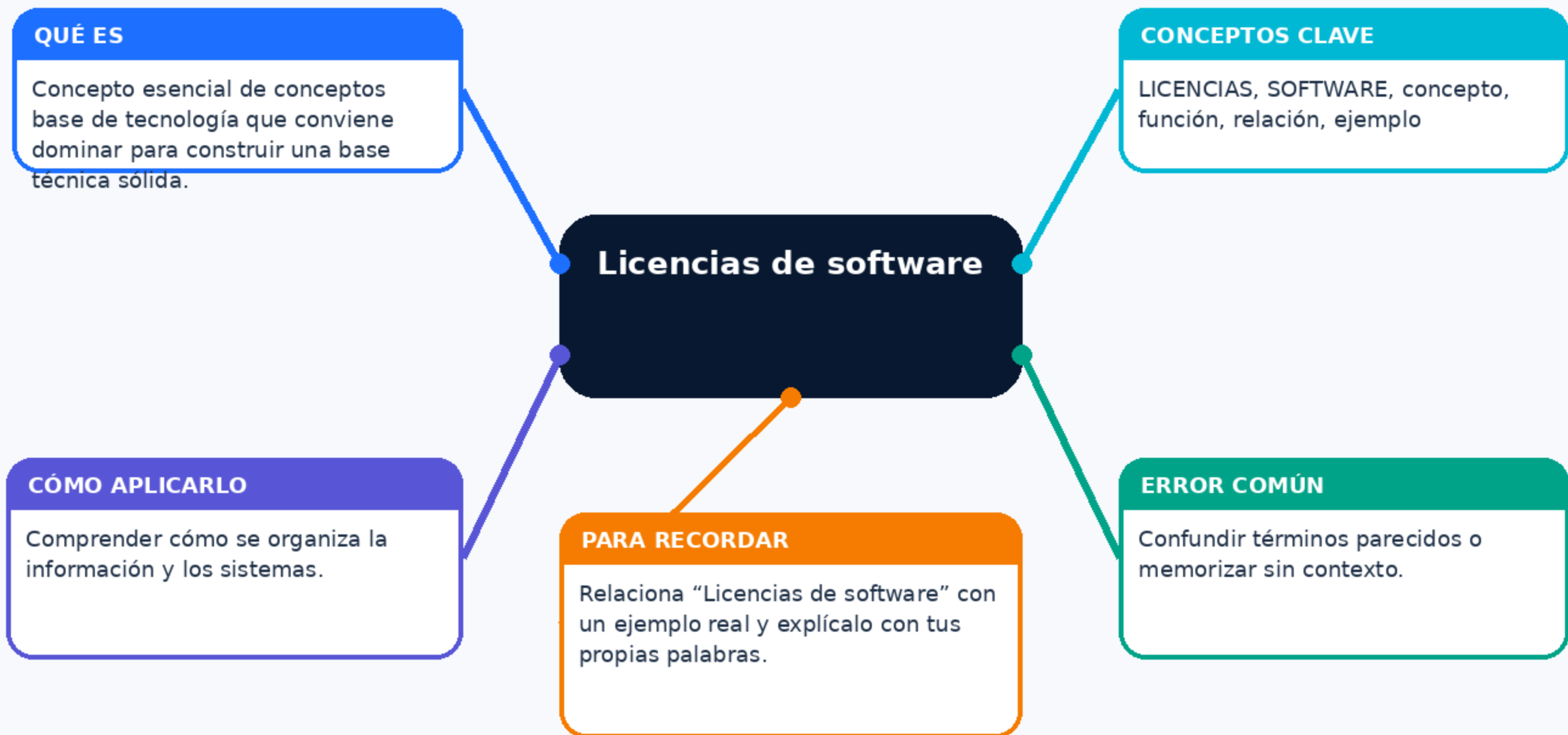
ERROR COMÚN

Confundir términos parecidos o memorizar sin contexto.

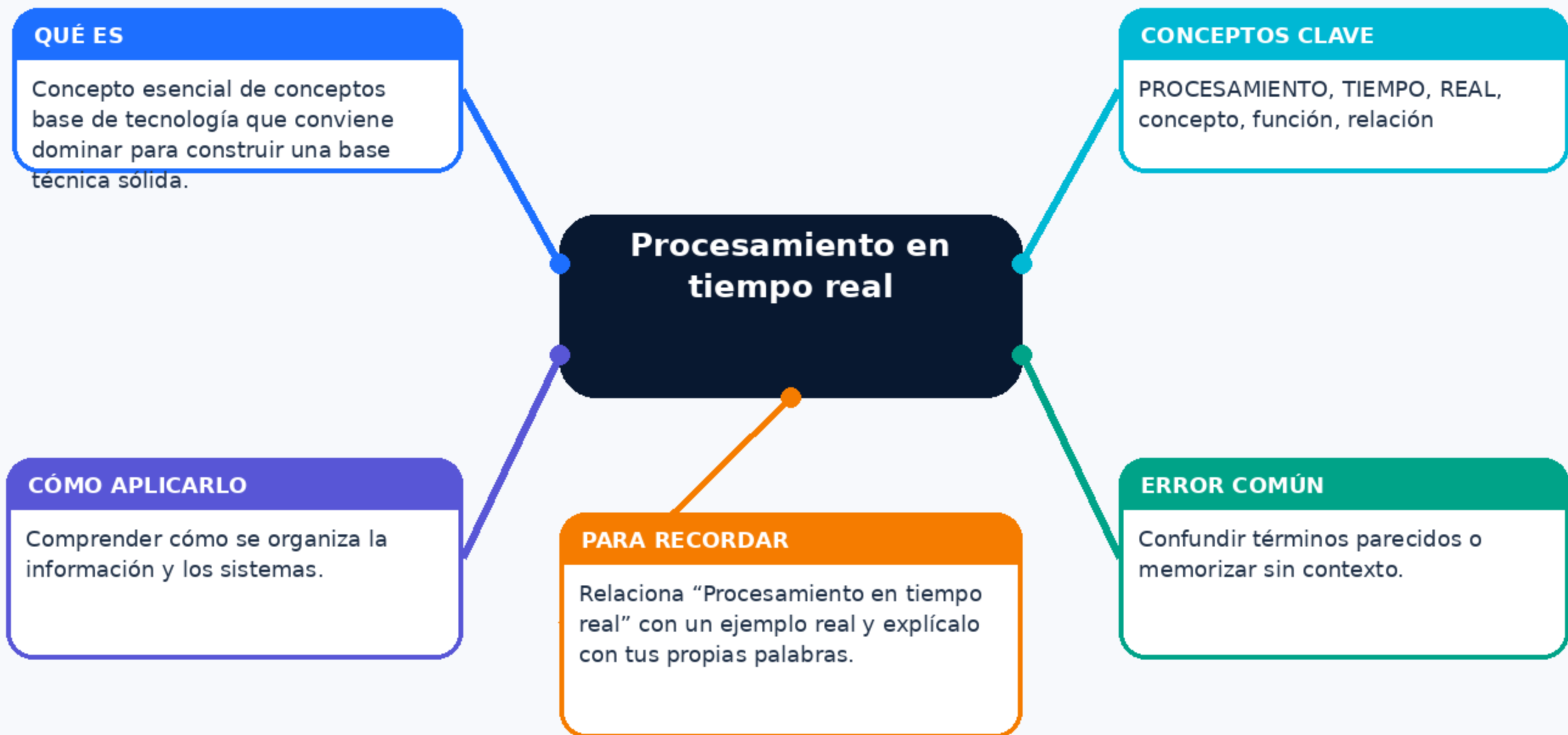
Interoperabilidad



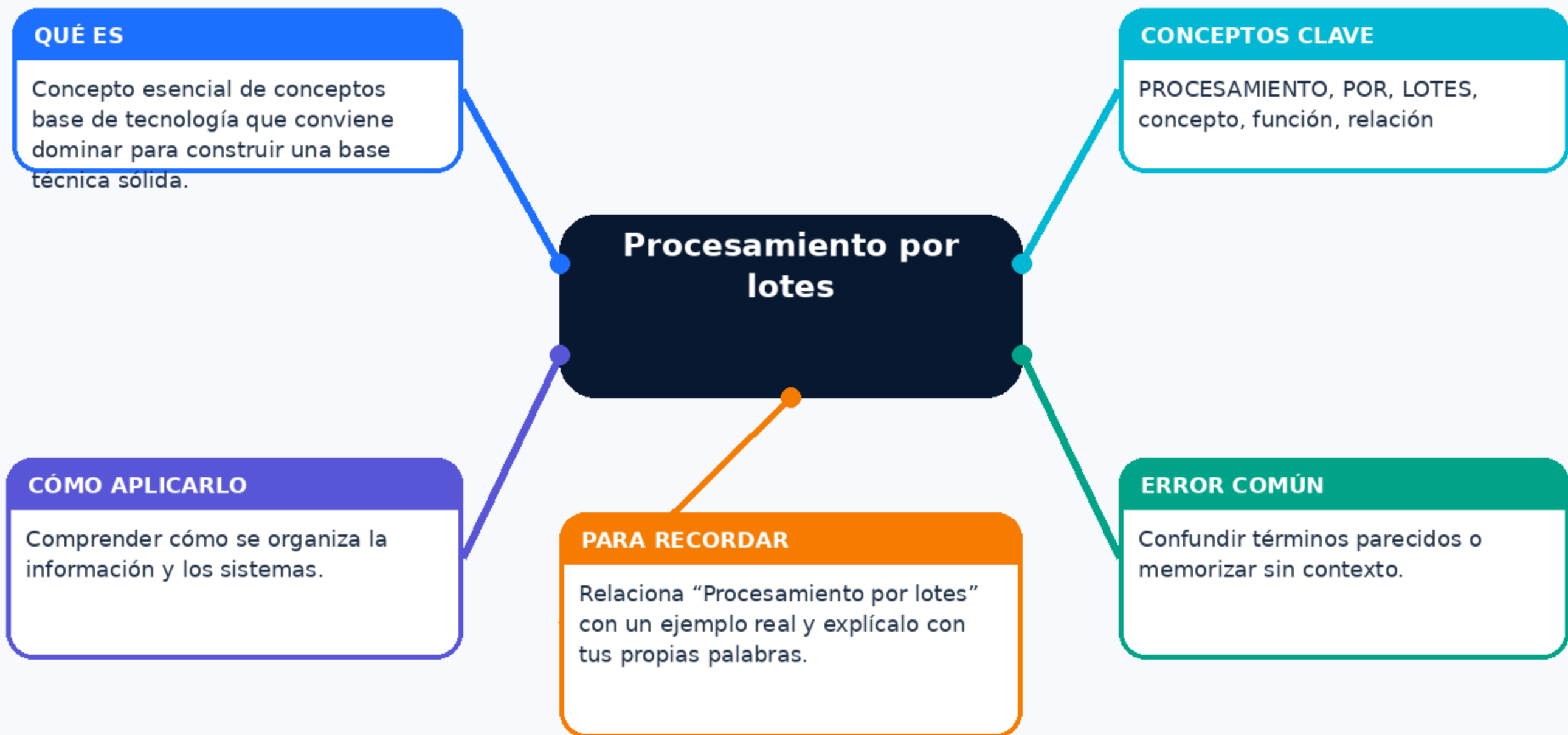
Licencias de software



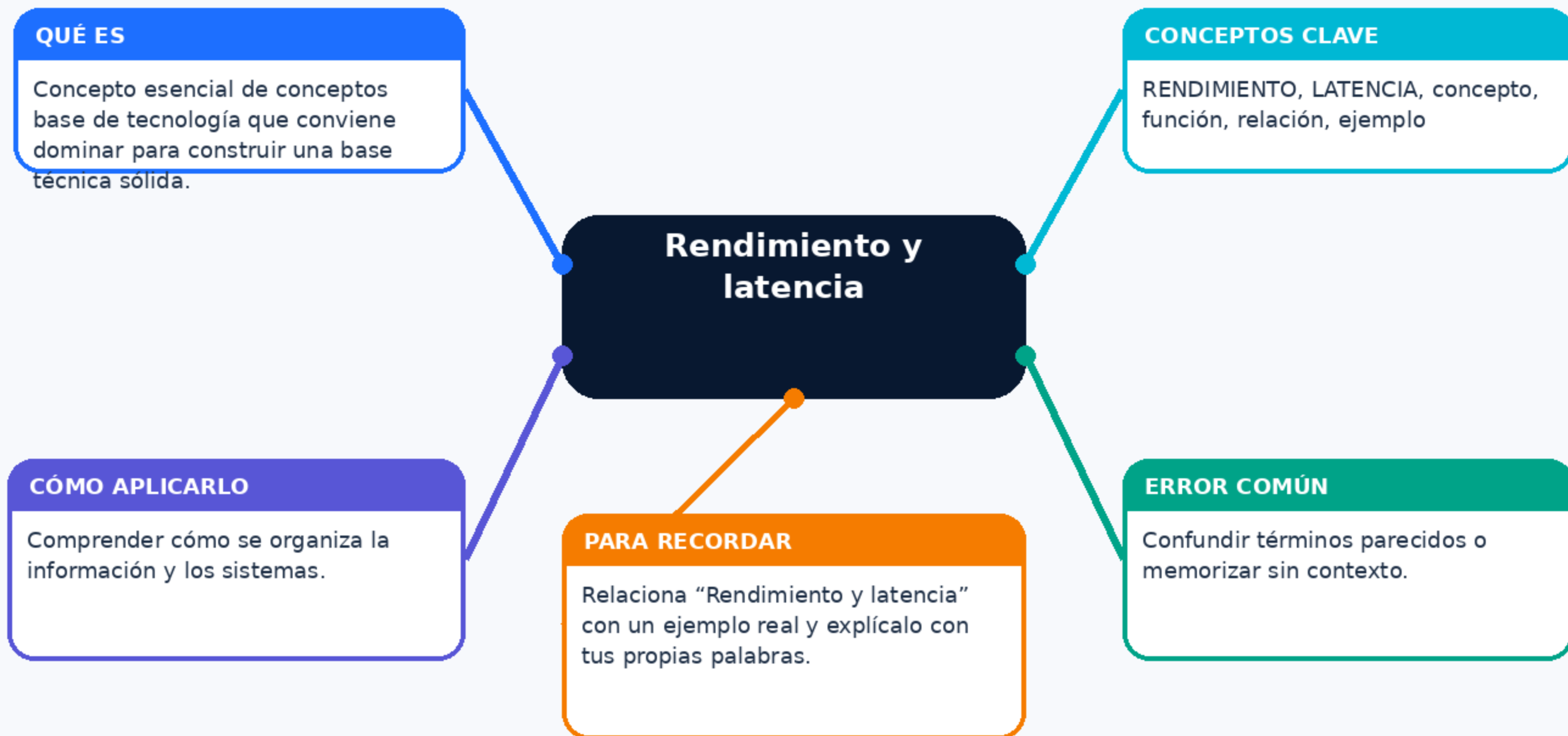
Procesamiento en tiempo real



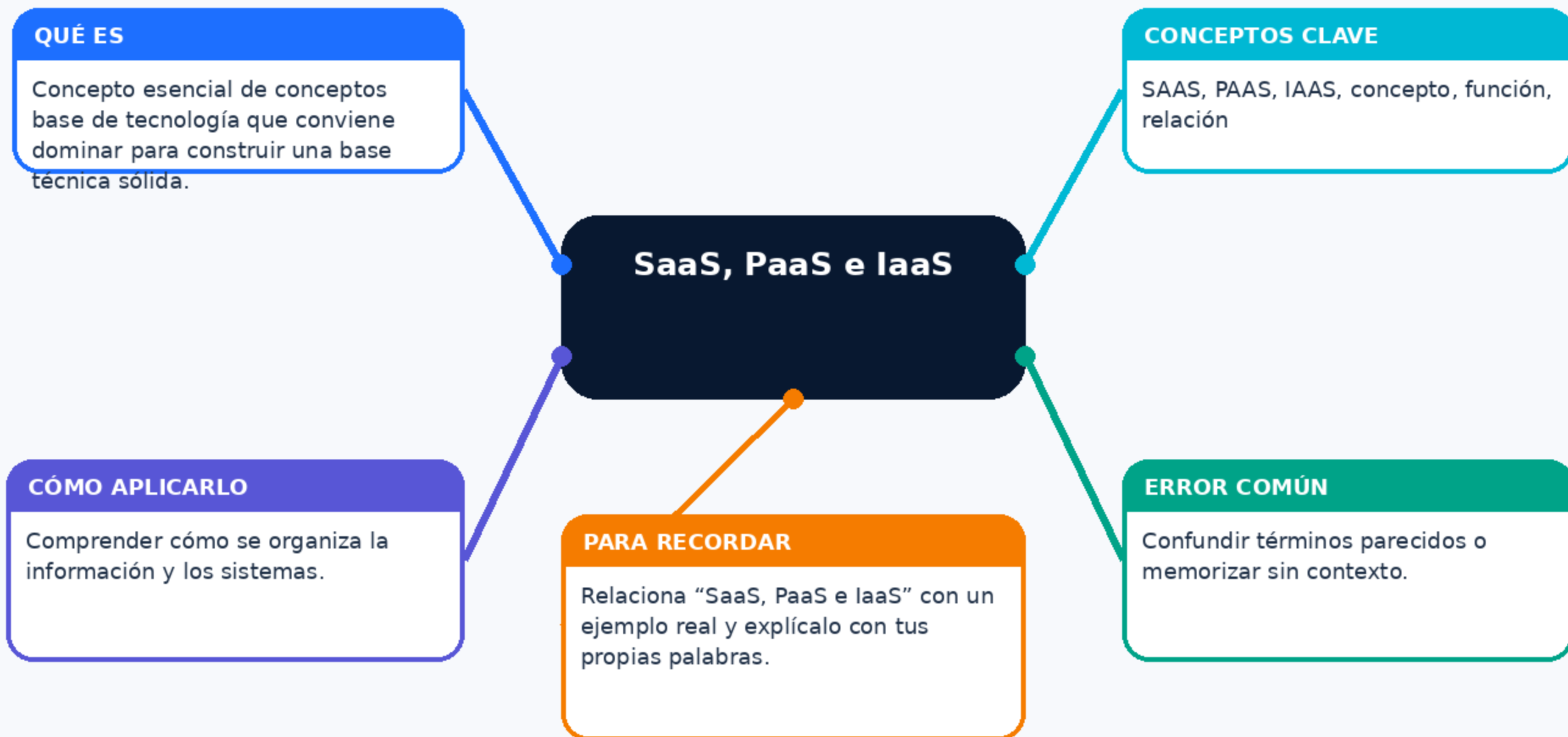
Procesamiento por lotes



Rendimiento y latencia



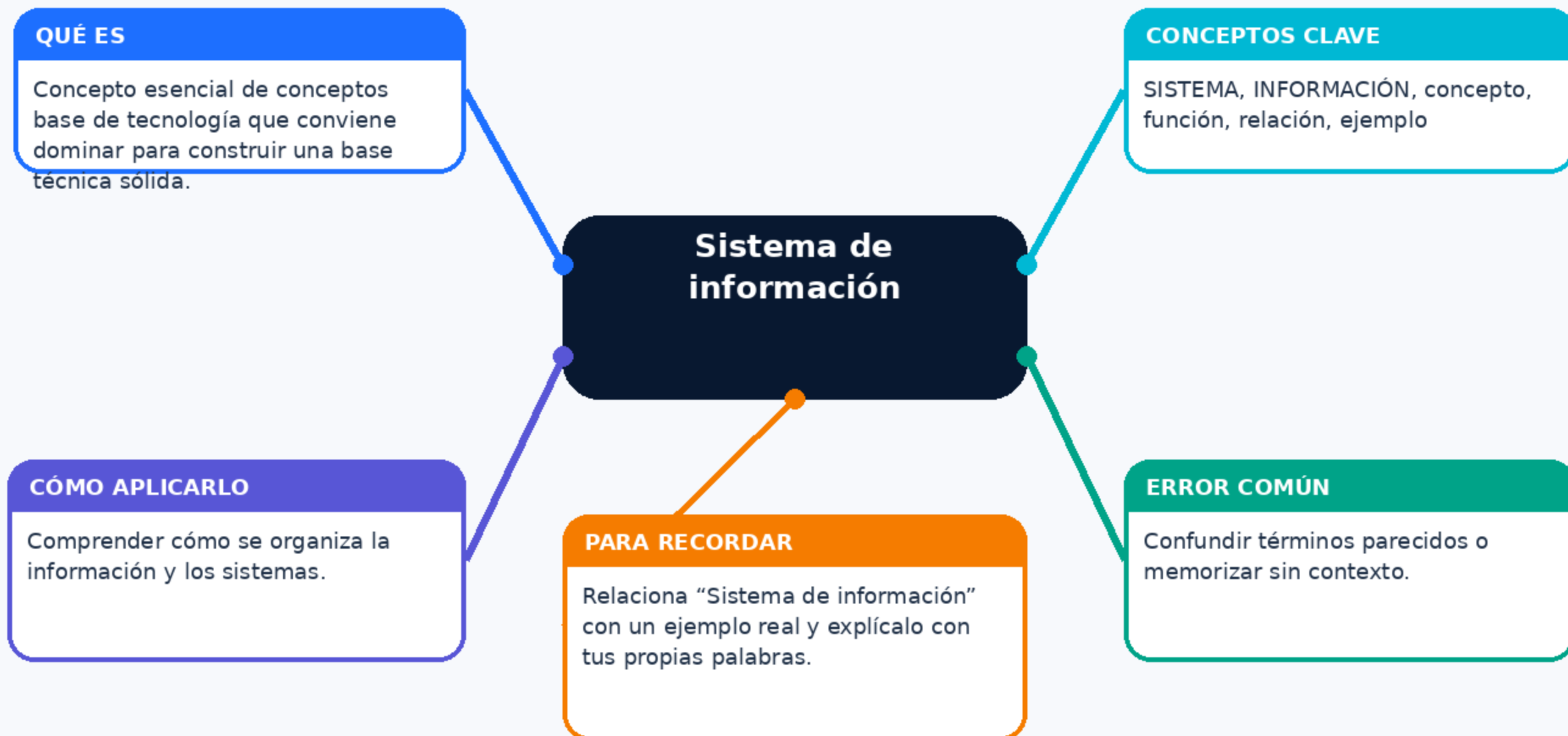
SaaS, PaaS e IaaS



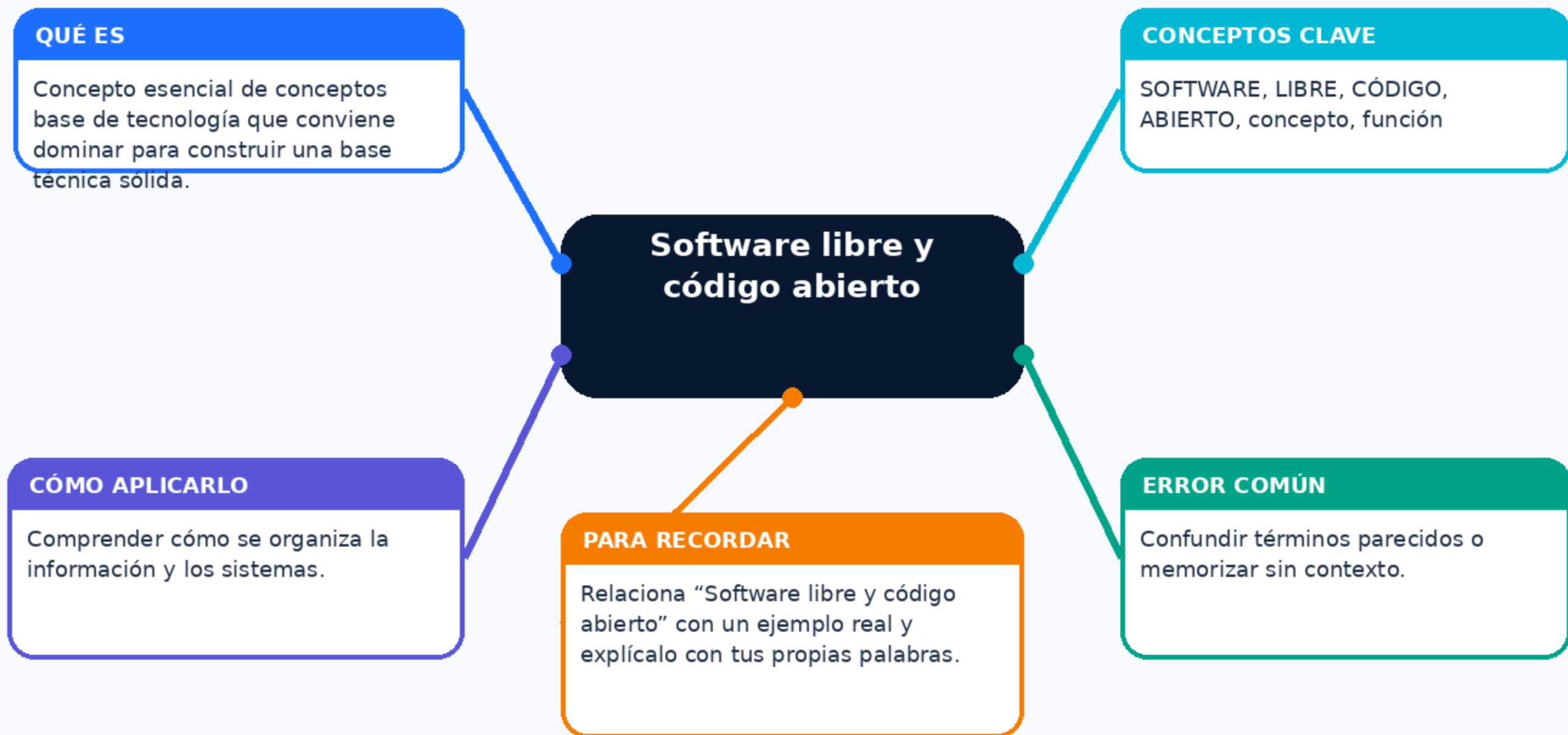
Sistema binario



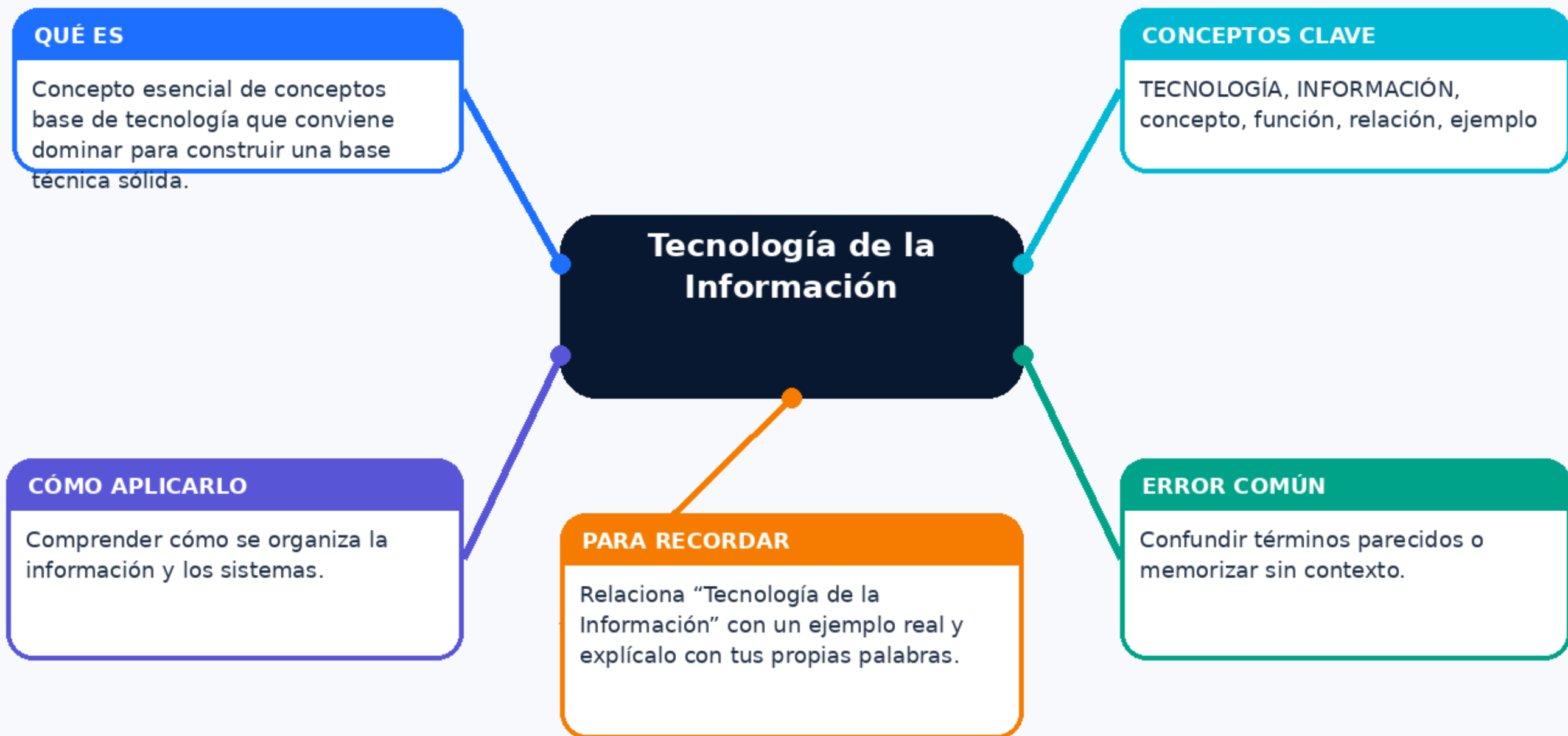
Sistema de información



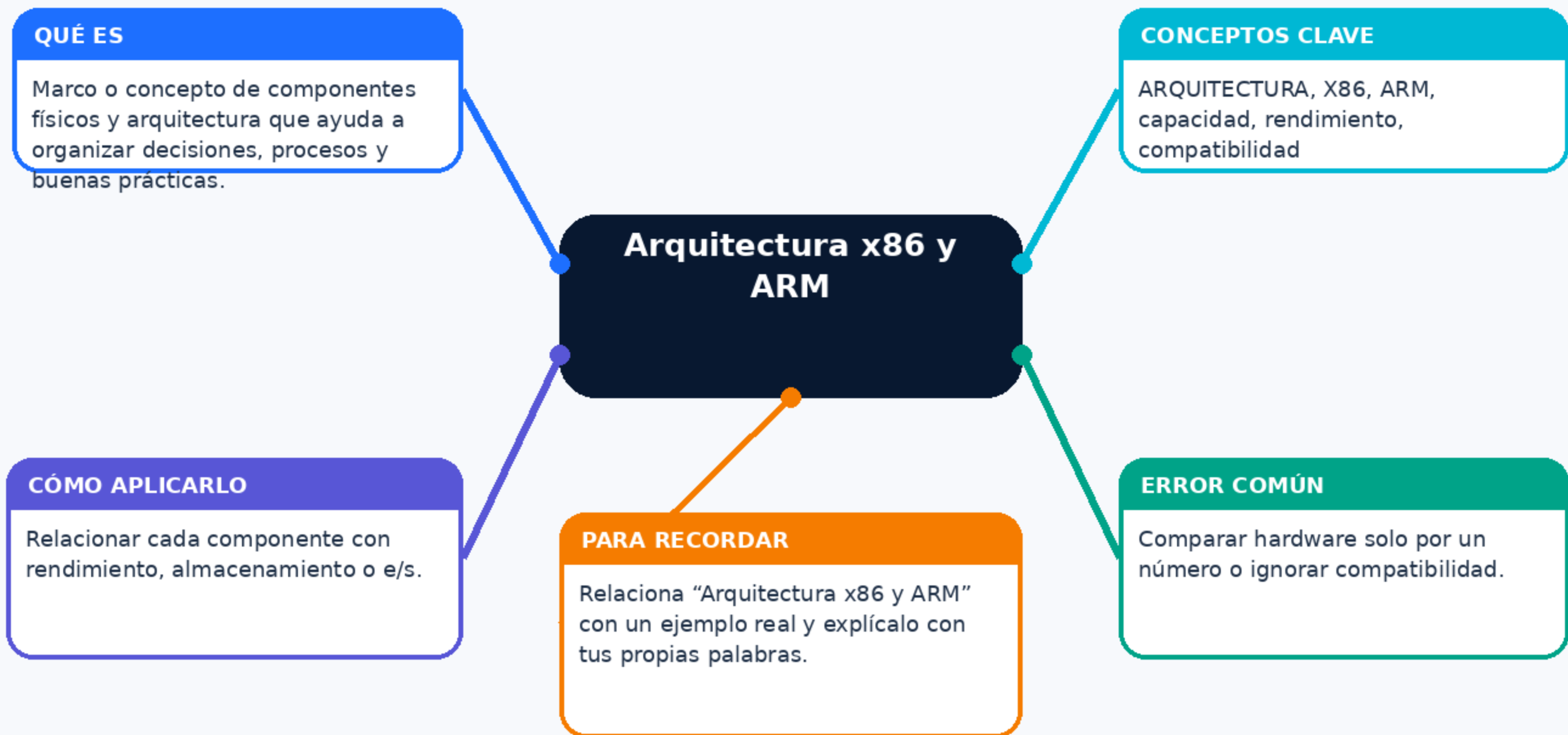
Software libre y código abierto



Tecnología de la Información



Arquitectura x86 y ARM



BIOS y UEFI

QUÉ ES

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

BIOS, UEFI, capacidad, rendimiento, compatibilidad, conexión

BIOS y UEFI

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "BIOS y UEFI" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Bus de datos

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

BUS, DATOS, capacidad, rendimiento, compatibilidad, conexión

Bus de datos

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "Bus de datos" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

CPU

QUÉ ES

Unidad central que ejecuta instrucciones y coordina operaciones del sistema.

CONCEPTOS CLAVE

ALU, unidad de control, registros, caché, núcleos

CPU

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "CPU" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Chipset

QUÉ ES

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

CHIPSET, capacidad, rendimiento, compatibilidad, conexión, uso

Chipset

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

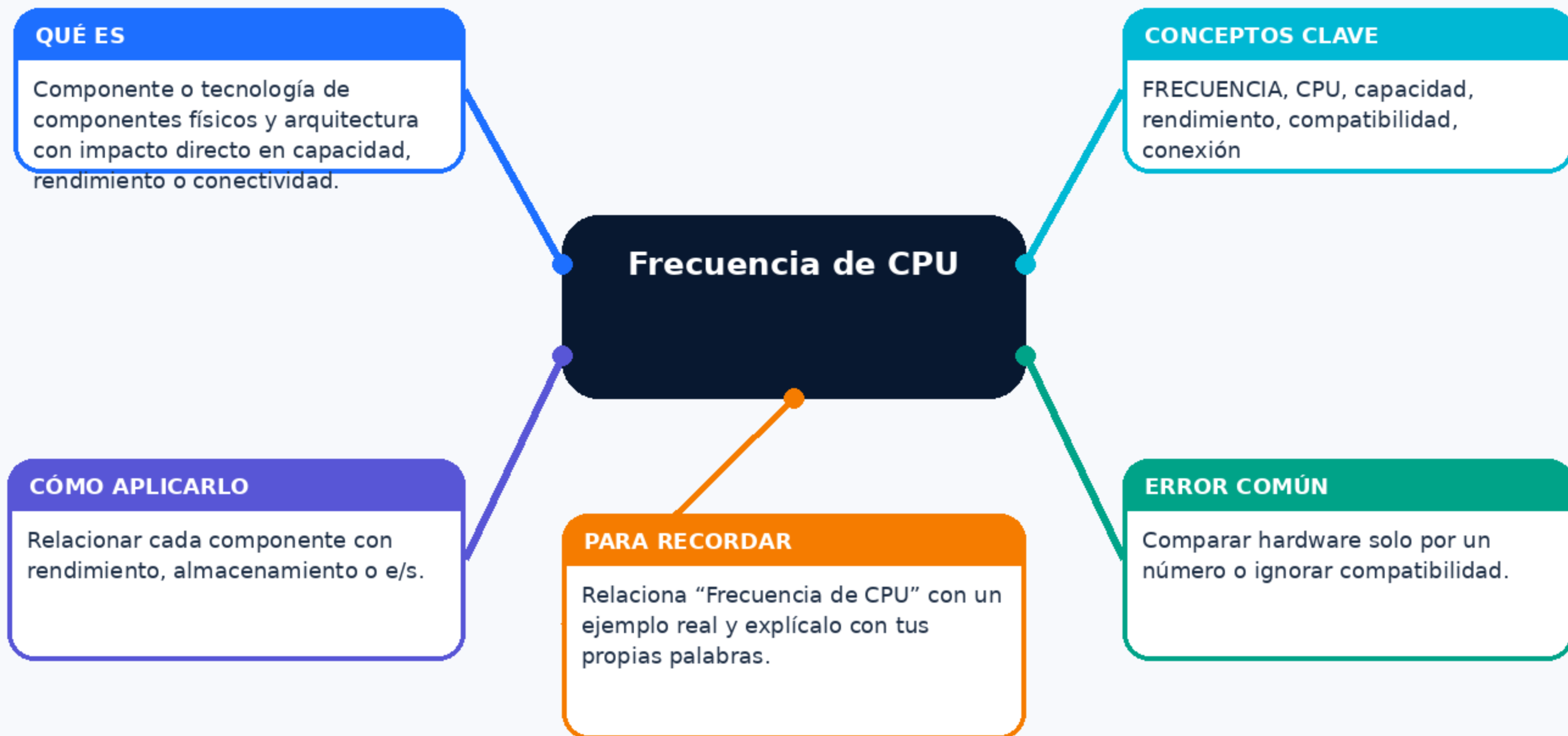
PARA RECORDAR

Relaciona "Chipset" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Frecuencia de CPU



Fuente de alimentación



GPU

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

GPU, capacidad, rendimiento, compatibilidad, conexión, uso

GPU

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "GPU" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

HDD

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

HDD, capacidad, rendimiento, compatibilidad, conexión, uso

HDD

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "HDD" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

HDMI y DisplayPort

QUÉ ES

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

HDMI, DISPLAYPORT, capacidad, rendimiento, compatibilidad, conexión

HDMI y DisplayPort

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "HDMI y DisplayPort" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Memoria RAM

QUÉ ES

Memoria volátil usada por programas y sistema operativo mientras están en ejecución.

CONCEPTOS CLAVE

capacidad, velocidad, DDR, latencia, dual channel

Memoria RAM

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "Memoria RAM" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Memoria caché

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

MEMORIA, CACHÉ, capacidad, rendimiento, compatibilidad, conexión

Memoria caché

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "Memoria caché" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Núcleos e hilos

QUÉ ES

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

NÚCLEOS, HILOS, capacidad, rendimiento, compatibilidad, conexión

Núcleos e hilos

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

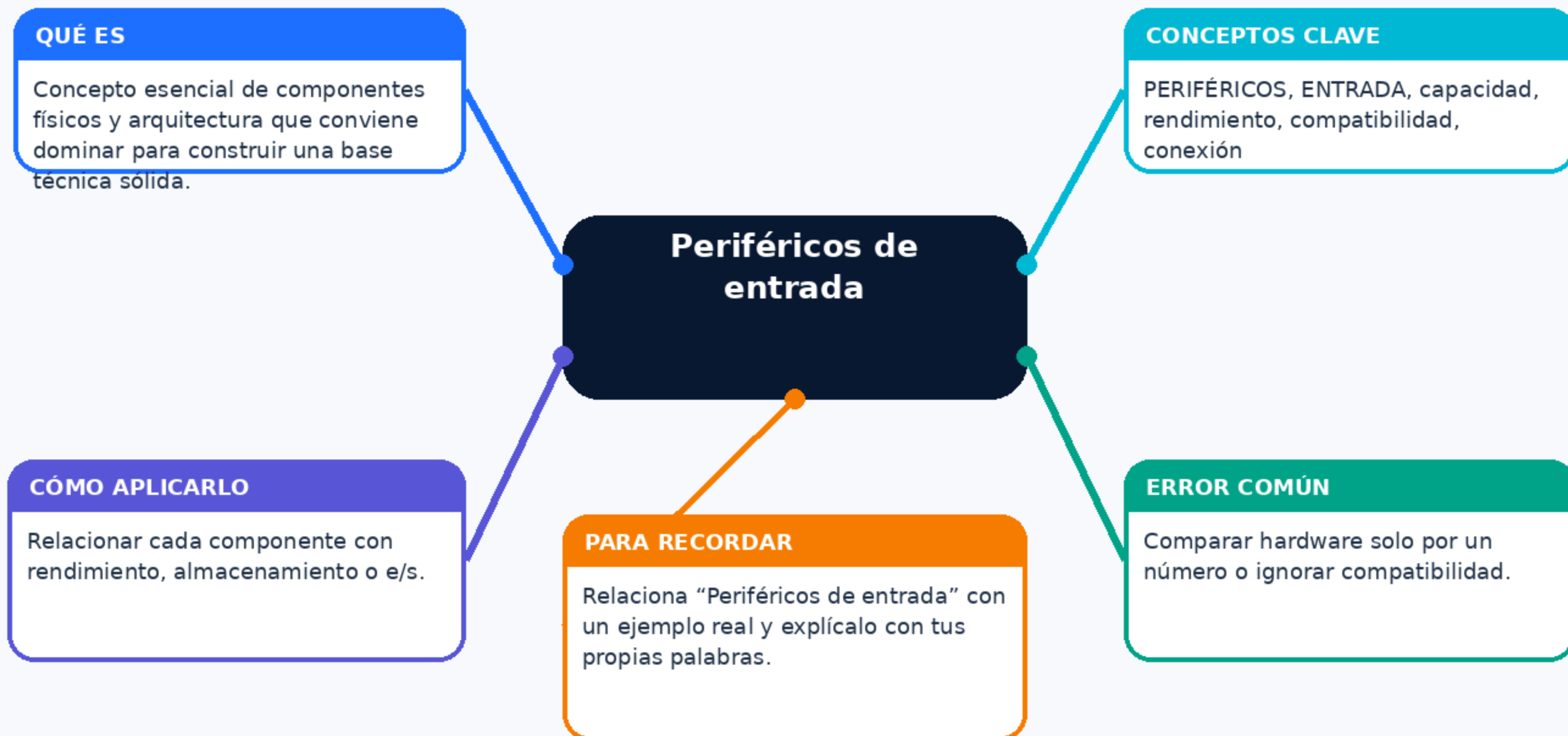
PARA RECORDAR

Relaciona "Núcleos e hilos" con un ejemplo real y explícalo con tus propias palabras.

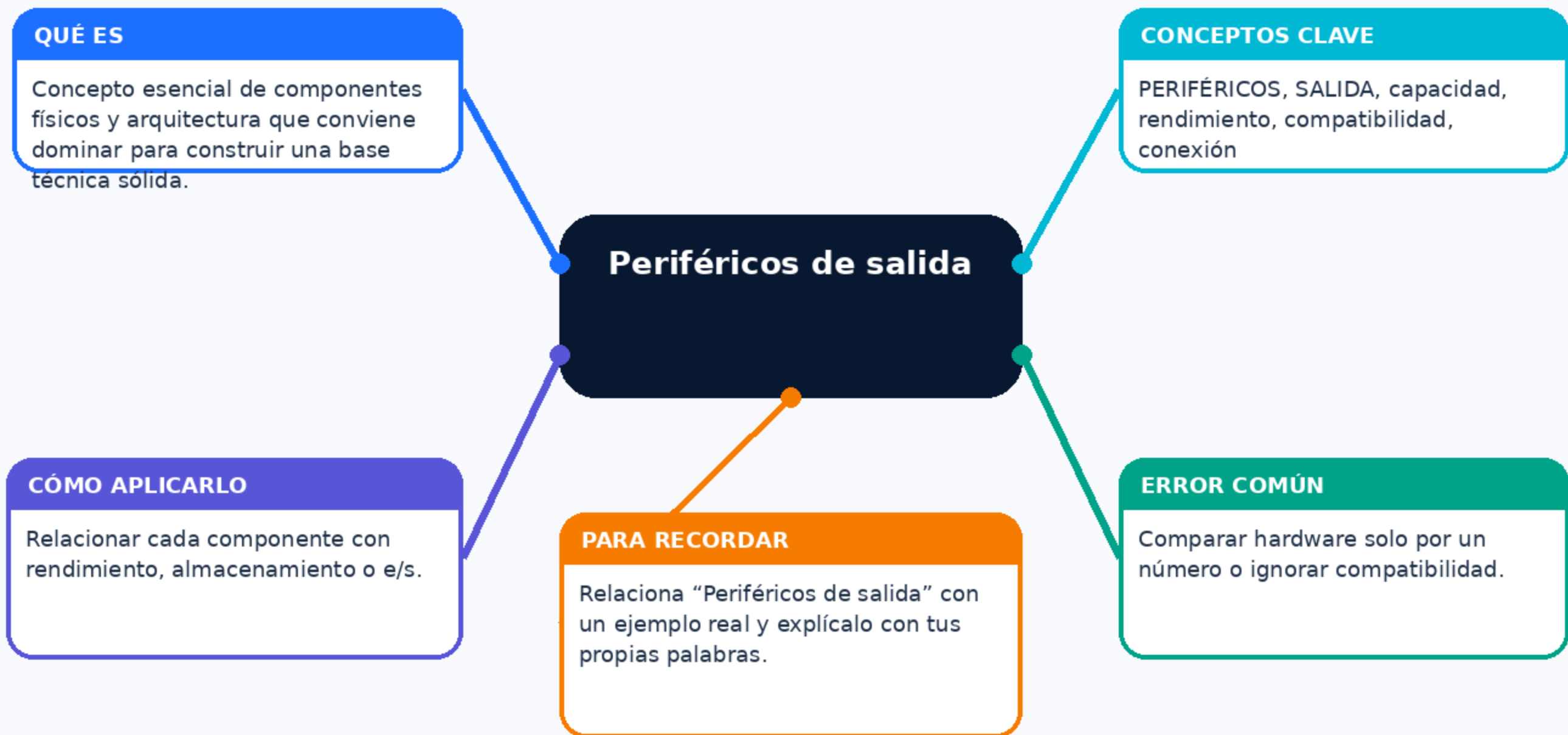
ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Periféricos de entrada



Periféricos de salida



Placa madre

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

PLACA, MADRE, capacidad, rendimiento, compatibilidad, conexión

Placa madre

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "Placa madre" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Puertos USB

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

PUERTOS, USB, capacidad, rendimiento, compatibilidad, conexión

Puertos USB

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "Puertos USB" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

RAID

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

RAID, capacidad, rendimiento, compatibilidad, conexión, uso

RAID

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "RAID" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

ROM y firmware

QUÉ ES

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ROM, FIRMWARE, capacidad, rendimiento, compatibilidad, conexión

ROM y firmware

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "ROM y firmware" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

SSD NVMe

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

SSD, NVME, capacidad, rendimiento, compatibilidad, conexión

SSD NVMe

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "SSD NVMe" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

SSD SATA

QUÉ ES

Componente o tecnología de componentes físicos y arquitectura con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

SSD, SATA, capacidad, rendimiento, compatibilidad, conexión

SSD SATA

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "SSD SATA" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Virtualización por hardware

QUÉ ES

Concepto esencial de componentes físicos y arquitectura que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

VIRTUALIZACIÓN, POR, HARDWARE, capacidad, rendimiento, compatibilidad

Virtualización por hardware

CÓMO APLICARLO

Relacionar cada componente con rendimiento, almacenamiento o e/s.

PARA RECORDAR

Relaciona "Virtualización por hardware" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Comparar hardware solo por un número o ignorar compatibilidad.

Arranque del sistema



Bash

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

BASH, proceso, memoria, archivo, usuario, servicio

Bash

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "Bash" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

CLI vs. GUI

QUÉ ES

Comparación entre conceptos relacionados de gestión de recursos del sistema, destacando cuándo conviene cada uno.

CONCEPTOS CLAVE

CLI, GUI, proceso, memoria, archivo, usuario

CLI vs. GUI

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "CLI vs. GUI" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

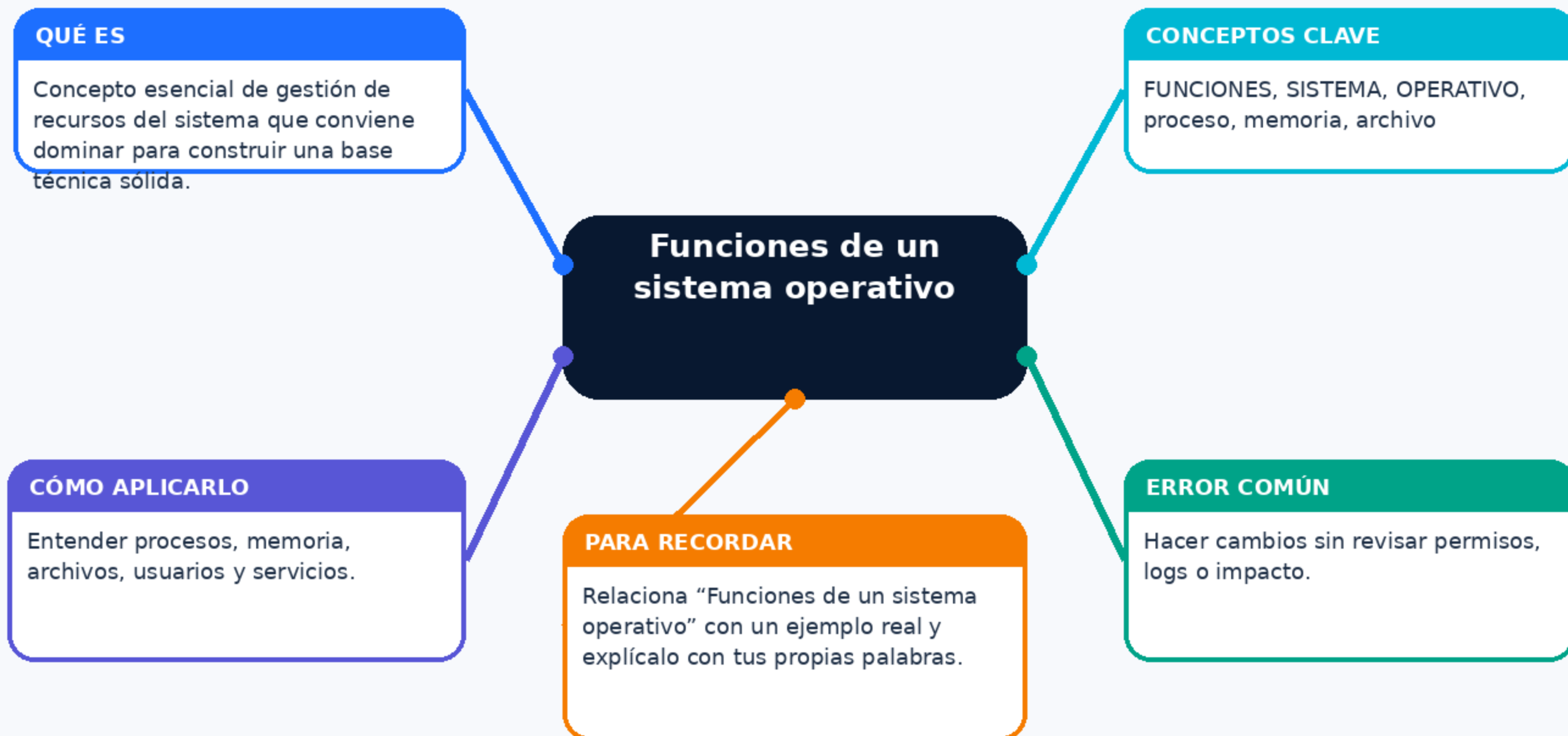
Contenedores vs. máquinas virtuales



Estados de un proceso



Funciones de un sistema operativo



Gestión de paquetes

QUÉ ES

Marco o concepto de gestión de recursos del sistema que ayuda a organizar decisiones, procesos y buenas prácticas.

CONCEPTOS CLAVE

GESTIÓN, PAQUETES, proceso, memoria, archivo, usuario

Gestión de paquetes

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "Gestión de paquetes" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Hilos

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

HILOS, proceso, memoria, archivo, usuario, servicio

Hilos

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "Hilos" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Kernel

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

KERNEL, proceso, memoria, archivo, usuario, servicio

Kernel

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

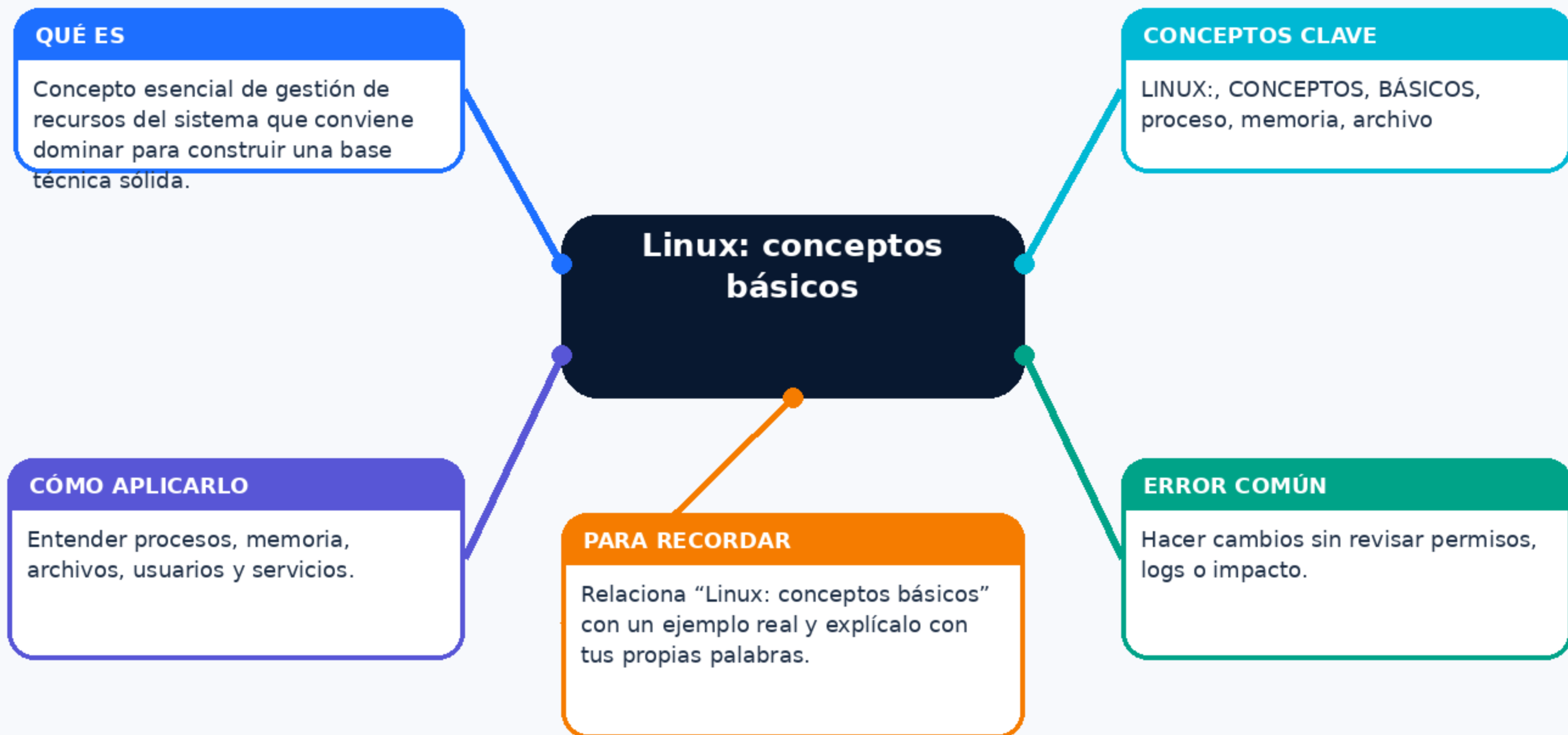
PARA RECORDAR

Relaciona "Kernel" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Linux: conceptos básicos



Logs del sistema



Memoria virtual

QUÉ ES

Componente o tecnología de gestión de recursos del sistema con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

MEMORIA, VIRTUAL, proceso, memoria, archivo, usuario

Memoria virtual

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "Memoria virtual" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Paginación

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

PAGINACIÓN, proceso, memoria, archivo, usuario, servicio

Paginación

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

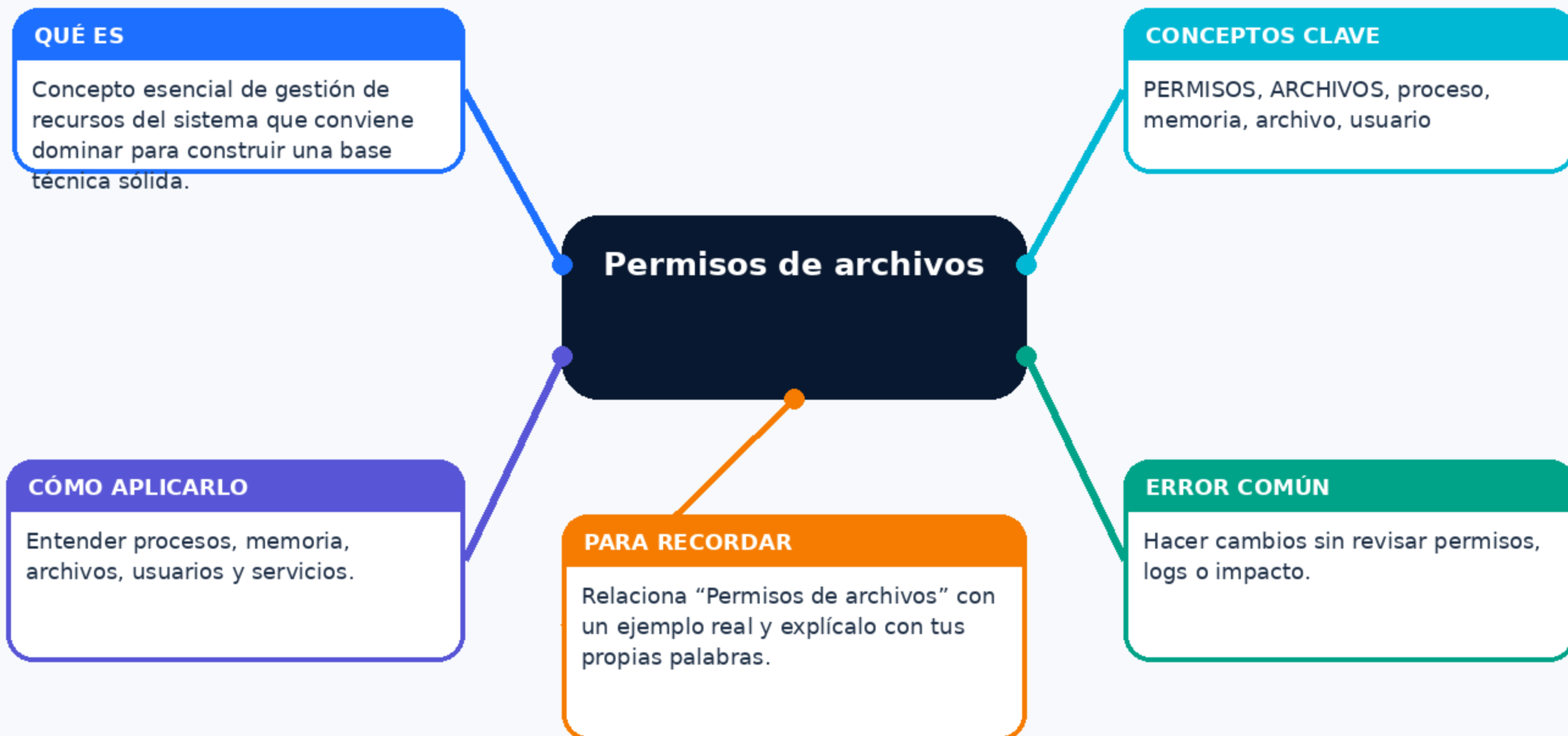
PARA RECORDAR

Relaciona "Paginación" con un ejemplo real y explícalo con tus propias palabras.

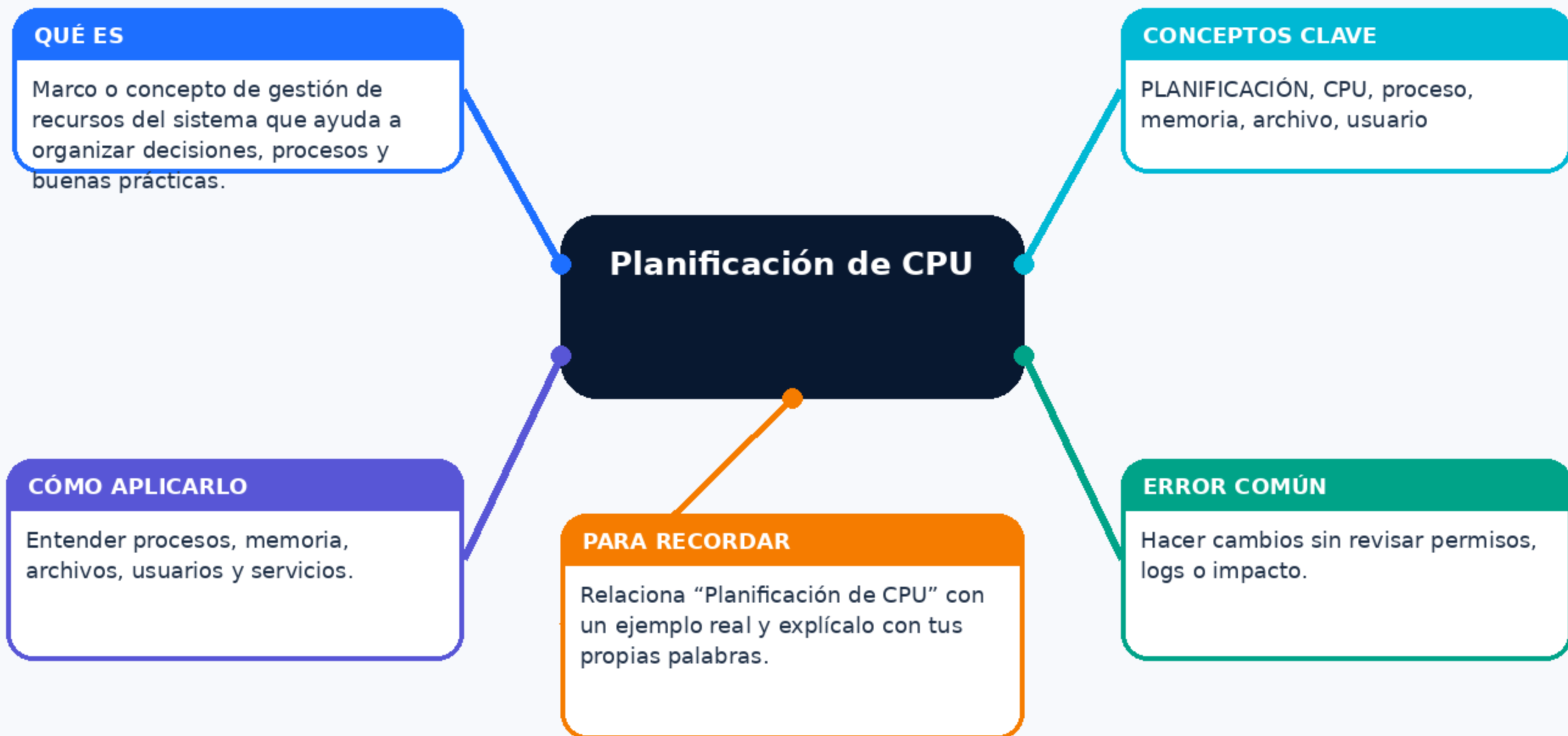
ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Permisos de archivos



Planificación de CPU



PowerShell

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

POWERSHELL, proceso, memoria, archivo, usuario, servicio

PowerShell

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "PowerShell" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Procesos

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

PROCESOS, proceso, memoria, archivo, usuario, servicio

Procesos

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "Procesos" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Servicios y demonios

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

SERVICIOS, DEMONIOS, proceso, memoria, archivo, usuario

Servicios y demonios

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "Servicios y demonios" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Shell

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

SHELL, proceso, memoria, archivo, usuario, servicio

Shell

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

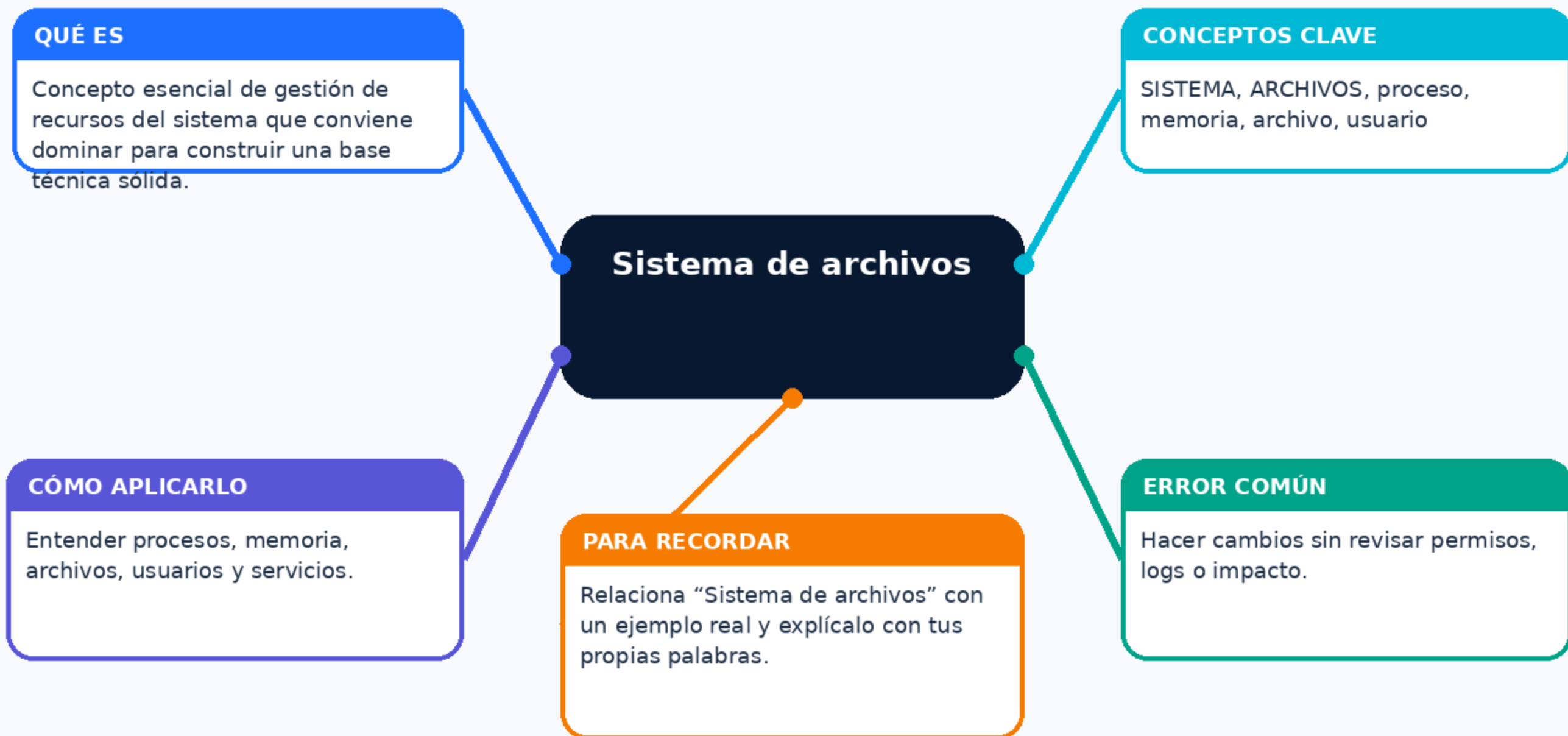
PARA RECORDAR

Relaciona "Shell" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

Sistema de archivos



Usuarios y grupos

QUÉ ES

Concepto esencial de gestión de recursos del sistema que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

USUARIOS, GRUPOS, proceso, memoria, archivo, usuario

Usuarios y grupos

CÓMO APLICARLO

Entender procesos, memoria, archivos, usuarios y servicios.

PARA RECORDAR

Relaciona "Usuarios y grupos" con un ejemplo real y explícalo con tus propias palabras.

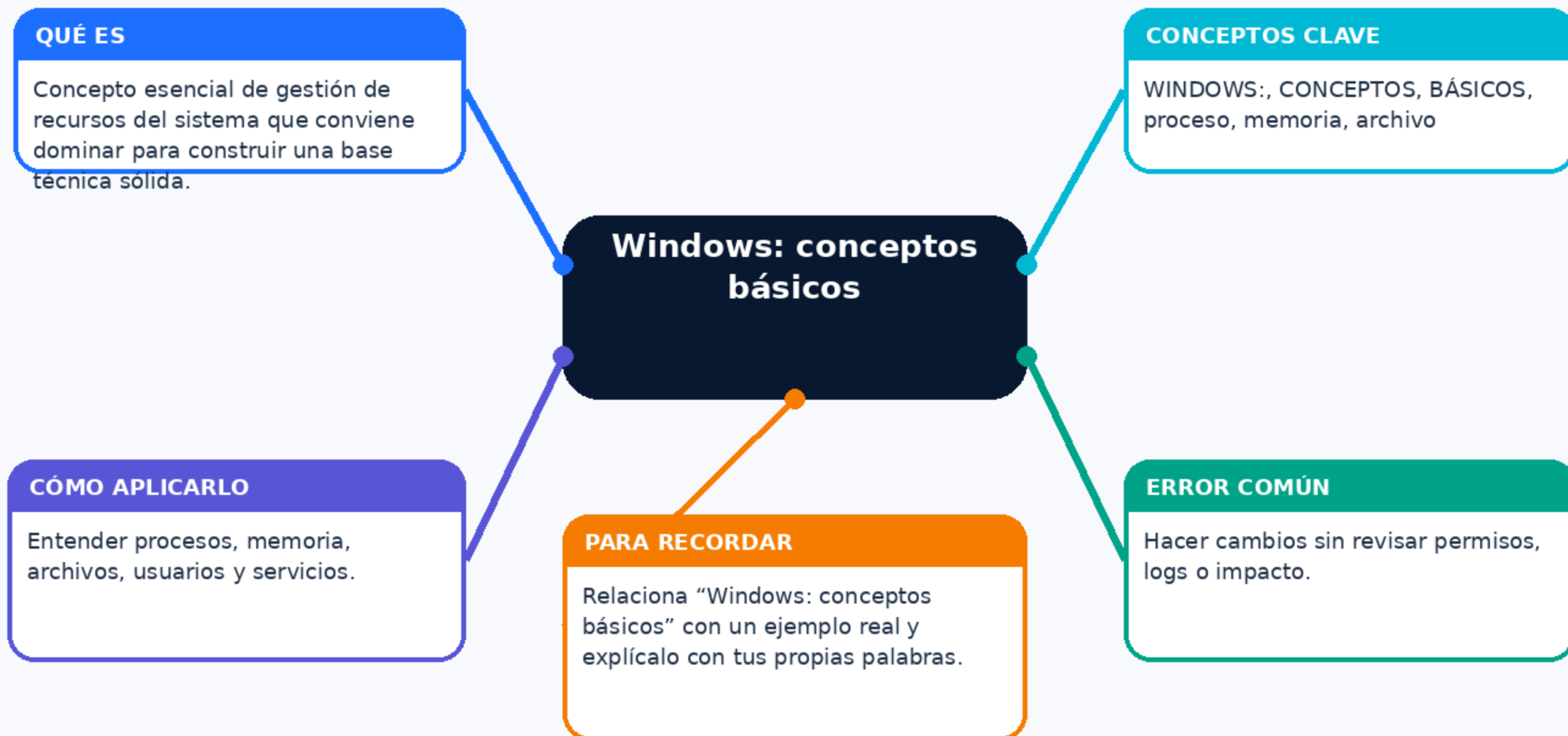
ERROR COMÚN

Hacer cambios sin revisar permisos, logs o impacto.

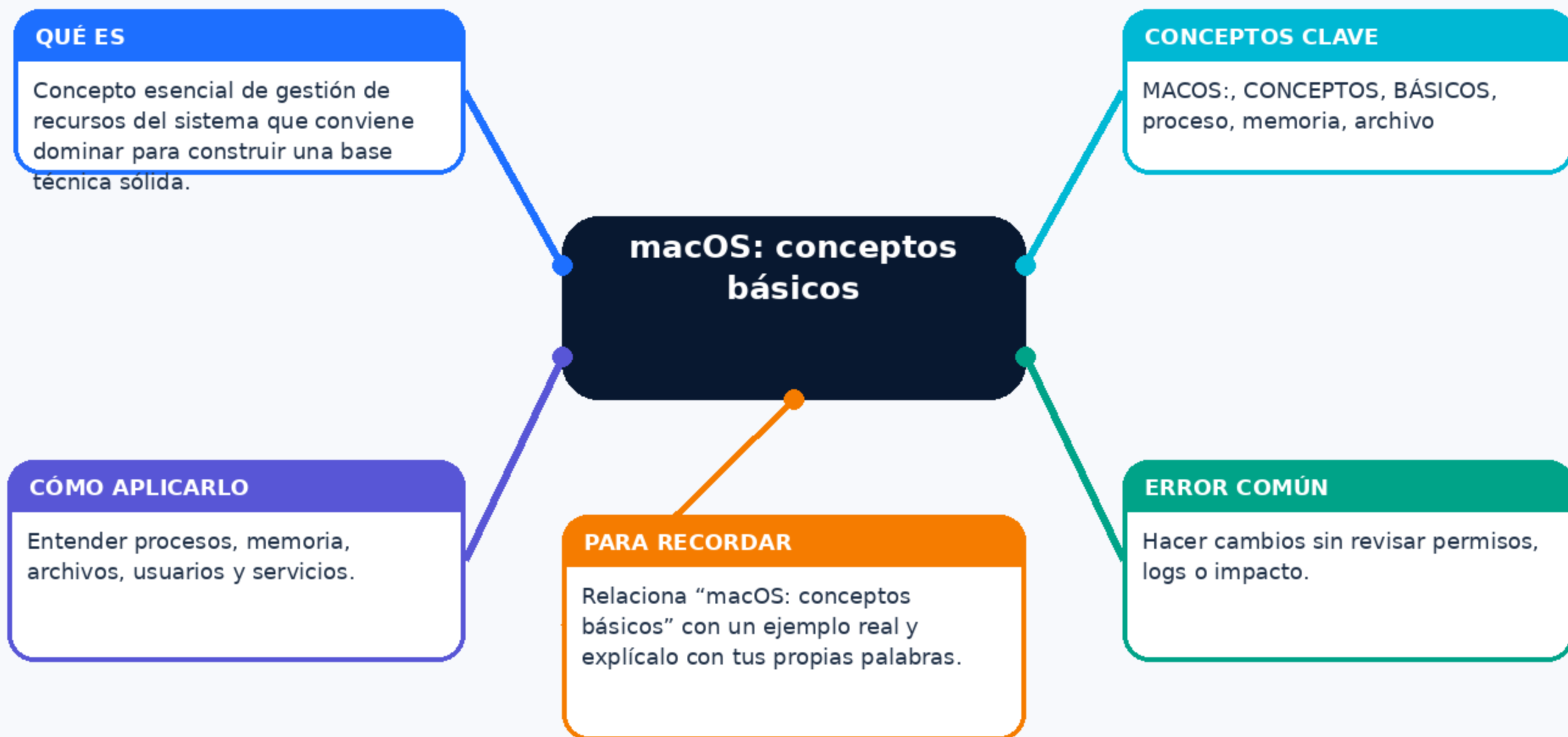
Variables de entorno



Windows: conceptos básicos



macOS: conceptos básicos



ARP

QUÉ ES

Concepto de comunicación dentro de comunicación entre dispositivos que define cómo se intercambia o interpreta información.

CONCEPTOS CLAVE

ARP, dirección, protocolo, capa, puerto, tráfico

ARP

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "ARP" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

Access Point

QUÉ ES

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ACCESS, POINT, dirección, protocolo, capa, puerto

Access Point

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

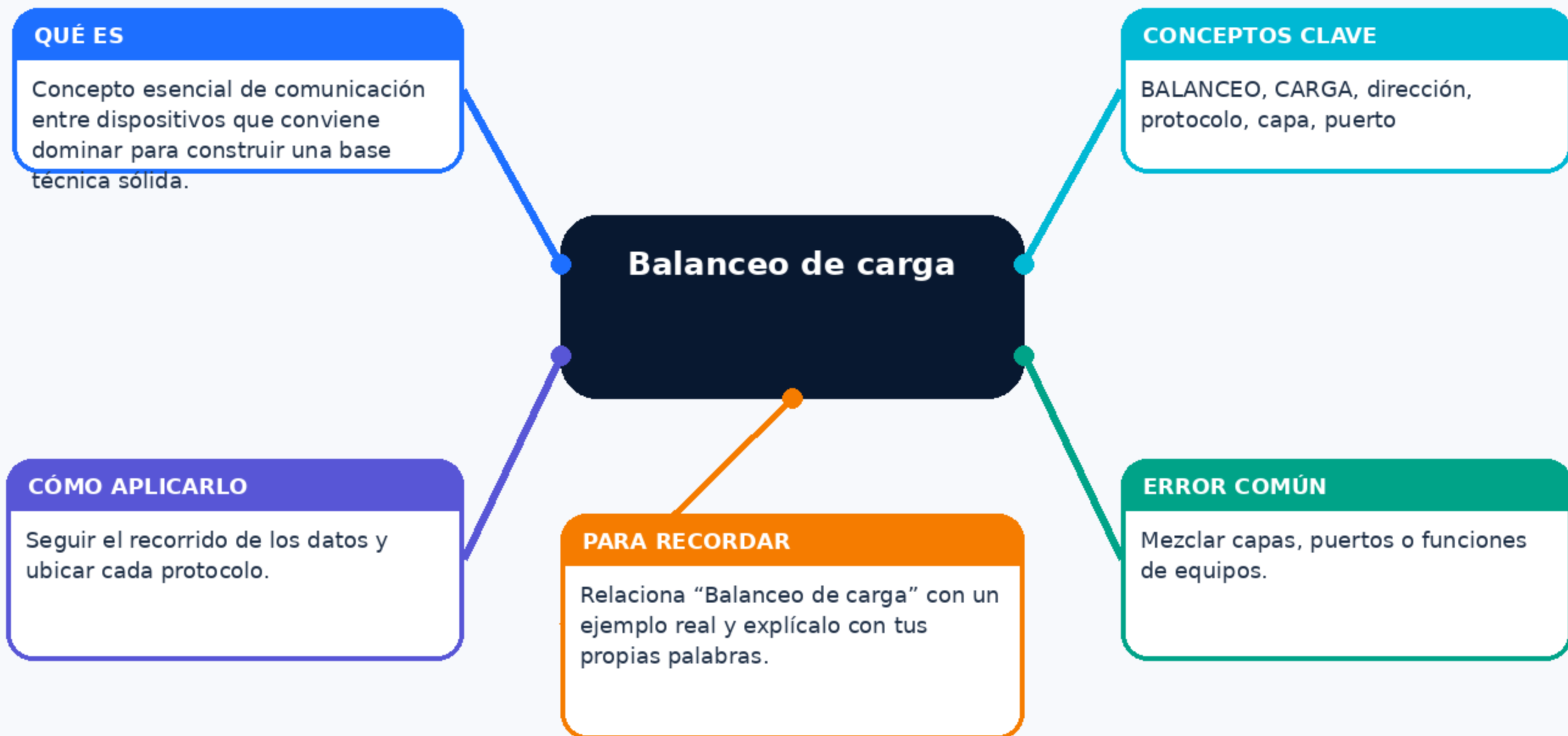
PARA RECORDAR

Relaciona "Access Point" con un ejemplo real y explícalo con tus propias palabras.

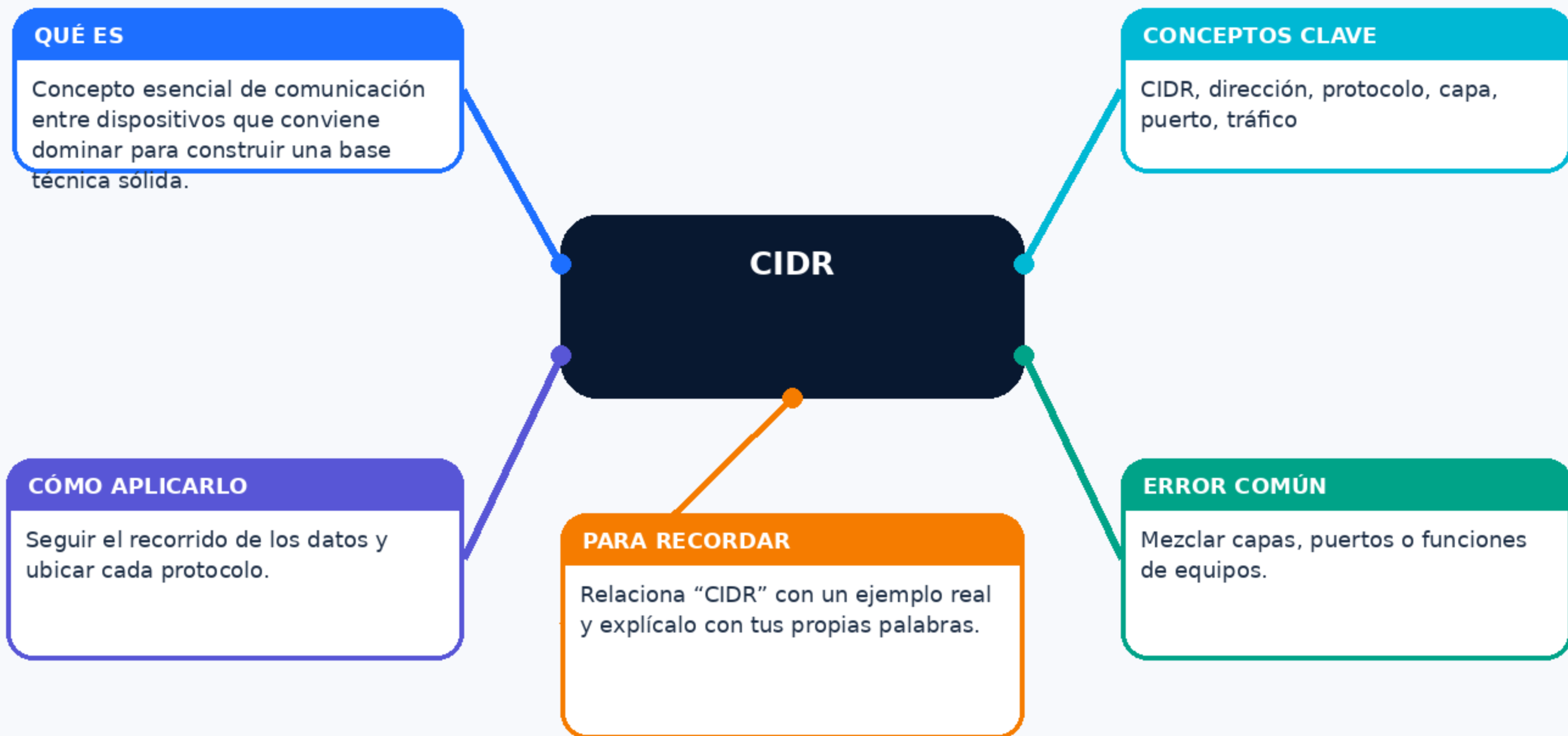
ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

Balanceo de carga



CIDR



DHCP

QUÉ ES

Protocolo que asigna automáticamente configuración IP a los dispositivos.

CONCEPTOS CLAVE

Discover, Offer, Request, Acknowledge, lease

DHCP

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "DHCP" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

DNS

QUÉ ES

Sistema que traduce nombres de dominio a direcciones IP y otros registros.

CONCEPTOS CLAVE

A, AAAA, CNAME, MX, NS, TTL

DNS

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "DNS" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

Dirección IPv4

QUÉ ES

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

DIRECCIÓN, IPV4, dirección, protocolo, capa, puerto

Dirección IPv4

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "Dirección IPv4" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

Dirección IPv6



FTP y SFTP

QUÉ ES

Concepto de comunicación dentro de comunicación entre dispositivos que define cómo se intercambia o interpreta información.

CONCEPTOS CLAVE

FTP, SFTP, dirección, protocolo, capa, puerto

FTP y SFTP

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "FTP y SFTP" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

Gateway predeterminado



HTTP

QUÉ ES

Protocolo de aplicación para intercambio de recursos en la Web.

CONCEPTOS CLAVE

request, response, métodos, headers, status codes

HTTP

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "HTTP" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

HTTPS

QUÉ ES

HTTP protegido mediante TLS para aportar cifrado, integridad y autenticación.

CONCEPTOS CLAVE

TLS, certificado, puerto 443, cifrado, identidad

HTTPS

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

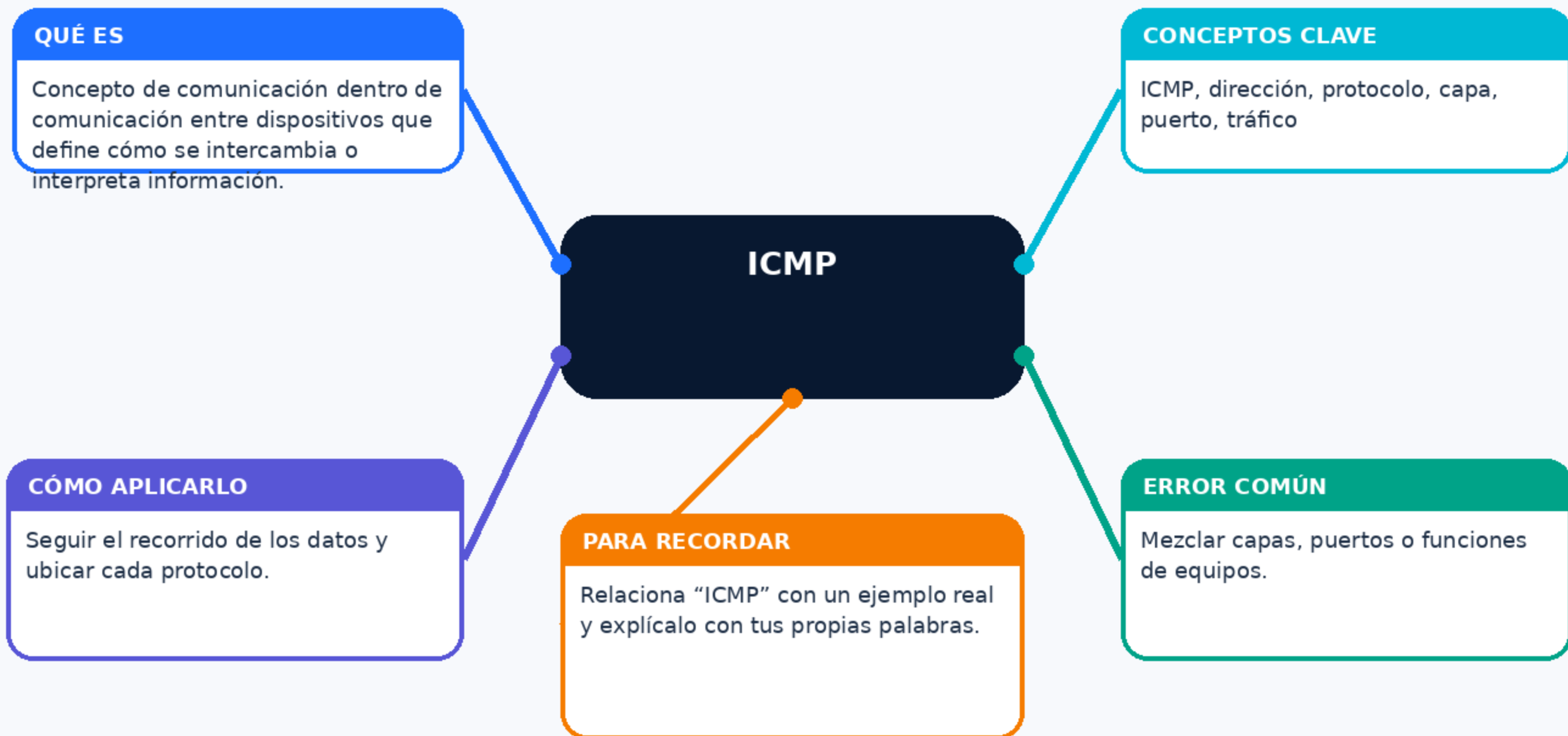
PARA RECORDAR

Relaciona "HTTPS" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

ICMP



LAN, MAN y WAN

QUÉ ES

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

LAN, MAN, WAN, dirección, protocolo, capa

LAN, MAN y WAN

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "LAN, MAN y WAN" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

MAC address

QUÉ ES

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

MAC, ADDRESS, dirección, protocolo, capa, puerto

MAC address

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

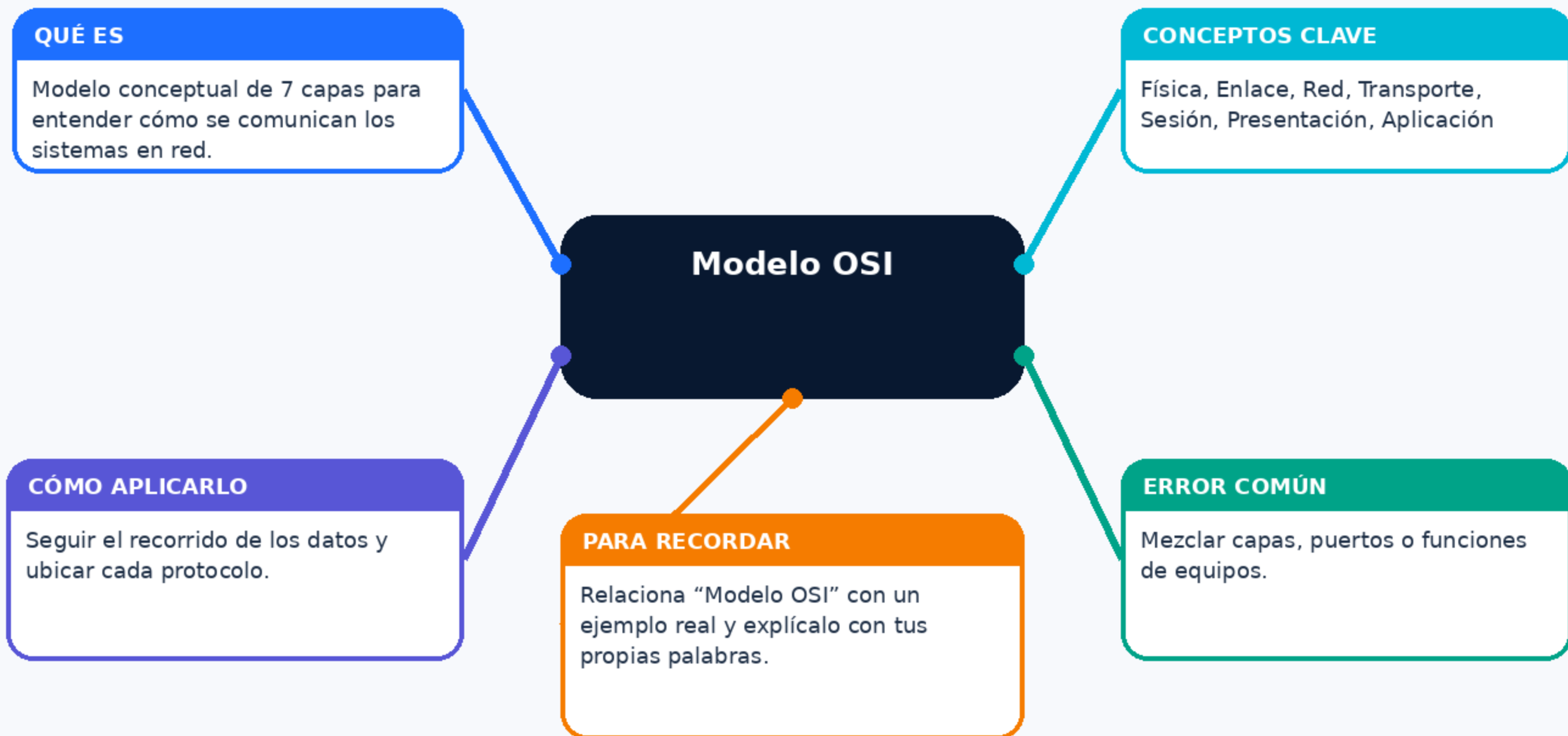
PARA RECORDAR

Relaciona "MAC address" con un ejemplo real y explícalo con tus propias palabras.

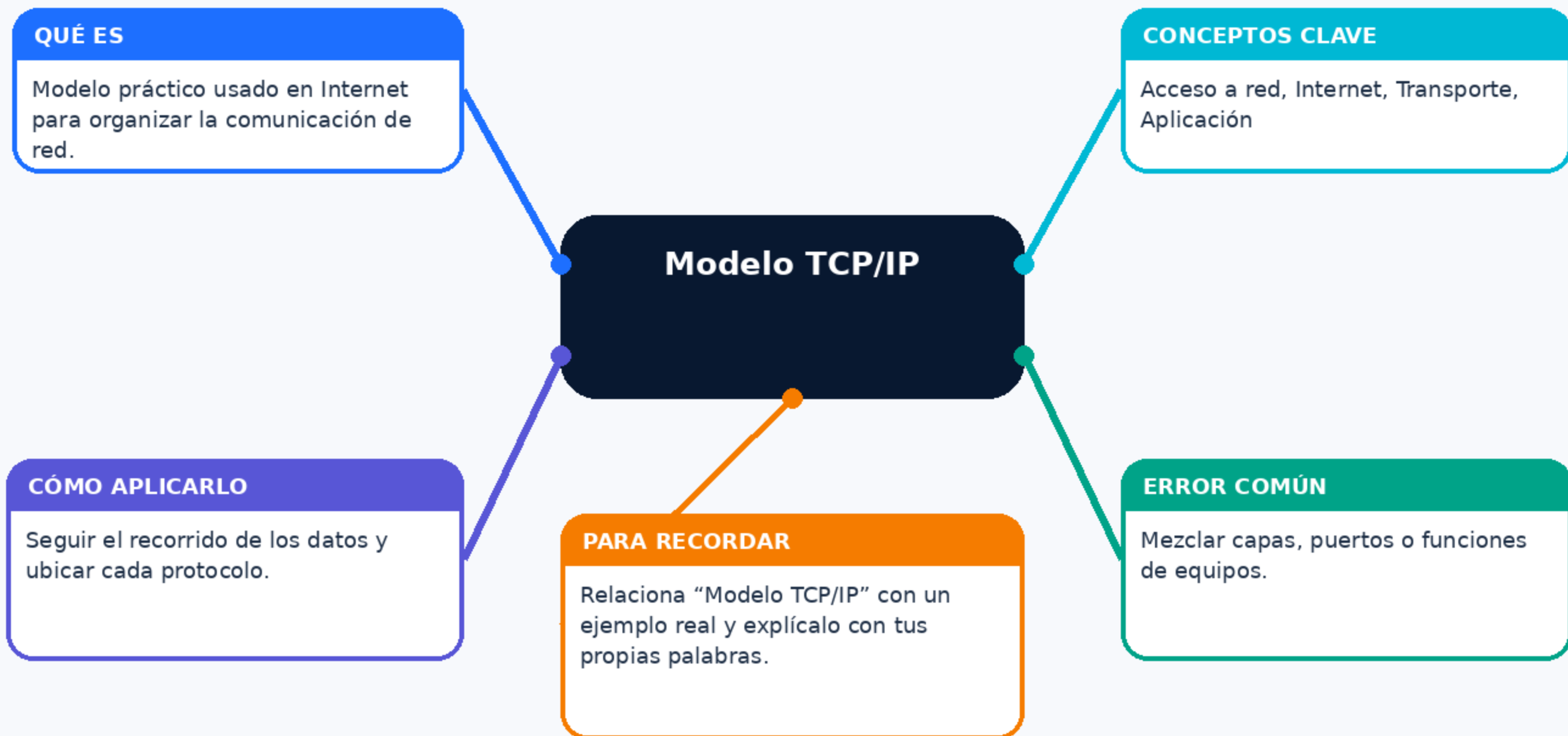
ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

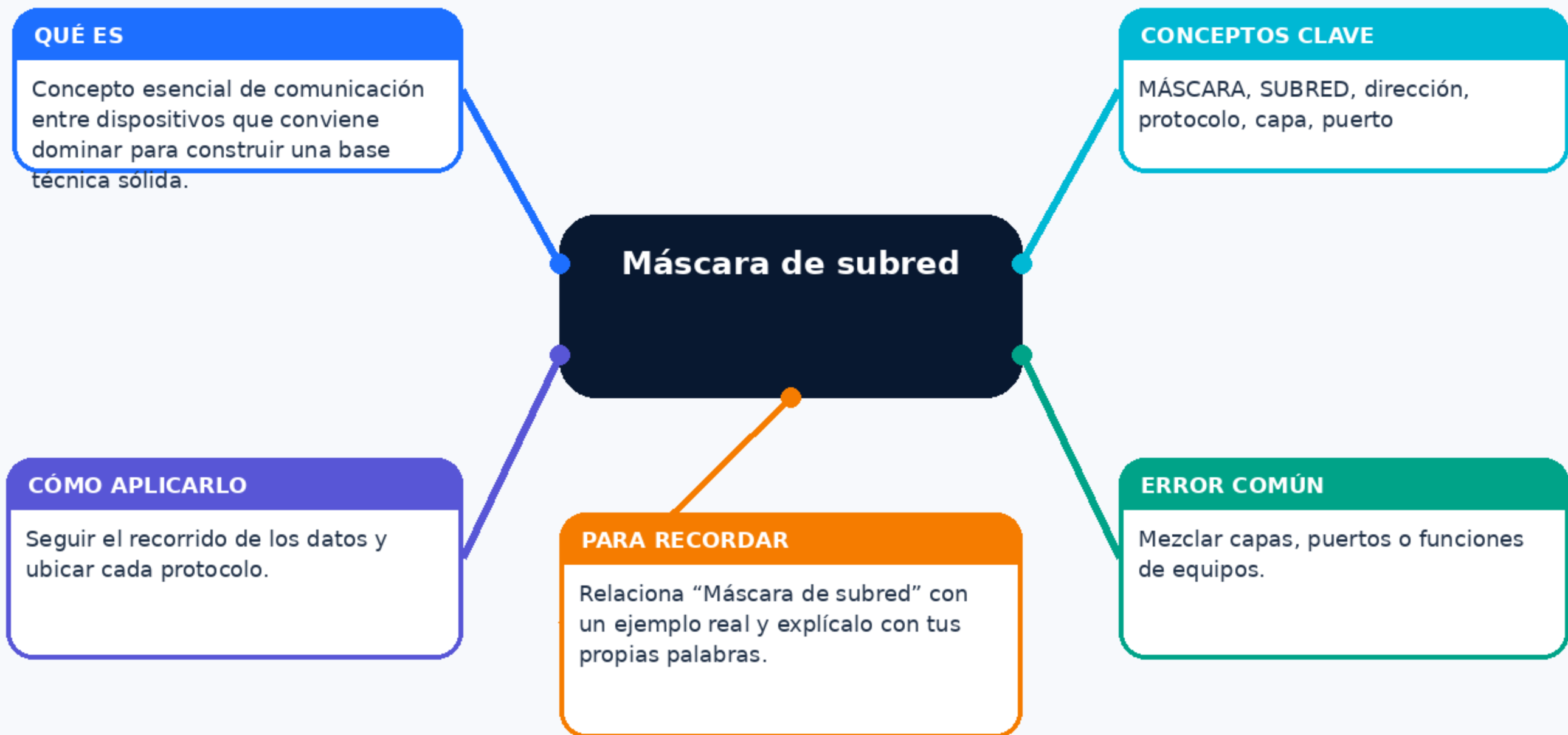
Modelo OSI



Modelo TCP/IP



Máscara de subred



NAT

QUÉ ES

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

NAT, dirección, protocolo, capa, puerto, tráfico

NAT

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

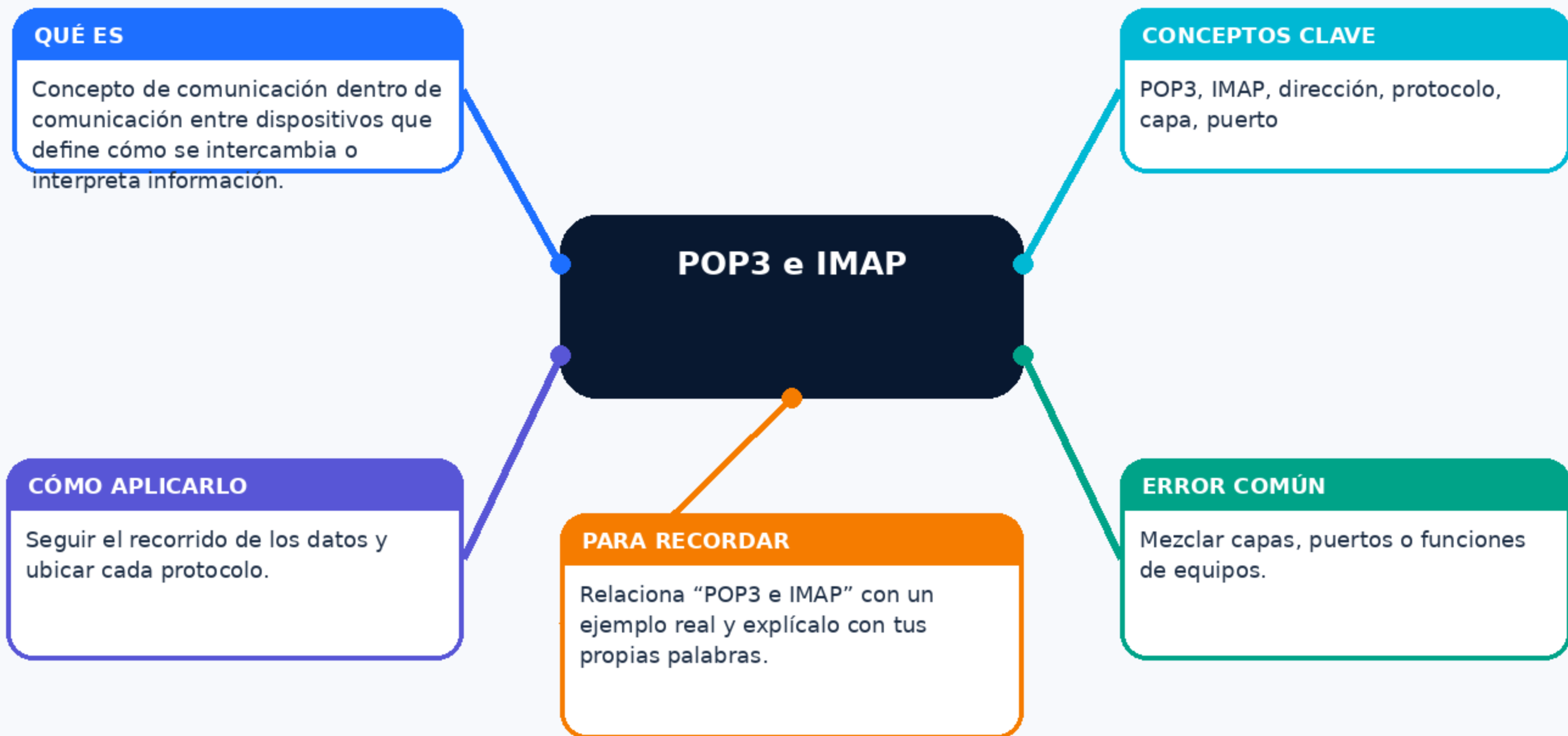
PARA RECORDAR

Relaciona "NAT" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

POP3 e IMAP



Puertos de red

QUÉ ES

Componente o tecnología de comunicación entre dispositivos con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

PUERTOS, RED, dirección, protocolo, capa, puerto

Puertos de red

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "Puertos de red" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

Router

QUÉ ES

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ROUTER, dirección, protocolo, capa, puerto, tráfico

Router

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

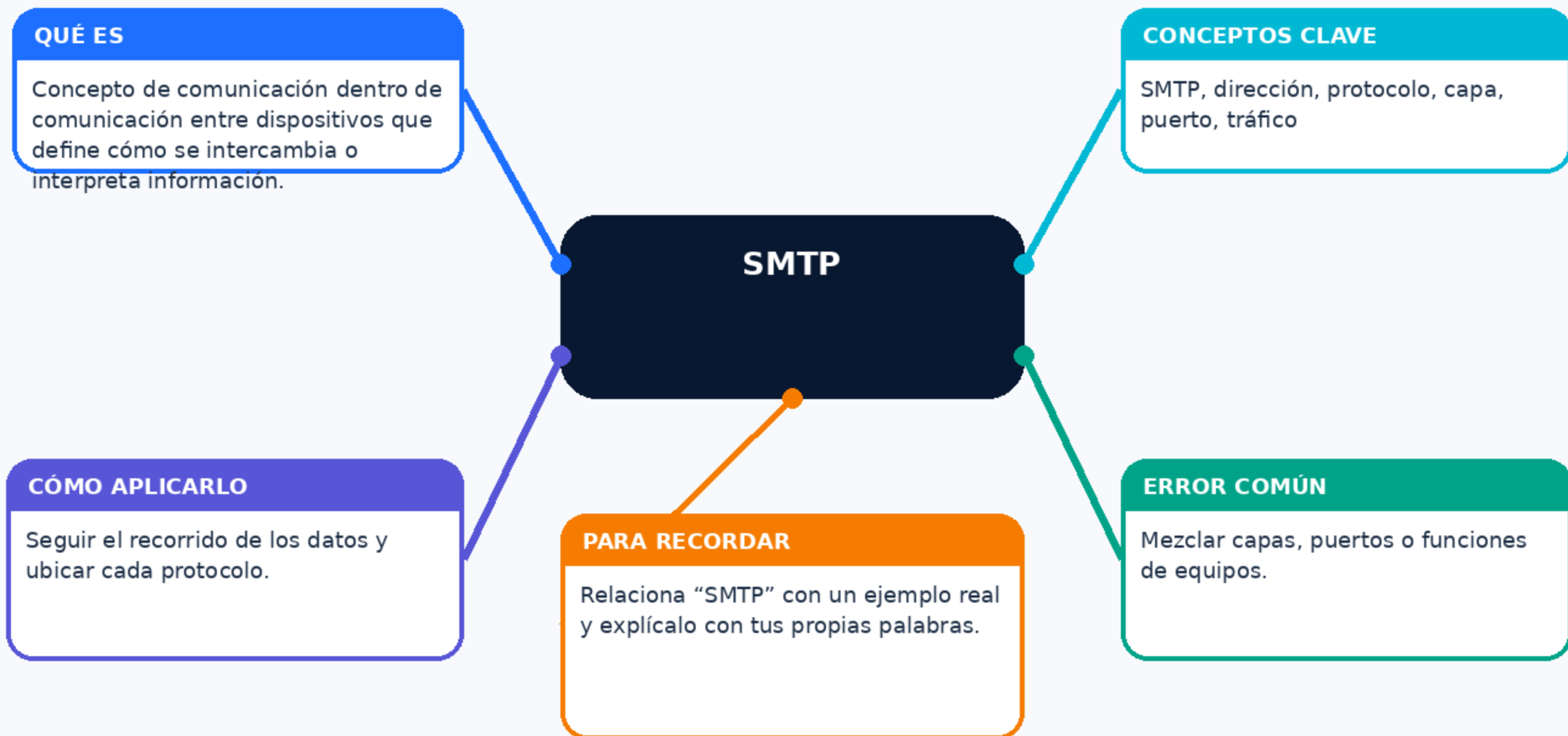
PARA RECORDAR

Relaciona "Router" con un ejemplo real y explícalo con tus propias palabras.

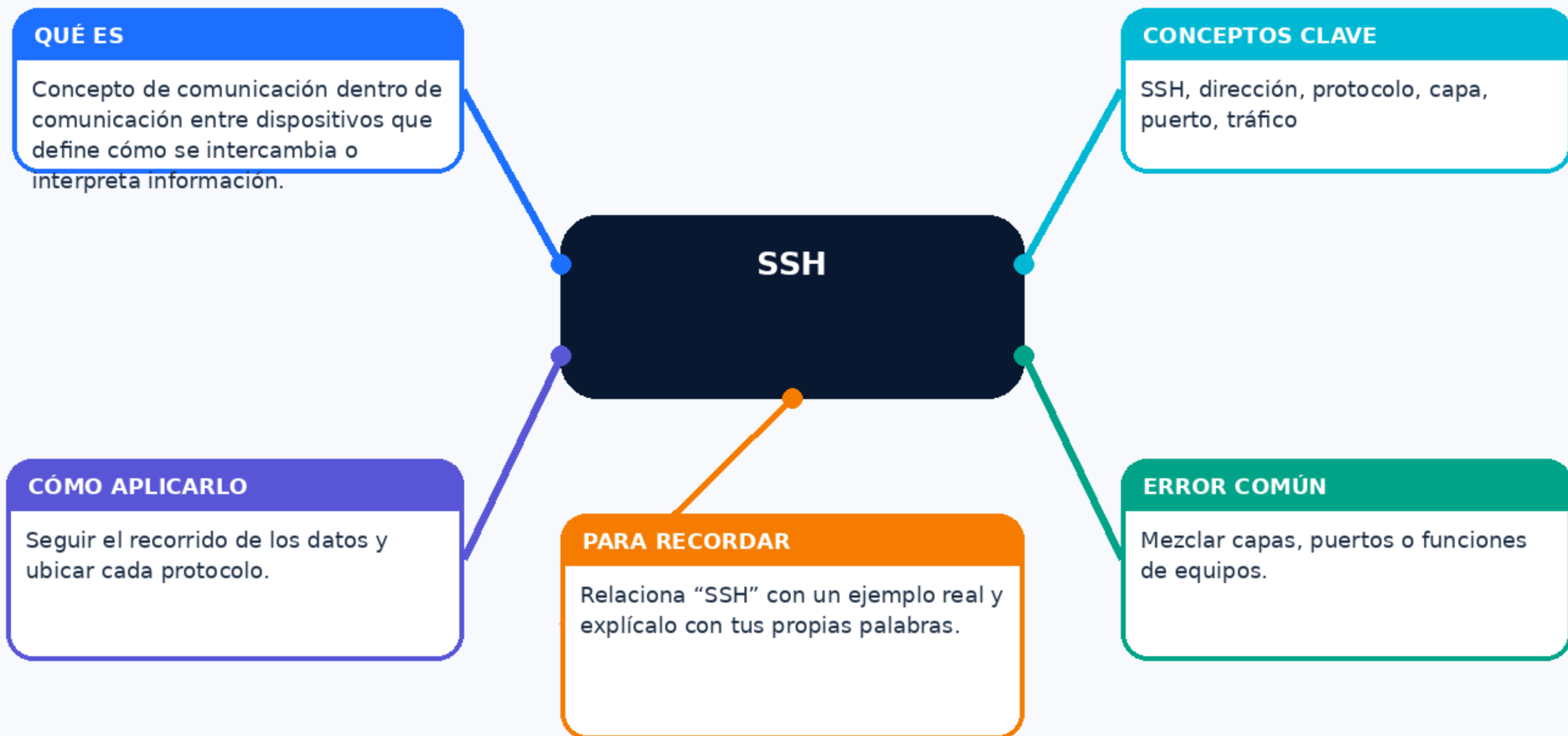
ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

SMTP



SSH



Subnetting

QUÉ ES

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

SUBNETTING, dirección, protocolo, capa, puerto, tráfico

Subnetting

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

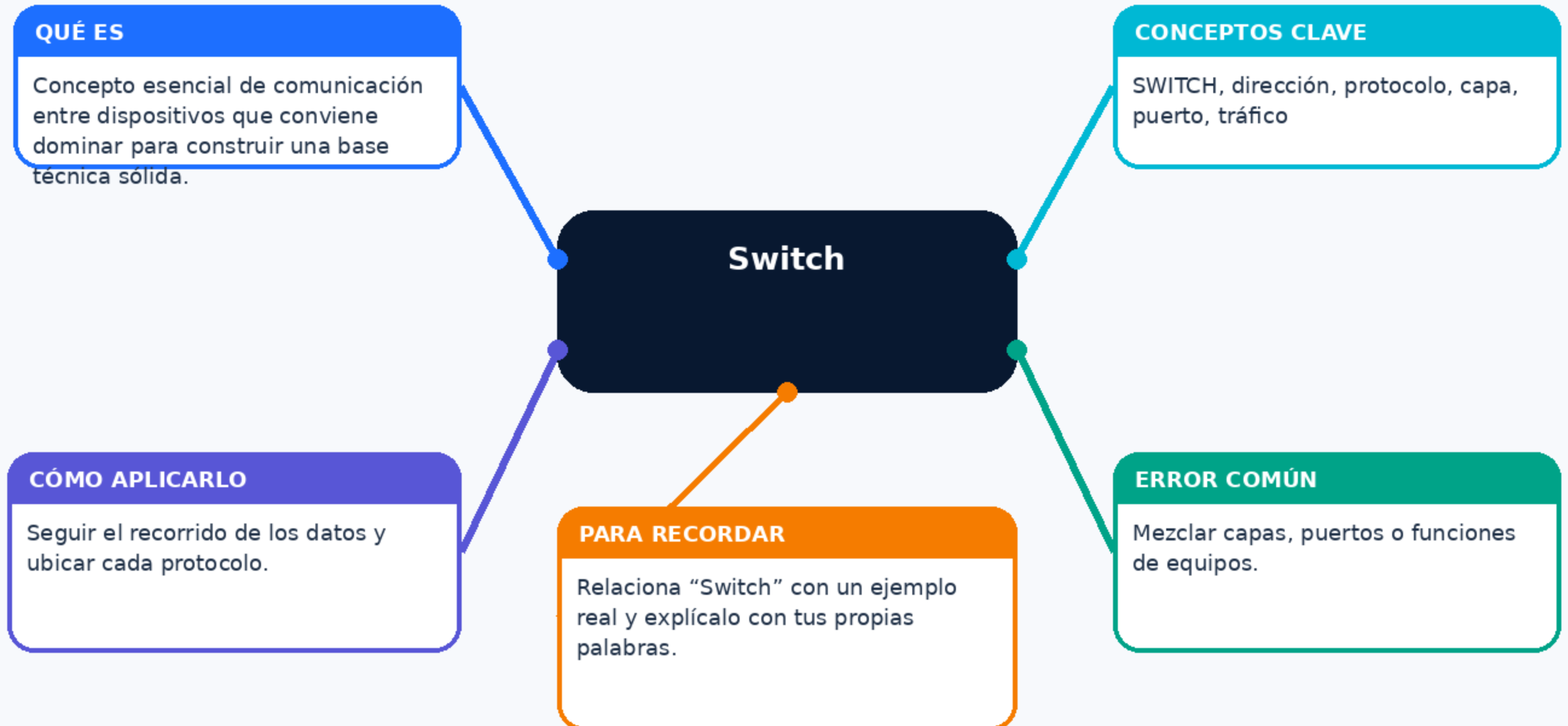
PARA RECORDAR

Relaciona "Subnetting" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

Switch



TCP

QUÉ ES

Protocolo de transporte orientado a conexión, confiable y con control de entrega.

CONCEPTOS CLAVE

handshake, ACK, retransmisión, orden, control de flujo

TCP

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

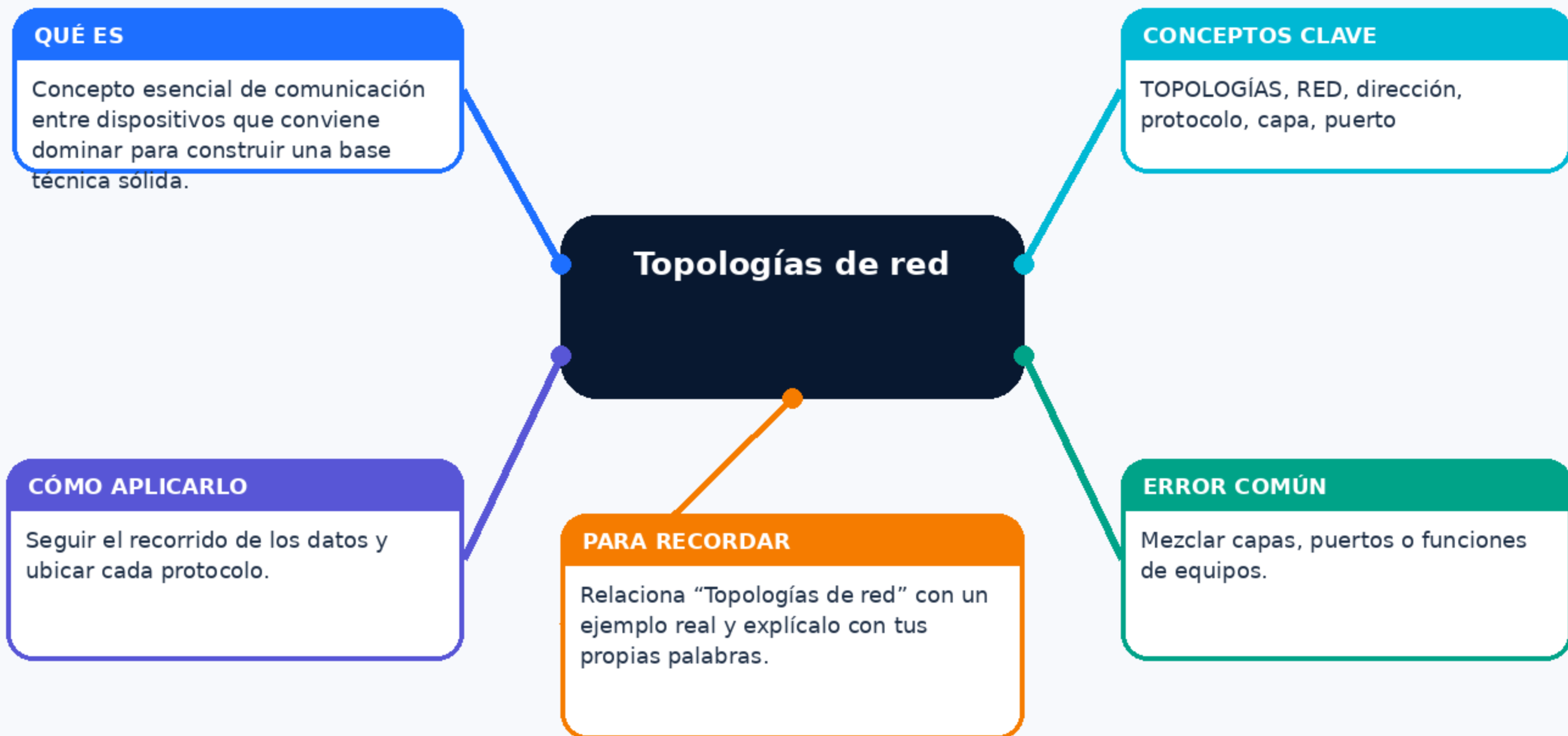
PARA RECORDAR

Relaciona "TCP" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

Topologías de red



UDP

QUÉ ES

Protocolo de transporte sin conexión, ligero y sin garantía de entrega.

CONCEPTOS CLAVE

datagramas, baja latencia, streaming, DNS, VoIP

UDP

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "UDP" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

VLAN

QUÉ ES

Concepto esencial de comunicación entre dispositivos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

VLAN, dirección, protocolo, capa, puerto, tráfico

VLAN

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

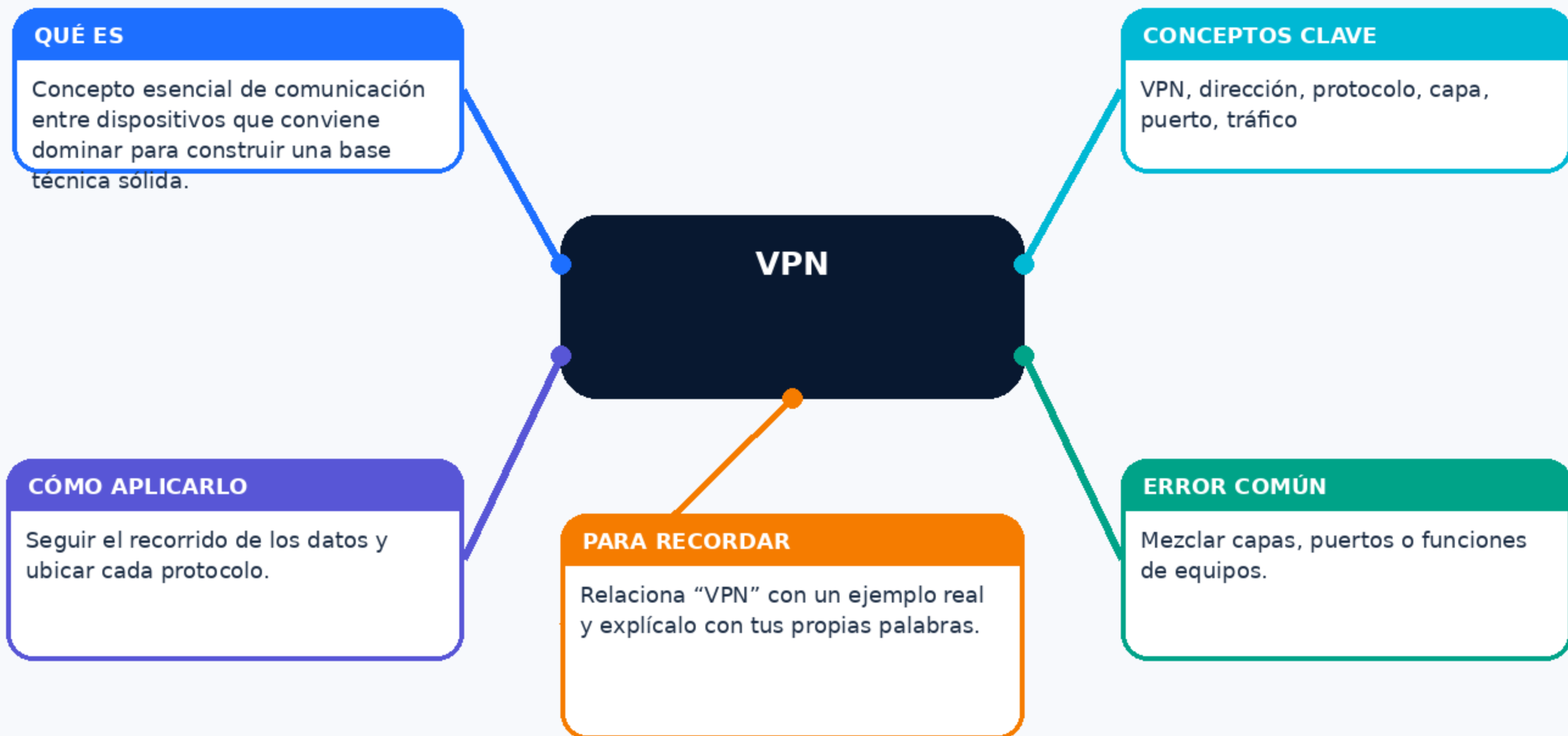
PARA RECORDAR

Relaciona "VLAN" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

VPN



Wi-Fi 2.4 vs. 5 GHz

QUÉ ES

Comparación entre conceptos relacionados de comunicación entre dispositivos, destacando cuándo conviene cada uno.

CONCEPTOS CLAVE

WI-FI, 2.4, GHZ, dirección, protocolo, capa

Wi-Fi 2.4 vs. 5 GHz

CÓMO APLICARLO

Seguir el recorrido de los datos y ubicar cada protocolo.

PARA RECORDAR

Relaciona "Wi-Fi 2.4 vs. 5 GHz" con un ejemplo real y explícalo con tus propias palabras.

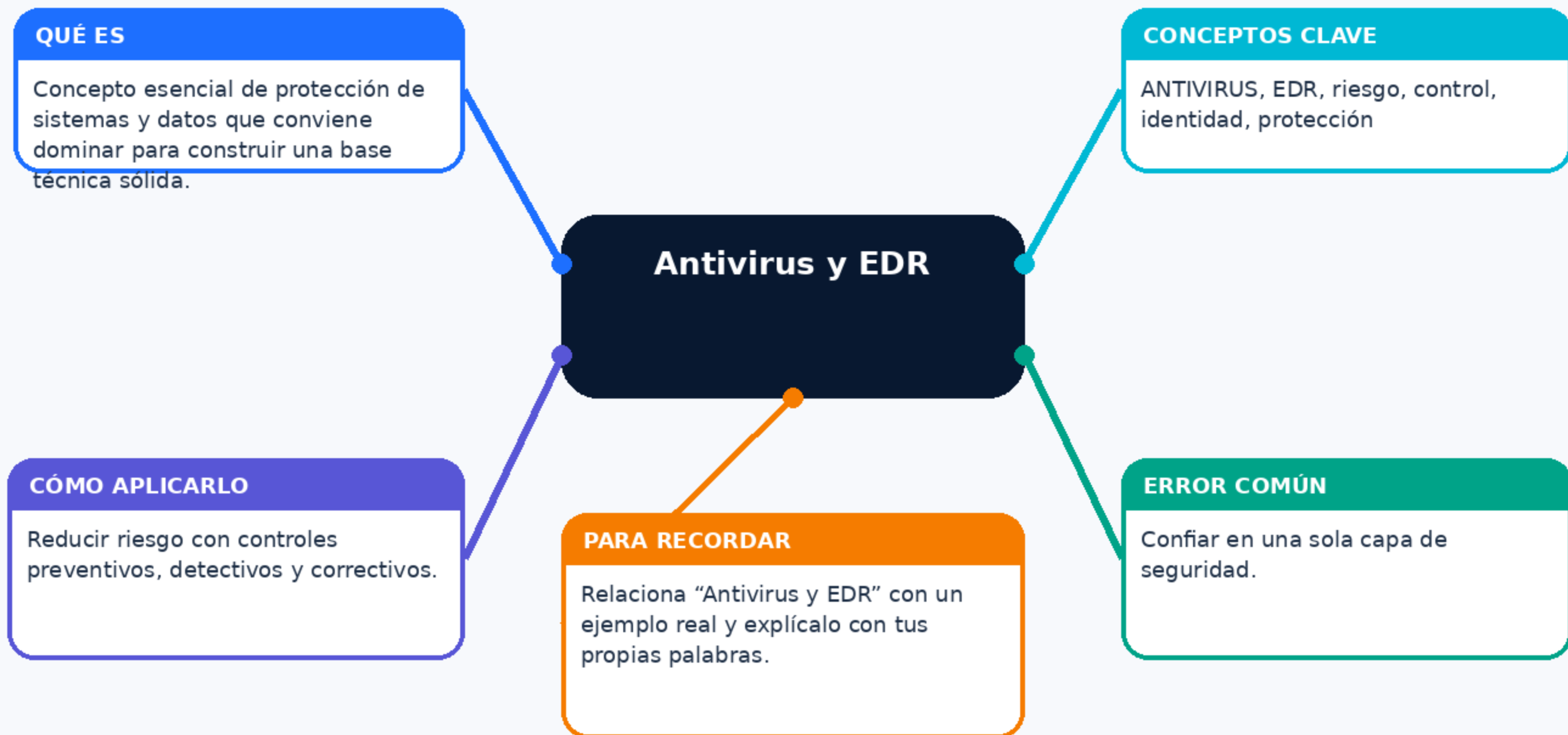
ERROR COMÚN

Mezclar capas, puertos o funciones de equipos.

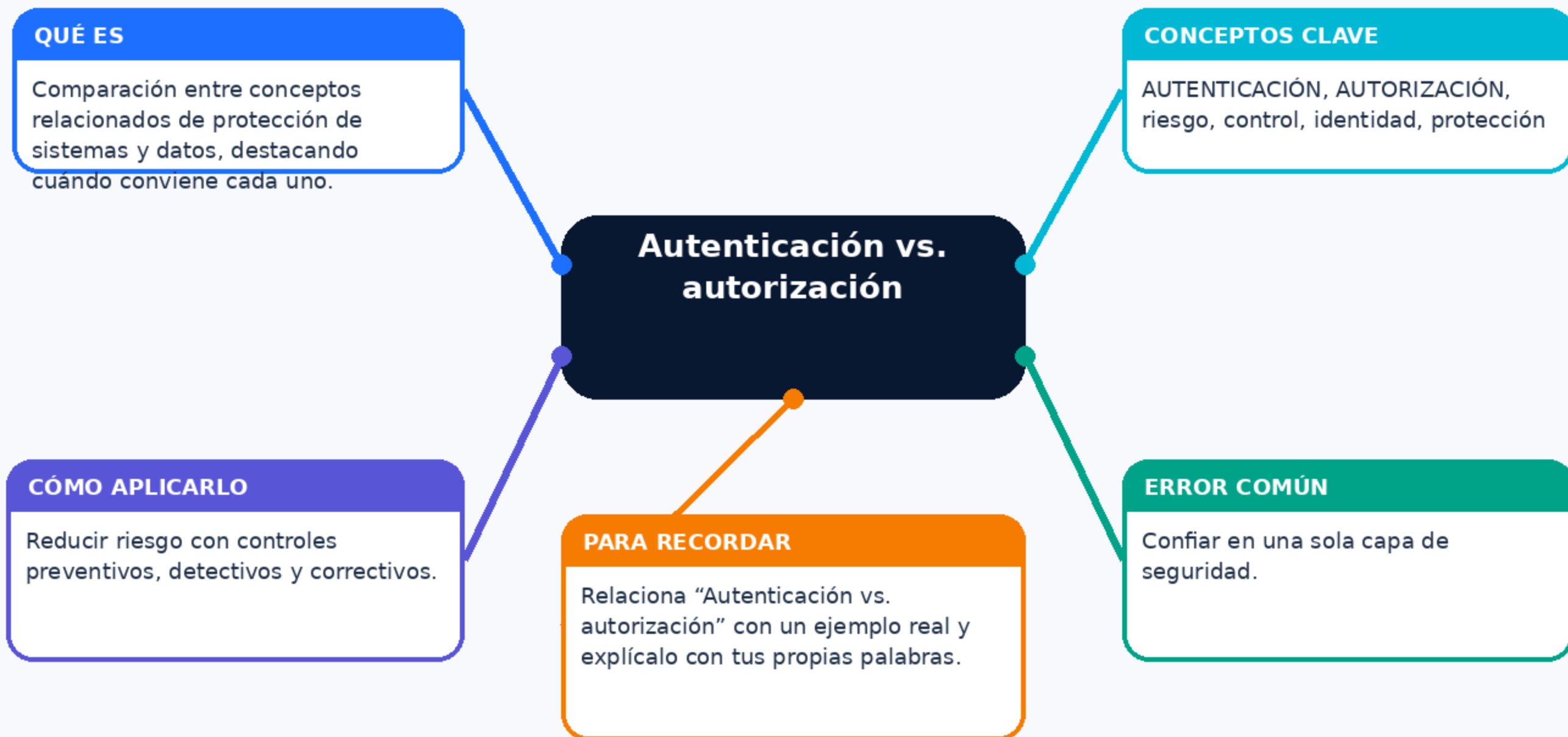
Amenaza, vulnerabilidad y riesgo



Antivirus y EDR



Autenticación vs. autorización



Backups 3-2-1

QUÉ ES

Estrategia de respaldo: 3 copias, 2 medios distintos y 1 copia fuera del sitio.

CONCEPTOS CLAVE

redundancia, offline/offsite, pruebas de restauración

Backups 3-2-1

CÓMO APLICARLO

Reducir riesgo con controles preventivos, detectivos y correctivos.

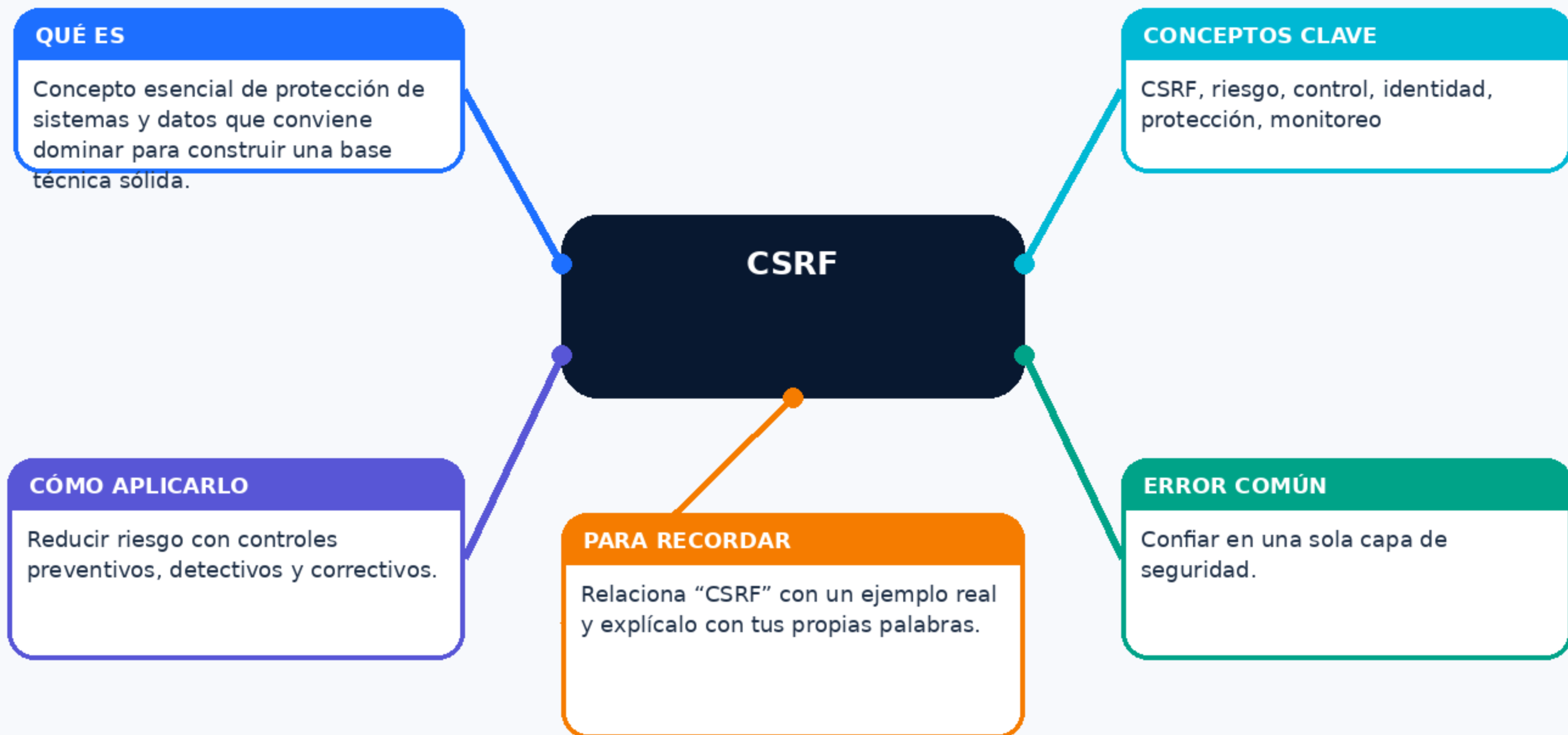
PARA RECORDAR

Relaciona "Backups 3-2-1" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confiar en una sola capa de seguridad.

CSRF



Cifrado asimétrico



Cifrado simétrico

QUÉ ES

Cifrado que usa la misma clave para cifrar y descifrar información.

CONCEPTOS CLAVE

AES, clave compartida, rapidez, gestión de claves

Cifrado simétrico

CÓMO APLICARLO

Reducir riesgo con controles preventivos, detectivos y correctivos.

PARA RECORDAR

Relaciona "Cifrado simétrico" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confiar en una sola capa de seguridad.

Confidencialidad, integridad y disponibilidad



Contraseñas seguras



Firewall

QUÉ ES

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

FIREWALL, riesgo, control, identidad, protección, monitoreo

Firewall

CÓMO APLICARLO

Reducir riesgo con controles preventivos, detectivos y correctivos.

PARA RECORDAR

Relaciona "Firewall" con un ejemplo real y explícalo con tus propias palabras.

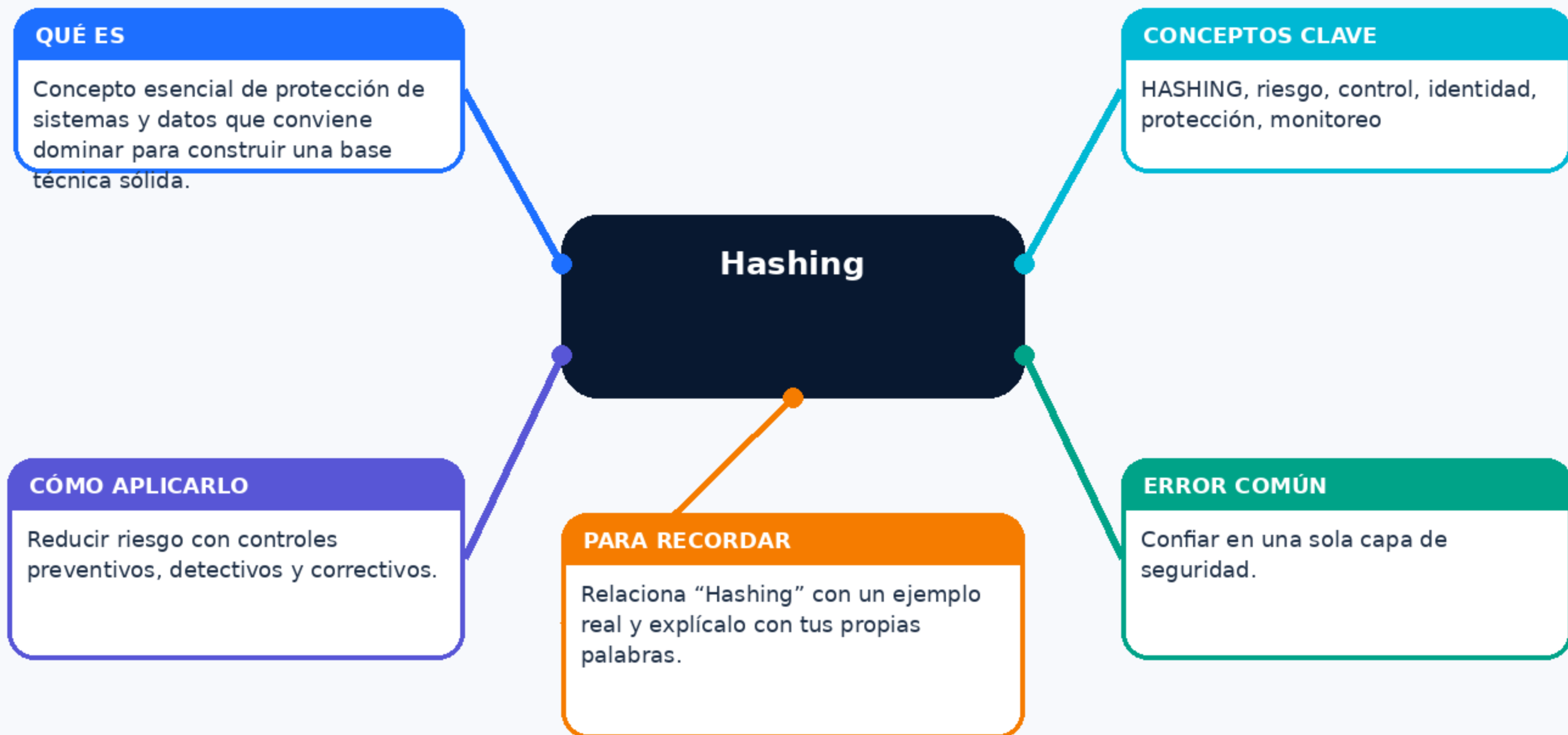
ERROR COMÚN

Confiar en una sola capa de seguridad.

Gestión de parches



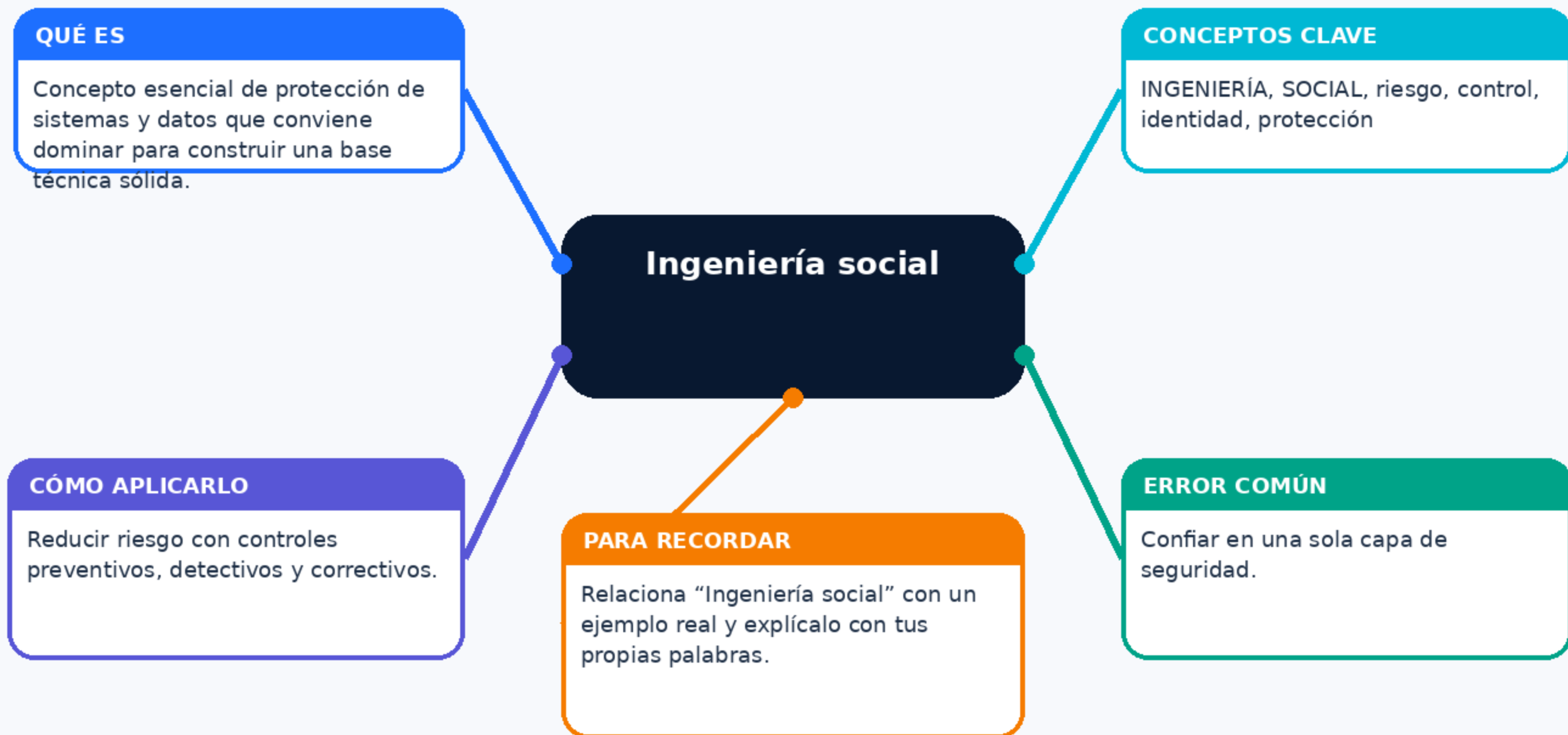
Hashing



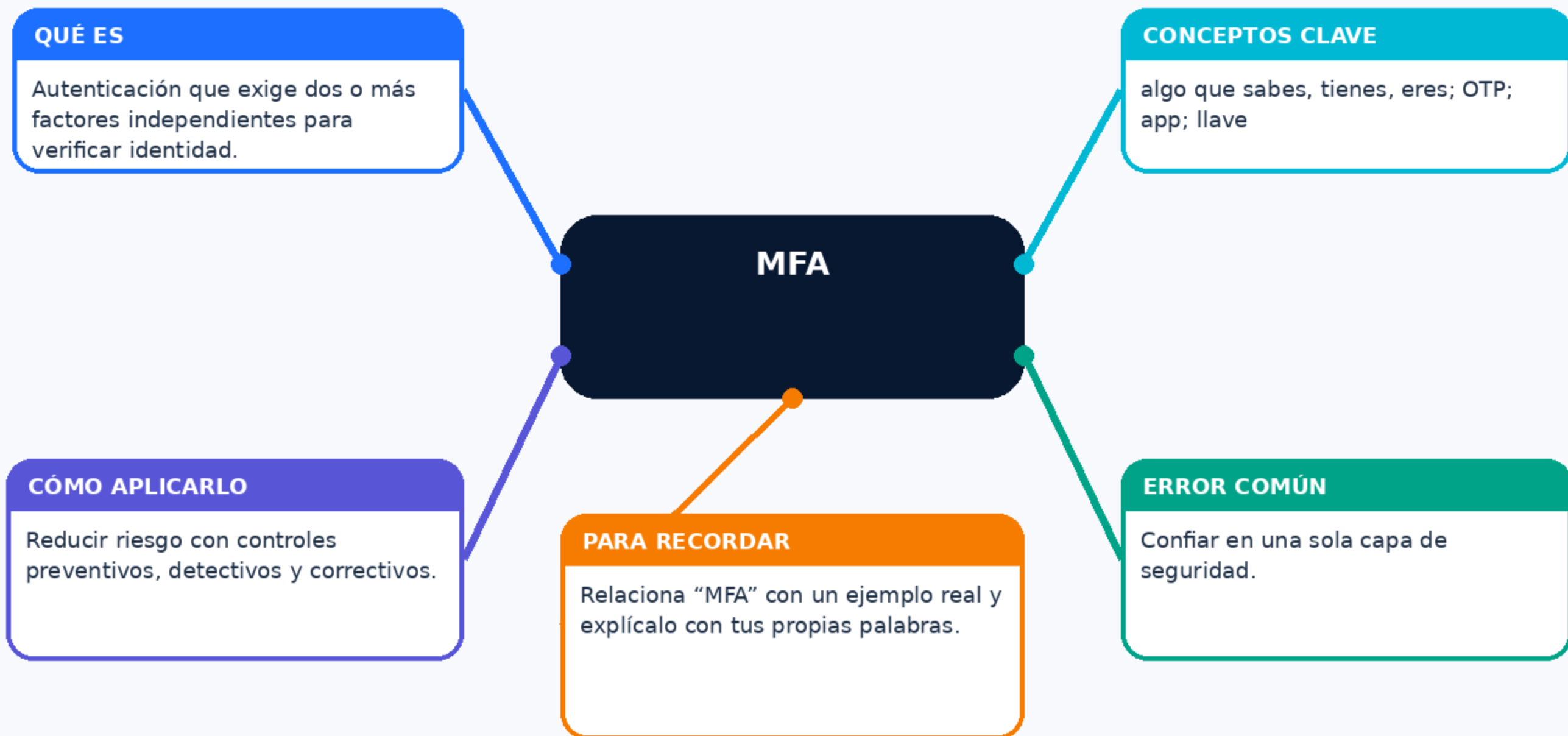
IDS e IPS



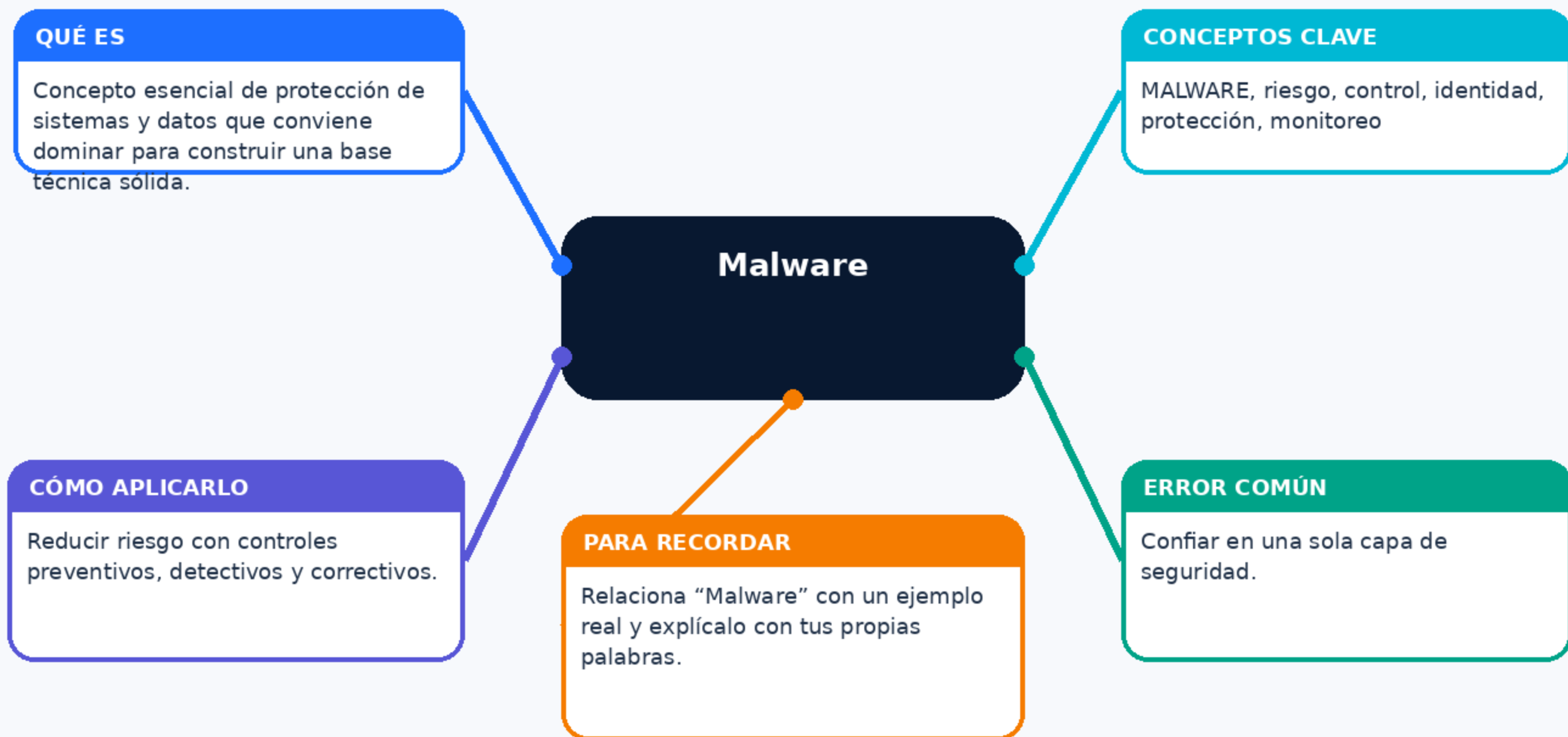
Ingeniería social



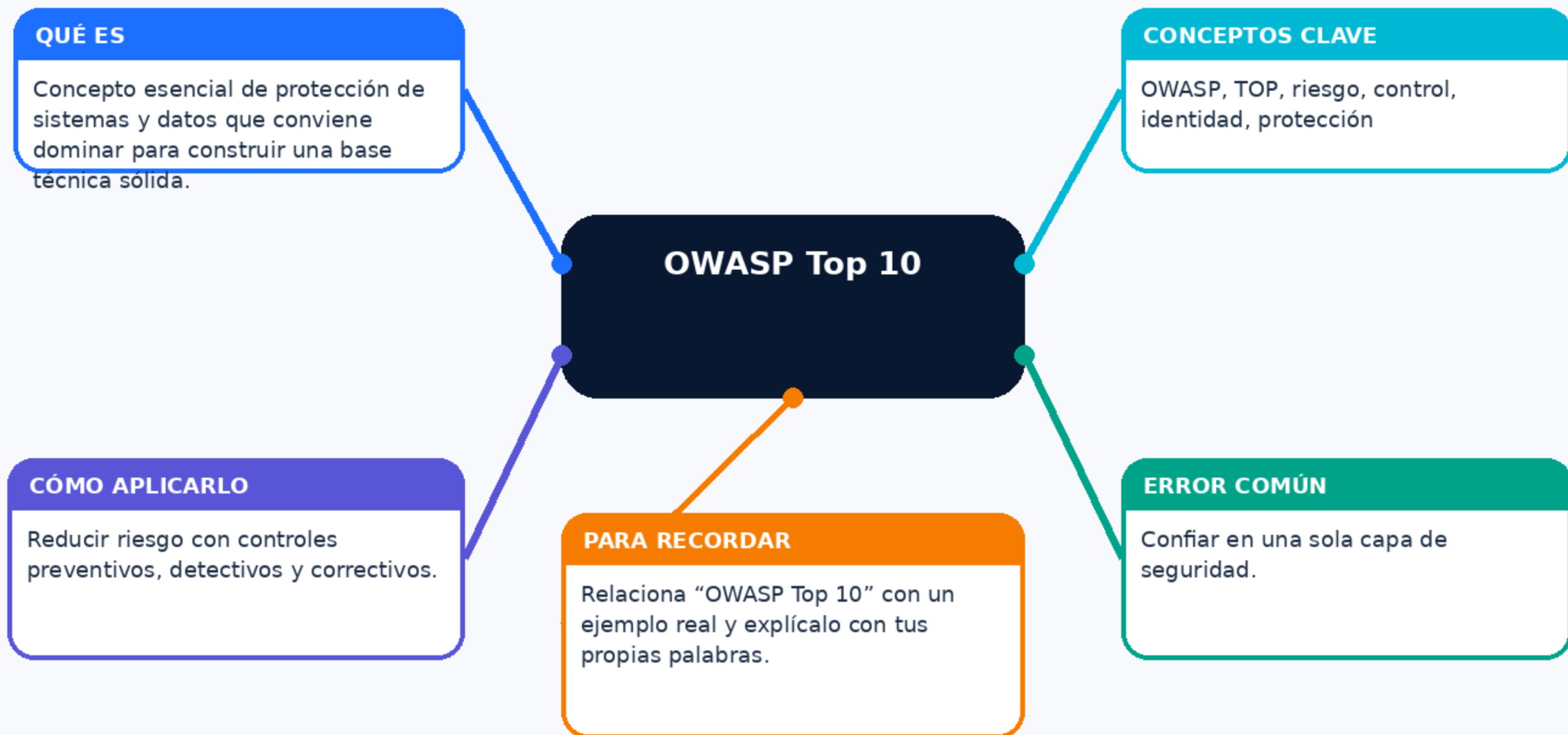
MFA



Malware



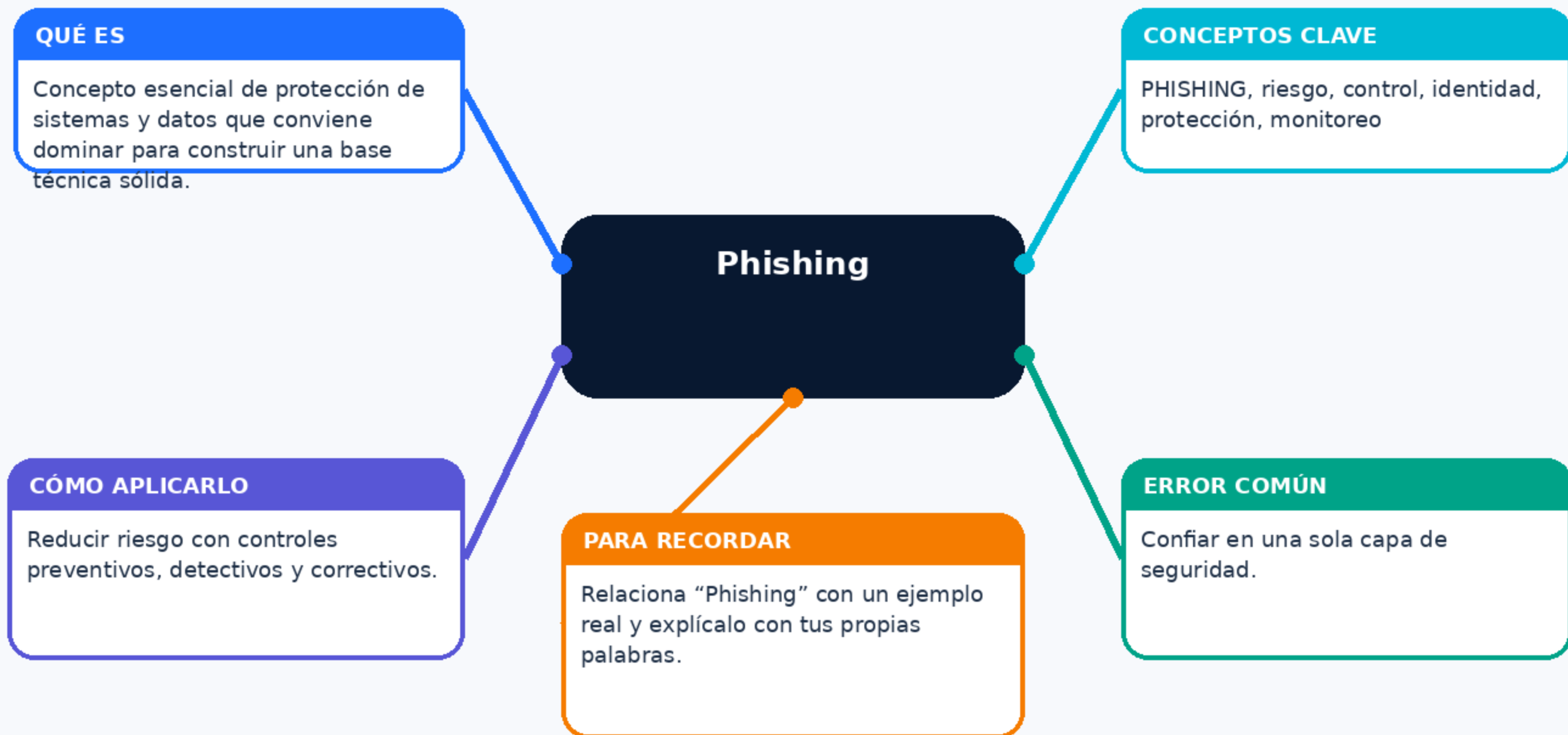
OWASP Top 10



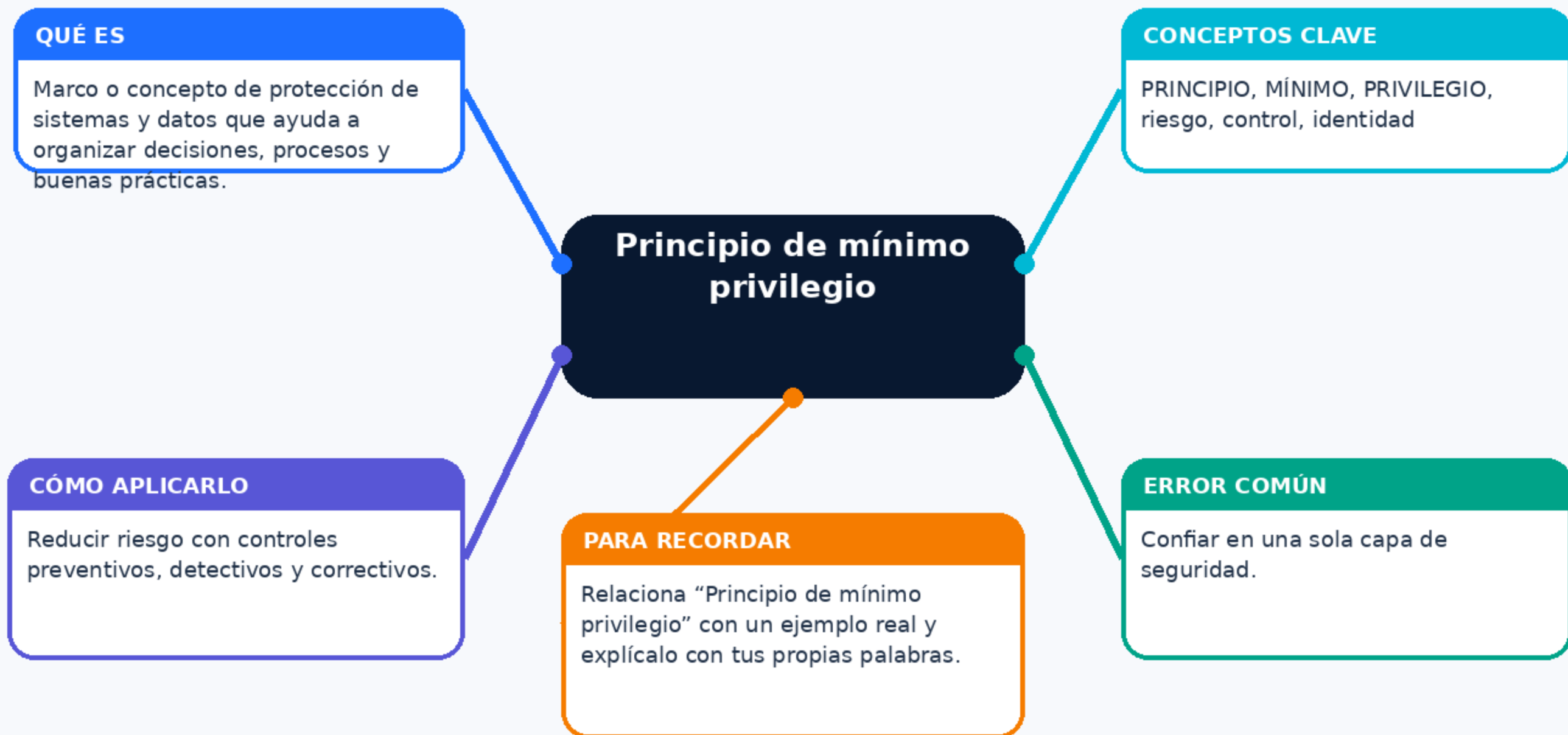
PKI y certificados digitales



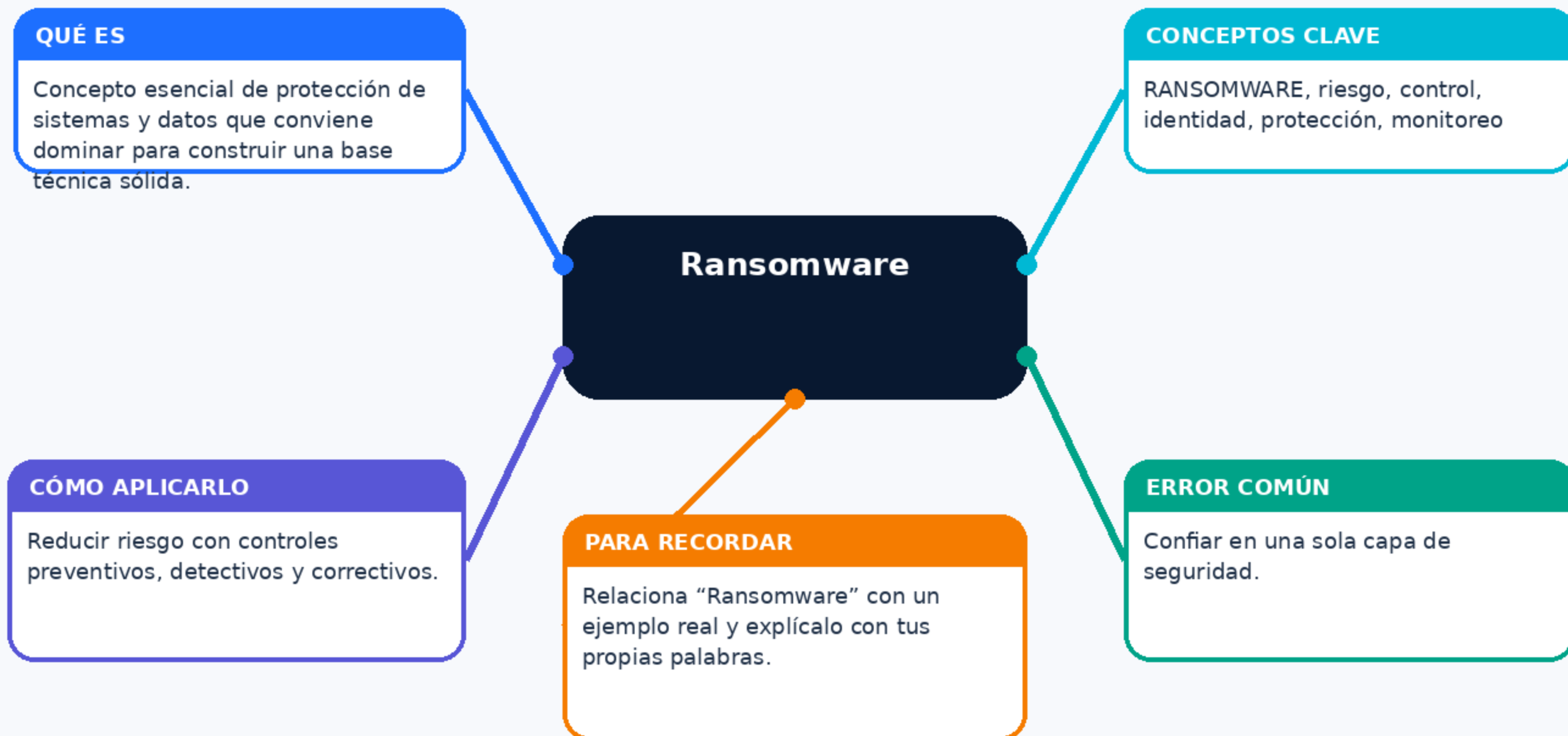
Phishing



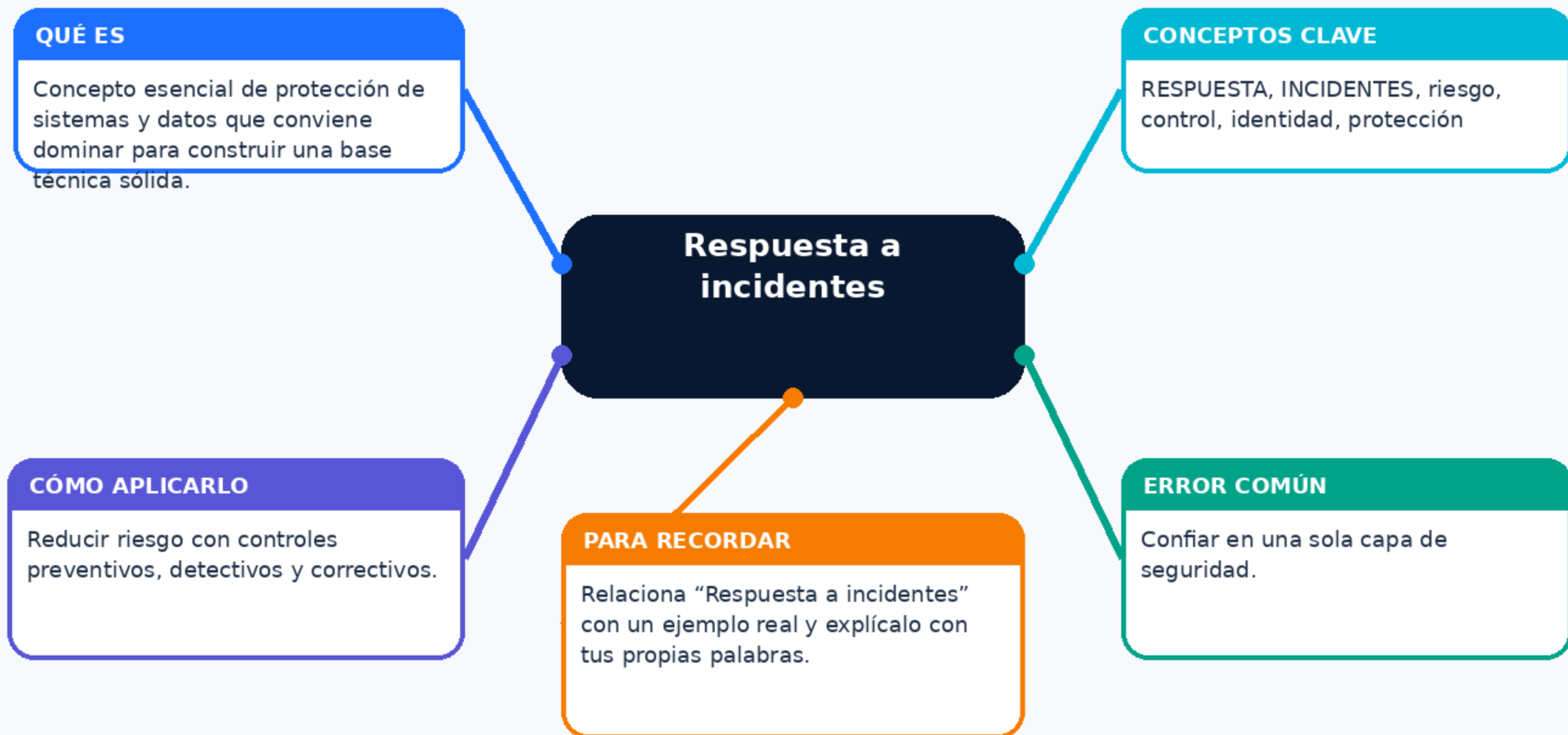
Principio de mínimo privilegio



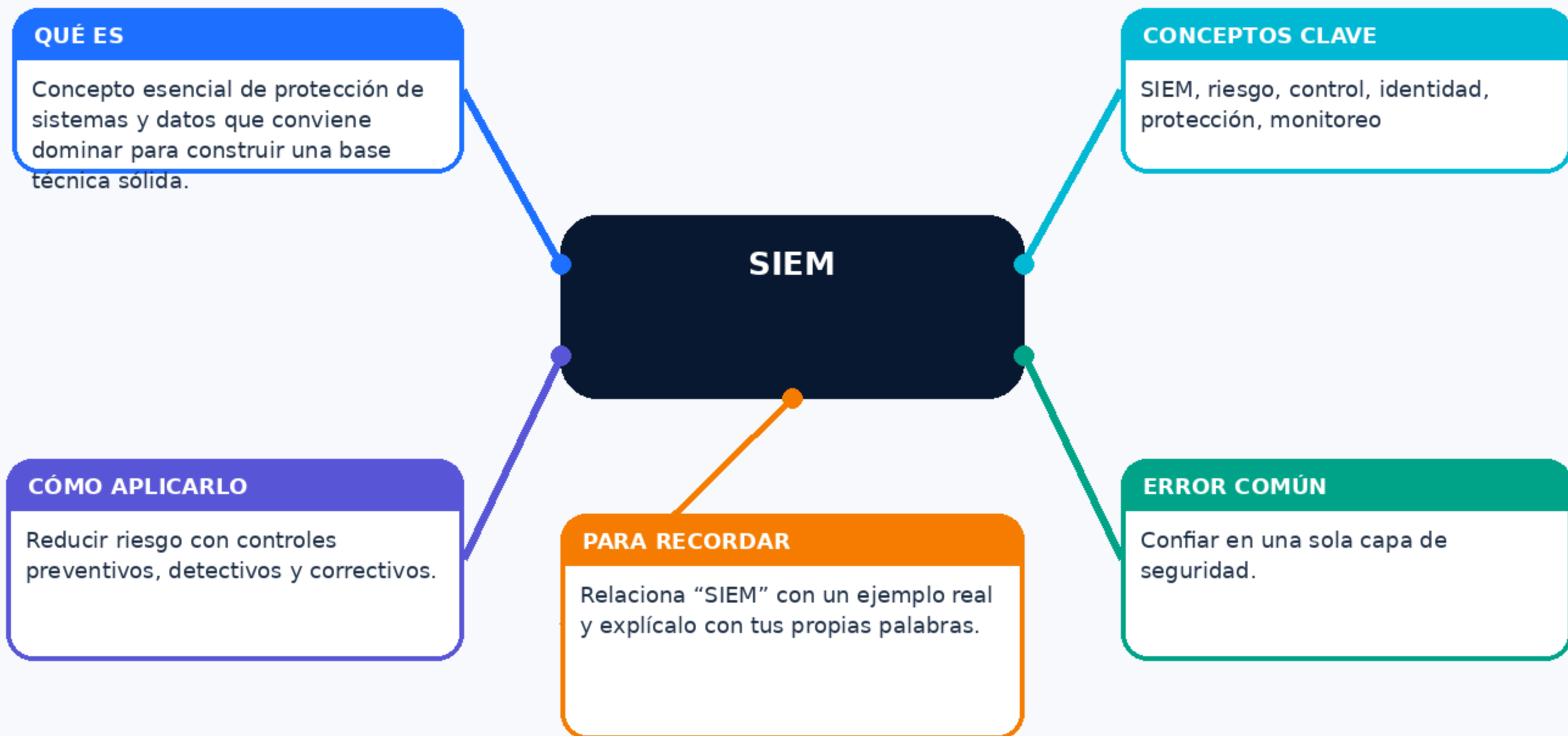
Ransomware



Respuesta a incidentes



SIEM



SQL Injection

QUÉ ES

Ataque que intenta alterar consultas SQL mediante entradas no tratadas correctamente.

CONCEPTOS CLAVE

consultas parametrizadas, validación, mínimo privilegio, ORM

SQL Injection

CÓMO APLICARLO

Reducir riesgo con controles preventivos, detectivos y correctivos.

PARA RECORDAR

Relaciona "SQL Injection" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confiar en una sola capa de seguridad.

Seguridad en Wi-Fi



TLS

QUÉ ES

Concepto esencial de protección de sistemas y datos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

TLS, riesgo, control, identidad, protección, monitoreo

TLS

CÓMO APLICARLO

Reducir riesgo con controles preventivos, detectivos y correctivos.

PARA RECORDAR

Relaciona "TLS" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confiar en una sola capa de seguridad.

XSS

QUÉ ES

Vulnerabilidad que permite ejecutar scripts no confiables en el navegador de otra persona.

CONCEPTOS CLAVE

stored, reflected, DOM, escape, CSP

XSS

CÓMO APLICARLO

Reducir riesgo con controles preventivos, detectivos y correctivos.

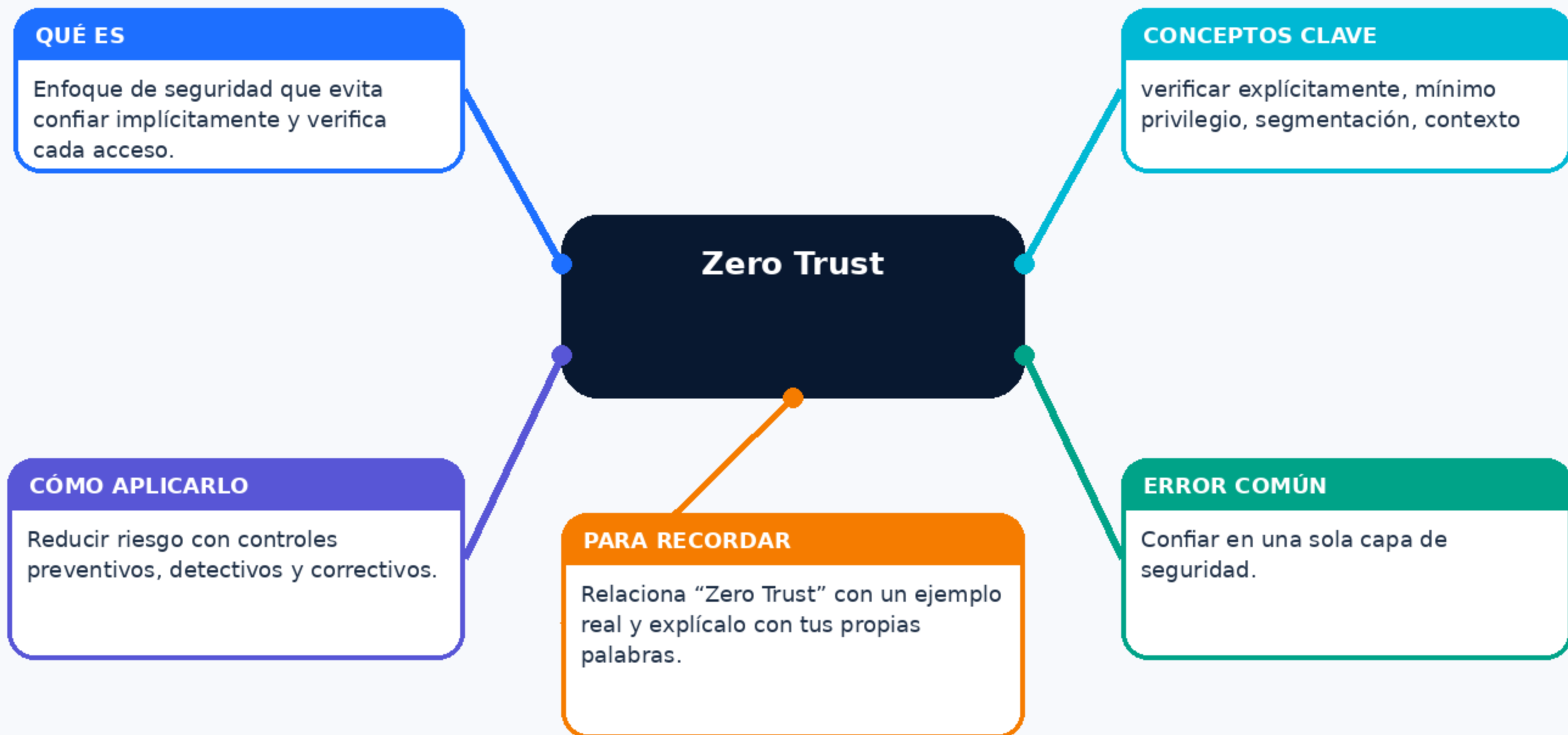
PARA RECORDAR

Relaciona "XSS" con un ejemplo real y explícalo con tus propias palabras.

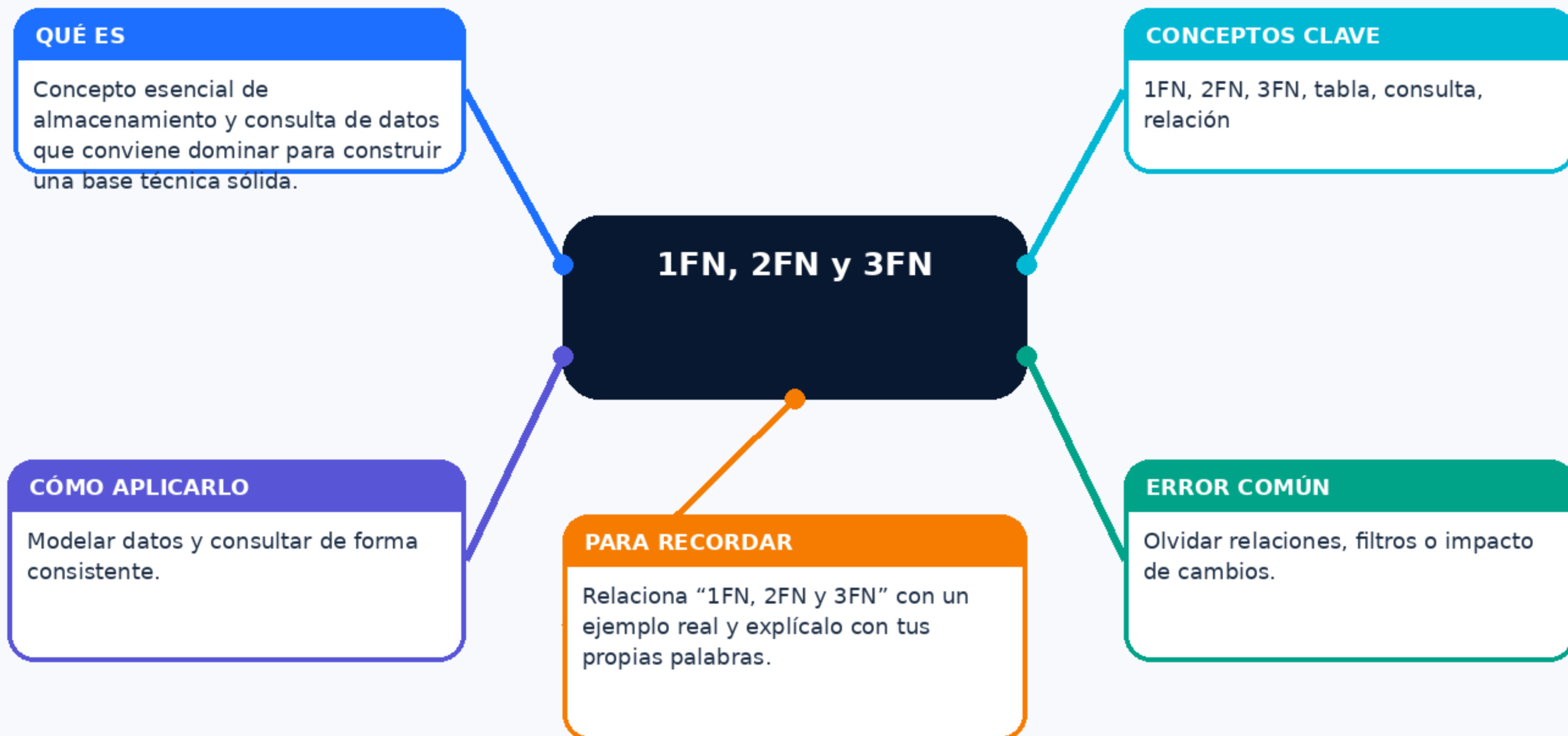
ERROR COMÚN

Confiar en una sola capa de seguridad.

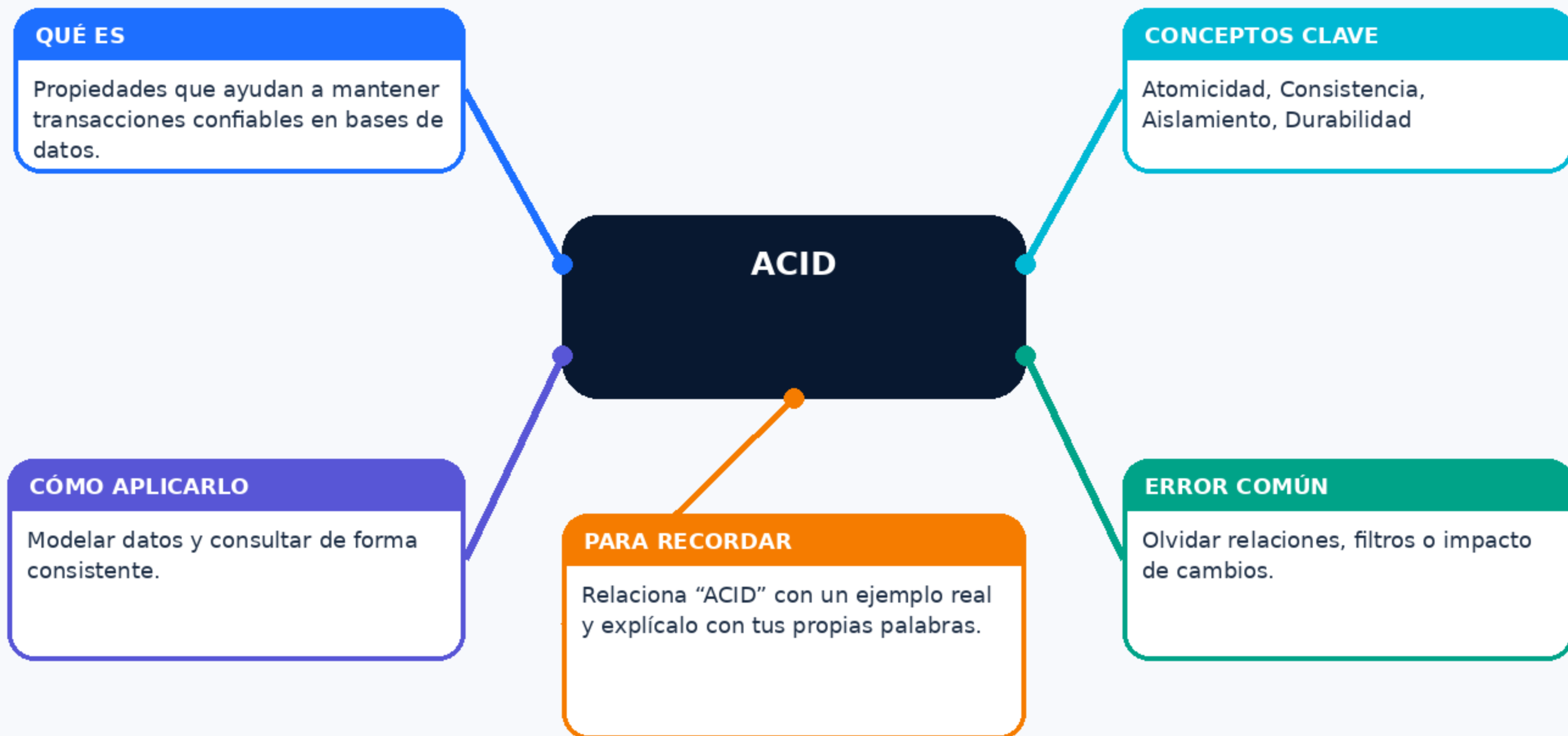
Zero Trust



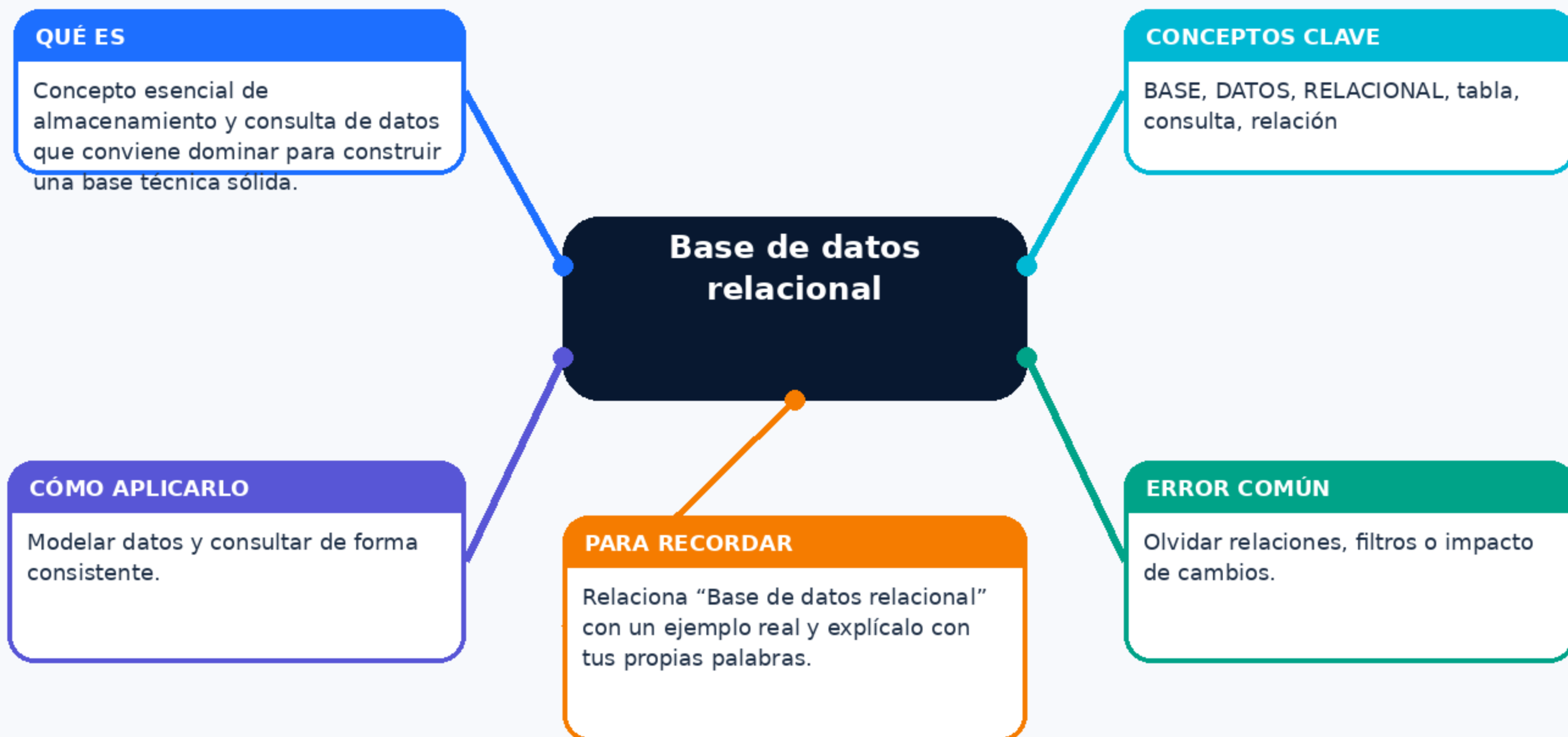
1FN, 2FN y 3FN



ACID



Base de datos relacional



Clave foránea

QUÉ ES

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

CLAVE, FORÁNEA, tabla, consulta, relación, transacción

Clave foránea

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

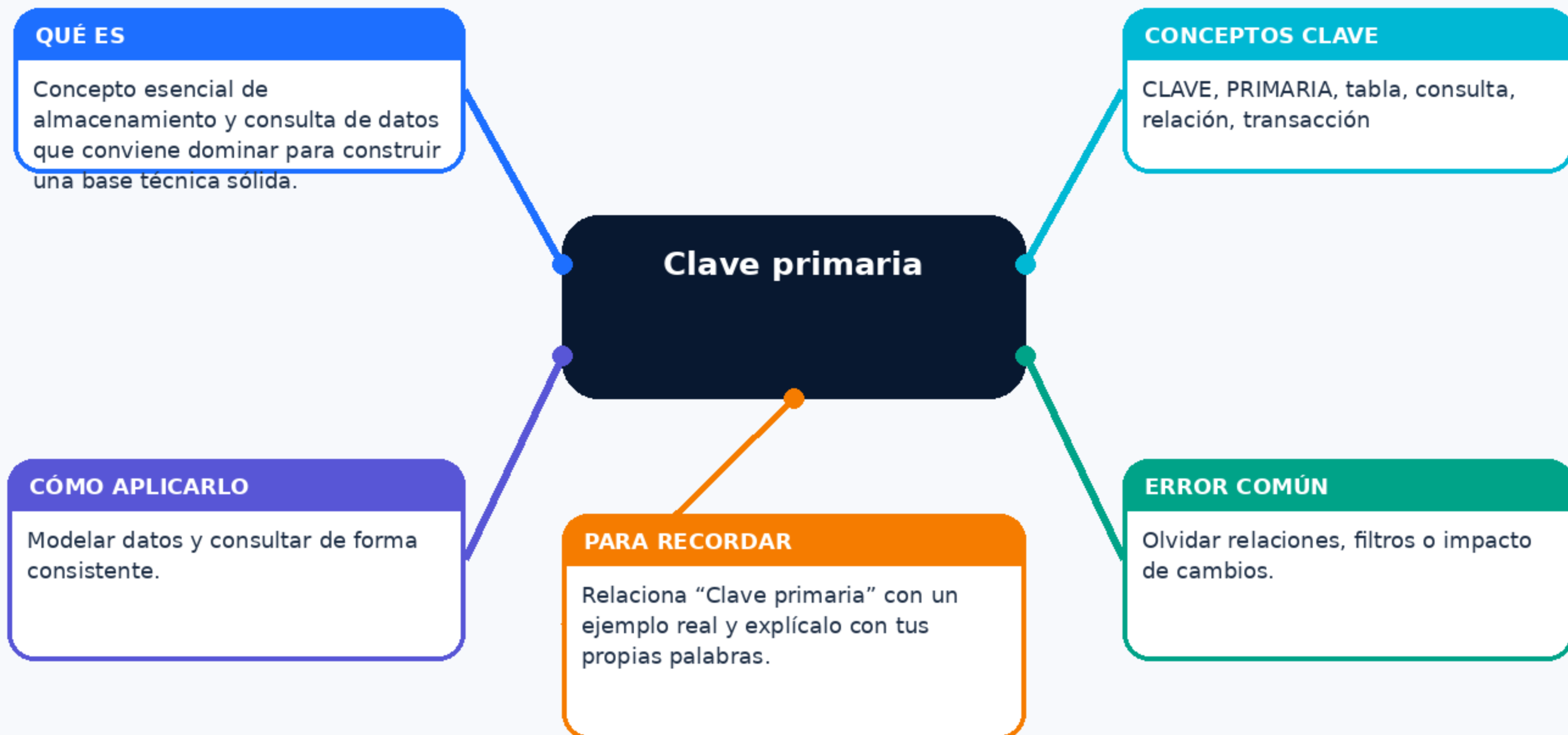
PARA RECORDAR

Relaciona "Clave foránea" con un ejemplo real y explícalo con tus propias palabras.

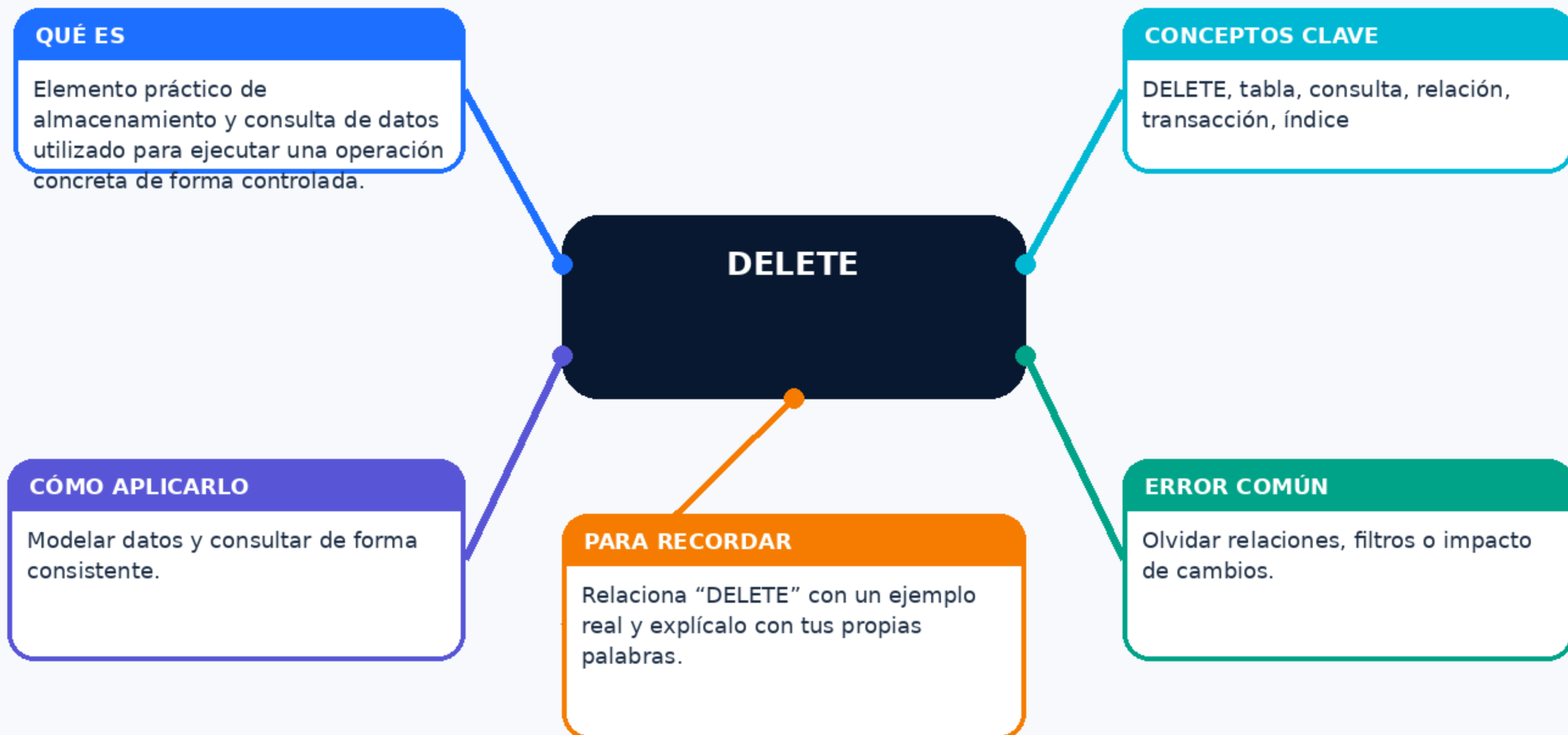
ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

Clave primaria



DELETE



Funciones agregadas



GROUP BY

QUÉ ES

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

GROUP, tabla, consulta, relación, transacción, índice

GROUP BY

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

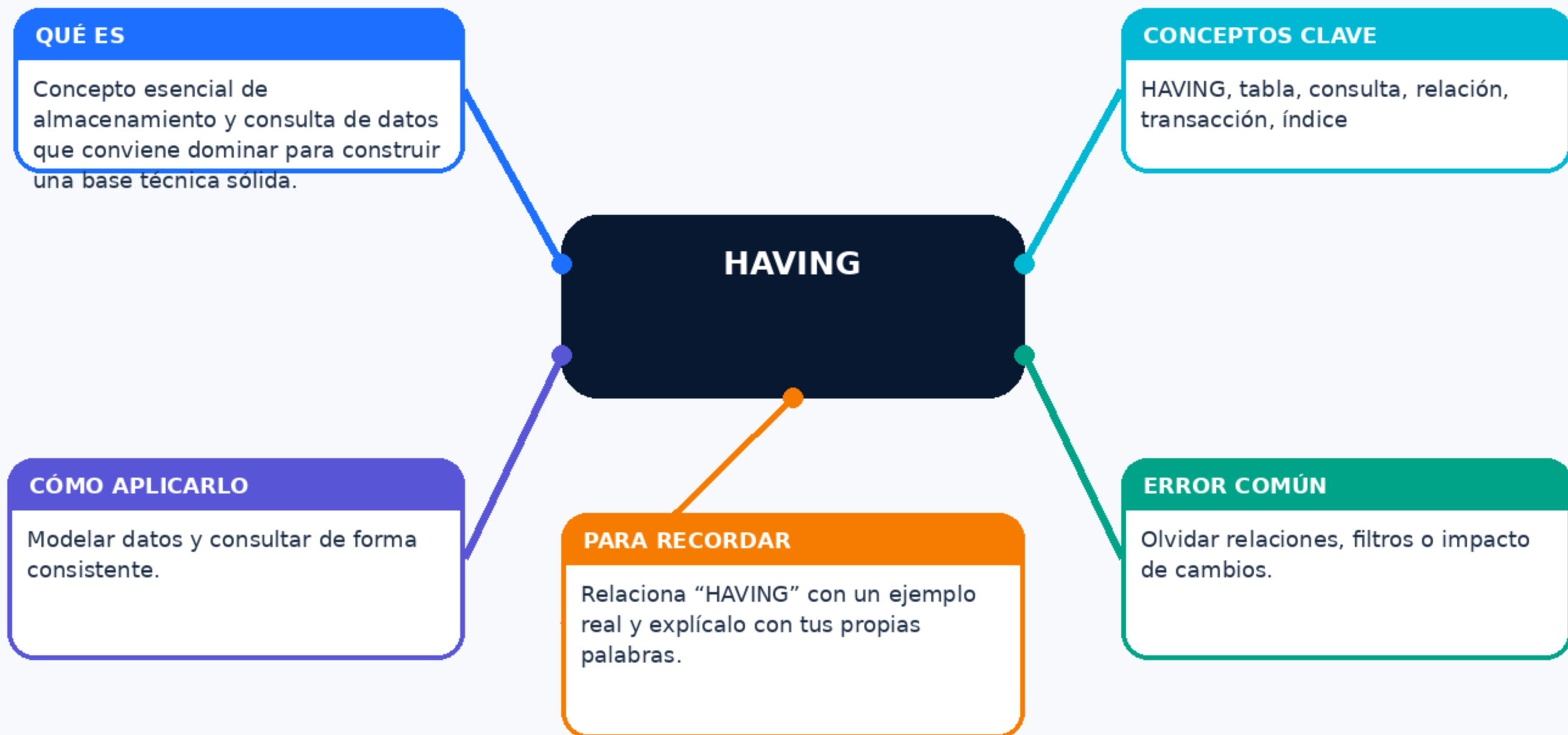
PARA RECORDAR

Relaciona "GROUP BY" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

HAVING



INNER JOIN

QUÉ ES

Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.

CONCEPTOS CLAVE

INNER, JOIN, tabla, consulta, relación, transacción

INNER JOIN

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

PARA RECORDAR

Relaciona "INNER JOIN" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

INSERT

QUÉ ES

Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.

CONCEPTOS CLAVE

INSERT, tabla, consulta, relación, transacción, índice

INSERT

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

PARA RECORDAR

Relaciona "INSERT" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

JOIN

QUÉ ES

Operación SQL para combinar filas relacionadas entre tablas.

CONCEPTOS CLAVE

INNER, LEFT, RIGHT, FULL, condición ON

JOIN

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

PARA RECORDAR

Relaciona "JOIN" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

LEFT JOIN

QUÉ ES

Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.

CONCEPTOS CLAVE

LEFT, JOIN, tabla, consulta, relación, transacción

LEFT JOIN

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

PARA RECORDAR

Relaciona "LEFT JOIN" con un ejemplo real y explícalo con tus propias palabras.

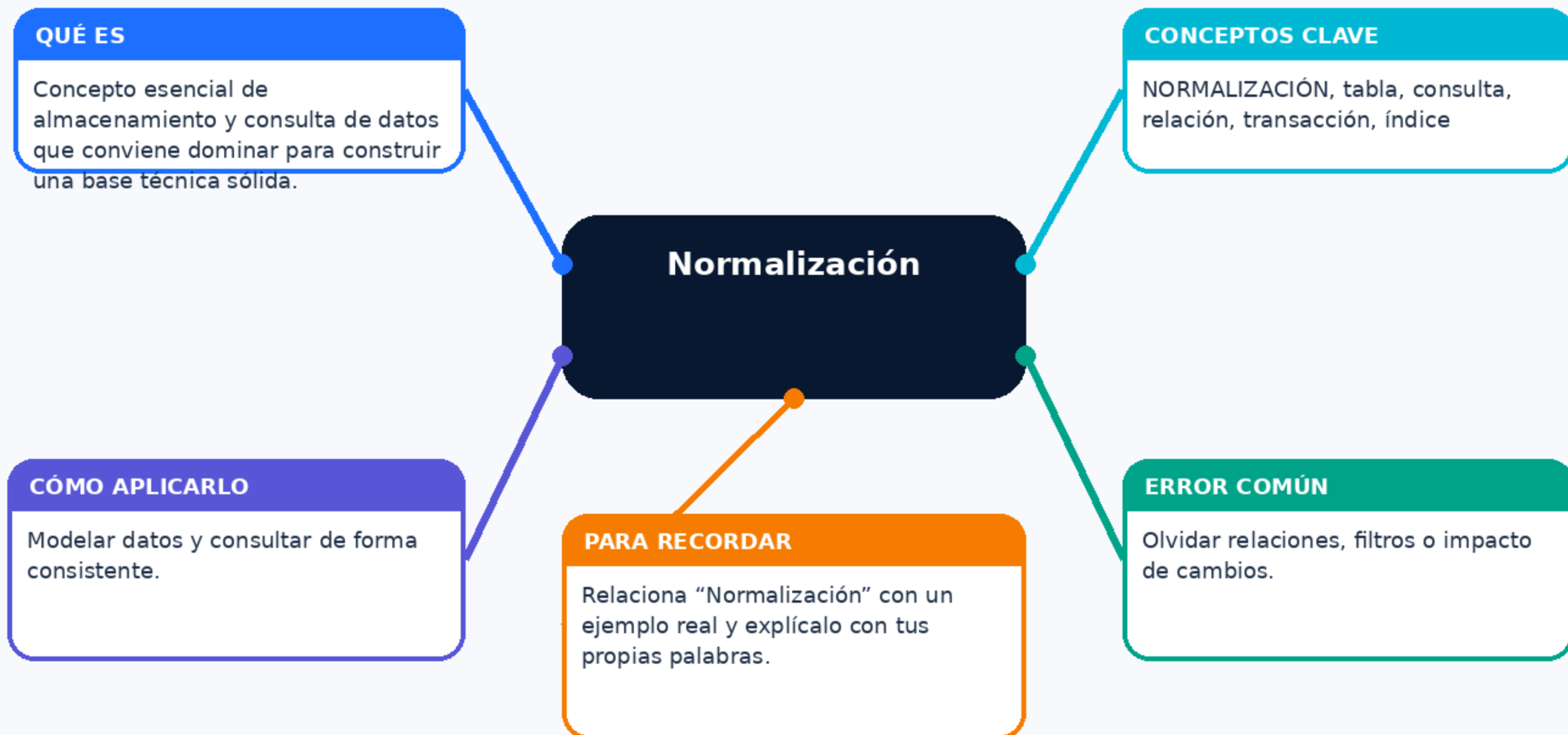
ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

NoSQL



Normalización



ORDER BY

QUÉ ES

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ORDER, tabla, consulta, relación, transacción, índice

ORDER BY

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

PARA RECORDAR

Relaciona "ORDER BY" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

Procedimientos almacenados



SELECT

QUÉ ES

Sentencia SQL usada para consultar datos de una o más tablas.

CONCEPTOS CLAVE

columnas, FROM, WHERE, JOIN, ORDER BY

SELECT

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

PARA RECORDAR

Relaciona "SELECT" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

Subconsultas

QUÉ ES

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

SUBCONSULTAS, tabla, consulta, relación, transacción, índice

Subconsultas

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

PARA RECORDAR

Relaciona "Subconsultas" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

Tabla, fila y columna

QUÉ ES

Concepto esencial de almacenamiento y consulta de datos que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

TABLA, FILA, COLUMNA, tabla, consulta, relación

Tabla, fila y columna

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

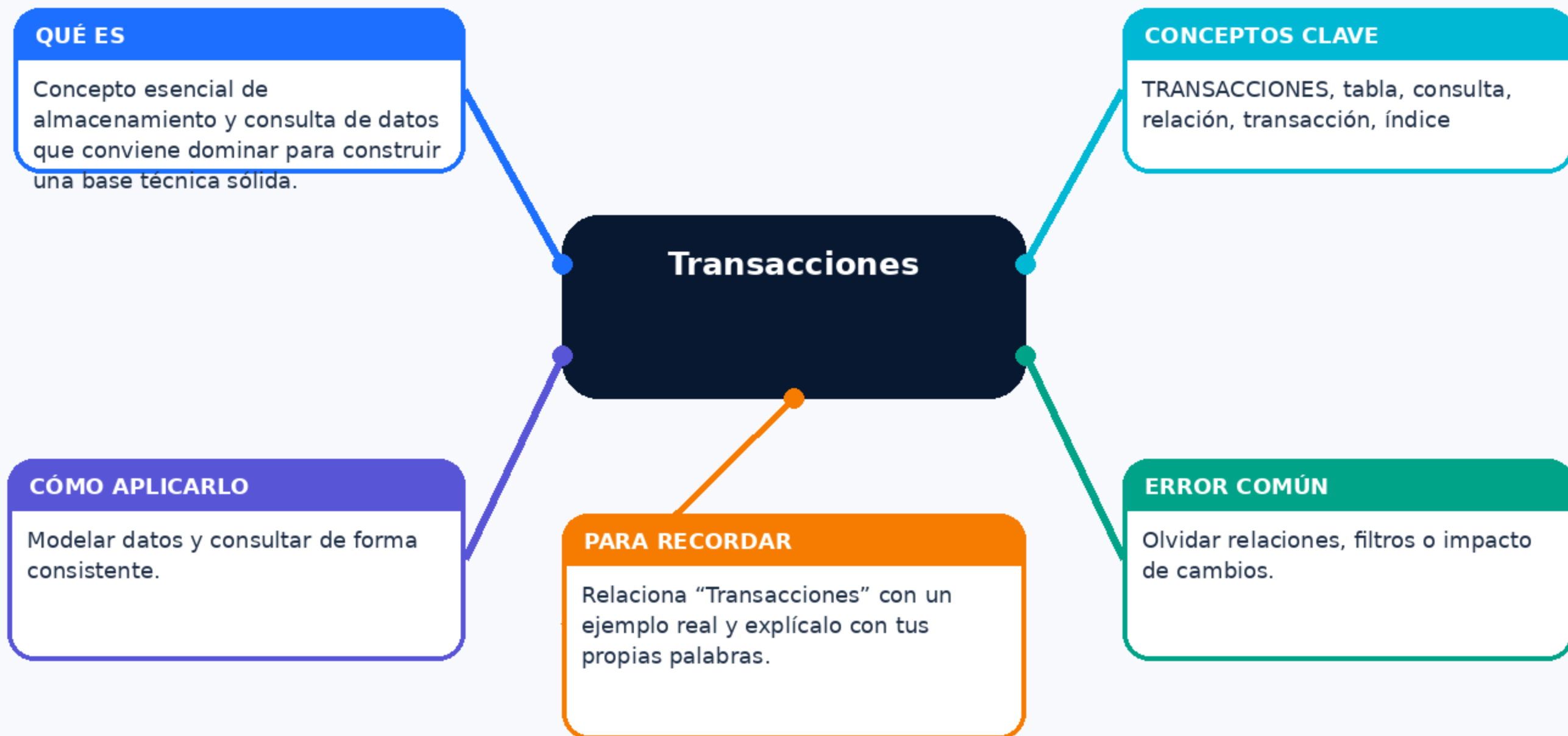
PARA RECORDAR

Relaciona "Tabla, fila y columna" con un ejemplo real y explícalo con tus propias palabras.

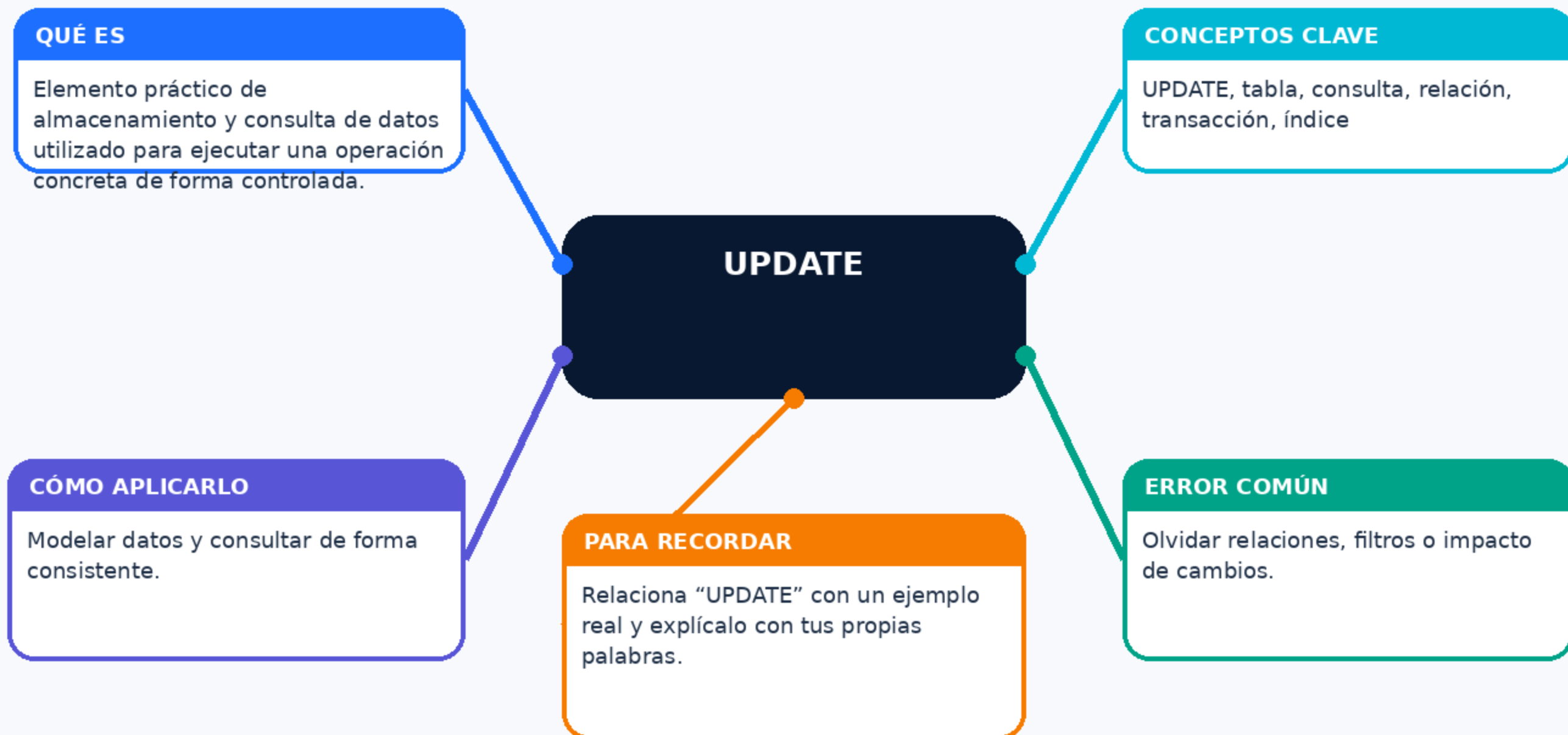
ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

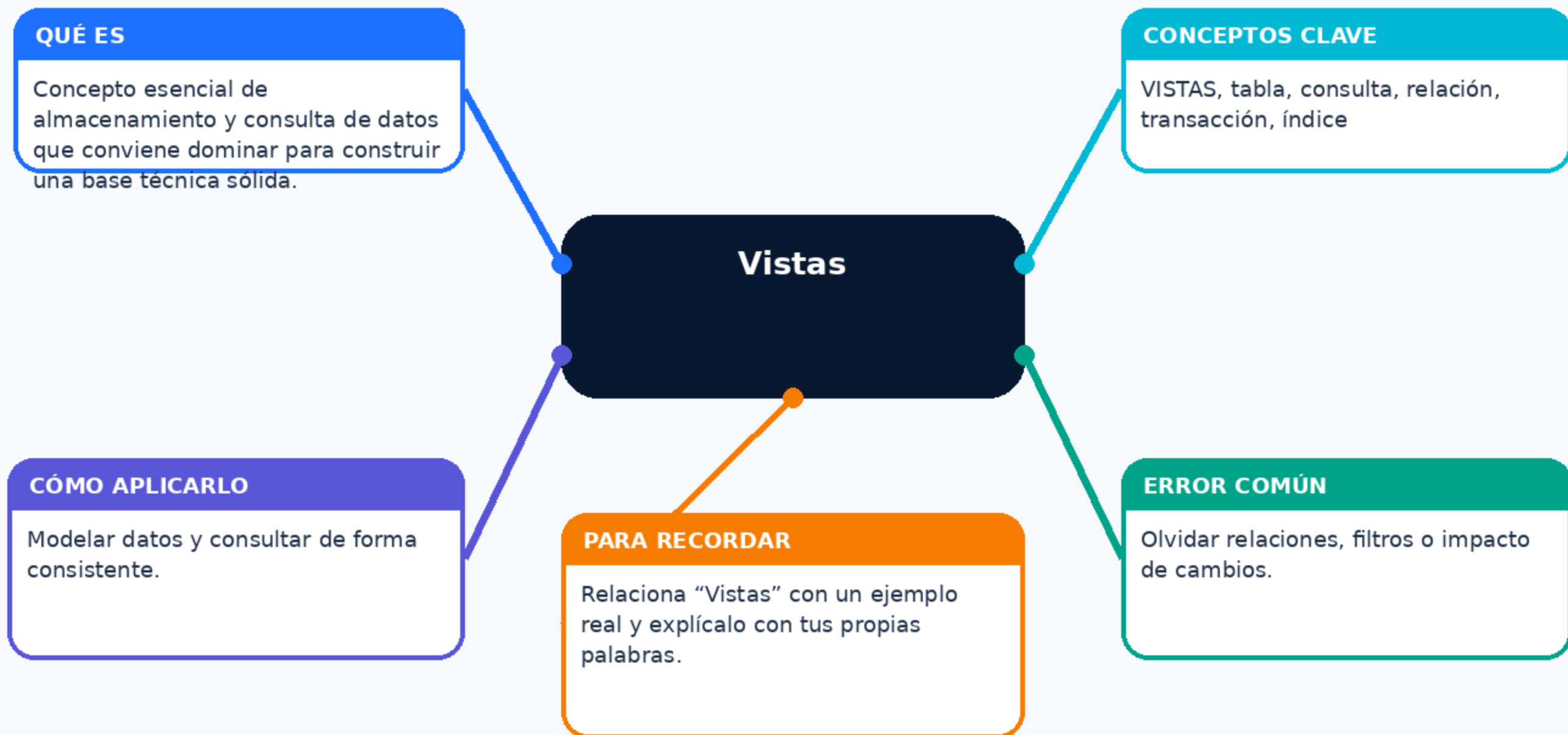
Transacciones



UPDATE



Vistas



WHERE

QUÉ ES

Elemento práctico de almacenamiento y consulta de datos utilizado para ejecutar una operación concreta de forma controlada.

CONCEPTOS CLAVE

WHERE, tabla, consulta, relación, transacción, índice

WHERE

CÓMO APLICARLO

Modelar datos y consultar de forma consistente.

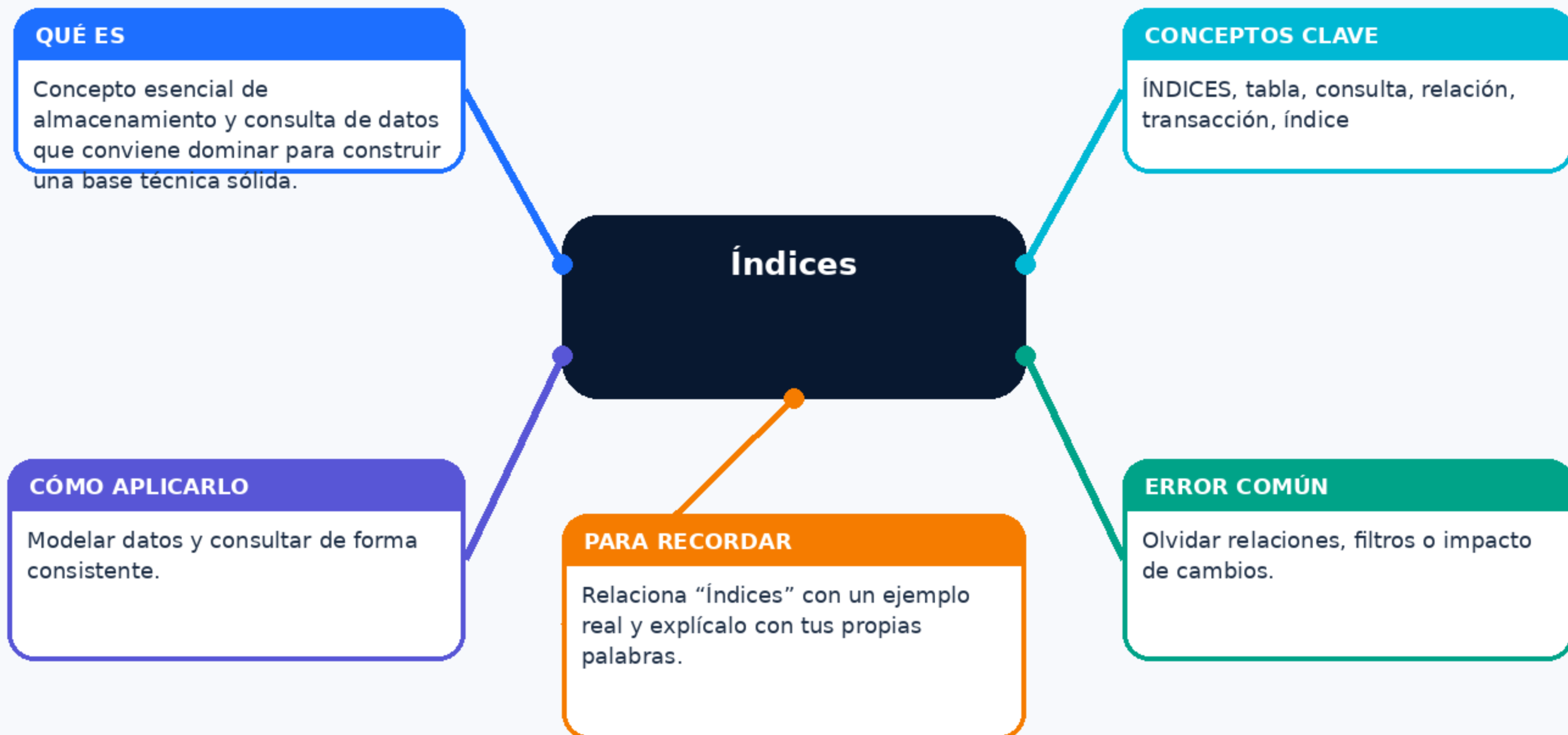
PARA RECORDAR

Relaciona "WHERE" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Olvidar relaciones, filtros o impacto de cambios.

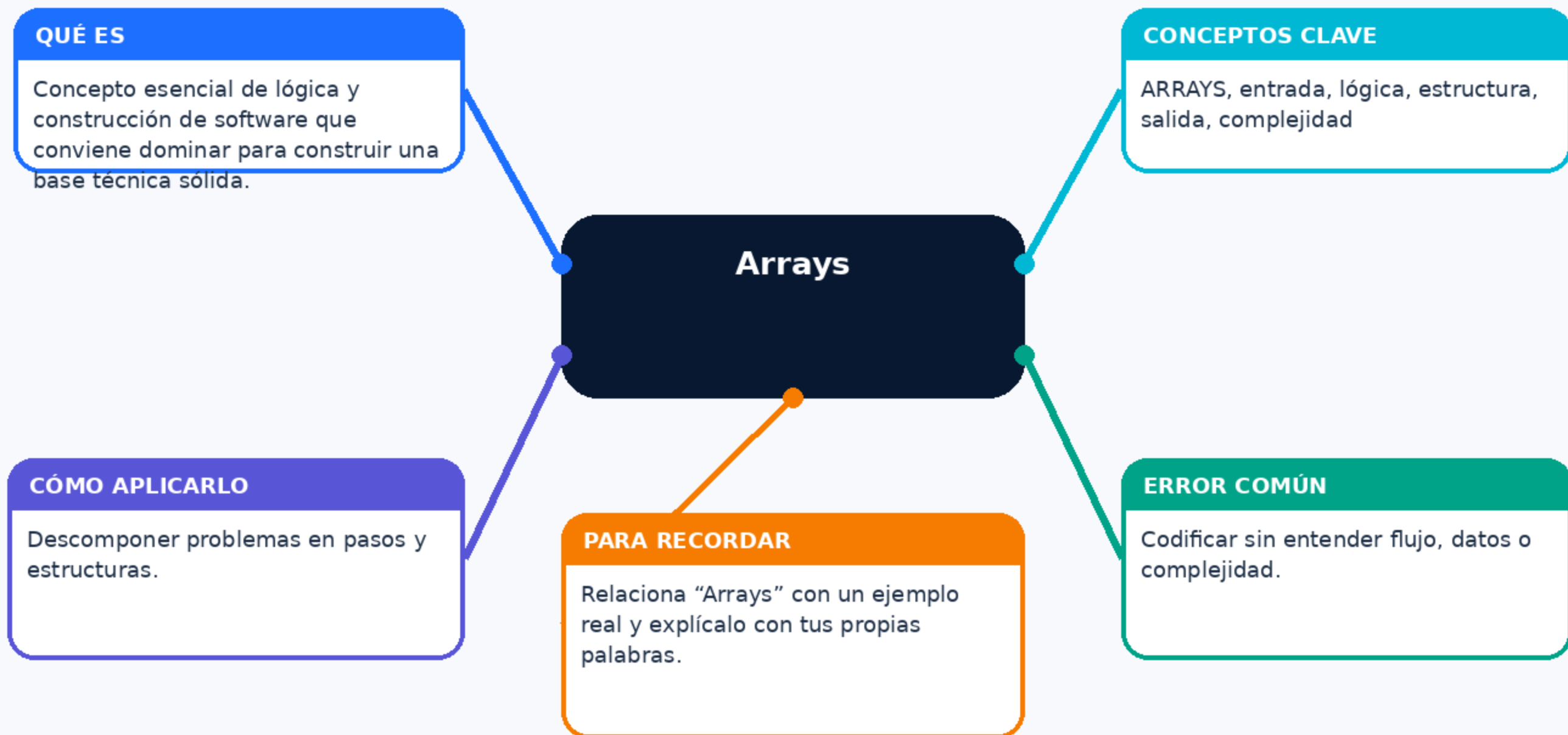
Índices



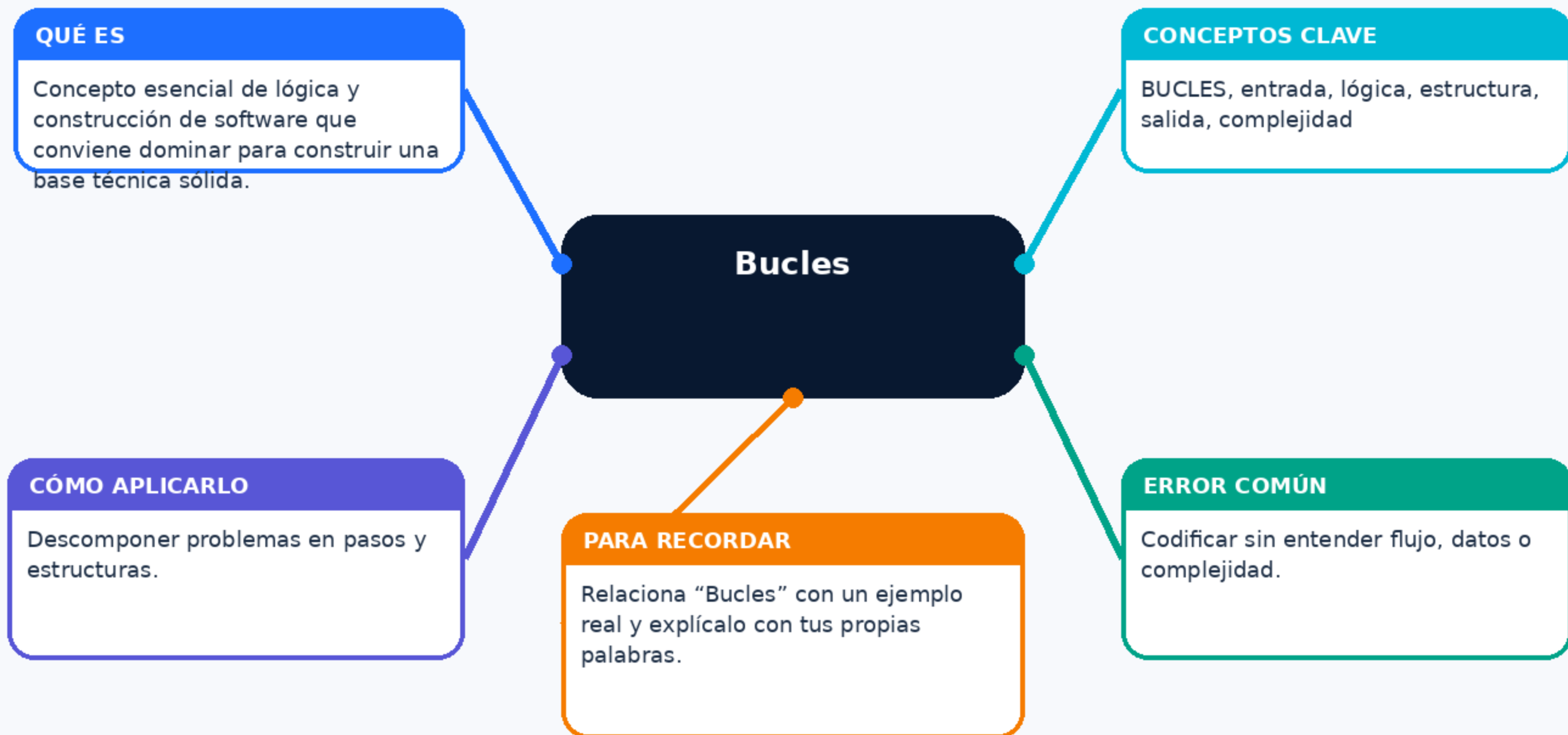
Algoritmo



Arrays



Bucles



Búsqueda binaria

QUÉ ES

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

BÚSQUEDA, BINARIA, entrada, lógica, estructura, salida

Búsqueda binaria

CÓMO APLICARLO

Descomponer problemas en pasos y estructuras.

PARA RECORDAR

Relaciona "Búsqueda binaria" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Codificar sin entender flujo, datos o complejidad.

Búsqueda lineal

QUÉ ES

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

BÚSQUEDA, LINEAL, entrada, lógica, estructura, salida

Búsqueda lineal

CÓMO APLICARLO

Descomponer problemas en pasos y estructuras.

PARA RECORDAR

Relaciona "Búsqueda lineal" con un ejemplo real y explícalo con tus propias palabras.

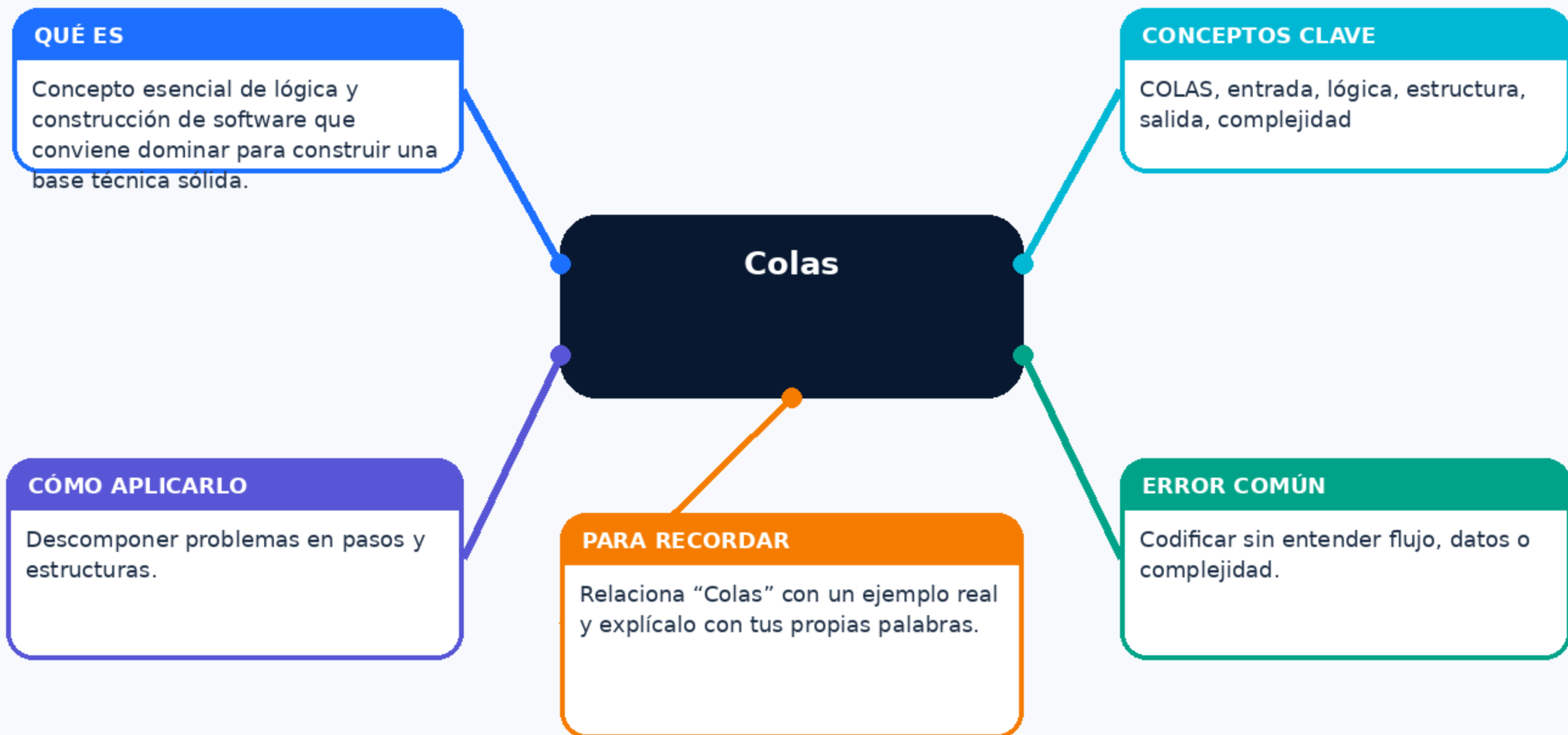
ERROR COMÚN

Codificar sin entender flujo, datos o complejidad.

Clases y objetos



Colas



Complejidad Big O

QUÉ ES

Notación que describe cómo crece el costo de un algoritmo según el tamaño de entrada.

CONCEPTOS CLAVE

$O(1)$, $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n^2)$

Complejidad Big O

CÓMO APLICARLO

Descomponer problemas en pasos y estructuras.

PARA RECORDAR

Relaciona "Complejidad Big O" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Codificar sin entender flujo, datos o complejidad.

Condicionales

QUÉ ES

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

CONDICIONALES, entrada, lógica, estructura, salida, complejidad

Condicionales

CÓMO APLICARLO

Descomponer problemas en pasos y estructuras.

PARA RECORDAR

Relaciona "Condicionales" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Codificar sin entender flujo, datos o complejidad.

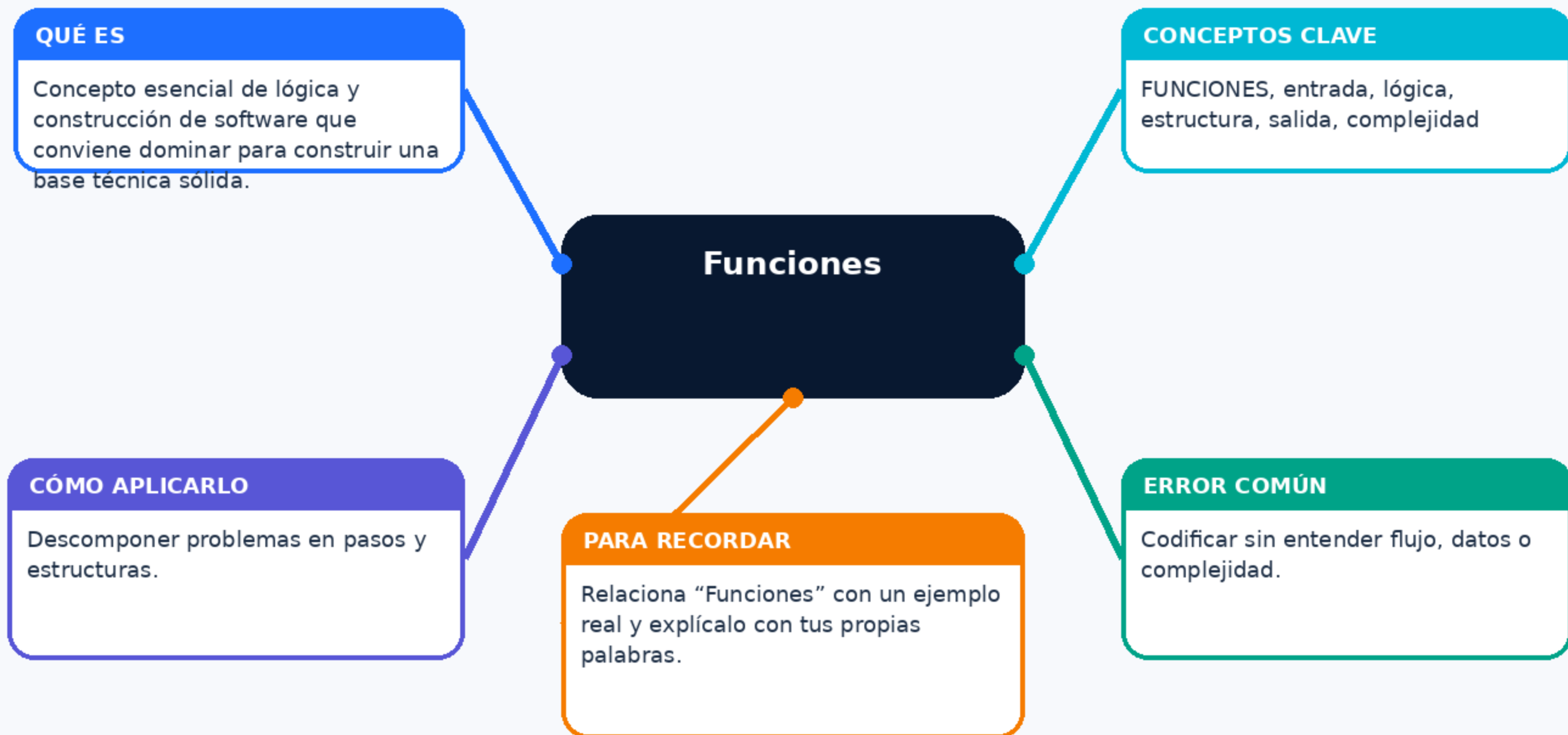
Conjuntos



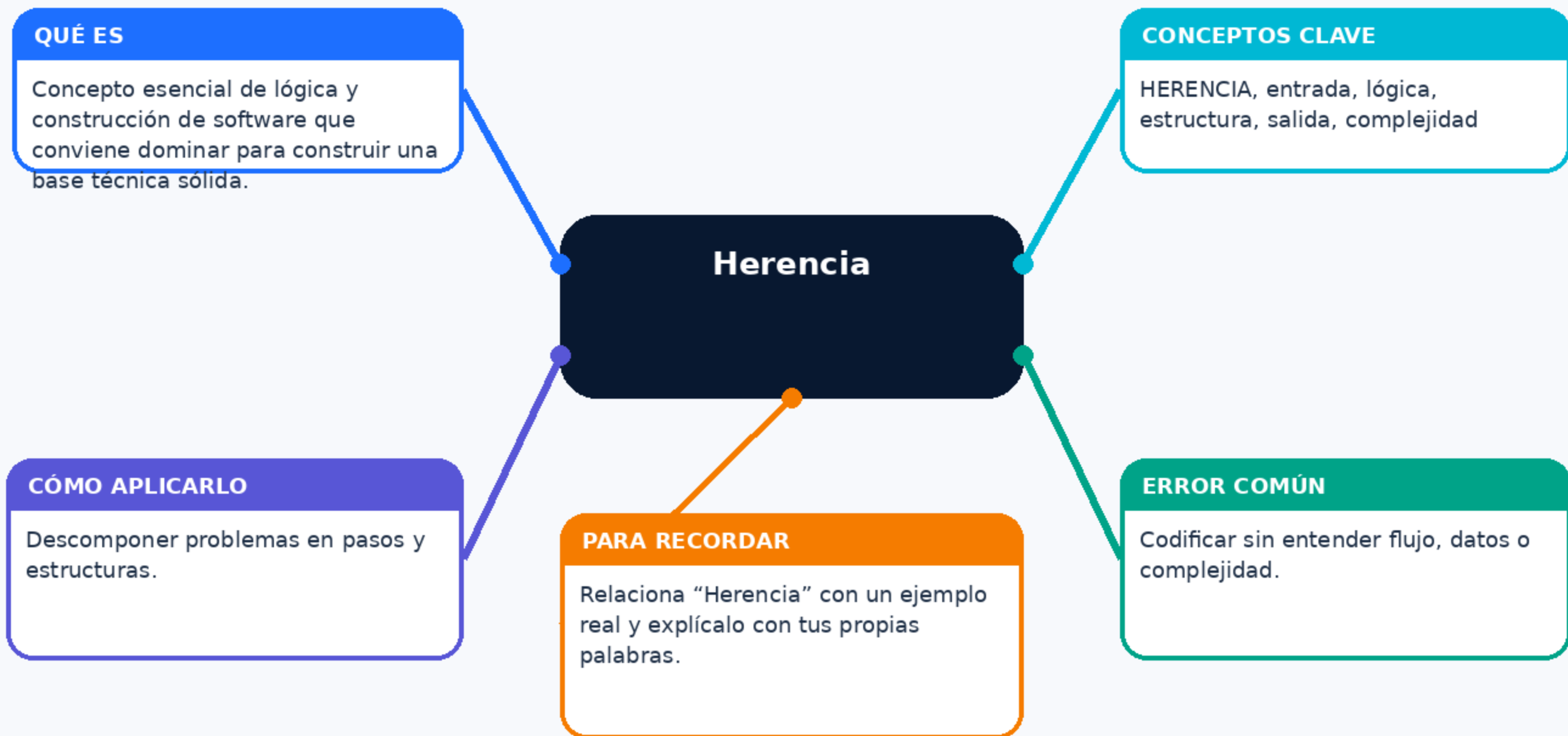
Diccionarios y mapas



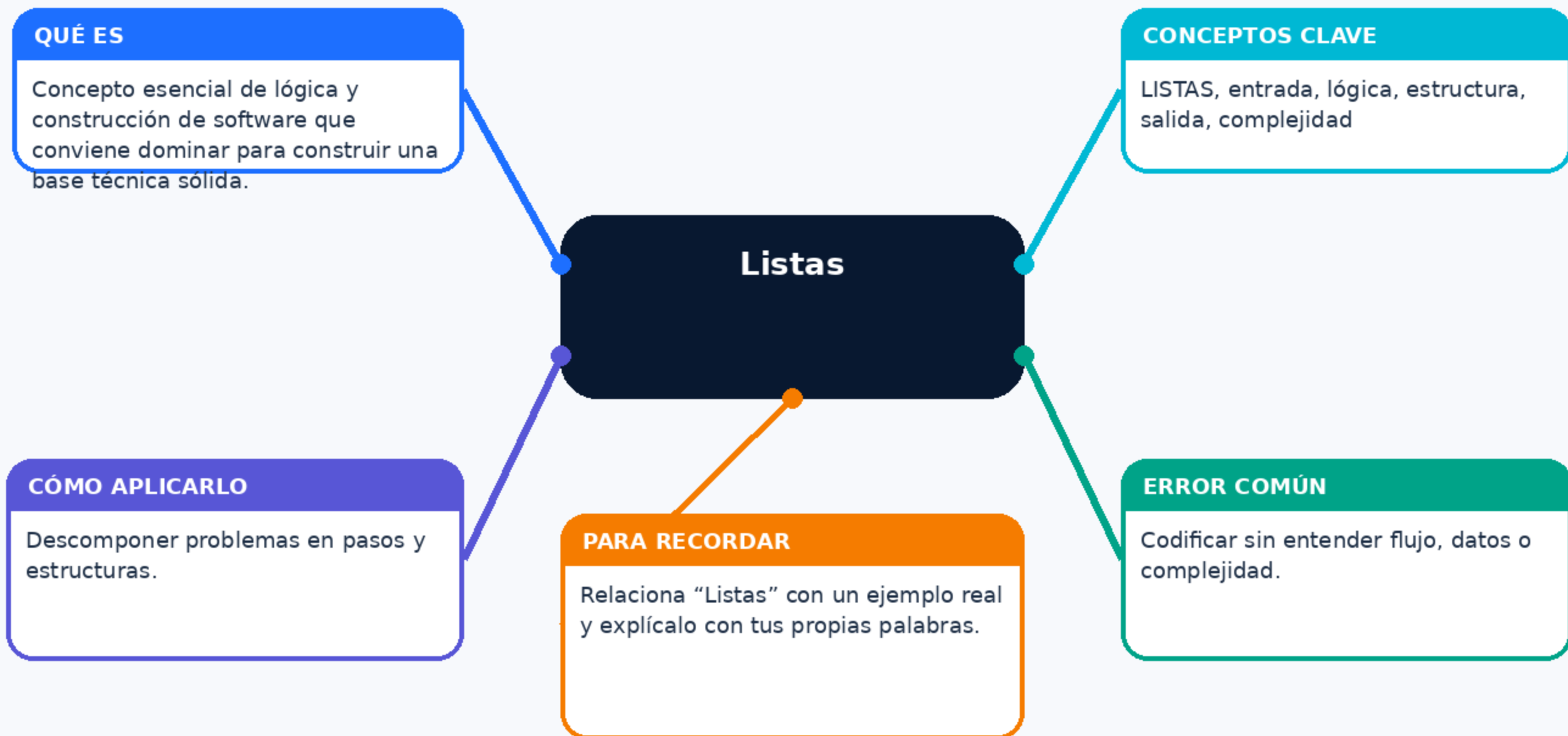
Funciones



Herencia



Listas



Manejo de excepciones



Operadores

QUÉ ES

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

OPERADORES, entrada, lógica, estructura, salida, complejidad

Operadores

CÓMO APLICARLO

Descomponer problemas en pasos y estructuras.

PARA RECORDAR

Relaciona "Operadores" con un ejemplo real y explícalo con tus propias palabras.

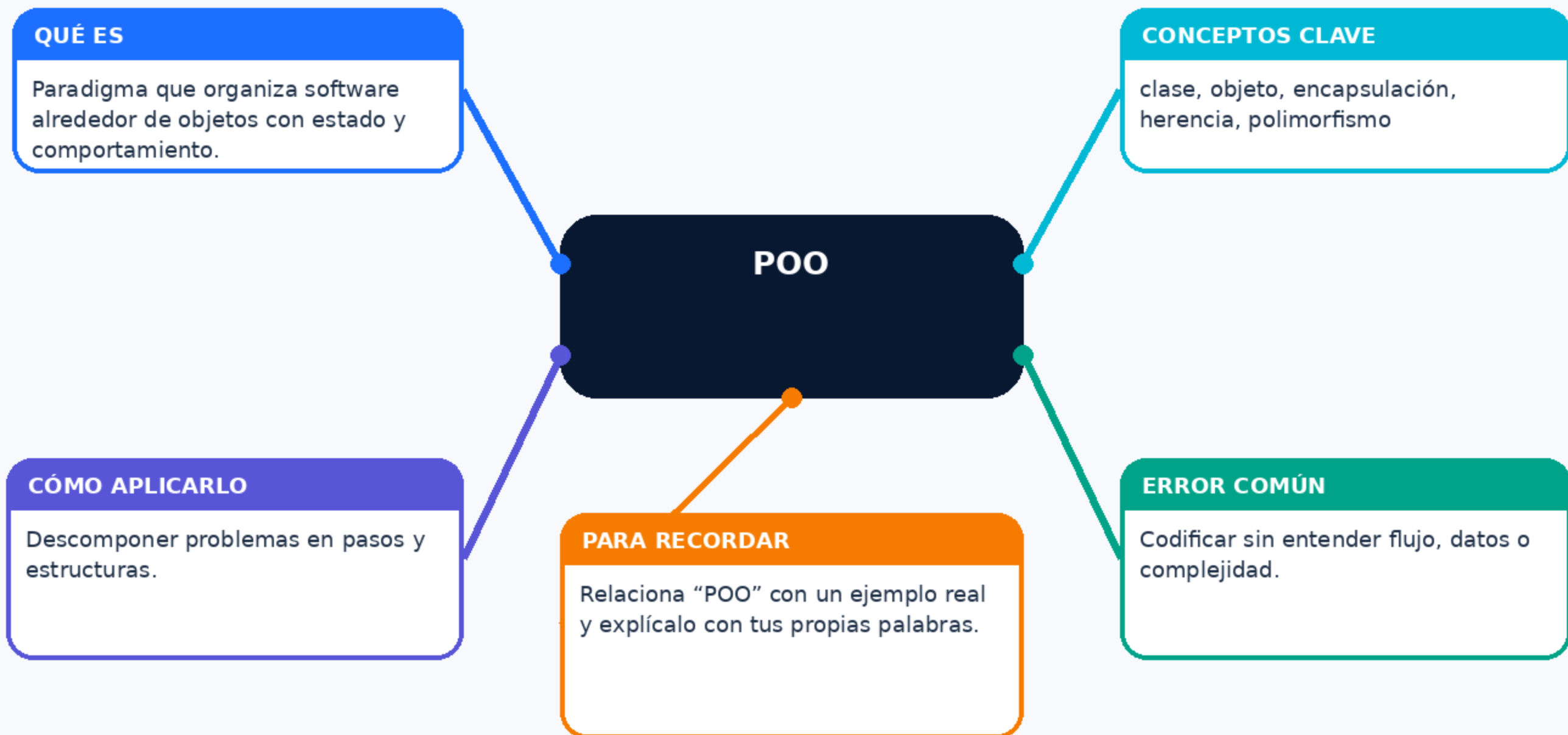
ERROR COMÚN

Codificar sin entender flujo, datos o complejidad.

Ordenamiento



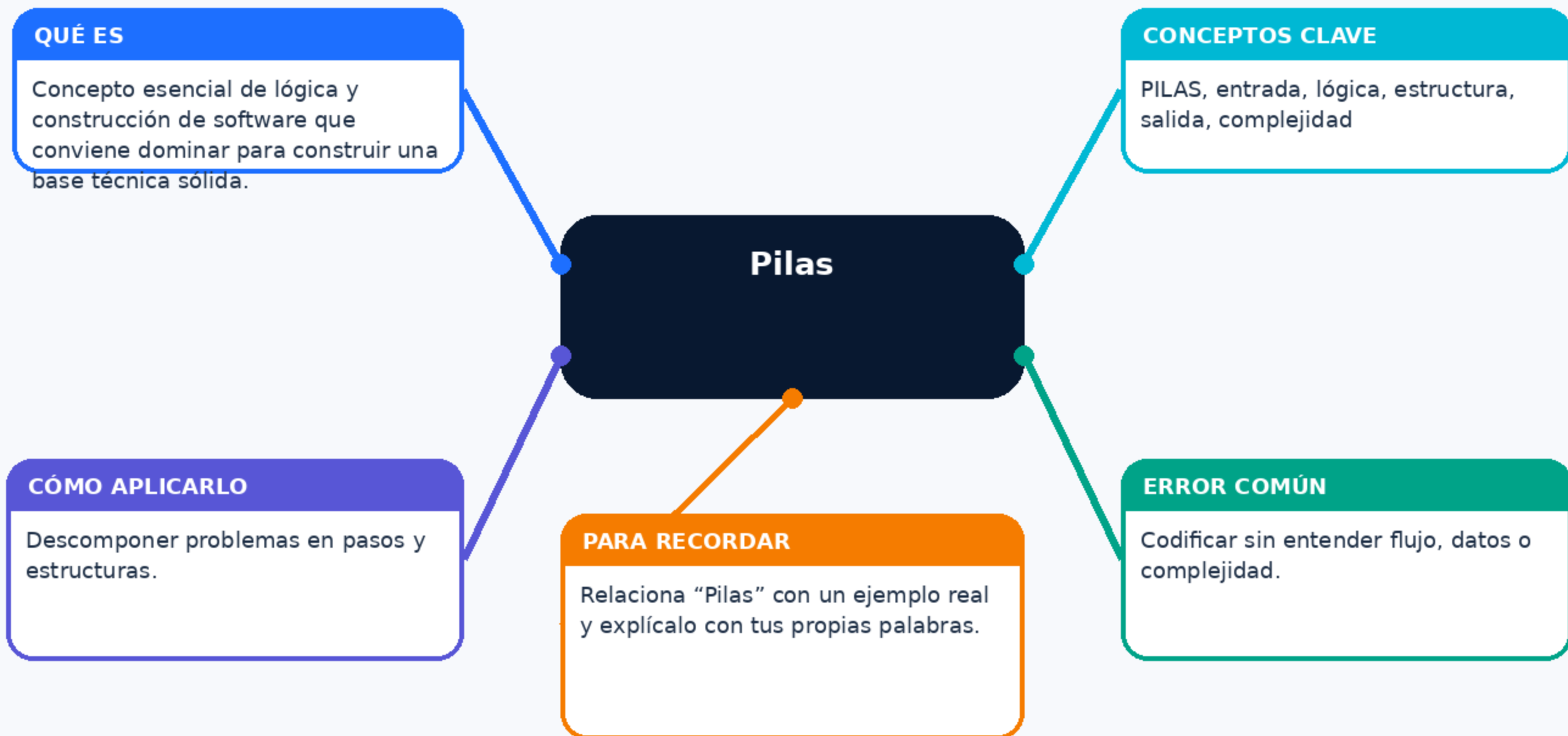
POO



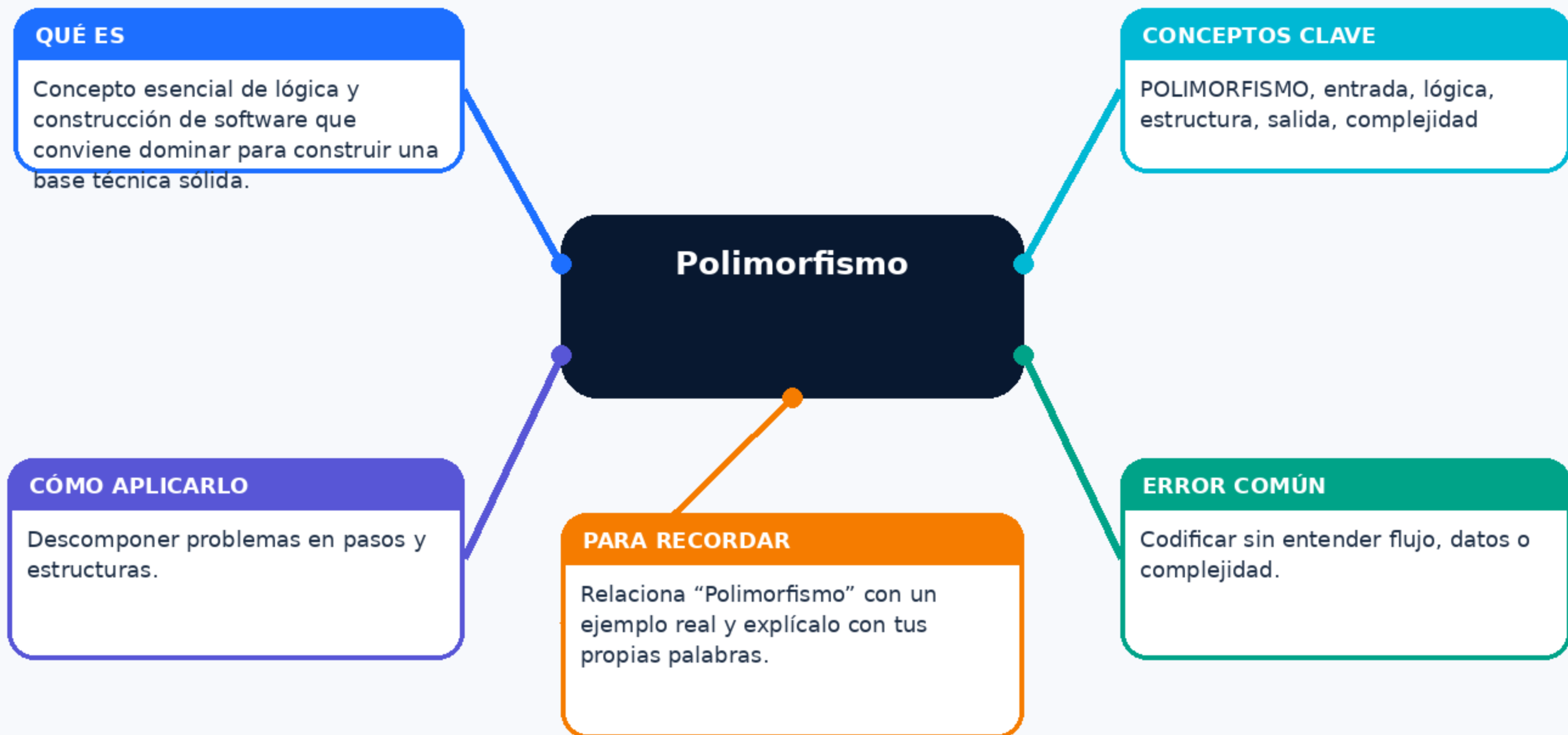
Parámetros y retorno



Pilas



Polimorfismo



Pseudocódigo

QUÉ ES

Concepto esencial de lógica y construcción de software que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

PSEUDOCÓDIGO, entrada, lógica, estructura, salida, complejidad

Pseudocódigo

CÓMO APLICARLO

Descomponer problemas en pasos y estructuras.

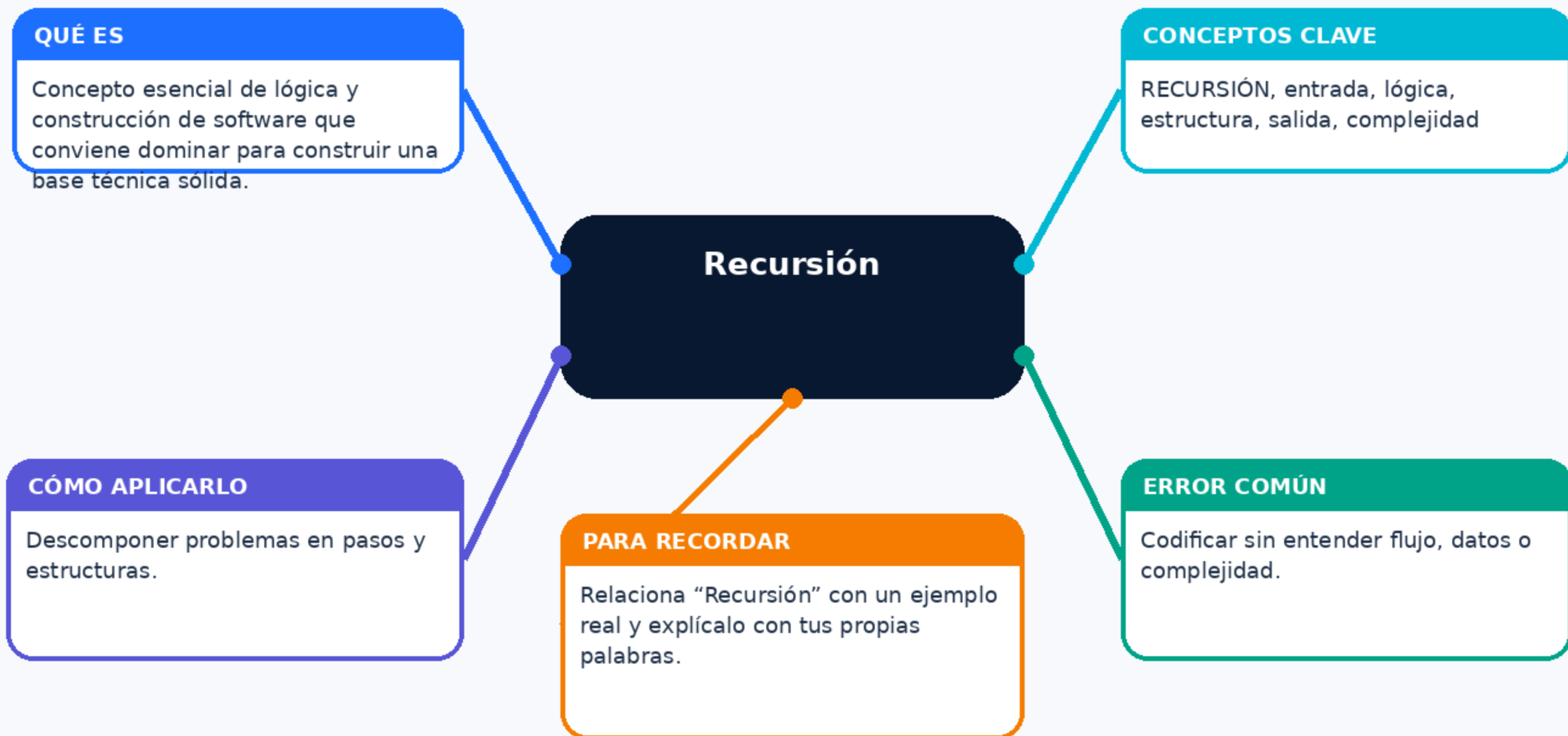
PARA RECORDAR

Relaciona "Pseudocódigo" con un ejemplo real y explícalo con tus propias palabras.

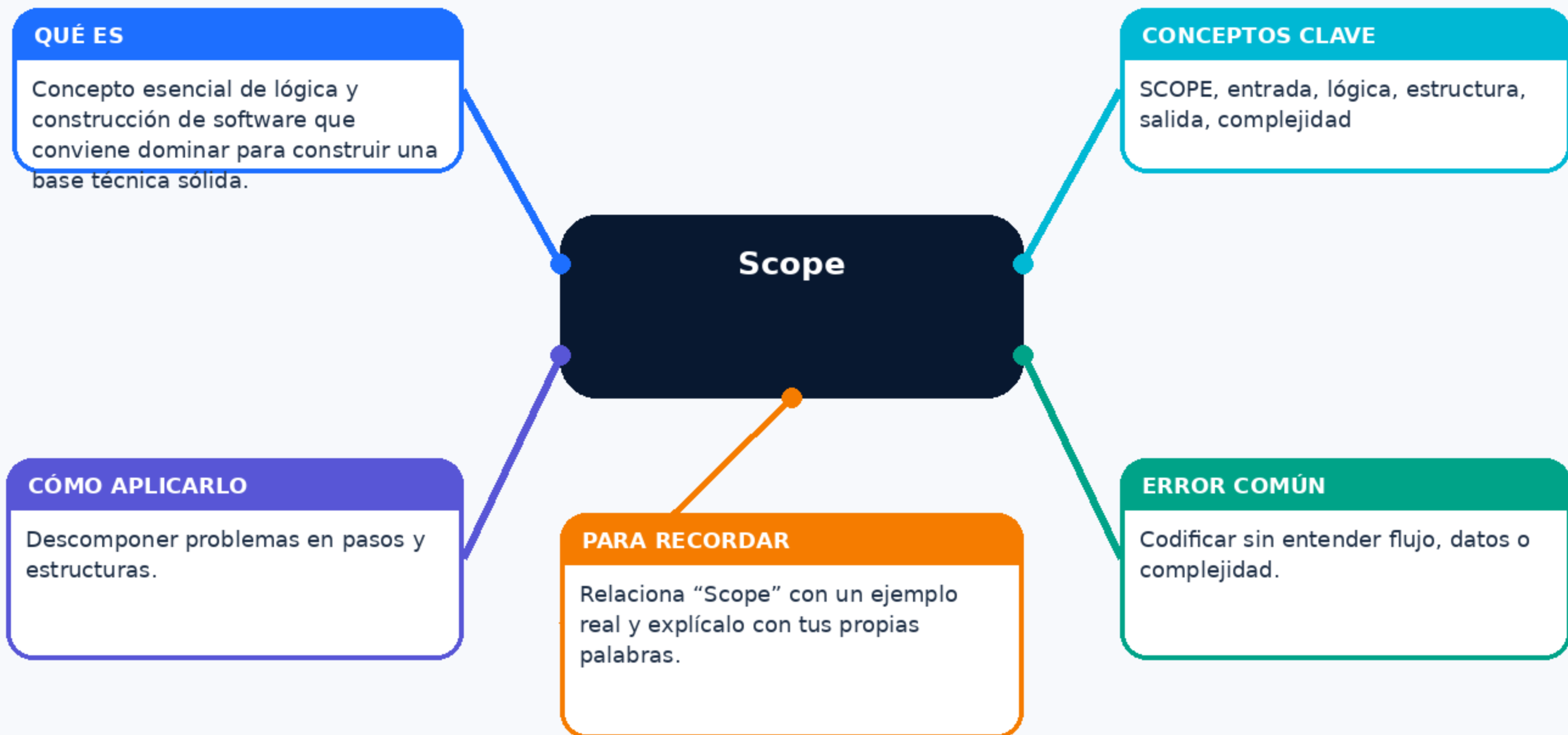
ERROR COMÚN

Codificar sin entender flujo, datos o complejidad.

Recursión



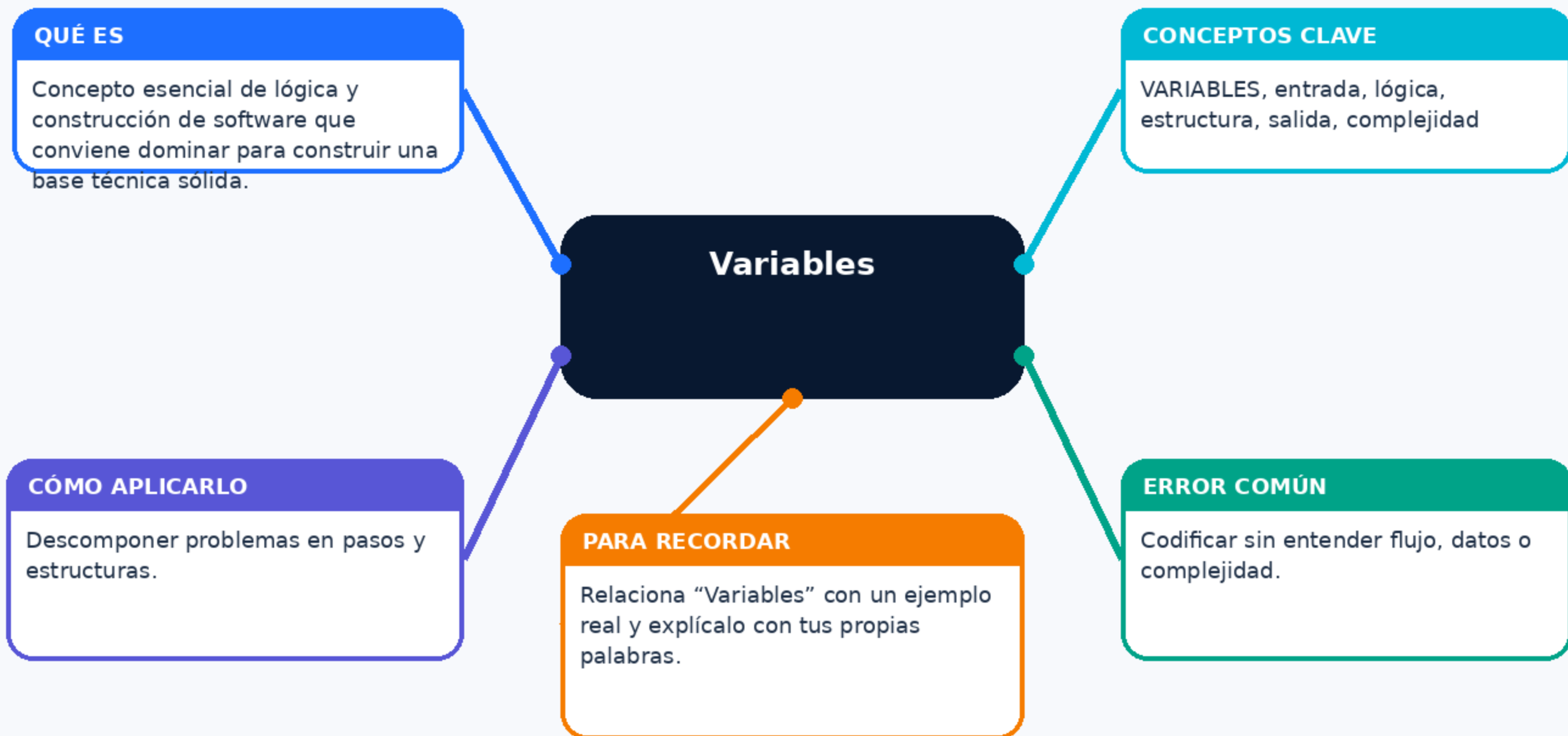
Scope



Tipos de datos



Variables



API

QUÉ ES

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

API, cliente, servidor, request, response, datos

API

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

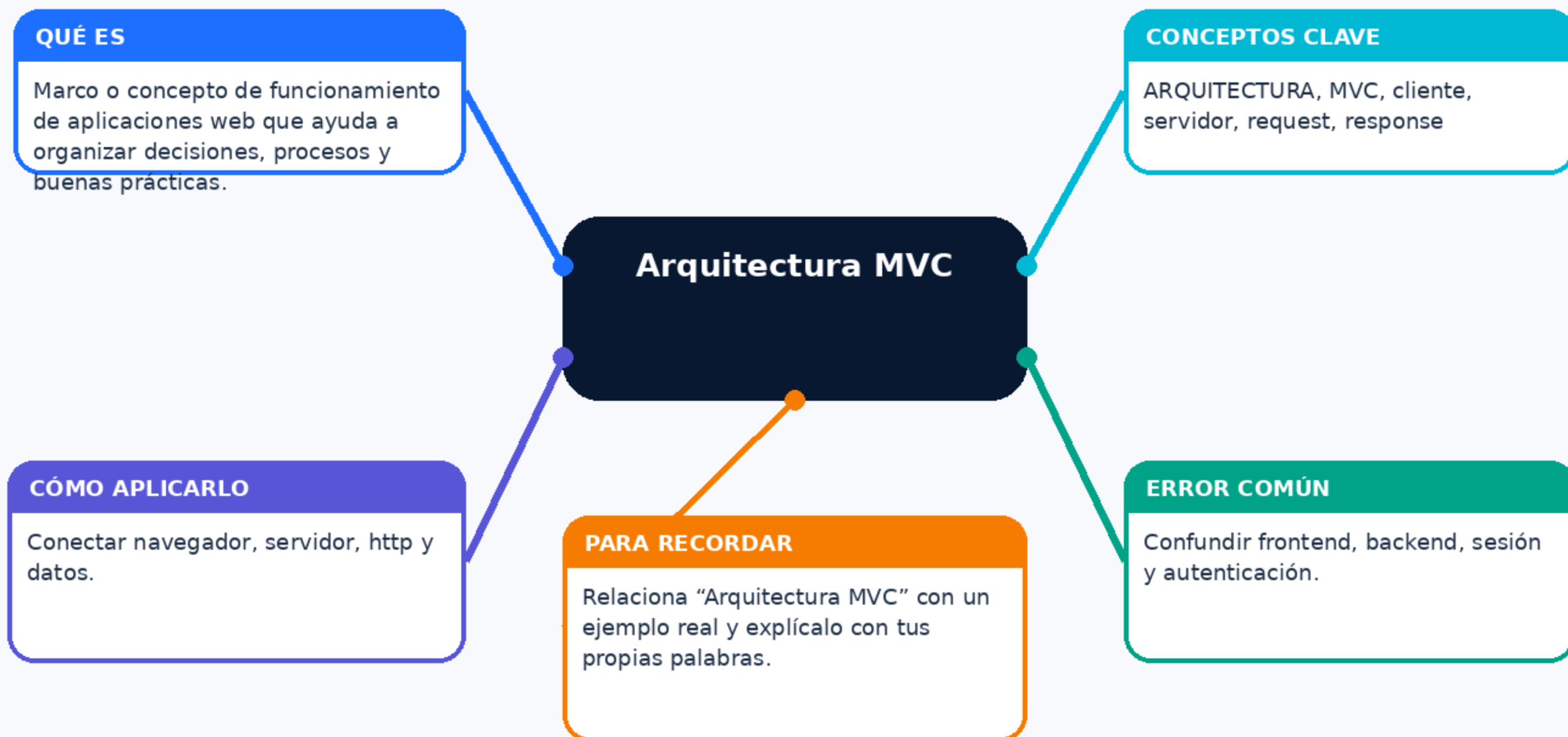
PARA RECORDAR

Relaciona "API" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

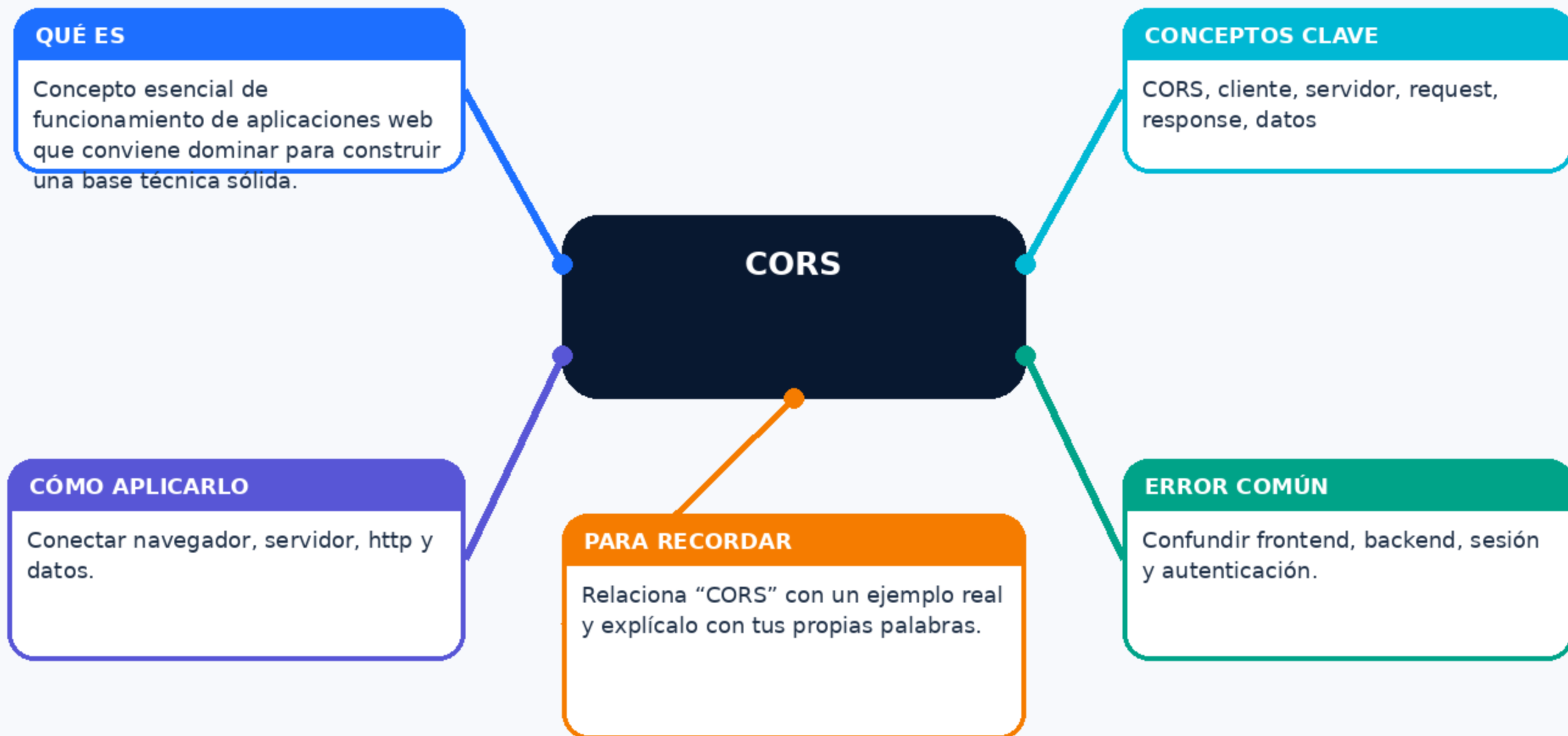
Arquitectura MVC



Autenticación con token



CORS



CSS

QUÉ ES

Lenguaje de estilos para controlar presentación y diseño visual en la Web.

CONCEPTOS CLAVE

selectores, cascada, box model, flexbox, grid

CSS

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

PARA RECORDAR

Relaciona "CSS" con un ejemplo real y explícalo con tus propias palabras.

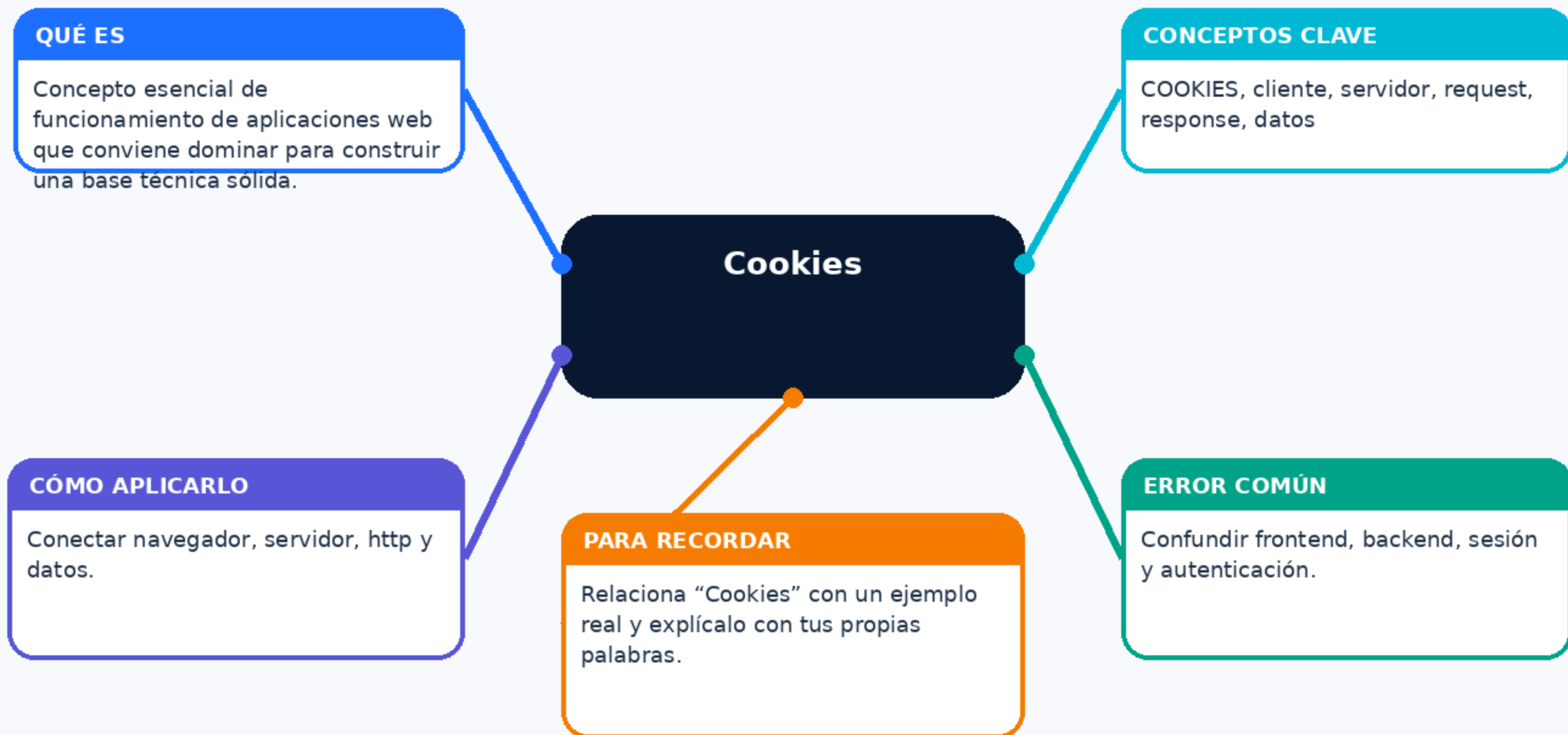
ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

Cliente y servidor



Cookies



Códigos de estado HTTP



Cómo funciona la Web



DOM

QUÉ ES

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

DOM, cliente, servidor, request, response, datos

DOM

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

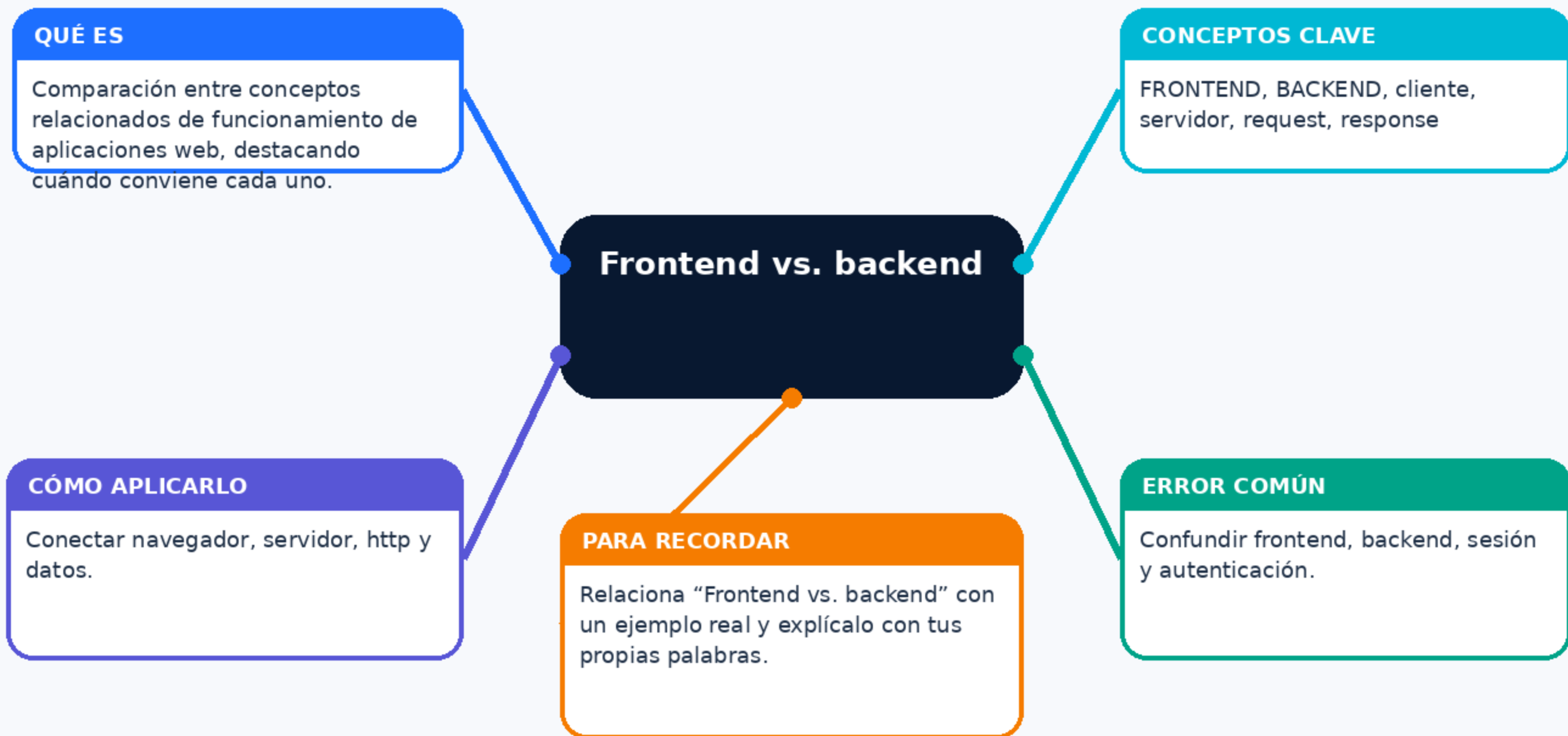
PARA RECORDAR

Relaciona "DOM" con un ejemplo real y explícalo con tus propias palabras.

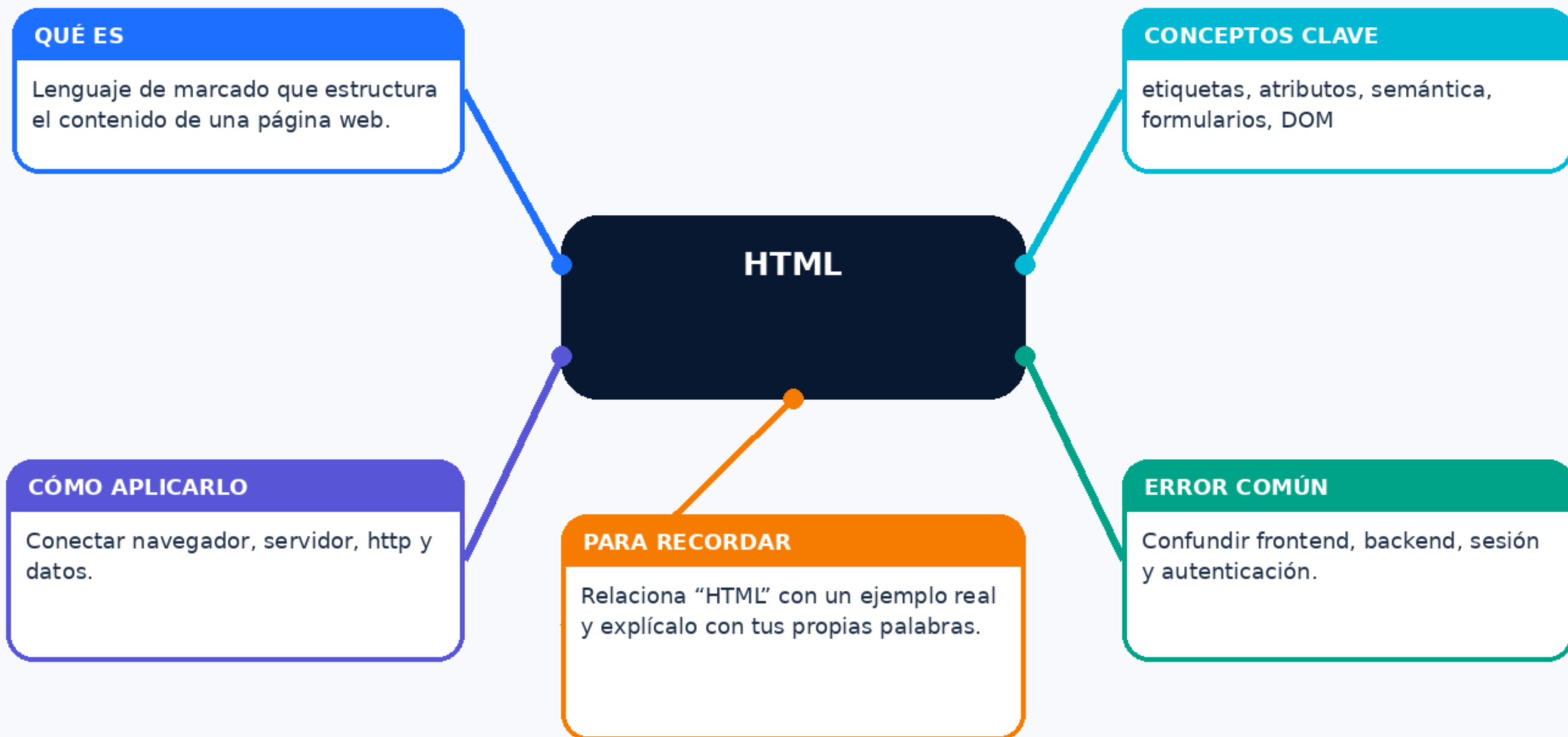
ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

Frontend vs. backend



HTML



JSON

QUÉ ES

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

JSON, cliente, servidor, request, response, datos

JSON

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

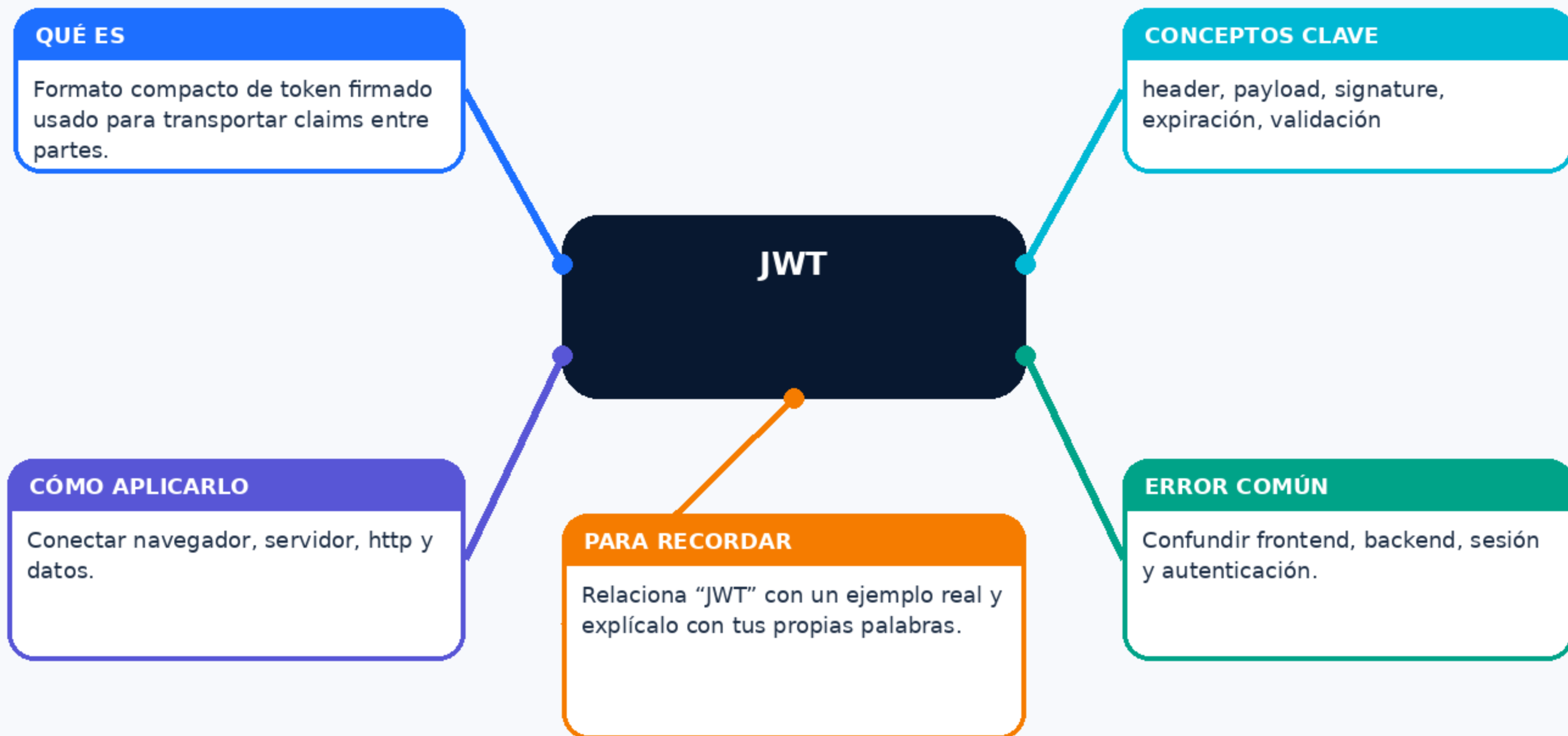
PARA RECORDAR

Relaciona "JSON" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

JWT



JavaScript

QUÉ ES

Lenguaje de programación usado ampliamente para crear comportamiento dinámico en aplicaciones web.

CONCEPTOS CLAVE

variables, funciones, DOM, eventos, async

JavaScript

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

PARA RECORDAR

Relaciona "JavaScript" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

Local Storage

QUÉ ES

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

LOCAL, STORAGE, cliente, servidor, request, response

Local Storage

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

PARA RECORDAR

Relaciona "Local Storage" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

Métodos HTTP

QUÉ ES

Marco o concepto de funcionamiento de aplicaciones web que ayuda a organizar decisiones, procesos y buenas prácticas.

CONCEPTOS CLAVE

MÉTODOS, HTTP, cliente, servidor, request, response

Métodos HTTP

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

PARA RECORDAR

Relaciona "Métodos HTTP" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

REST

QUÉ ES

Estilo arquitectónico para APIs basado en recursos, HTTP y operaciones sin estado.

CONCEPTOS CLAVE

recursos, URIs, métodos HTTP, stateless, representaciones

REST

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

PARA RECORDAR

Relaciona "REST" con un ejemplo real y explícalo con tus propias palabras.

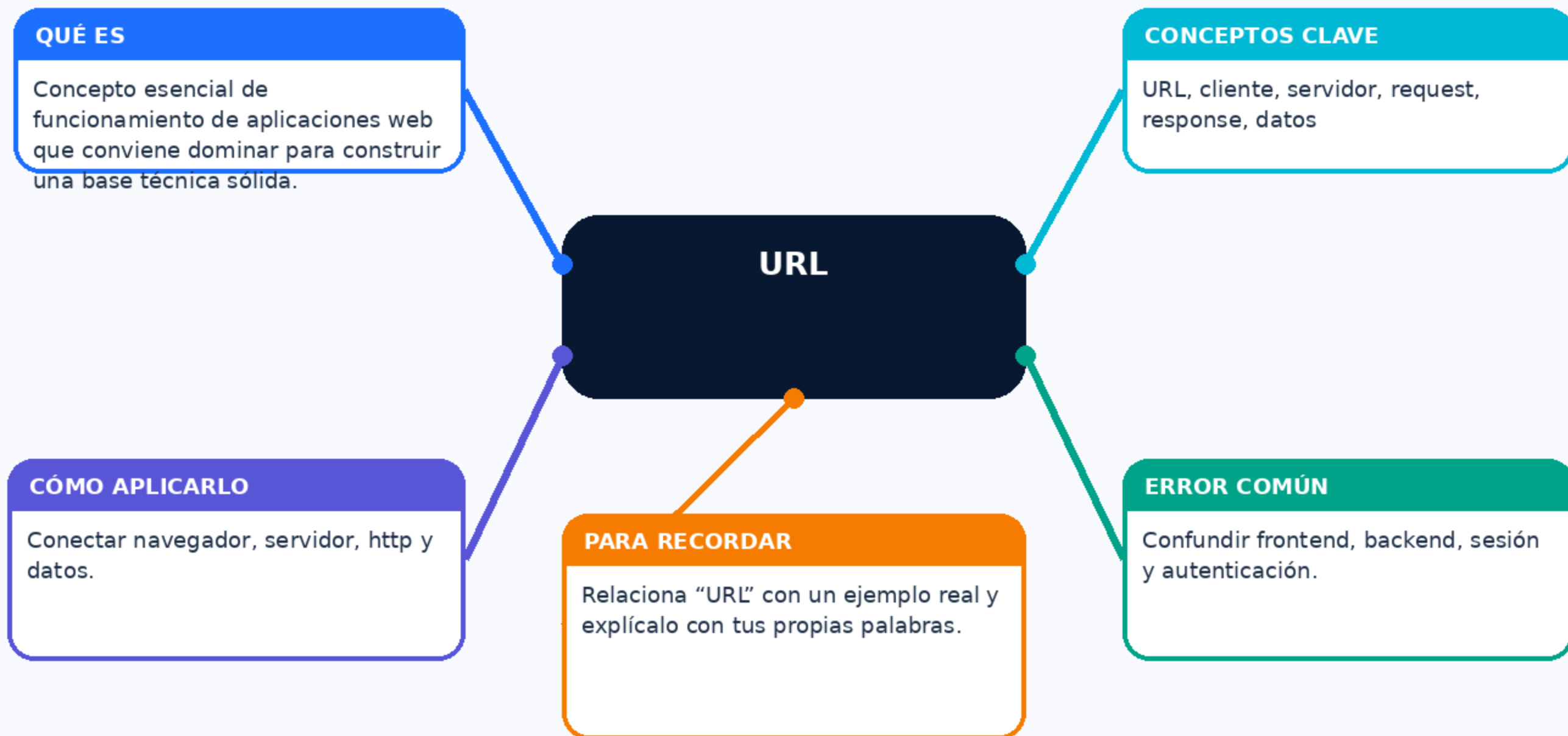
ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

Sesiones



URL



Webhooks



XML

QUÉ ES

Concepto esencial de funcionamiento de aplicaciones web que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

XML, cliente, servidor, request, response, datos

XML

CÓMO APLICARLO

Conectar navegador, servidor, http y datos.

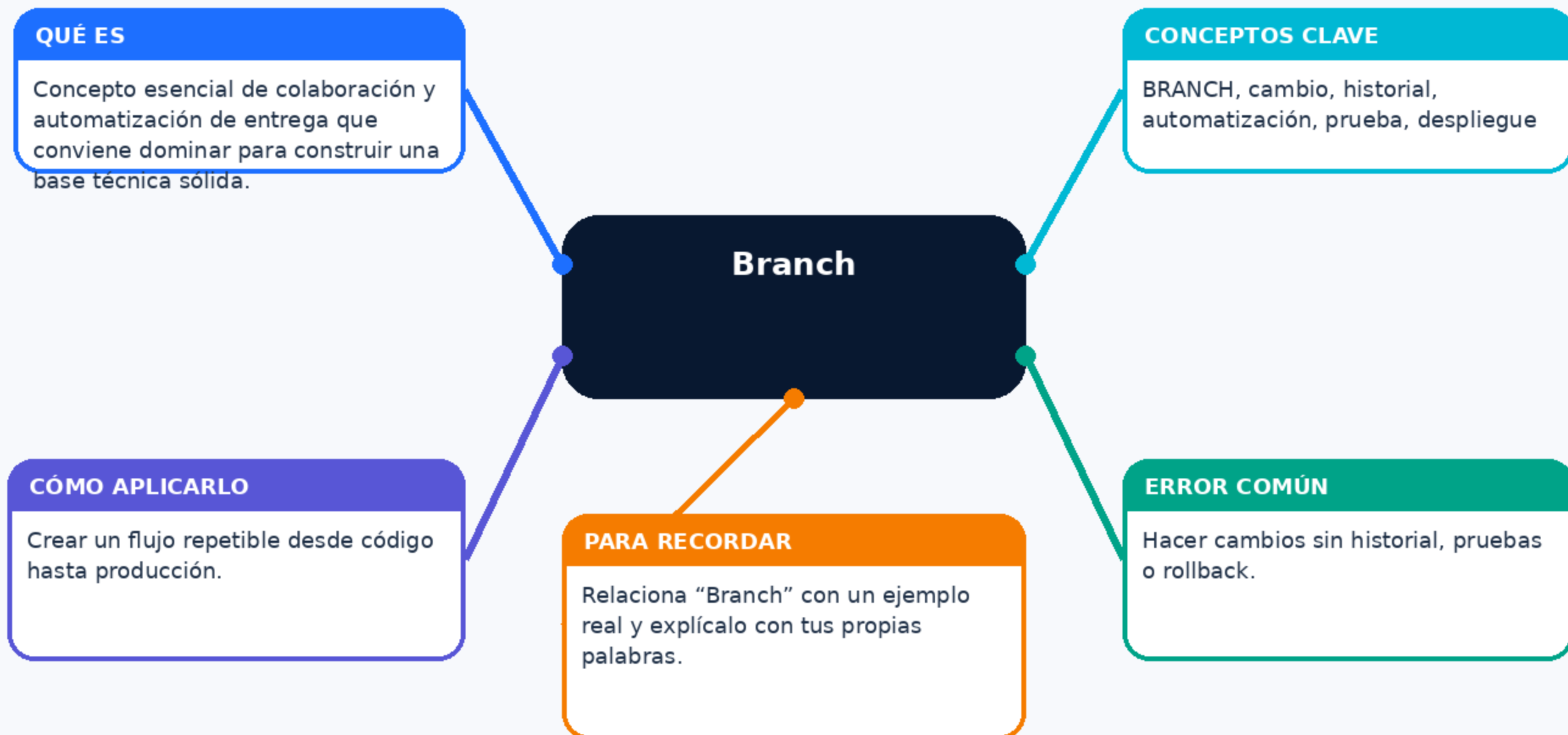
PARA RECORDAR

Relaciona "XML" con un ejemplo real y explícalo con tus propias palabras.

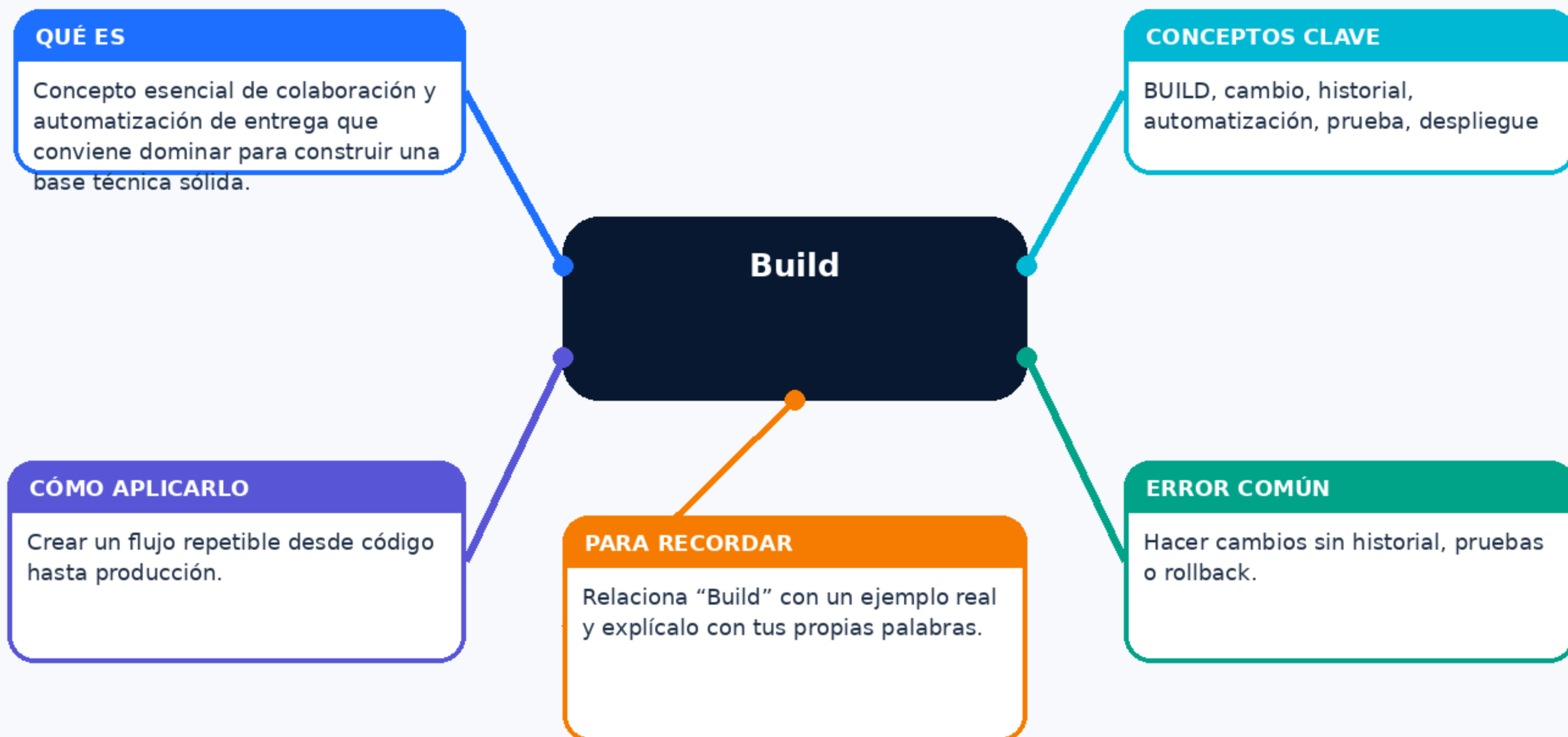
ERROR COMÚN

Confundir frontend, backend, sesión y autenticación.

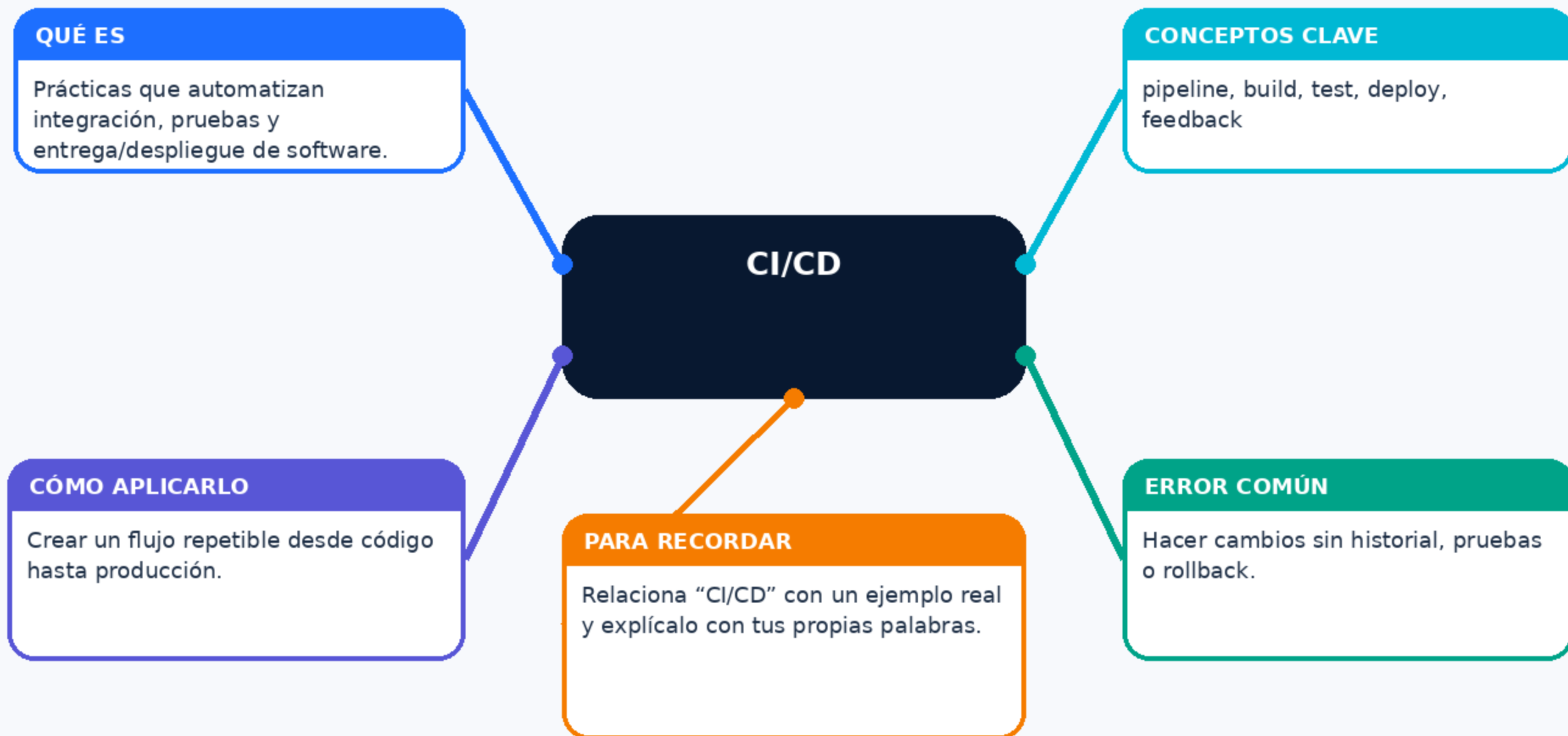
Branch



Build



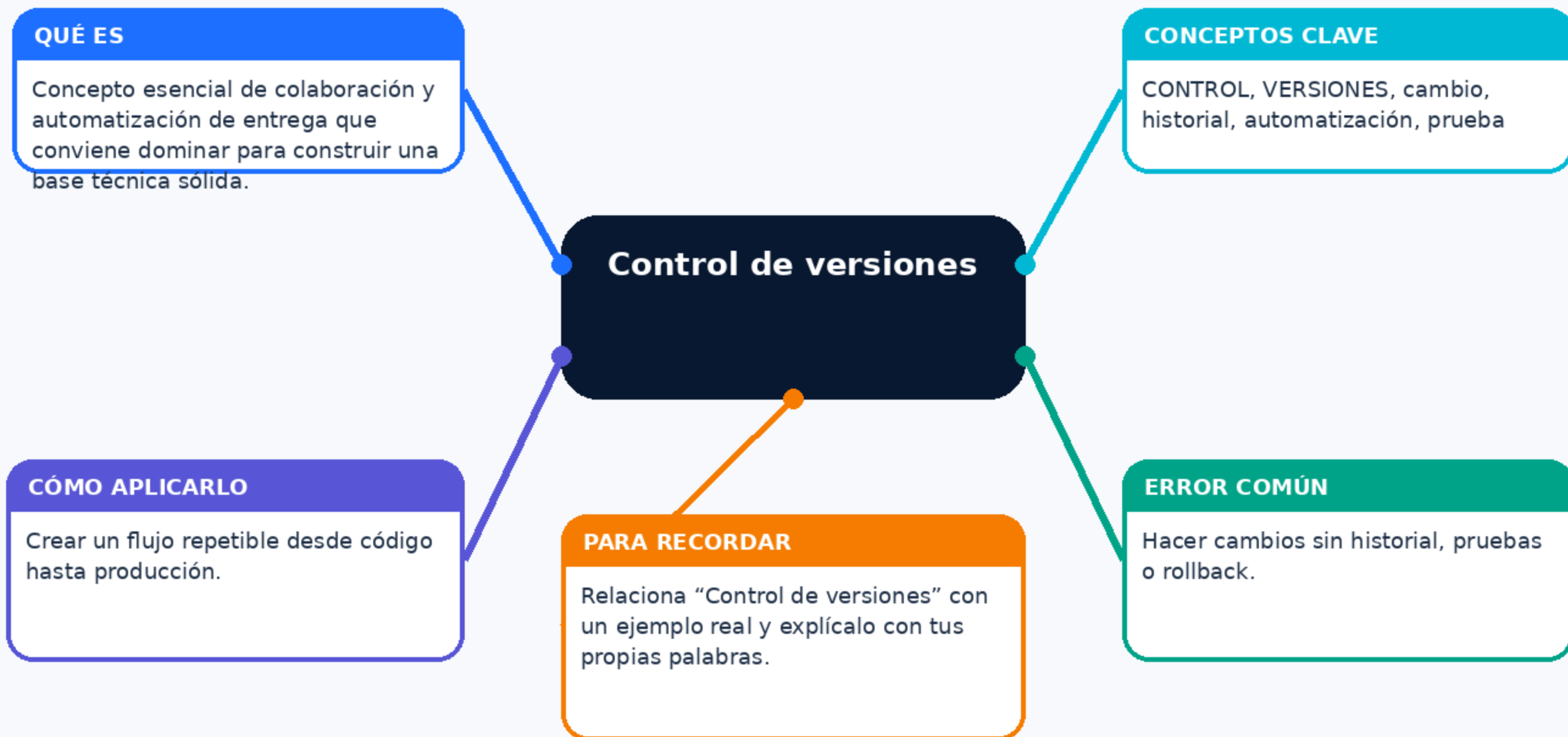
CI/CD



Commit



Control de versiones



Deploy

QUÉ ES

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

DEPLOY, cambio, historial, automatización, prueba, despliegue

Deploy

CÓMO APLICARLO

Crear un flujo repetible desde código hasta producción.

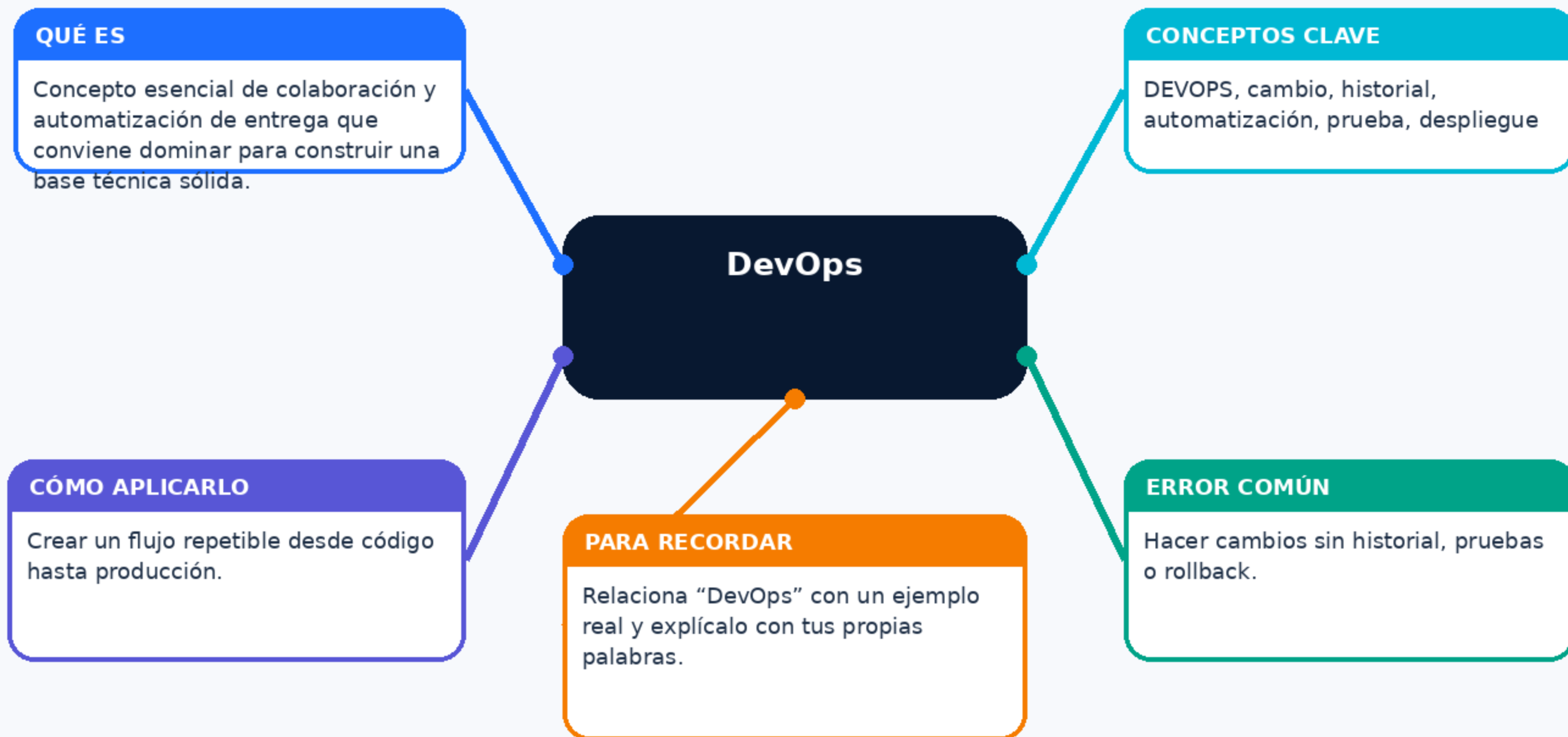
PARA RECORDAR

Relaciona "Deploy" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin historial, pruebas o rollback.

DevOps



Git

QUÉ ES

Sistema distribuido de control de versiones para rastrear cambios en código y colaborar.

CONCEPTOS CLAVE

commit, branch, merge, remote, history

Git

CÓMO APLICARLO

Crear un flujo repetible desde código hasta producción.

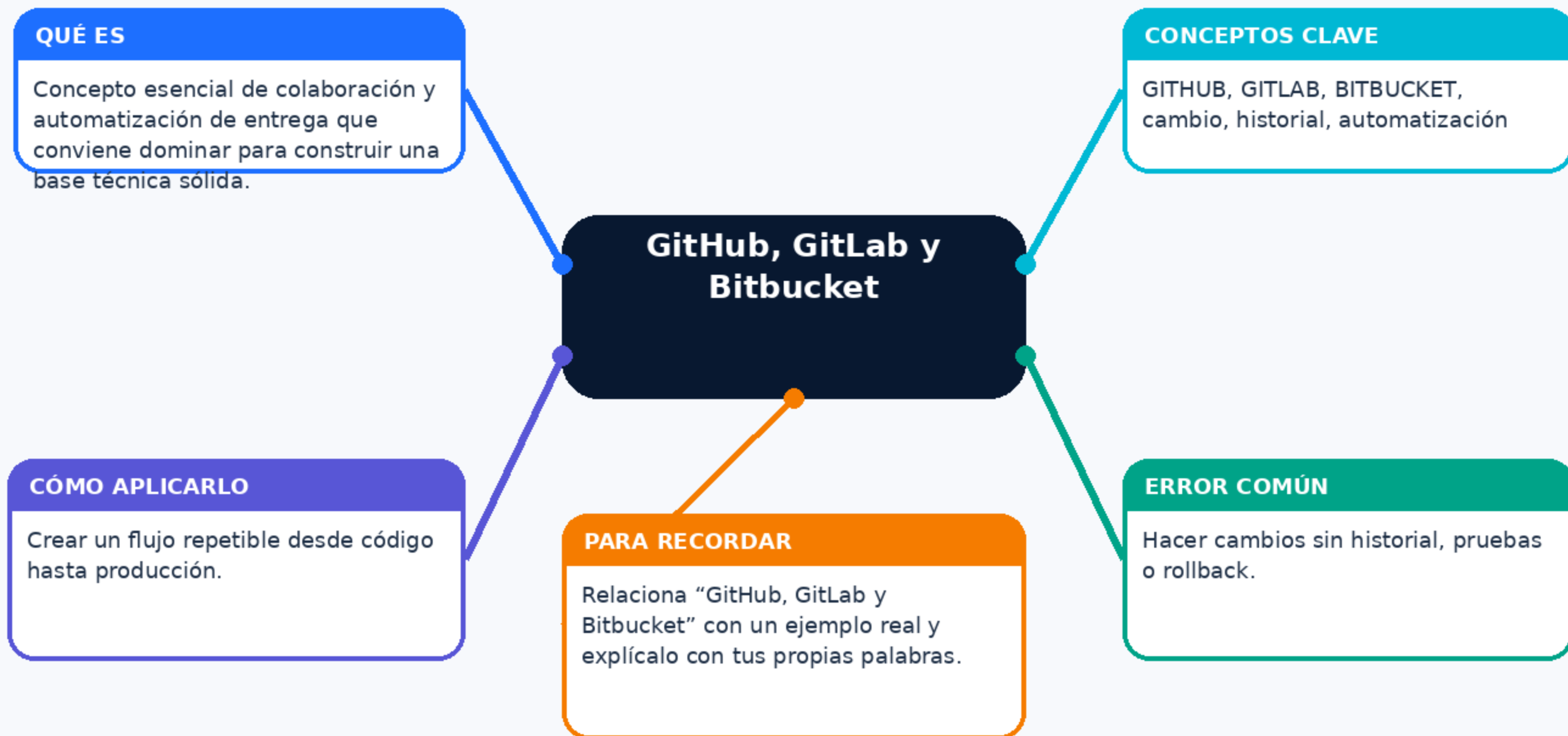
PARA RECORDAR

Relaciona "Git" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin historial, pruebas o rollback.

GitHub, GitLab y Bitbucket



Git clone, pull y push

QUÉ ES

Elemento práctico de colaboración y automatización de entrega utilizado para ejecutar una operación concreta de forma controlada.

CONCEPTOS CLAVE

GIT, CLONE, PULL, PUSH, cambio, historial

Git clone, pull y push

CÓMO APLICARLO

Crear un flujo repetible desde código hasta producción.

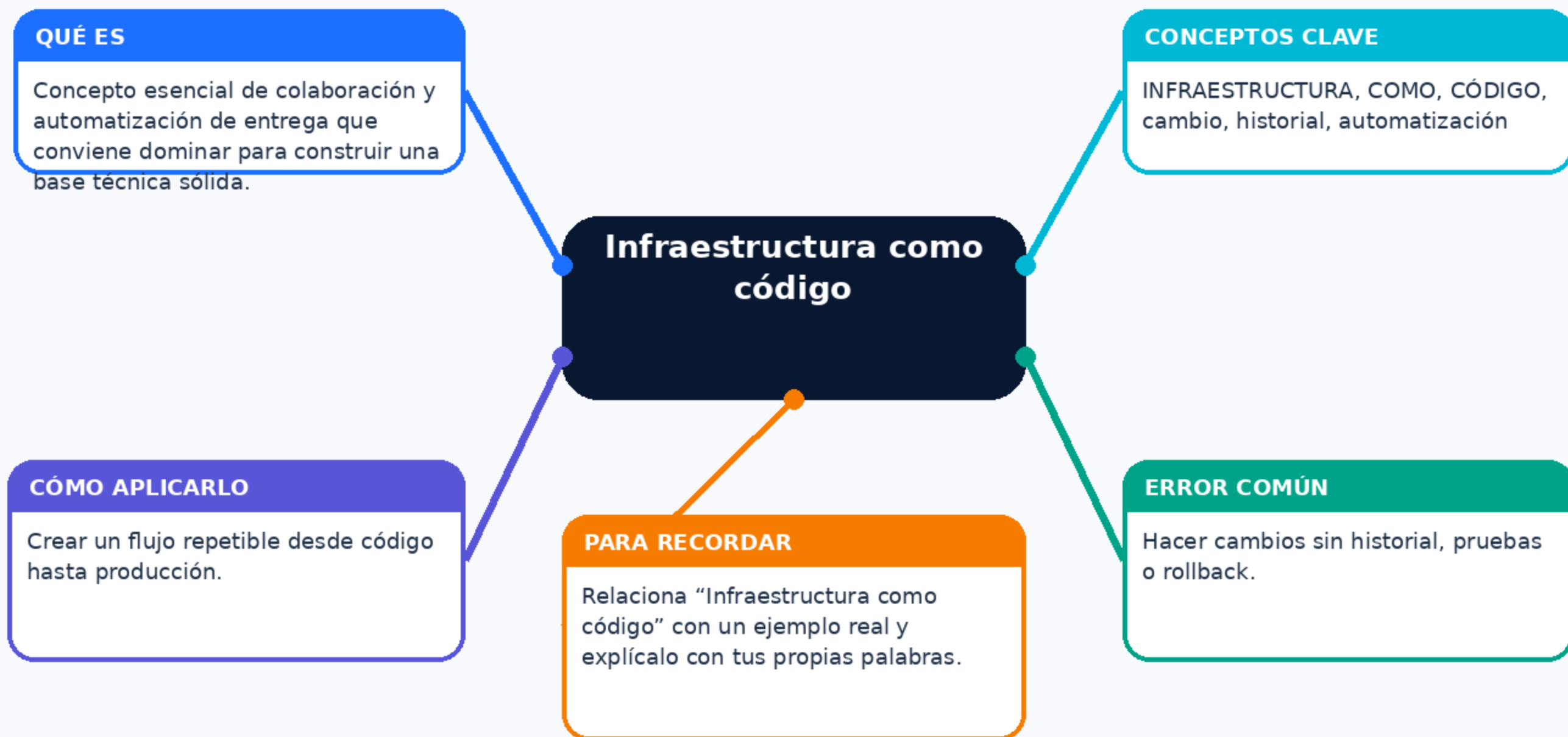
PARA RECORDAR

Relaciona "Git clone, pull y push" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin historial, pruebas o rollback.

Infraestructura como código



Merge

QUÉ ES

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

MERGE, cambio, historial, automatización, prueba, despliegue

Merge

CÓMO APLICARLO

Crear un flujo repetible desde código hasta producción.

PARA RECORDAR

Relaciona "Merge" con un ejemplo real y explícalo con tus propias palabras.

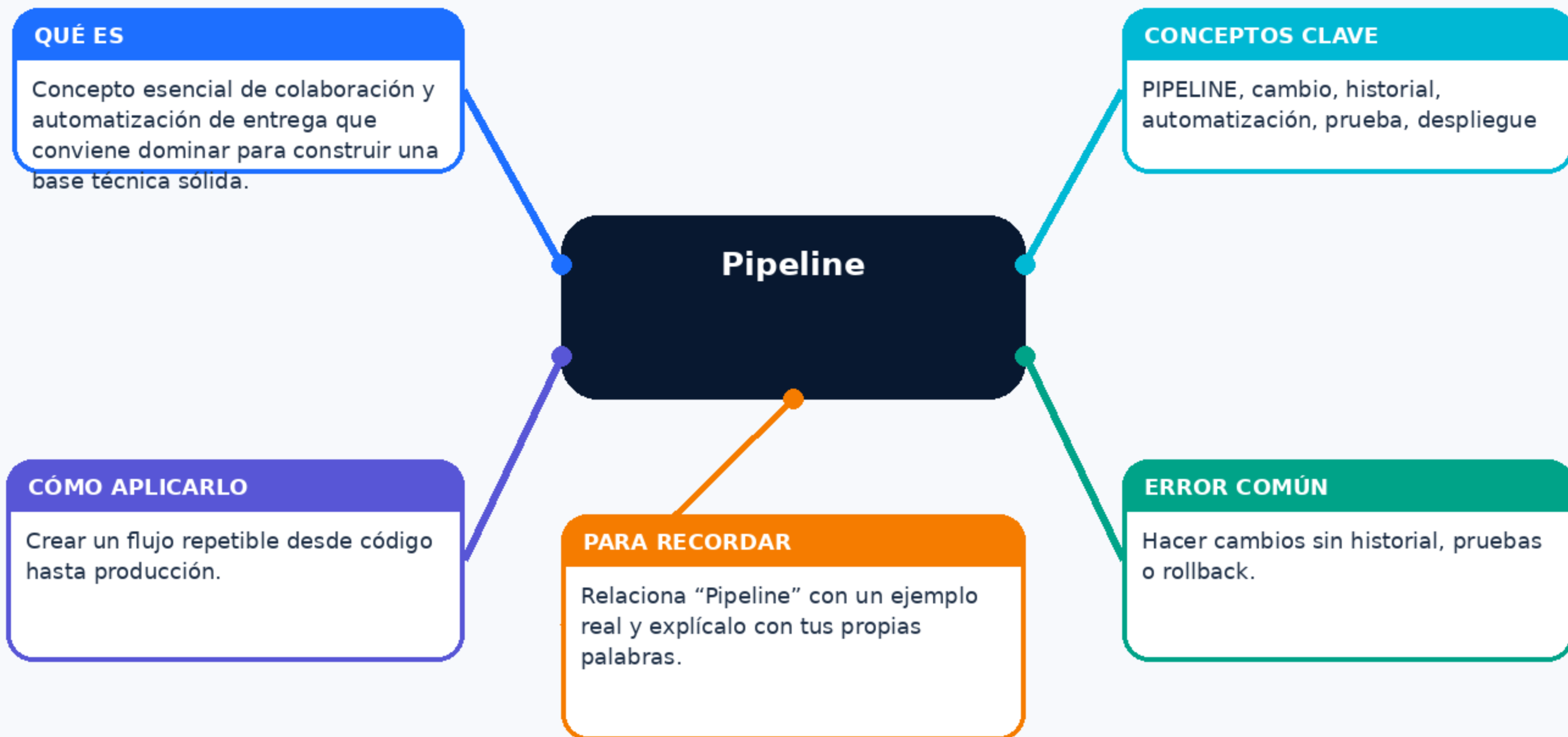
ERROR COMÚN

Hacer cambios sin historial, pruebas o rollback.

Observabilidad



Pipeline



Pull Request

QUÉ ES

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

PULL, REQUEST, cambio, historial, automatización, prueba

Pull Request

CÓMO APLICARLO

Crear un flujo repetible desde código hasta producción.

PARA RECORDAR

Relaciona "Pull Request" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin historial, pruebas o rollback.

Rebase

QUÉ ES

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

REBASE, cambio, historial, automatización, prueba, despliegue

Rebase

CÓMO APLICARLO

Crear un flujo repetible desde código hasta producción.

PARA RECORDAR

Relaciona "Rebase" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin historial, pruebas o rollback.

Repositorio local y remoto



Resolución de conflictos



Rollback

QUÉ ES

Concepto esencial de colaboración y automatización de entrega que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ROLLBACK, cambio, historial, automatización, prueba, despliegue

Rollback

CÓMO APLICARLO

Crear un flujo repetible desde código hasta producción.

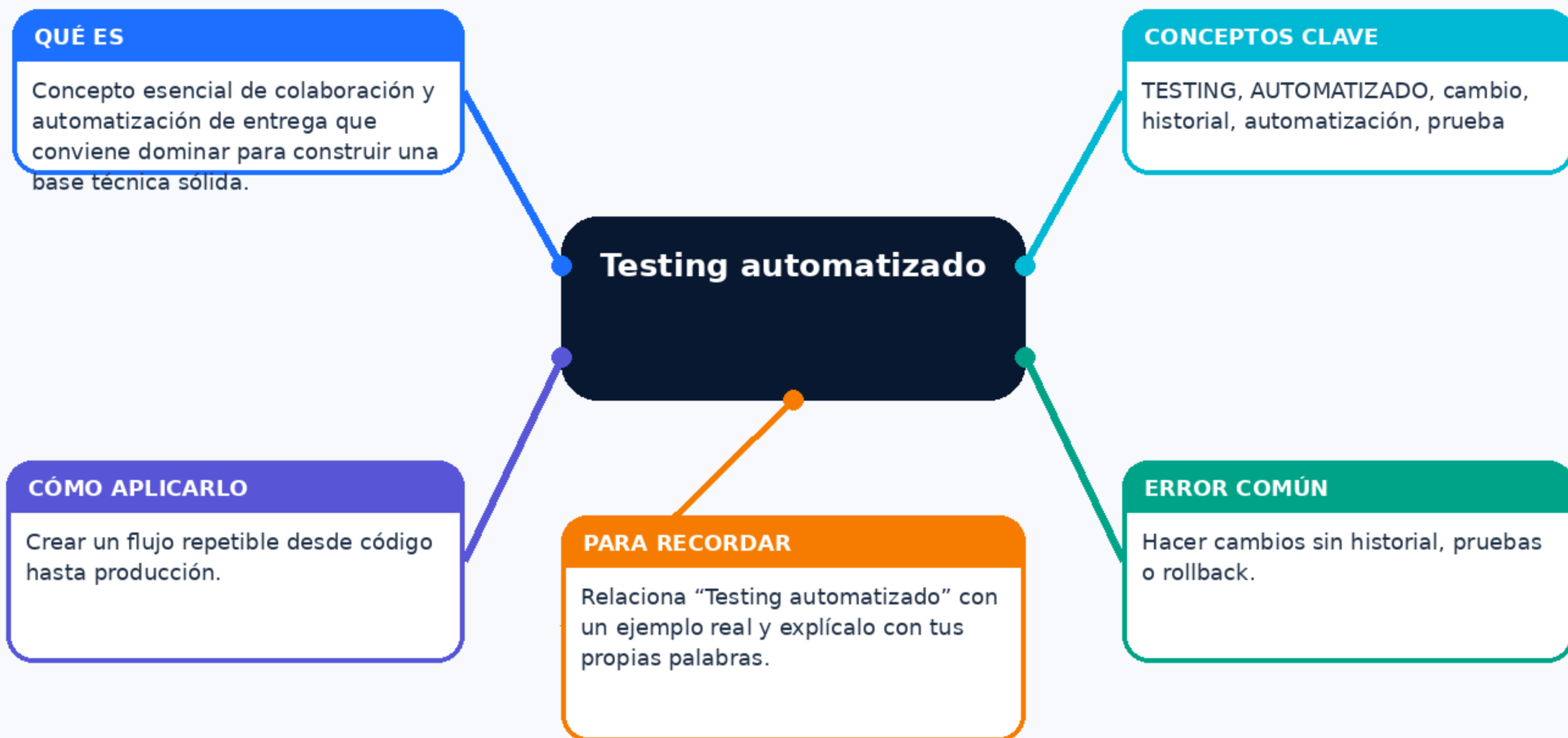
PARA RECORDAR

Relaciona "Rollback" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Hacer cambios sin historial, pruebas o rollback.

Testing automatizado



Almacenamiento en la nube



Alta disponibilidad

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ALTA, DISPONIBILIDAD, recurso, servicio, escalado, disponibilidad

Alta disponibilidad

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Alta disponibilidad" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

CDN

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

CDN, recurso, servicio, escalado, disponibilidad, costo

CDN

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "CDN" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Cloud Computing

QUÉ ES

Entrega bajo demanda de recursos de computación a través de una red.

CONCEPTOS CLAVE

elasticidad, pago por uso, autoservicio, regiones, servicios

Cloud Computing

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Cloud Computing" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Contenedores

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

CONTENEDORES, recurso, servicio, escalado, disponibilidad, costo

Contenedores

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Contenedores" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Disaster Recovery

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

DISASTER, RECOVERY, recurso, servicio, escalado, disponibilidad

Disaster Recovery

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Disaster Recovery" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Docker

QUÉ ES

Plataforma para empaquetar y ejecutar aplicaciones en contenedores aislados.

CONCEPTOS CLAVE

imagen, contenedor, Dockerfile, registry, volumen

Docker

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Docker" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Docker Compose

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

DOCKER, COMPOSE, recurso, servicio, escalado, disponibilidad

Docker Compose

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Docker Compose" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Dockerfile

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

DOCKERFILE, recurso, servicio, escalado, disponibilidad, costo

Dockerfile

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Dockerfile" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Elasticidad

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ELASTICIDAD, recurso, servicio, escalado, disponibilidad, costo

Elasticidad

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

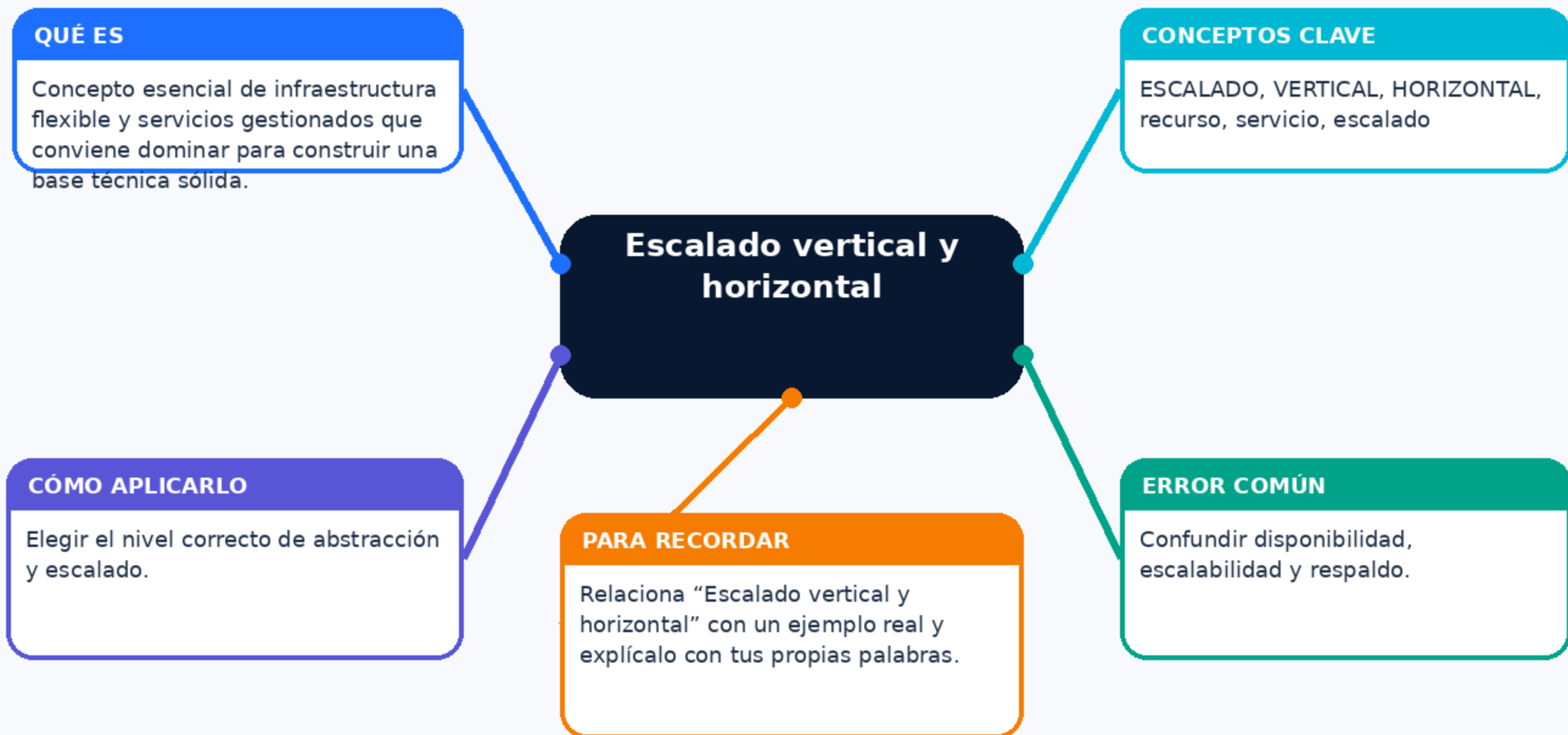
PARA RECORDAR

Relaciona "Elasticidad" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Escalado vertical y horizontal



Hipervisor

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

HIPERVISOR, recurso, servicio, escalado, disponibilidad, costo

Hipervisor

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Hypervisor" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

IaaS

QUÉ ES

Modelo cloud que ofrece cómputo, red y almacenamiento virtualizados.

CONCEPTOS CLAVE

VM, red, disco, control del SO, pago por uso

IaaS

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

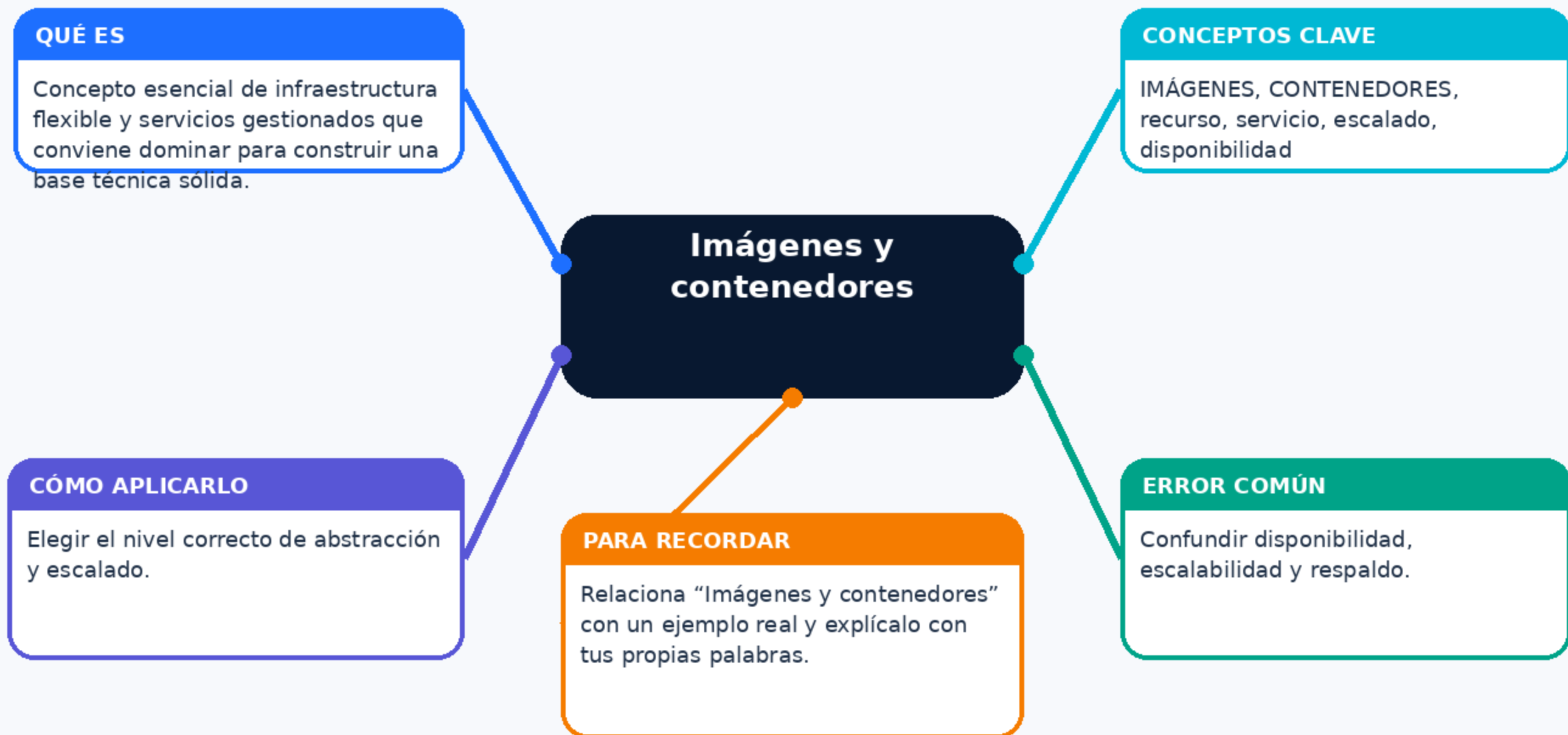
PARA RECORDAR

Relaciona "IaaS" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Imágenes y contenedores



Kubernetes

QUÉ ES

Orquestador de contenedores que automatiza despliegue, escalado y operación.

CONCEPTOS CLAVE

pod, deployment, service, ingress, namespace

Kubernetes

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Kubernetes" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Máquinas virtuales

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

MÁQUINAS, VIRTUALES, recurso, servicio, escalado, disponibilidad

Máquinas virtuales

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

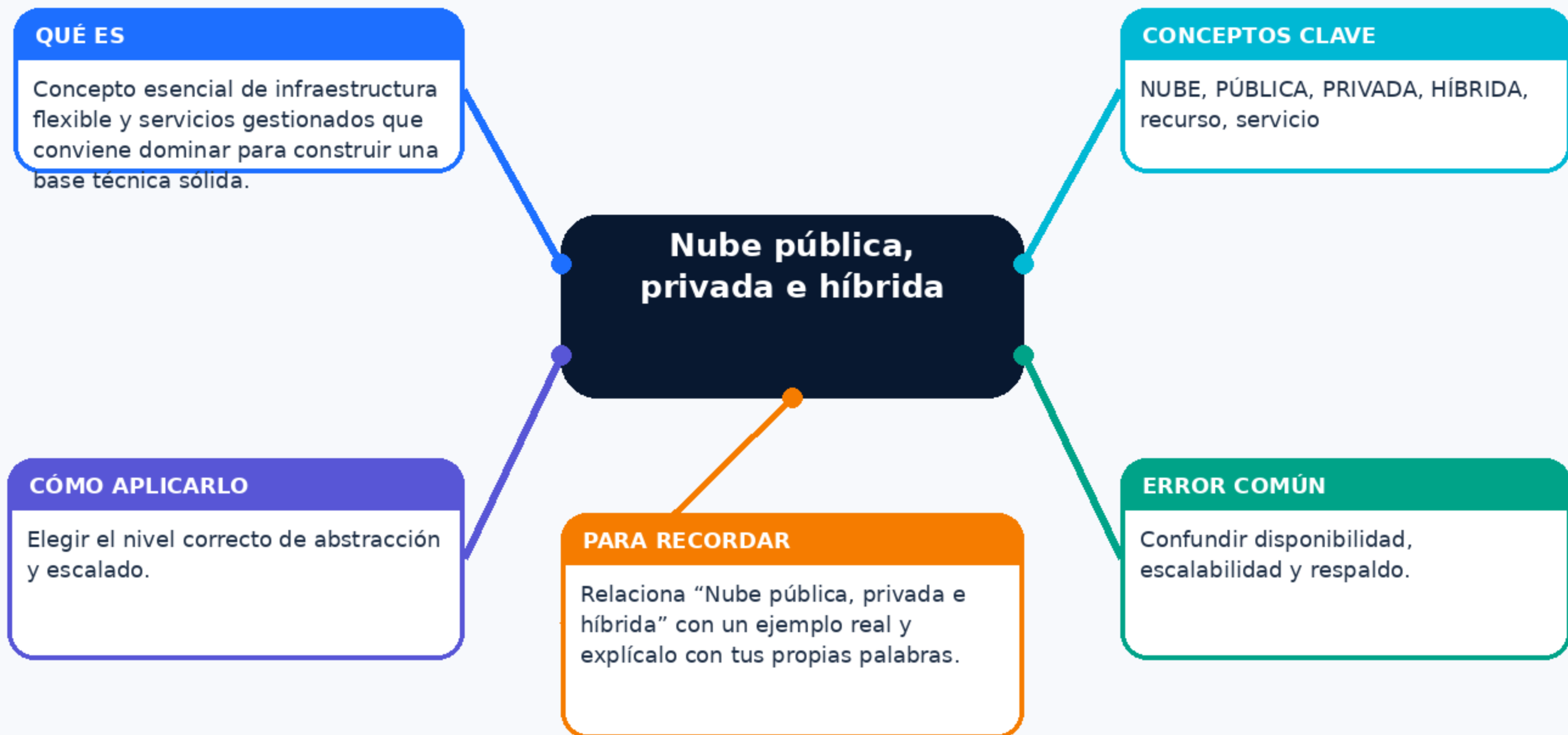
PARA RECORDAR

Relaciona "Máquinas virtuales" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Nube pública, privada e híbrida



Object Storage

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

OBJECT, STORAGE, recurso, servicio, escalado, disponibilidad

Object Storage

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Object Storage" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

PaaS

QUÉ ES

Modelo cloud que ofrece una plataforma gestionada para desplegar aplicaciones.

CONCEPTOS CLAVE

runtime, despliegue, escalado, servicios gestionados

PaaS

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "PaaS" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Pods

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

PODS, recurso, servicio, escalado, disponibilidad, costo

Pods

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

PARA RECORDAR

Relaciona "Pods" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

Regiones y zonas de disponibilidad



SaaS

QUÉ ES

Modelo en el que una aplicación completa se consume como servicio.

CONCEPTOS CLAVE

navegador, suscripción, proveedor
gestiona infraestructura

SaaS

CÓMO APLICARLO

Elegir el nivel correcto de abstracción
y escalado.

PARA RECORDAR

Relaciona "SaaS" con un ejemplo real
y explícalo con tus propias palabras.

ERROR COMÚN

Confundir disponibilidad,
escalabilidad y respaldo.

Serverless

QUÉ ES

Concepto esencial de infraestructura flexible y servicios gestionados que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

SERVERLESS, recurso, servicio, escalado, disponibilidad, costo

Serverless

CÓMO APLICARLO

Elegir el nivel correcto de abstracción y escalado.

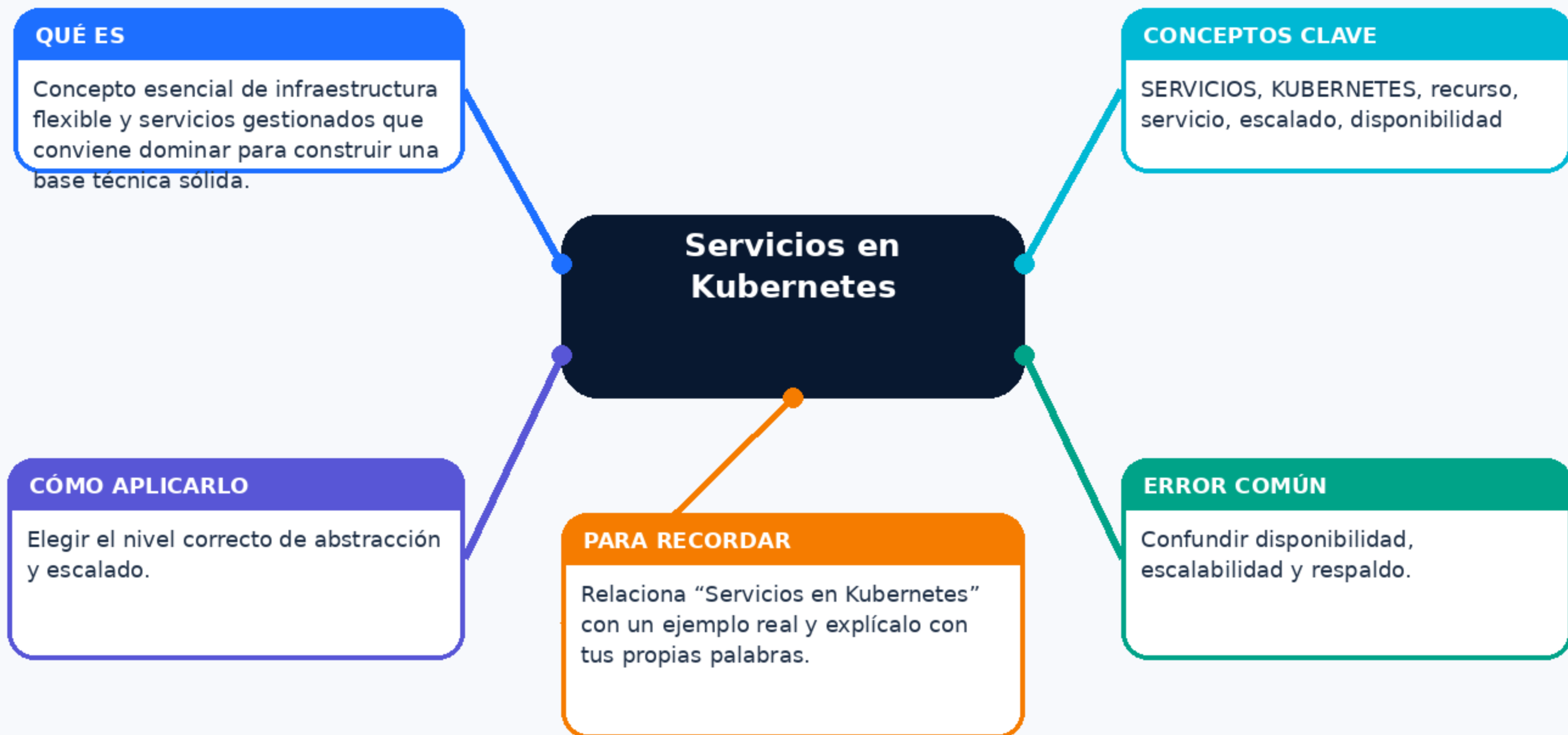
PARA RECORDAR

Relaciona "Serverless" con un ejemplo real y explícalo con tus propias palabras.

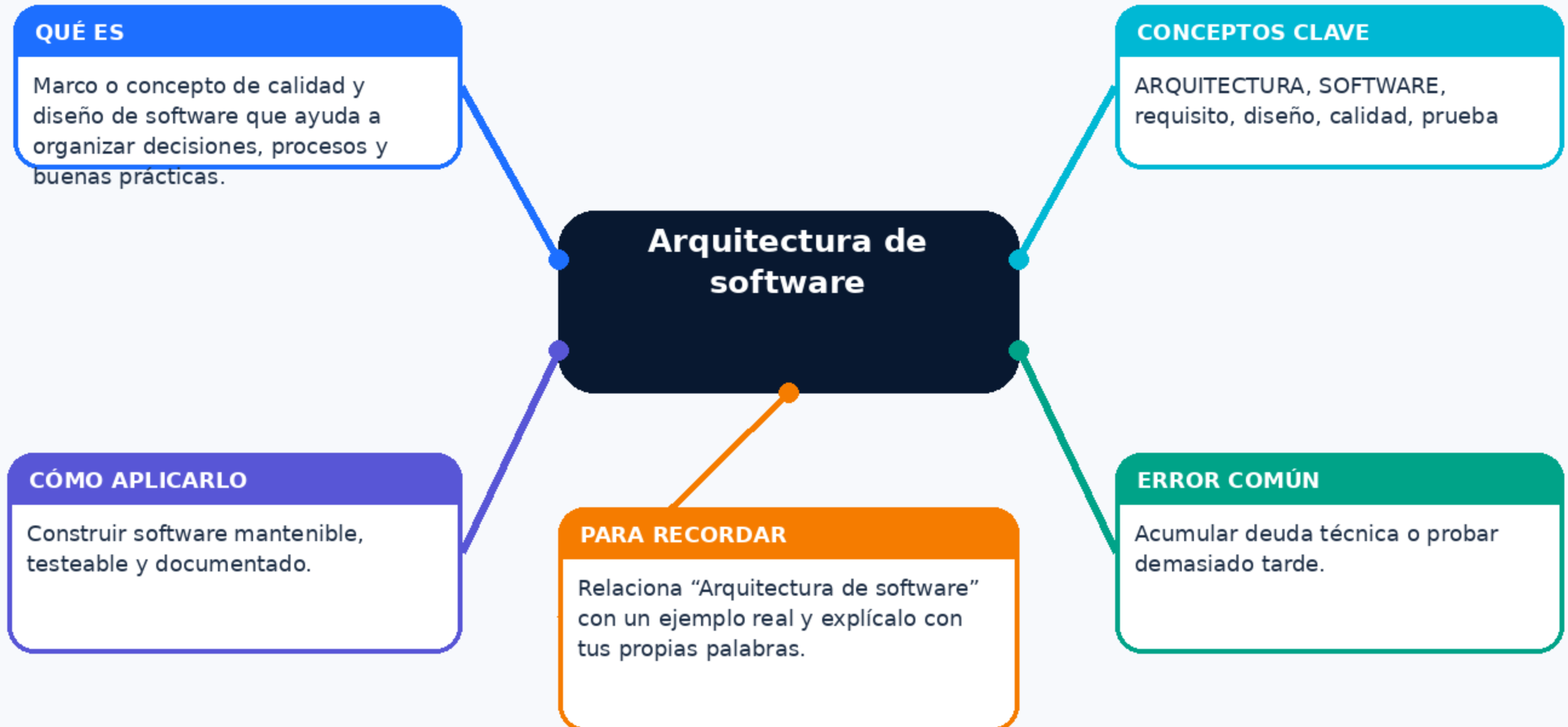
ERROR COMÚN

Confundir disponibilidad, escalabilidad y respaldo.

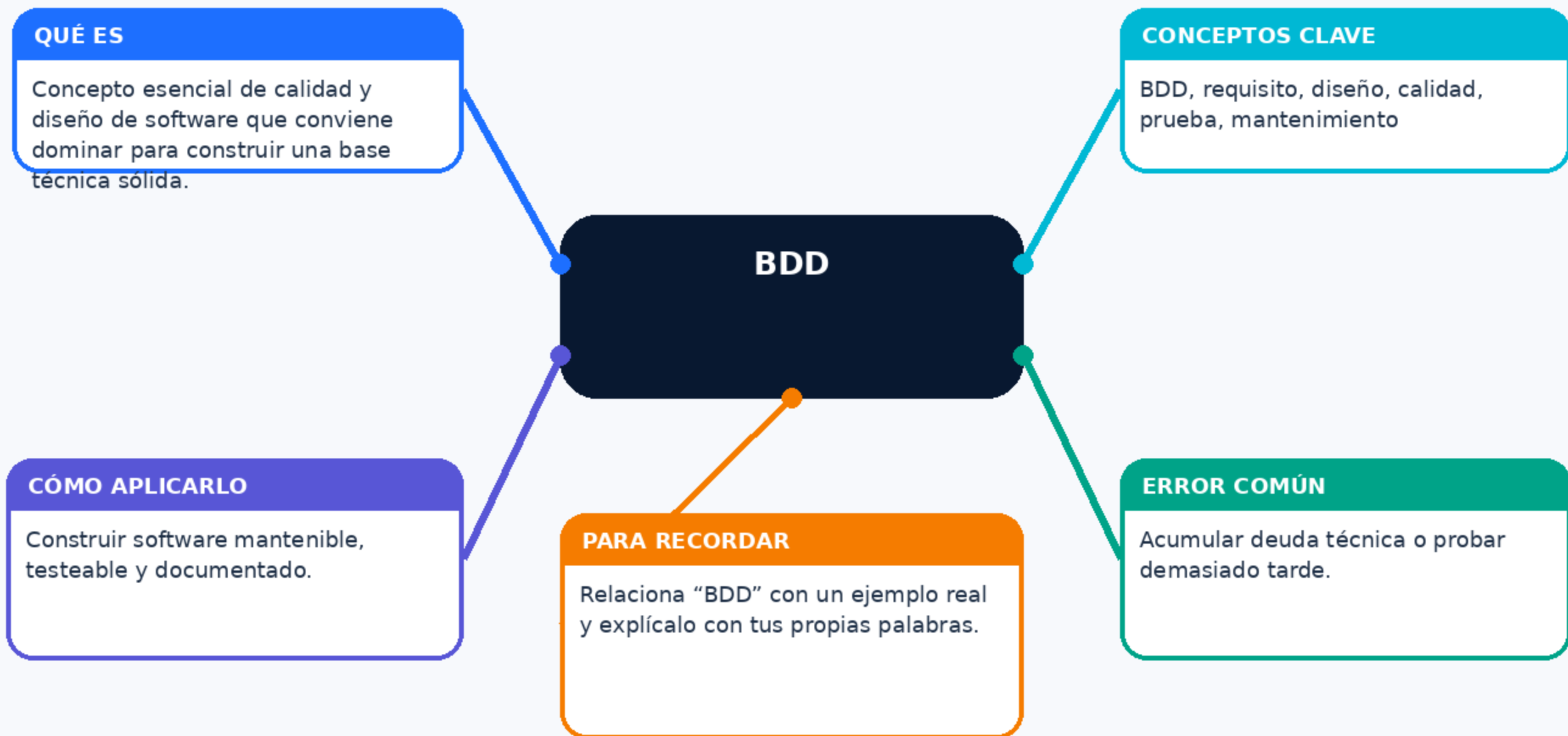
Servicios en Kubernetes



Arquitectura de software



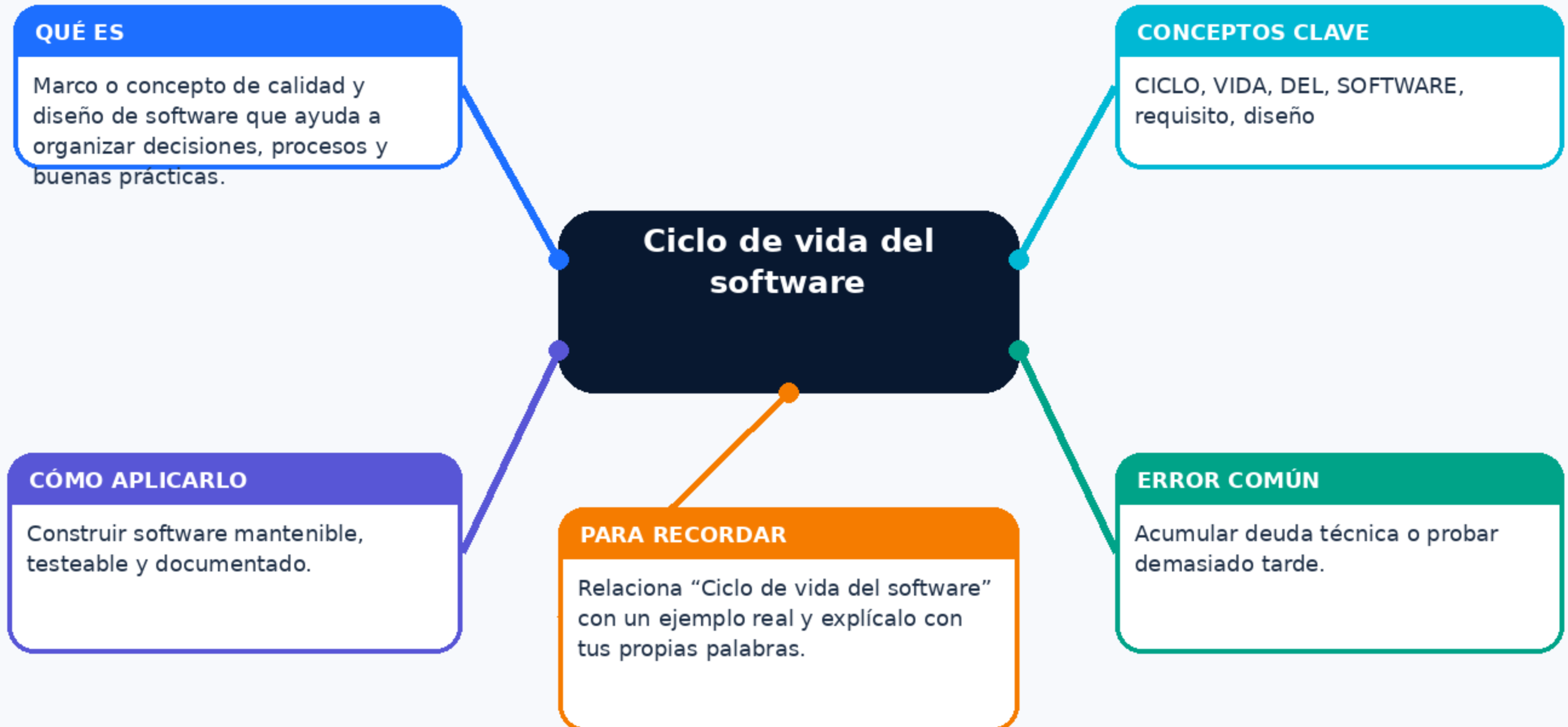
BDD



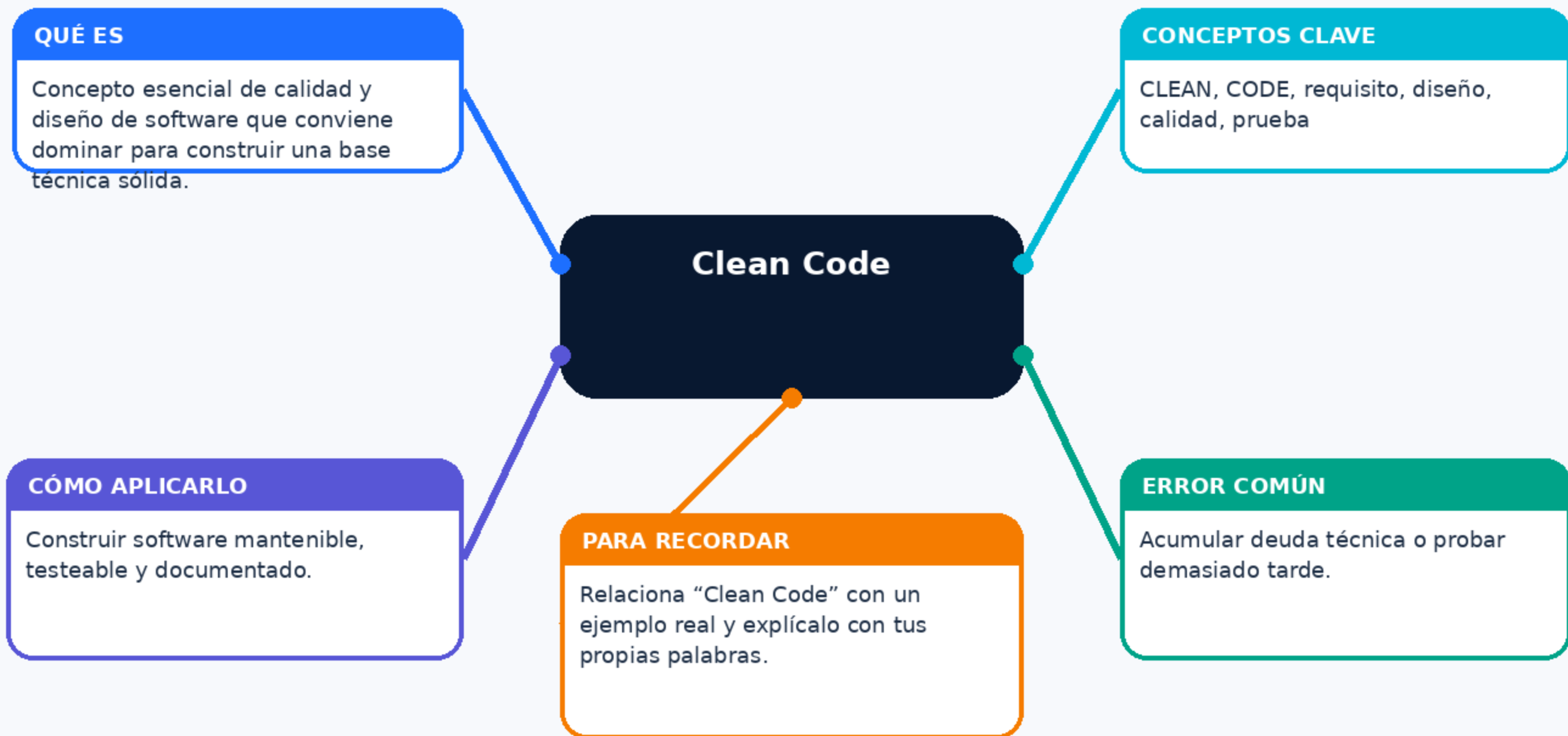
Casos de uso



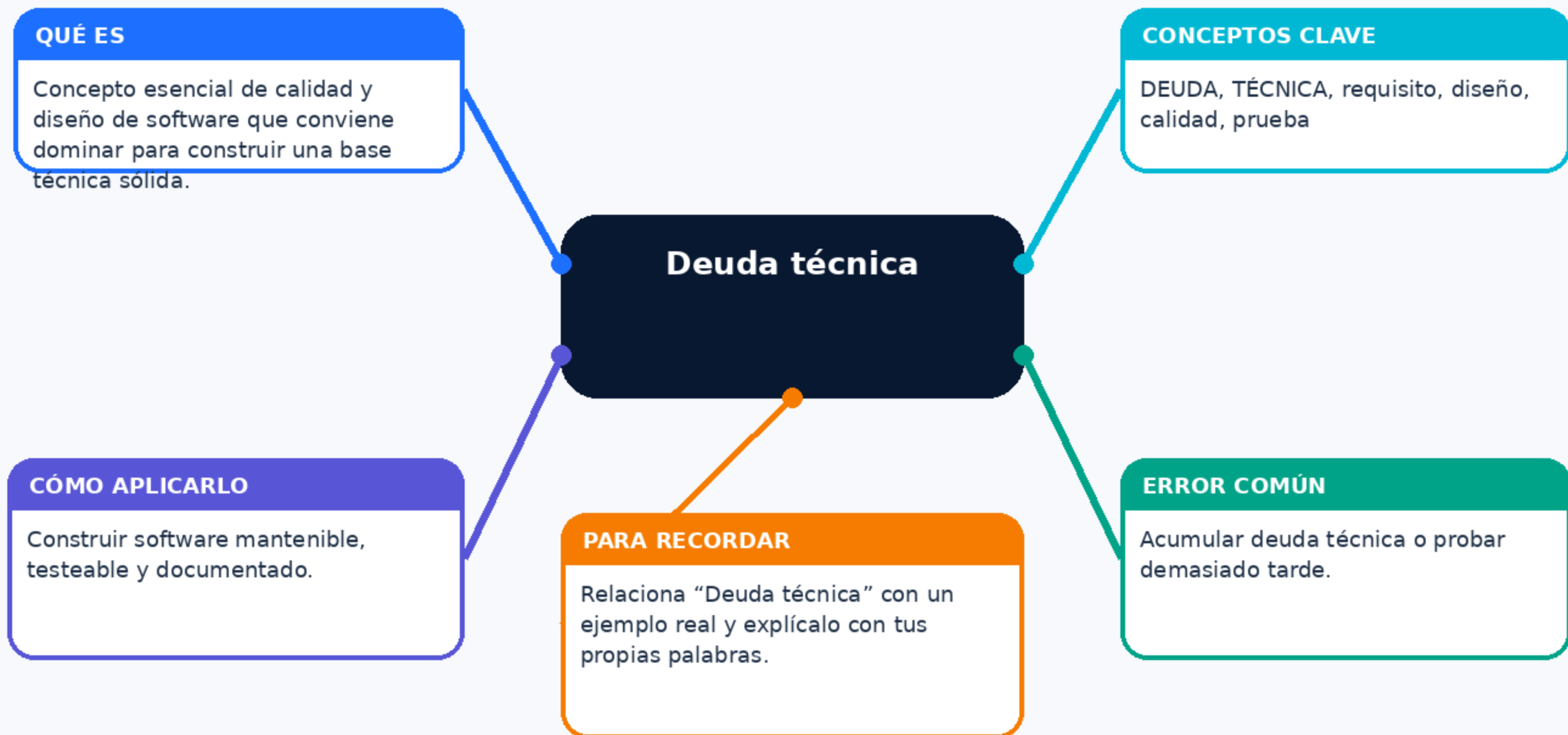
Ciclo de vida del software



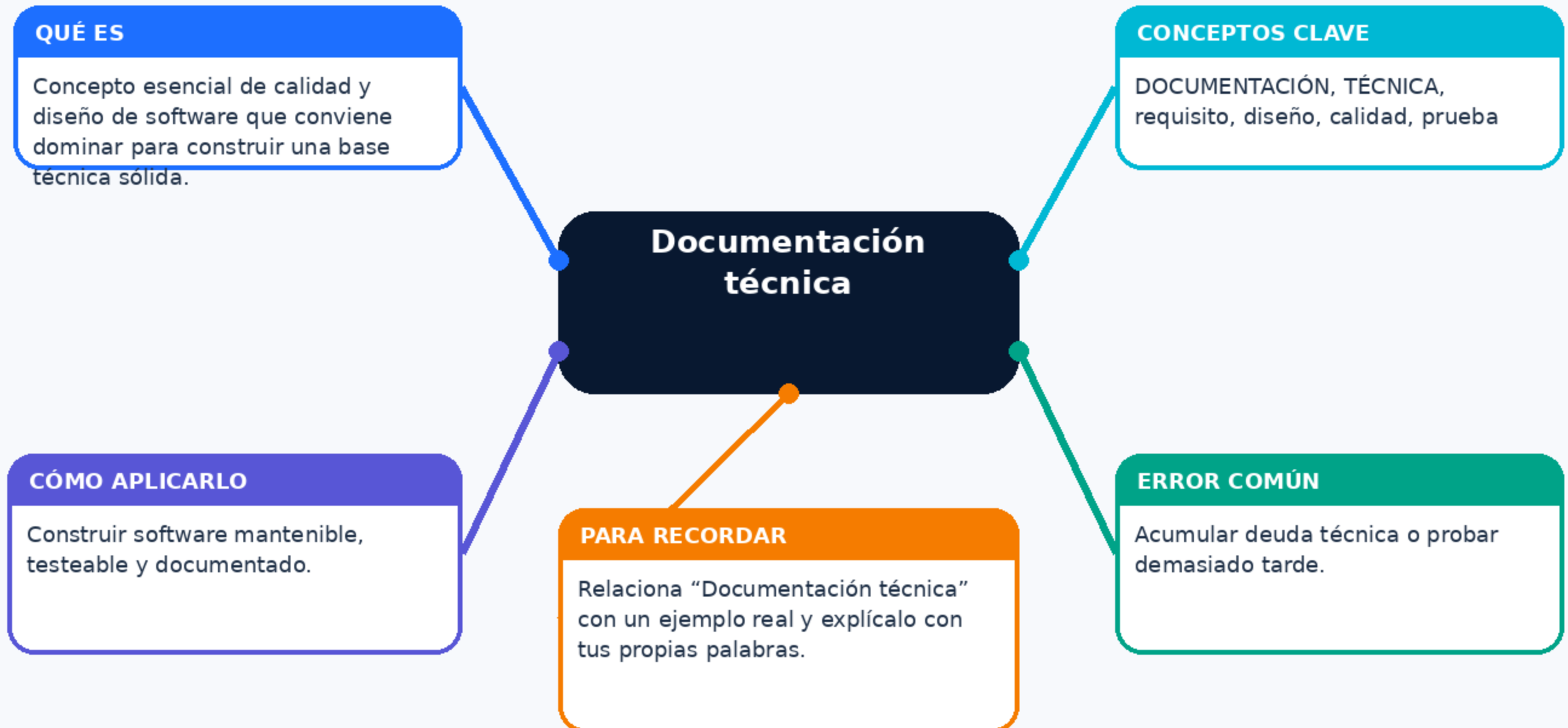
Clean Code



Deuda técnica



Documentación técnica



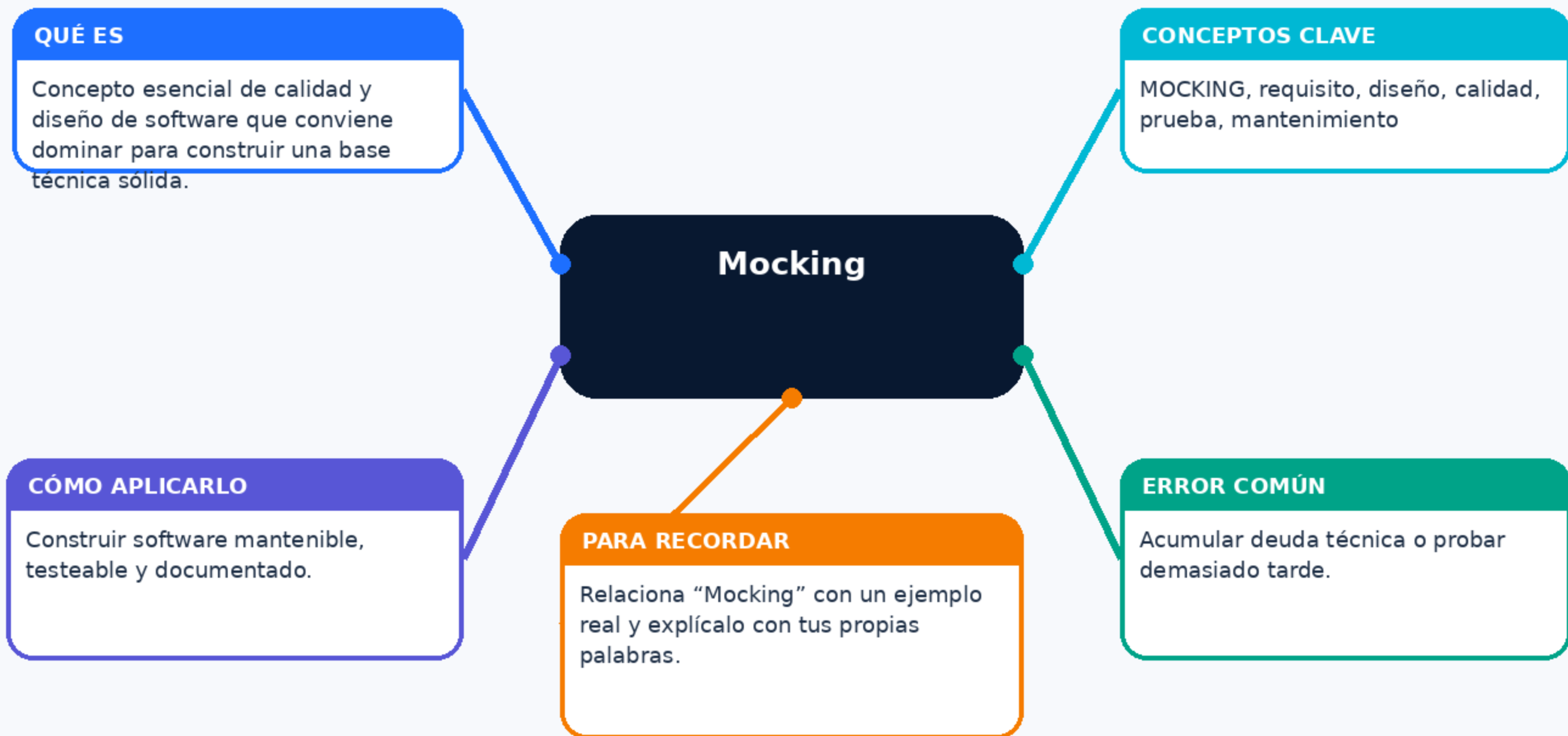
Historias de usuario



Mantenimiento de software



Mocking



Patrones de diseño



Pruebas de aceptación

QUÉ ES

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

PRUEBAS, ACEPTACIÓN, requisito, diseño, calidad, prueba

Pruebas de aceptación

CÓMO APLICARLO

Construir software mantenible, testeable y documentado.

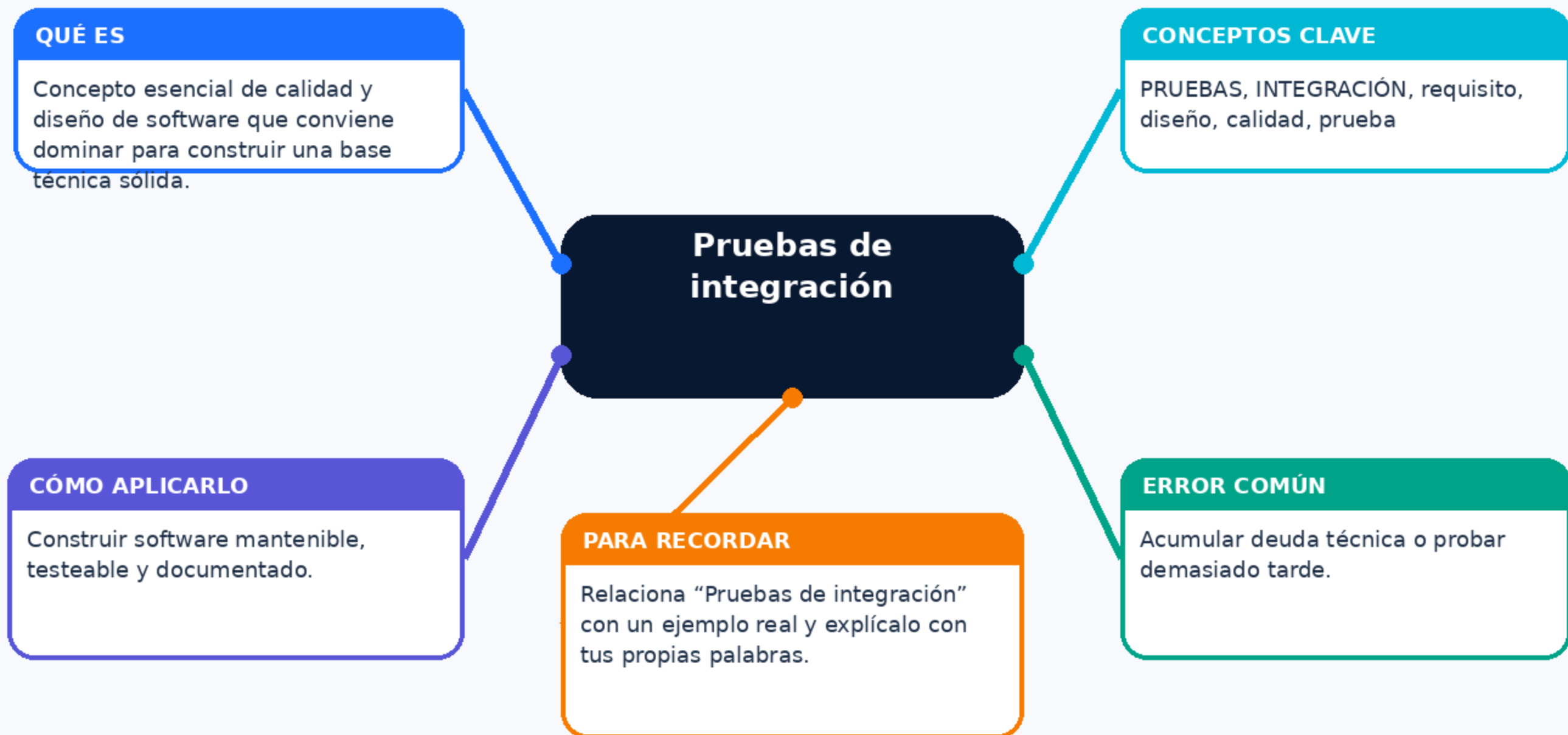
PARA RECORDAR

Relaciona "Pruebas de aceptación" con un ejemplo real y explícalo con tus propias palabras.

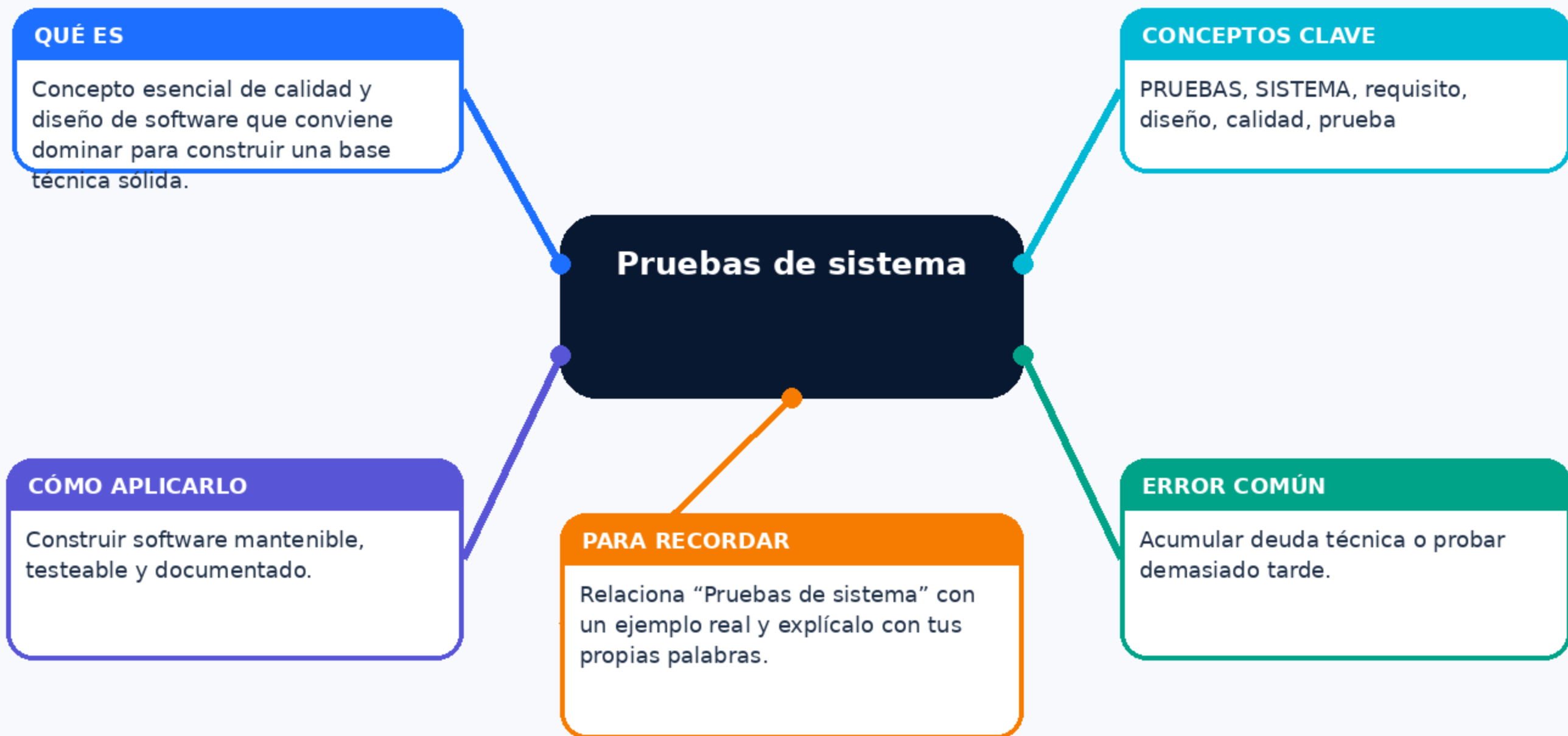
ERROR COMÚN

Acumular deuda técnica o probar demasiado tarde.

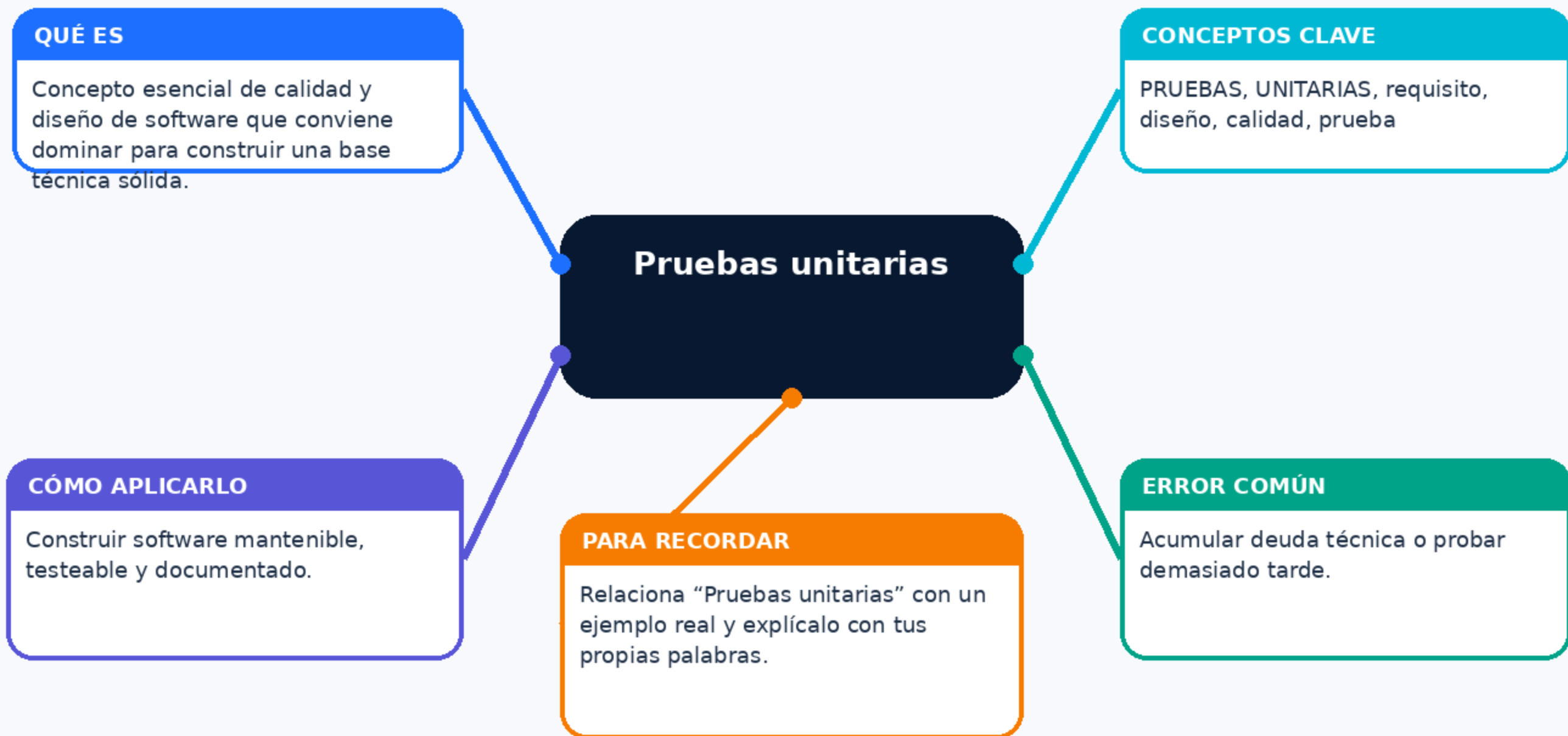
Pruebas de integración



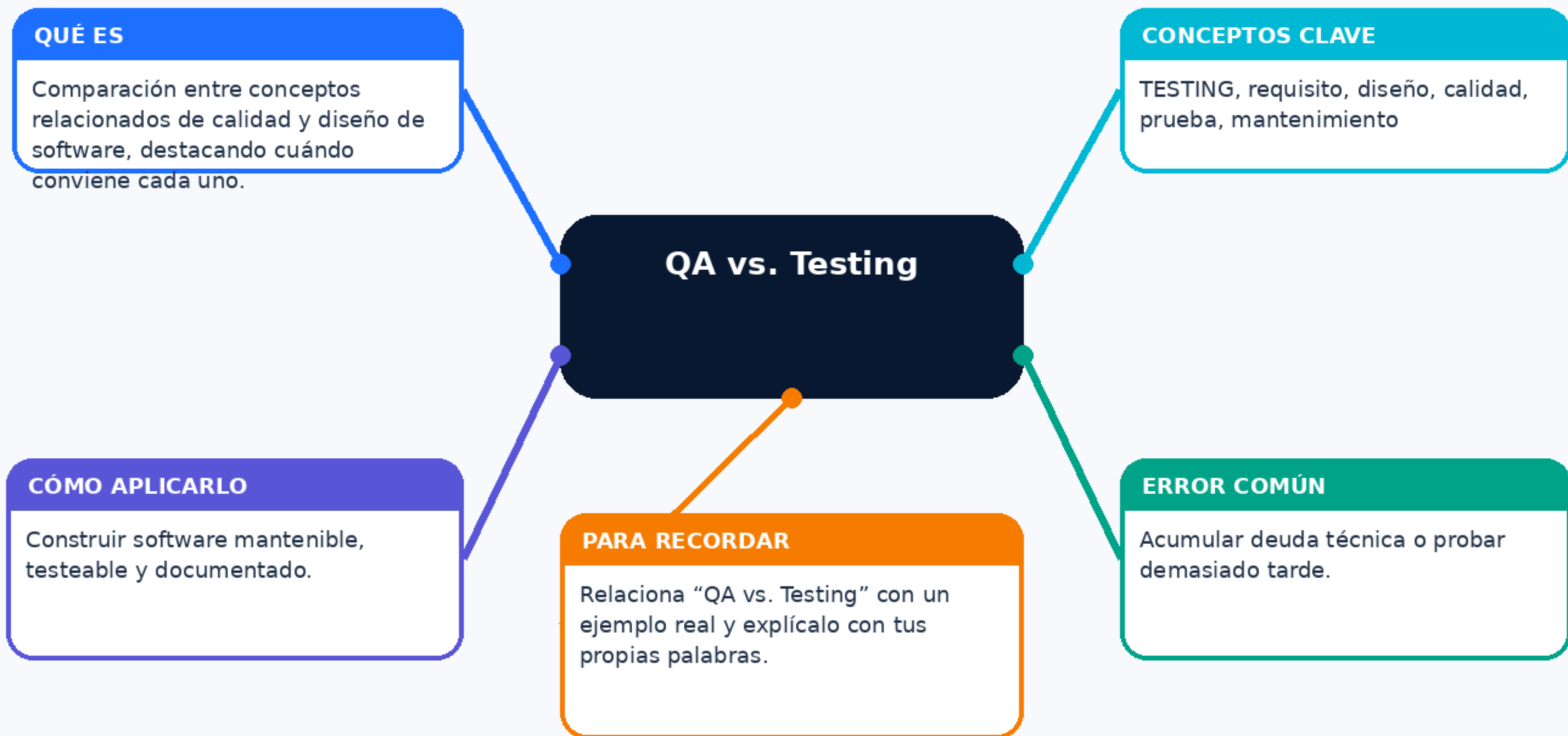
Pruebas de sistema



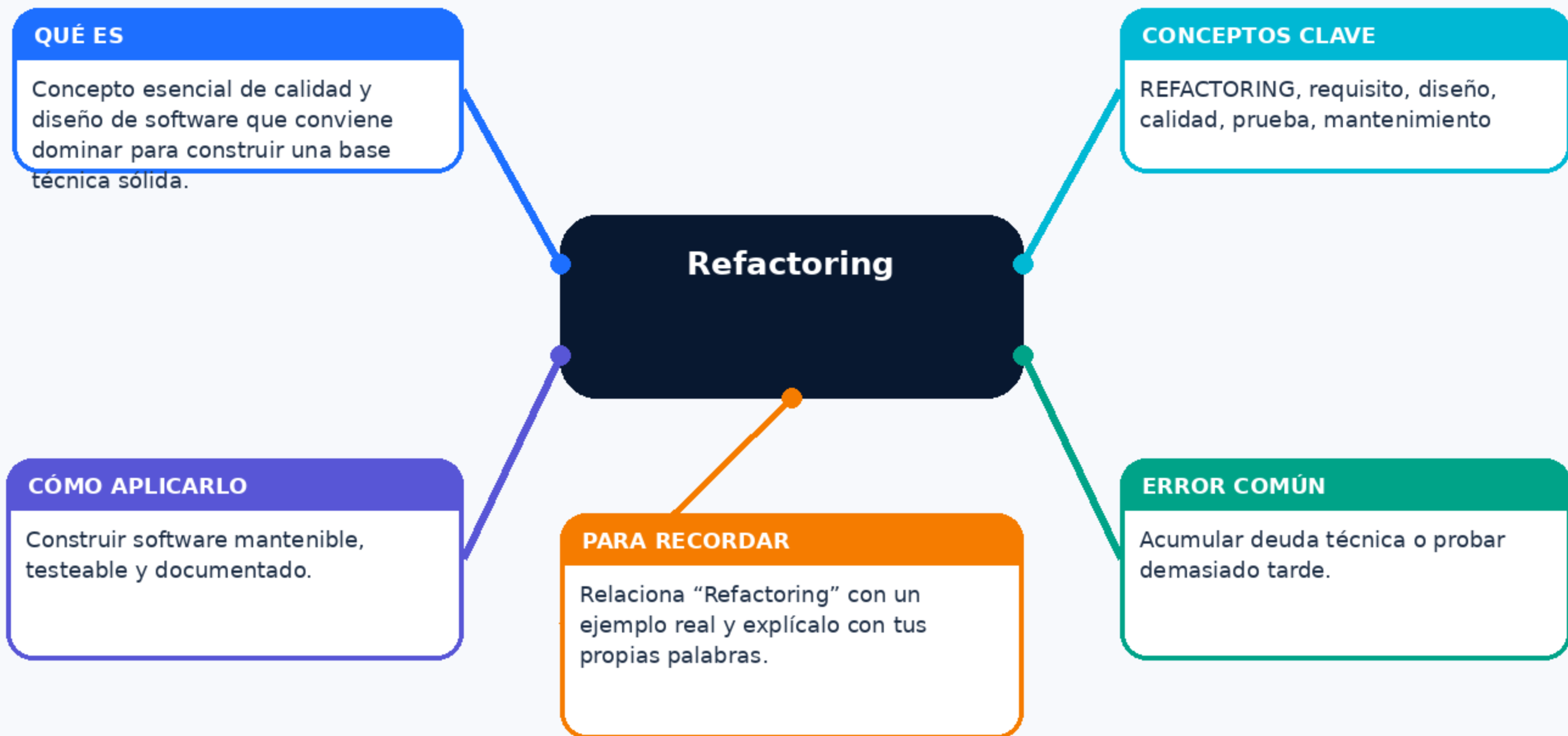
Pruebas unitarias



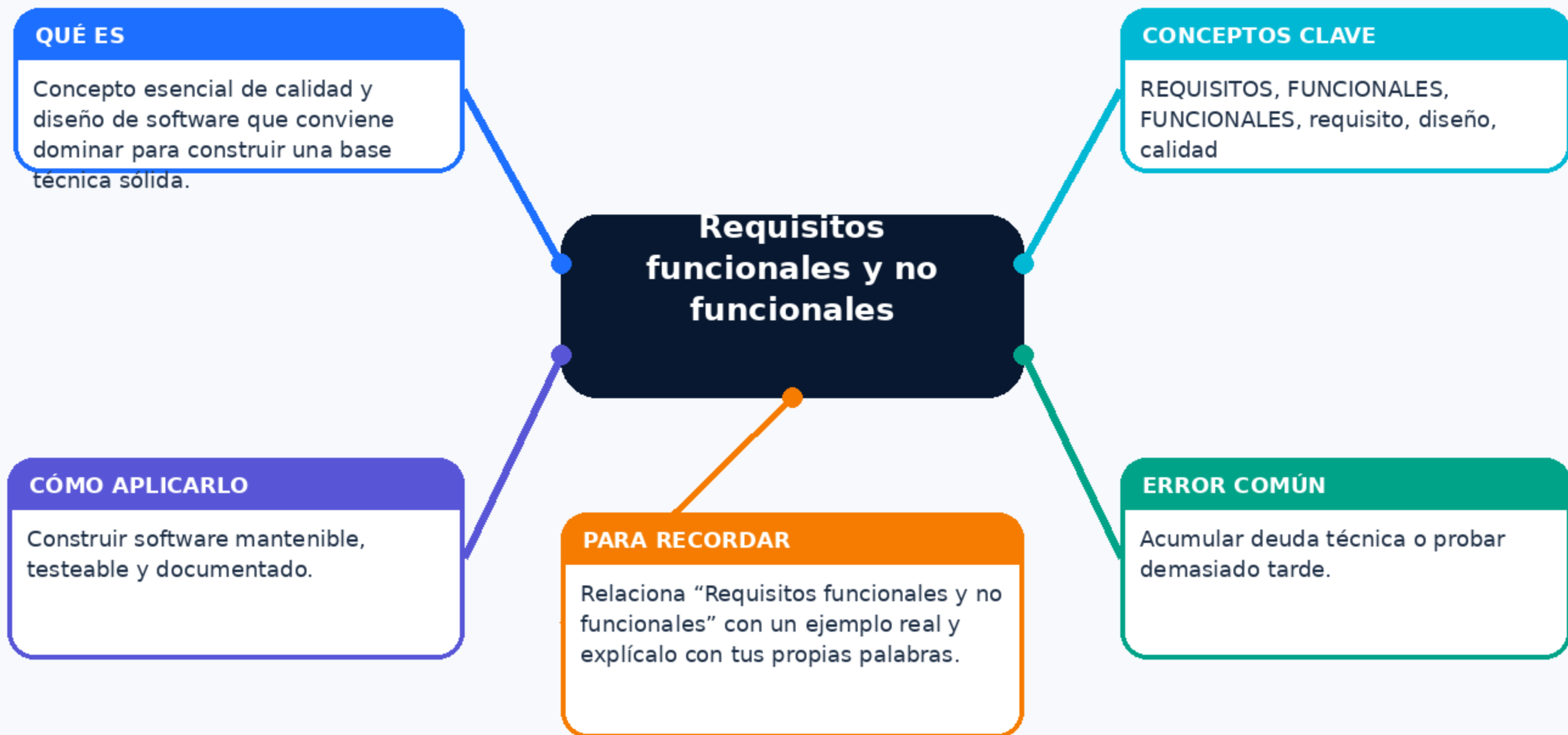
QA vs. Testing



Refactoring



Requisitos funcionales y no funcionales



SOLID

QUÉ ES

Concepto esencial de calidad y diseño de software que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

SOLID, requisito, diseño, calidad, prueba, mantenimiento

SOLID

CÓMO APLICARLO

Construir software mantenible, testeable y documentado.

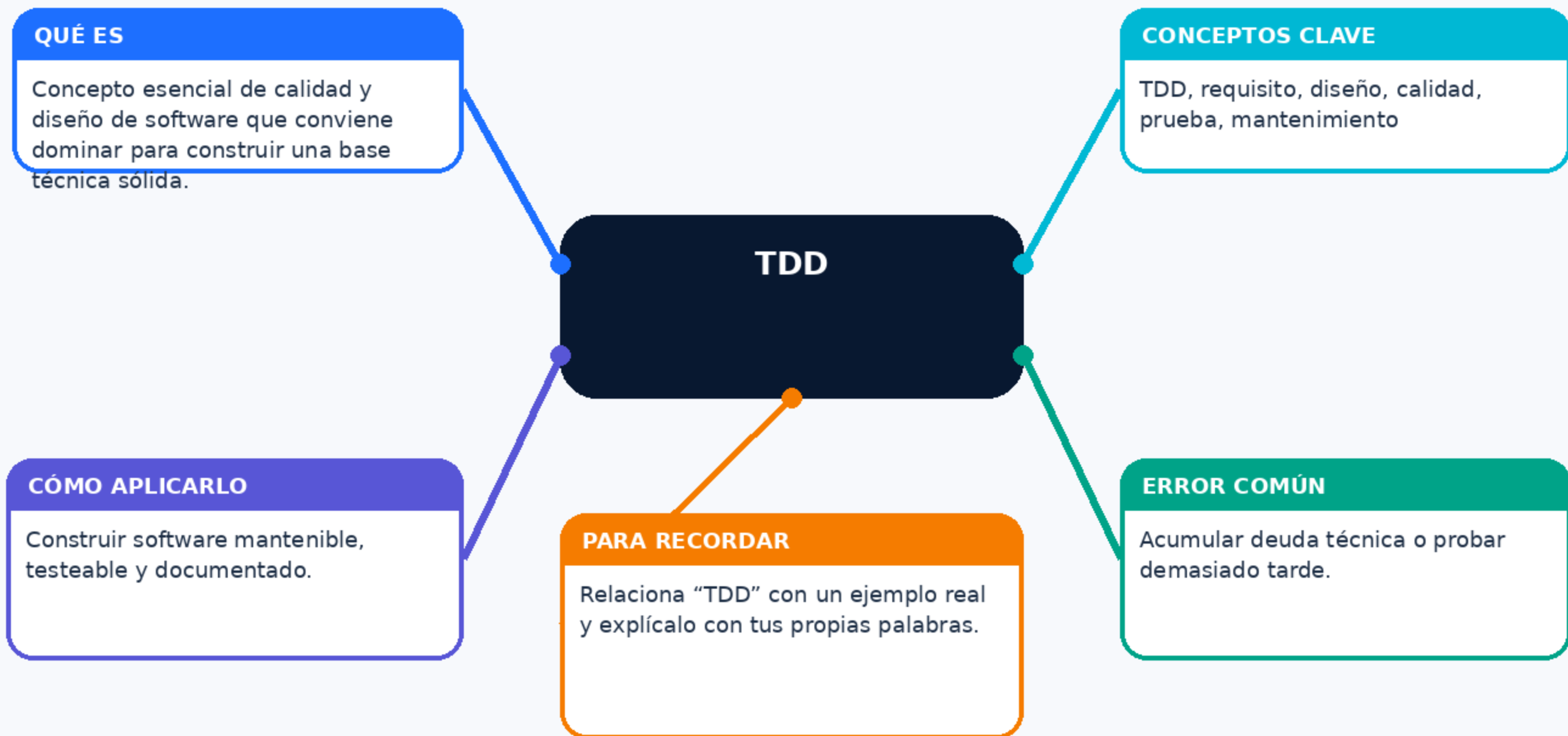
PARA RECORDAR

Relaciona "SOLID" con un ejemplo real y explícalo con tus propias palabras.

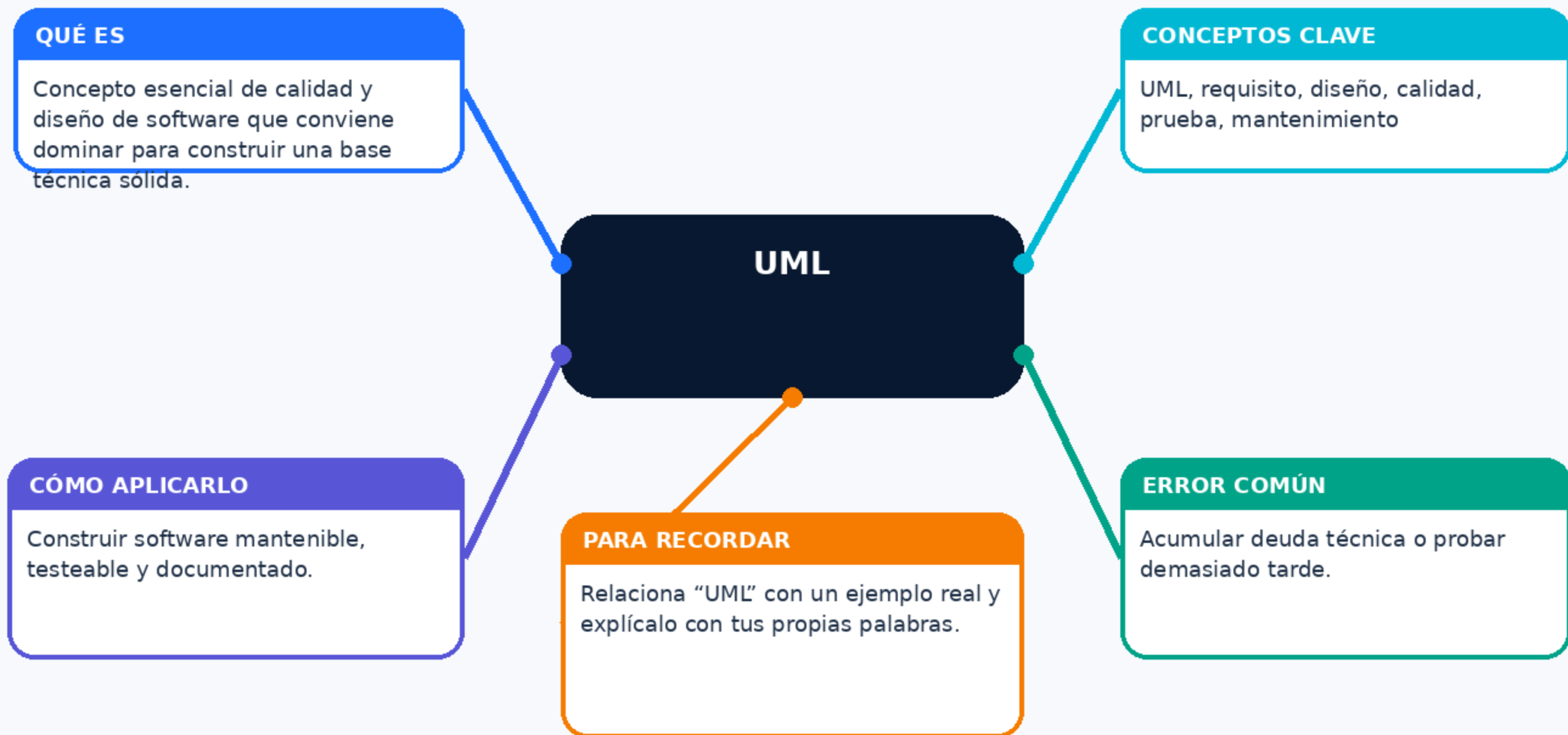
ERROR COMÚN

Acumular deuda técnica o probar demasiado tarde.

TDD



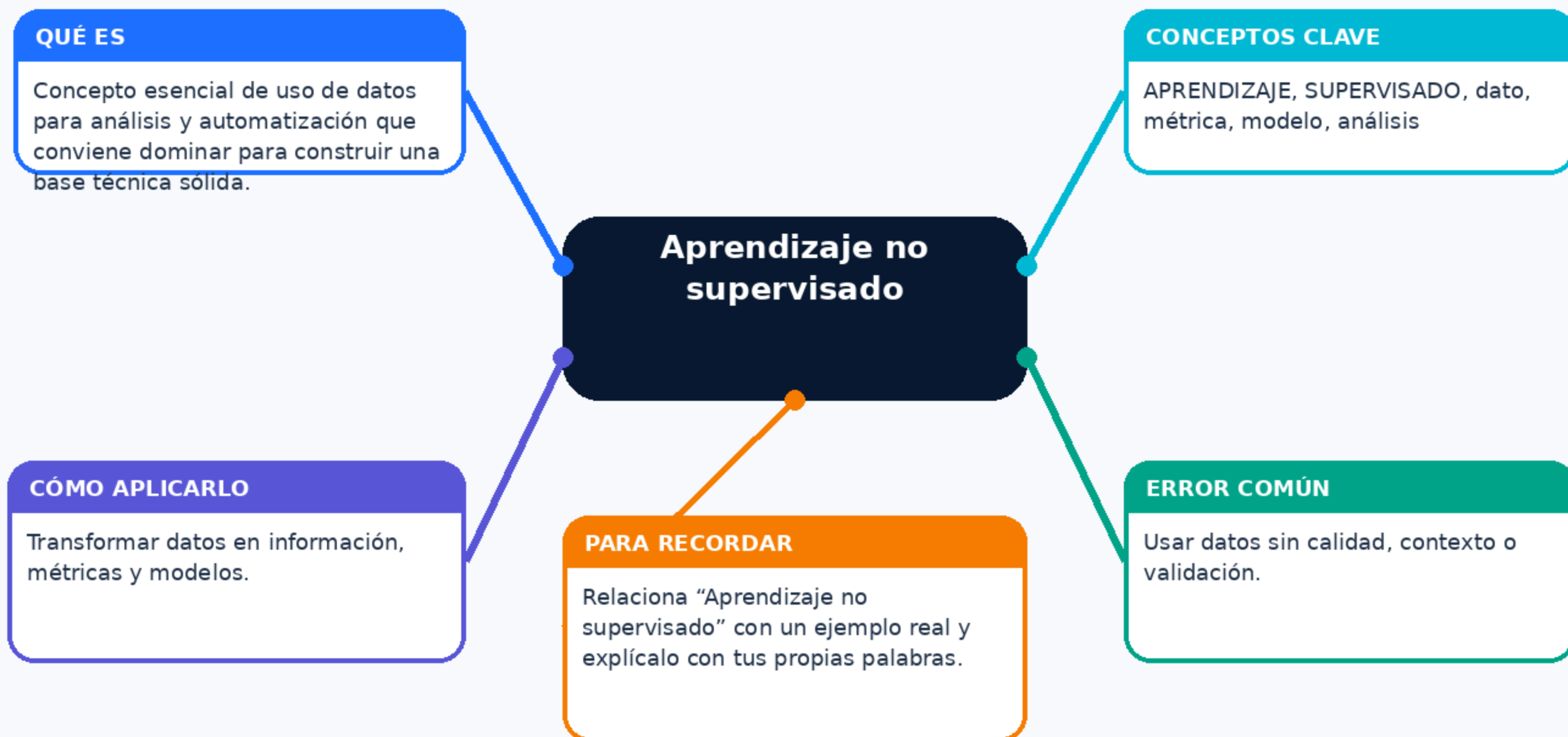
UML



Versionado semántico



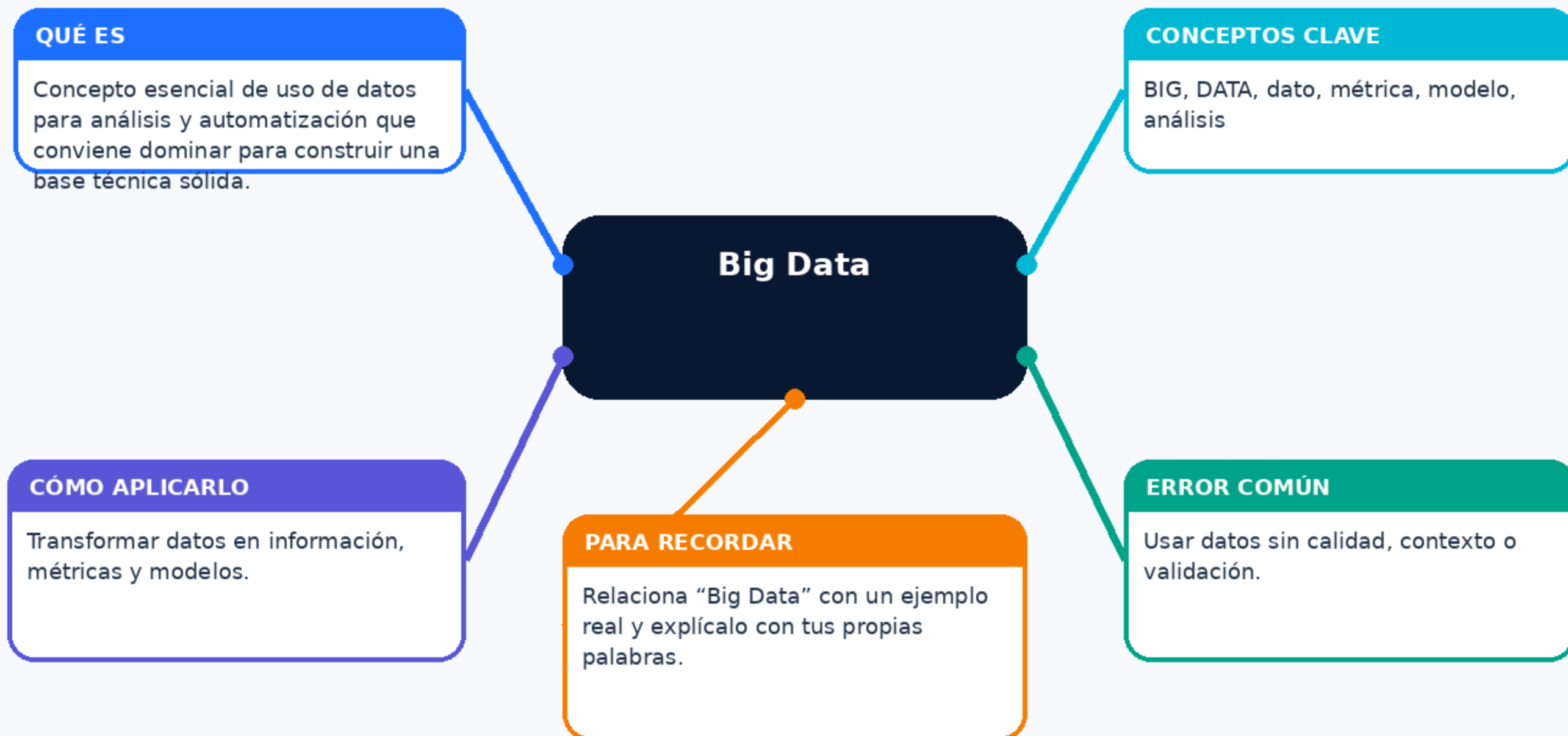
Aprendizaje no supervisado



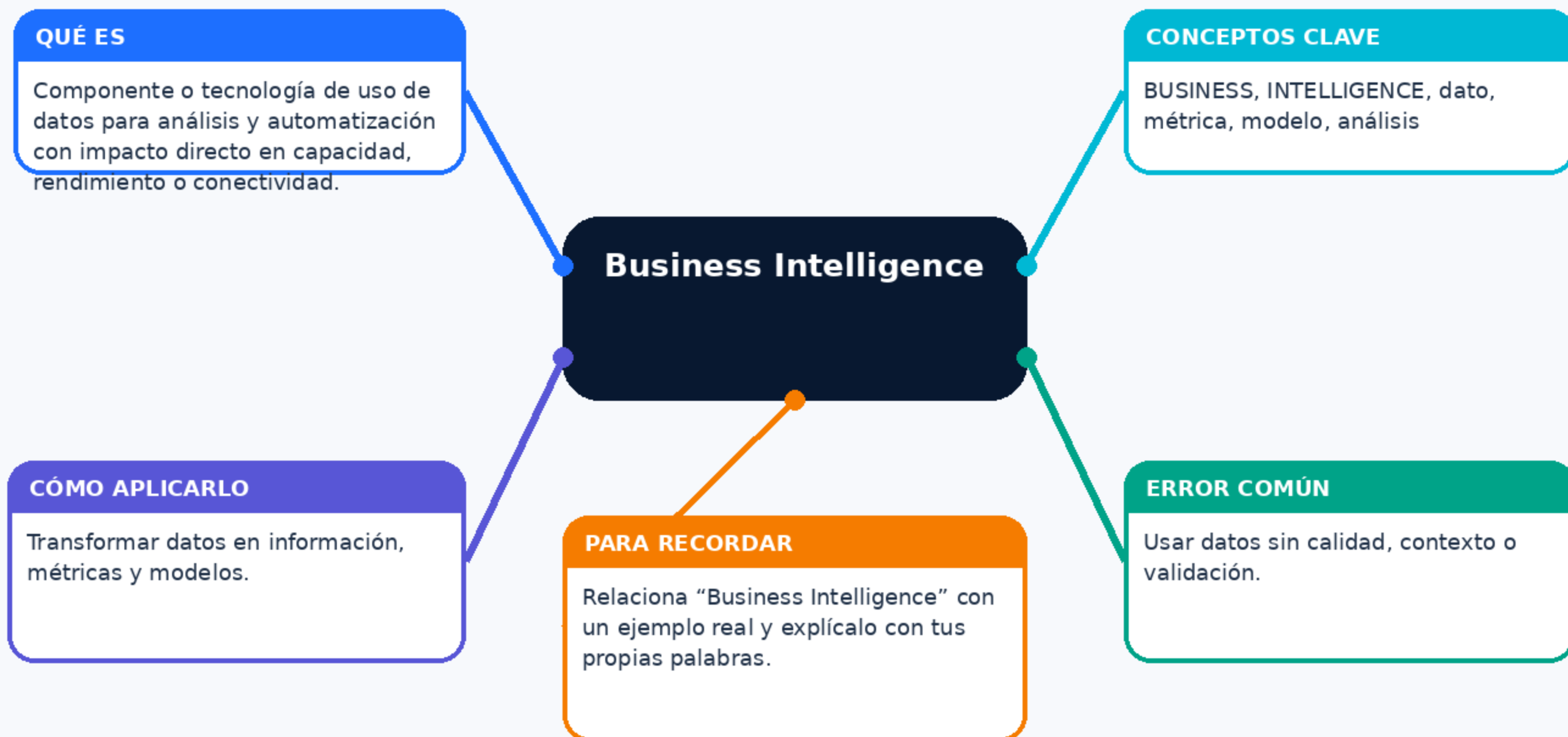
Aprendizaje supervisado



Big Data



Business Intelligence



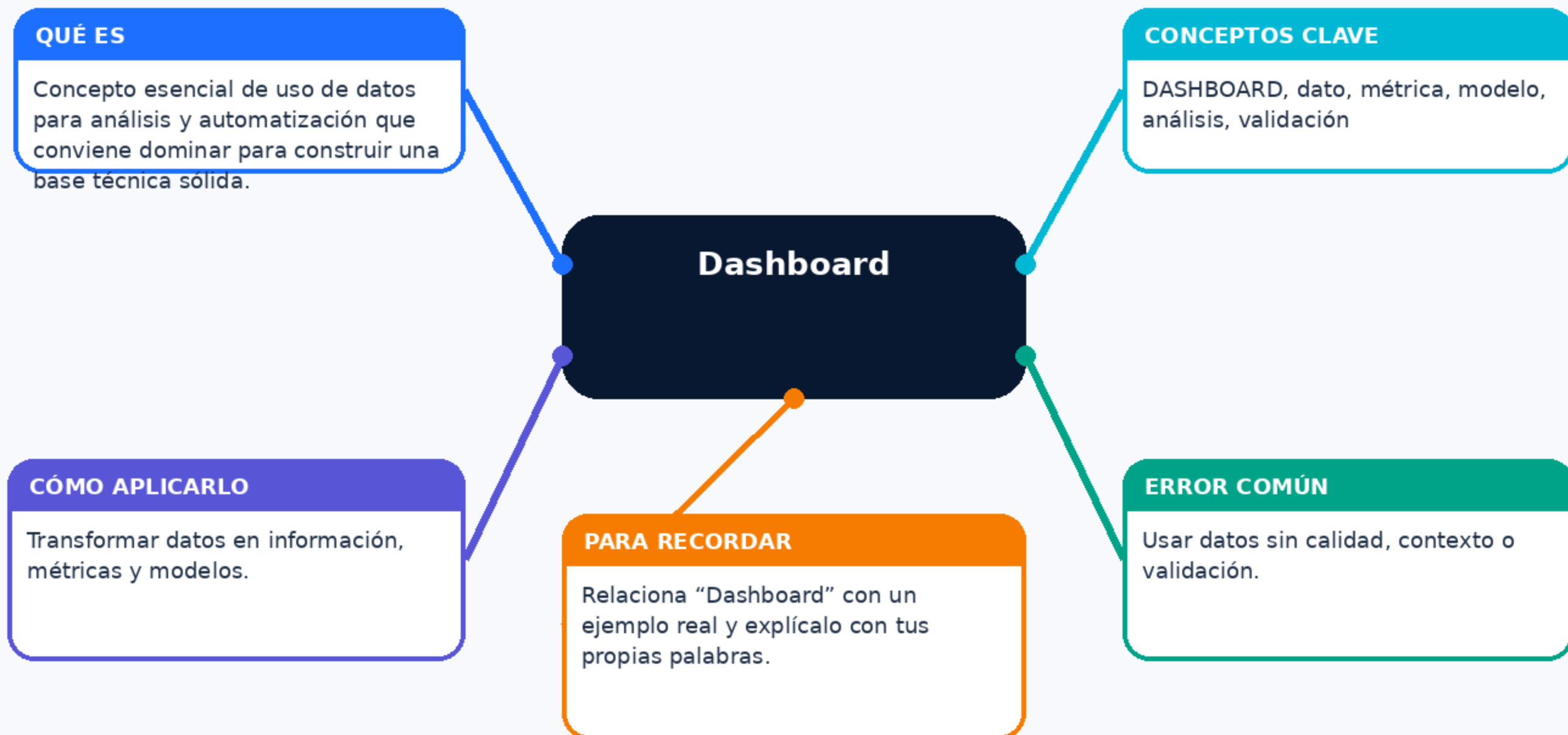
Clasificación



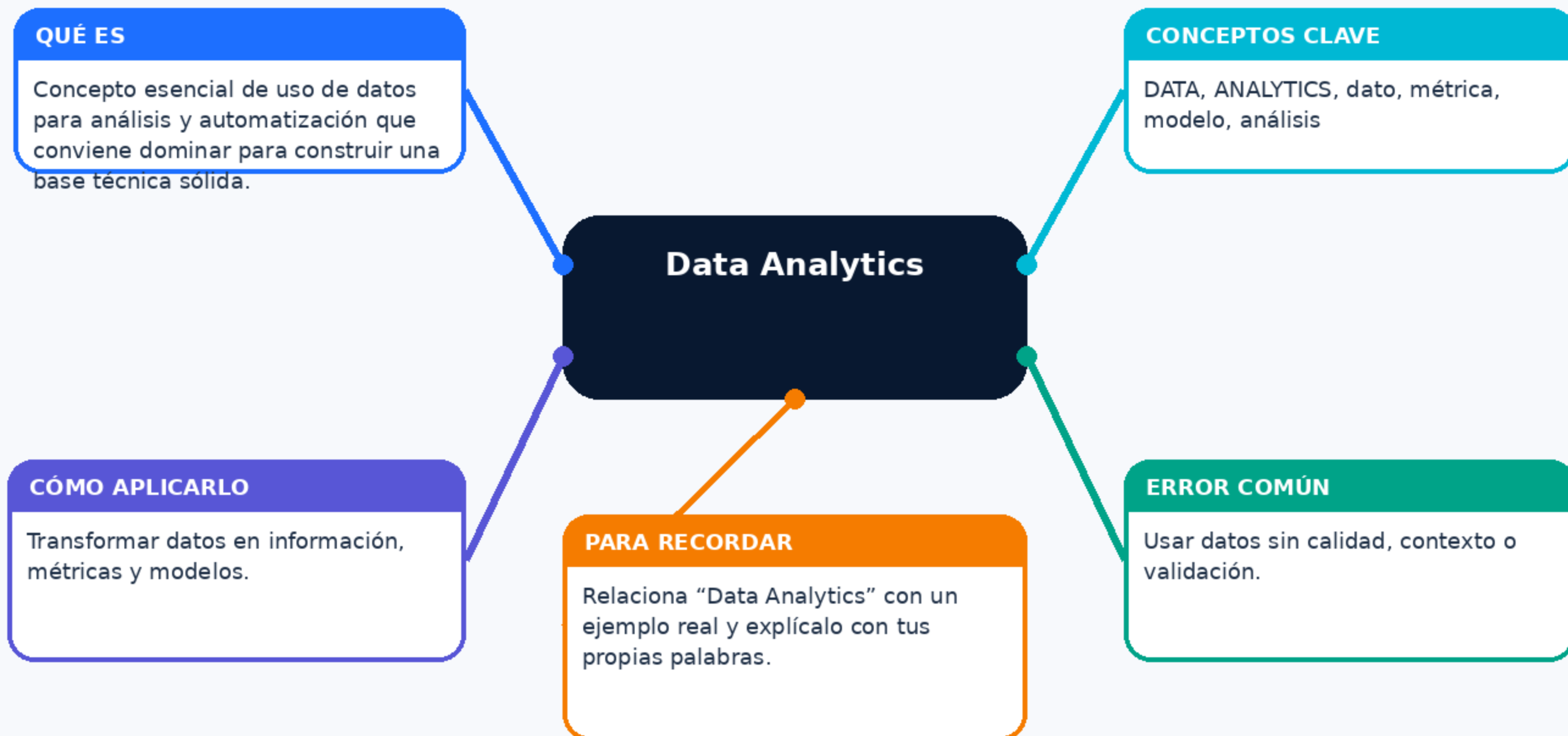
Clustering



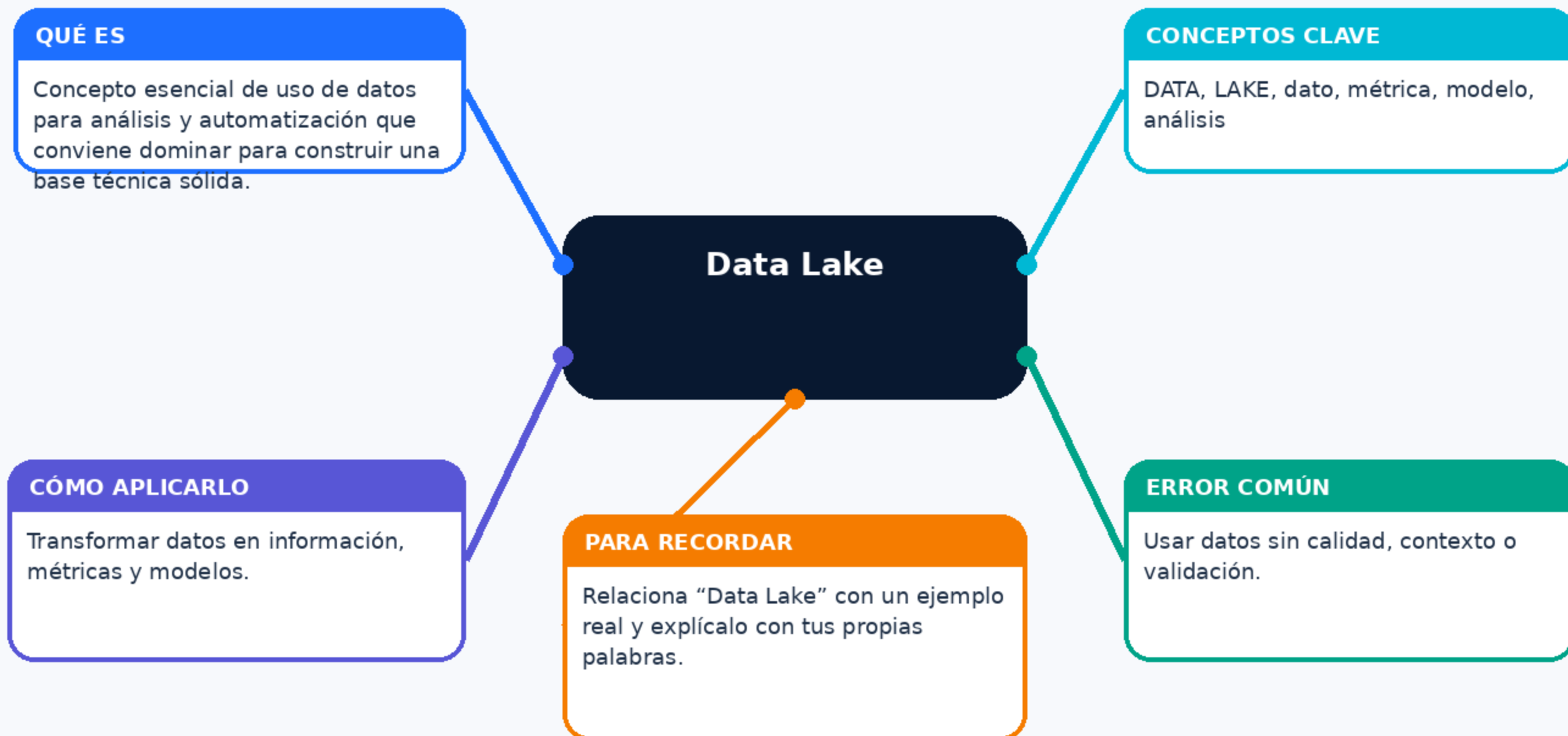
Dashboard



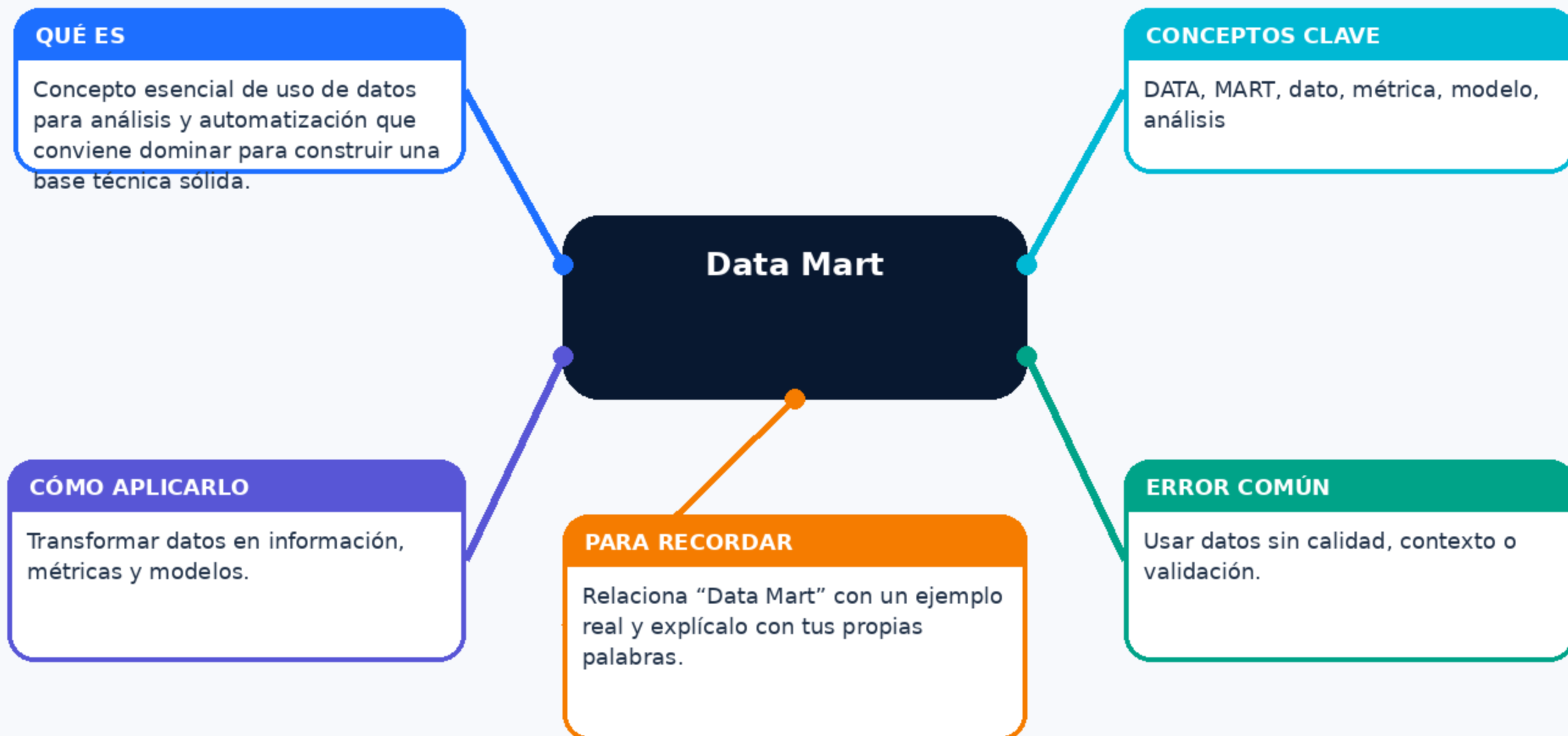
Data Analytics



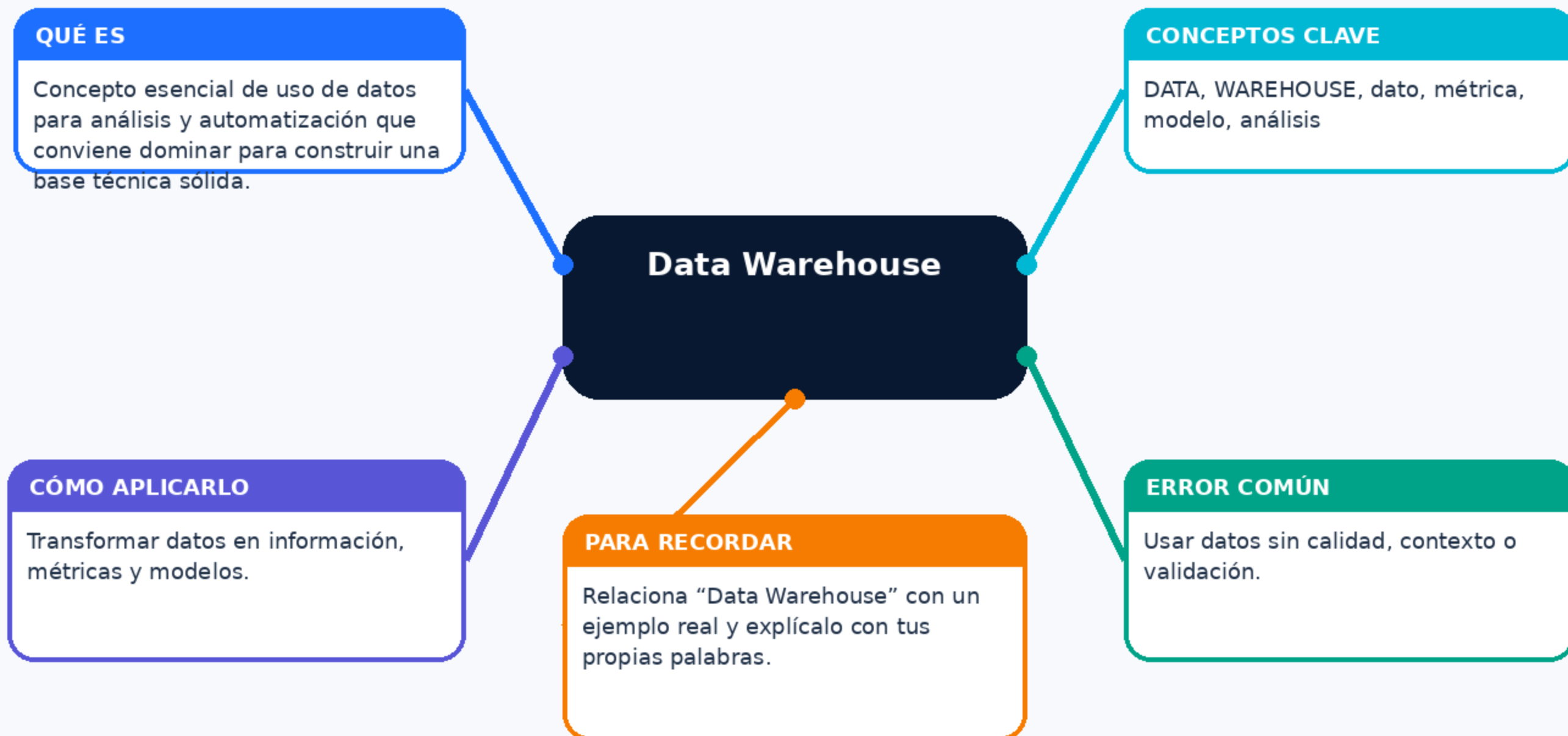
Data Lake



Data Mart



Data Warehouse



ELT

QUÉ ES

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ELT, dato, métrica, modelo, análisis, validación

ELT

CÓMO APLICARLO

Transformar datos en información, métricas y modelos.

PARA RECORDAR

Relaciona "ELT" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Usar datos sin calidad, contexto o validación.

ETL

QUÉ ES

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

ETL, dato, métrica, modelo, análisis, validación

ETL

CÓMO APLICARLO

Transformar datos en información, métricas y modelos.

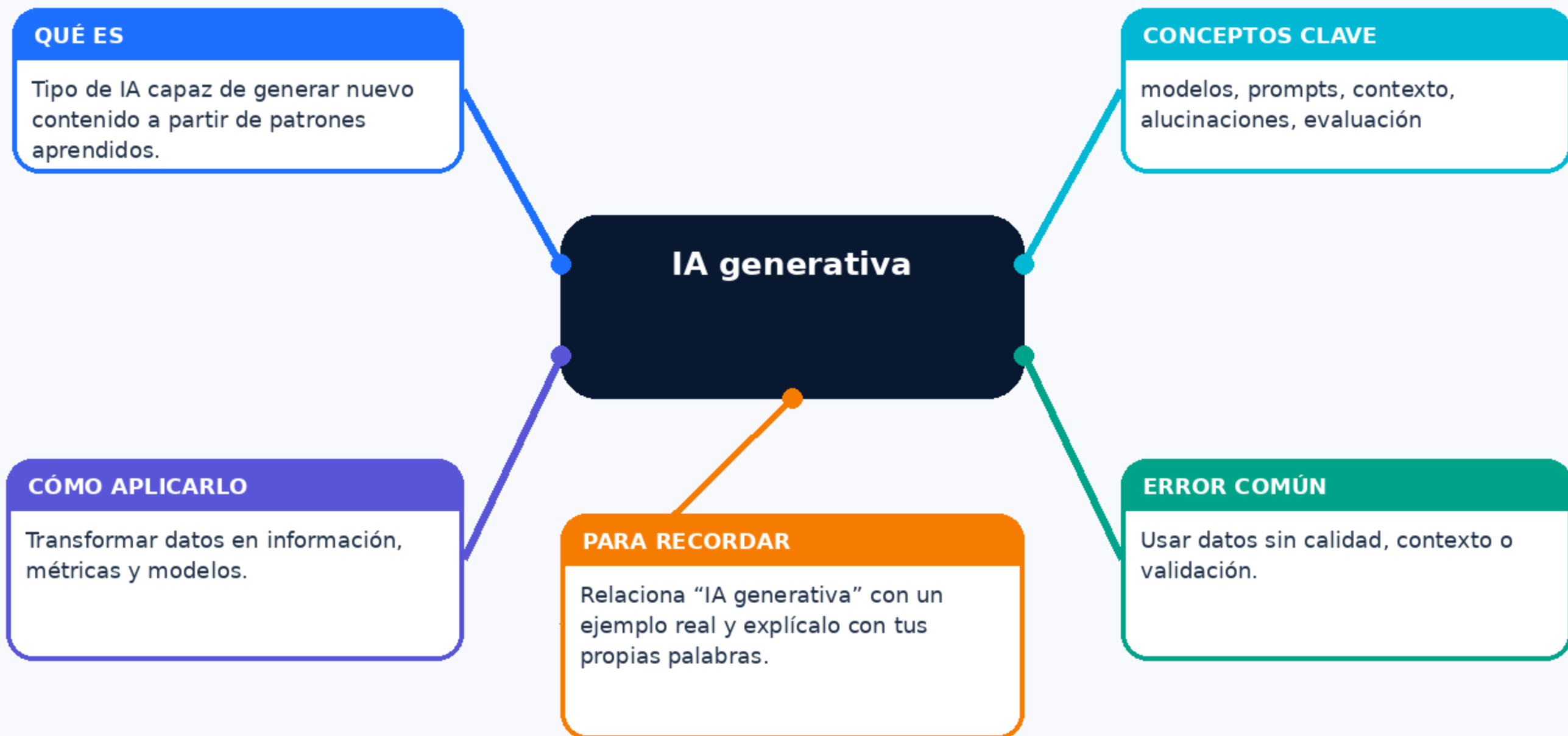
PARA RECORDAR

Relaciona "ETL" con un ejemplo real y explícalo con tus propias palabras.

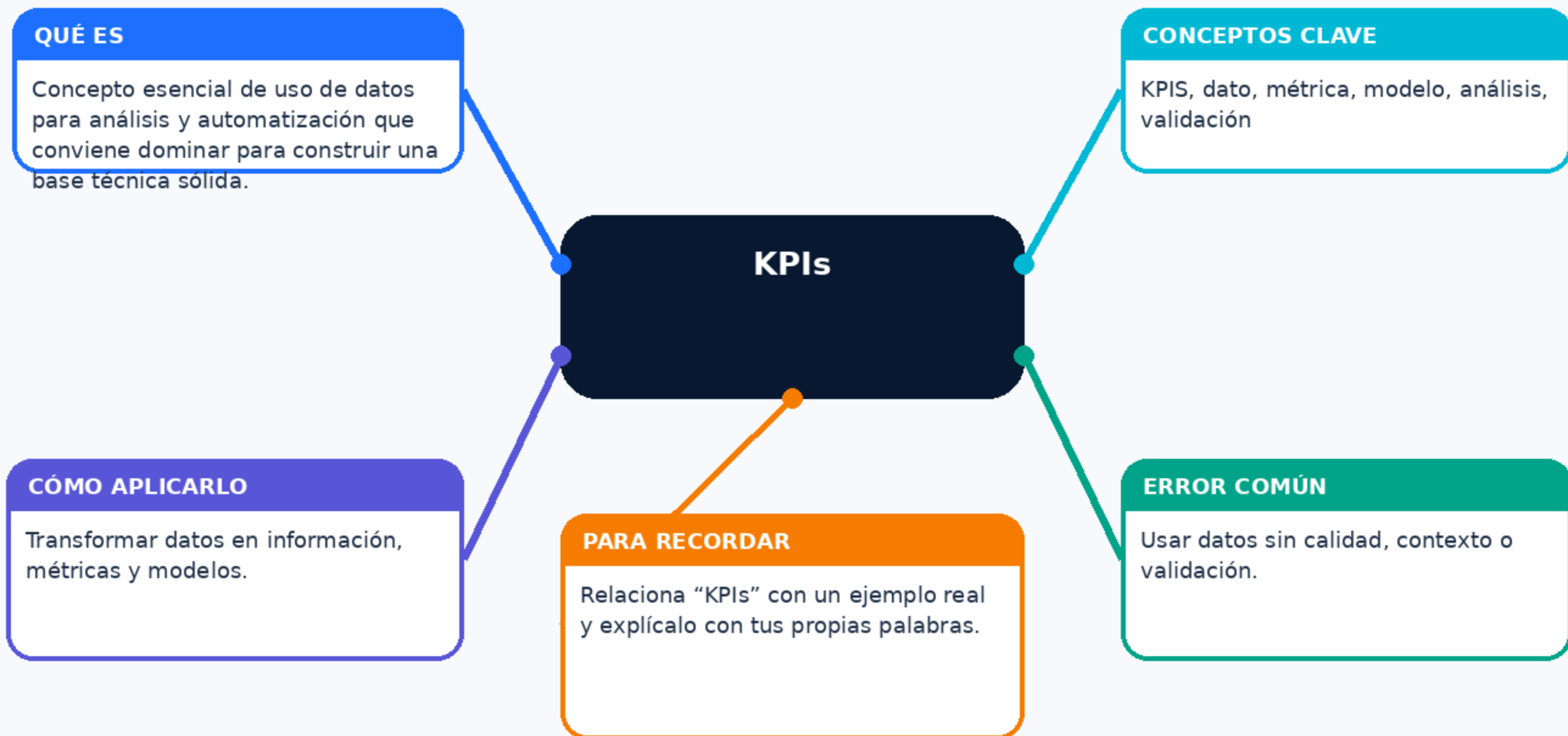
ERROR COMÚN

Usar datos sin calidad, contexto o validación.

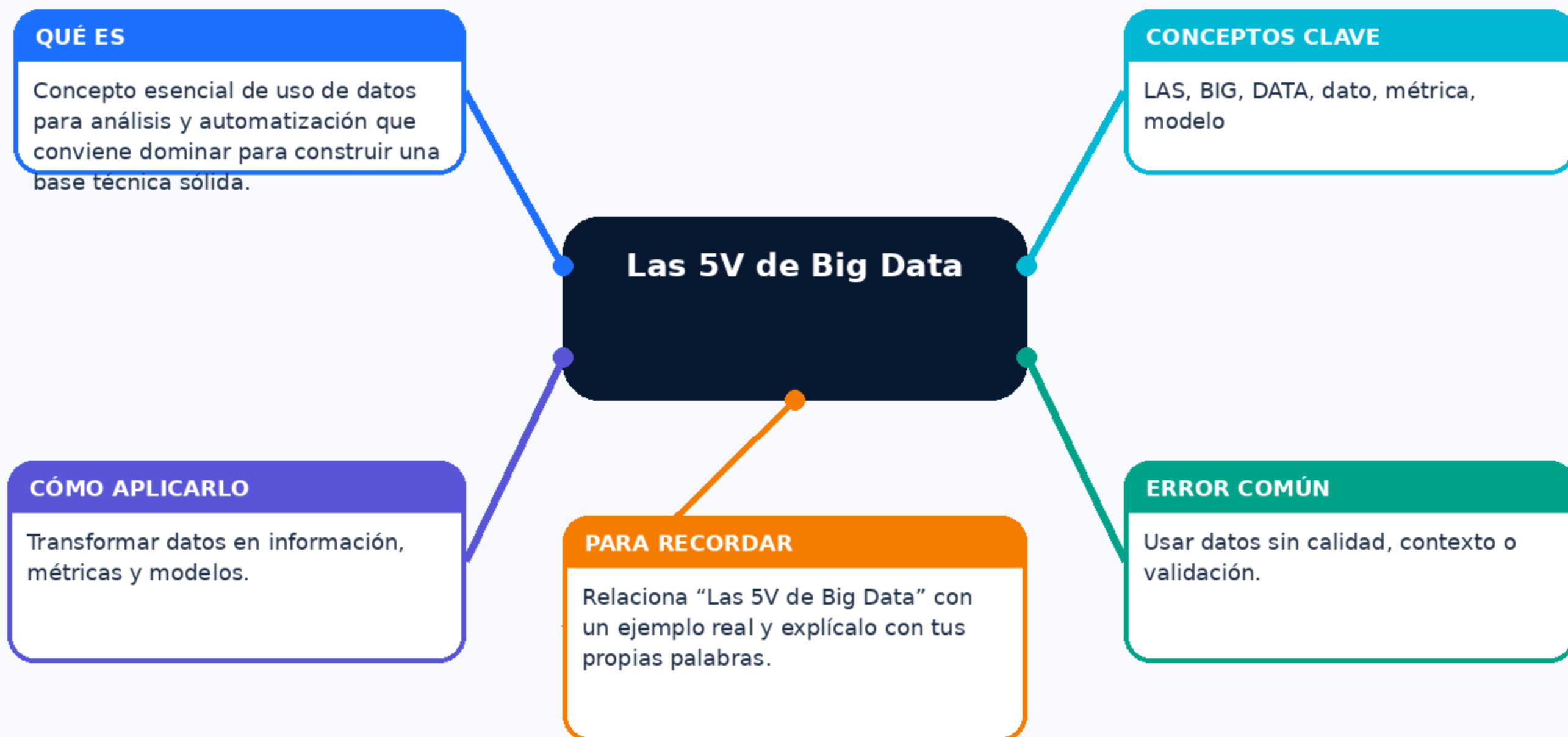
IA generativa



KPIs



Las 5V de Big Data



Machine Learning

QUÉ ES

Área de IA que crea modelos capaces de aprender patrones a partir de datos.

CONCEPTOS CLAVE

features, entrenamiento, validación, modelo, predicción

Machine Learning

CÓMO APLICARLO

Transformar datos en información, métricas y modelos.

PARA RECORDAR

Relaciona "Machine Learning" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Usar datos sin calidad, contexto o validación.

OLTP vs. OLAP

QUÉ ES

Comparación entre conceptos relacionados de uso de datos para análisis y automatización, destacando cuándo conviene cada uno.

CONCEPTOS CLAVE

OLTP, OLAP, dato, métrica, modelo, análisis

OLTP vs. OLAP

CÓMO APLICARLO

Transformar datos en información, métricas y modelos.

PARA RECORDAR

Relaciona "OLTP vs. OLAP" con un ejemplo real y explícalo con tus propias palabras.

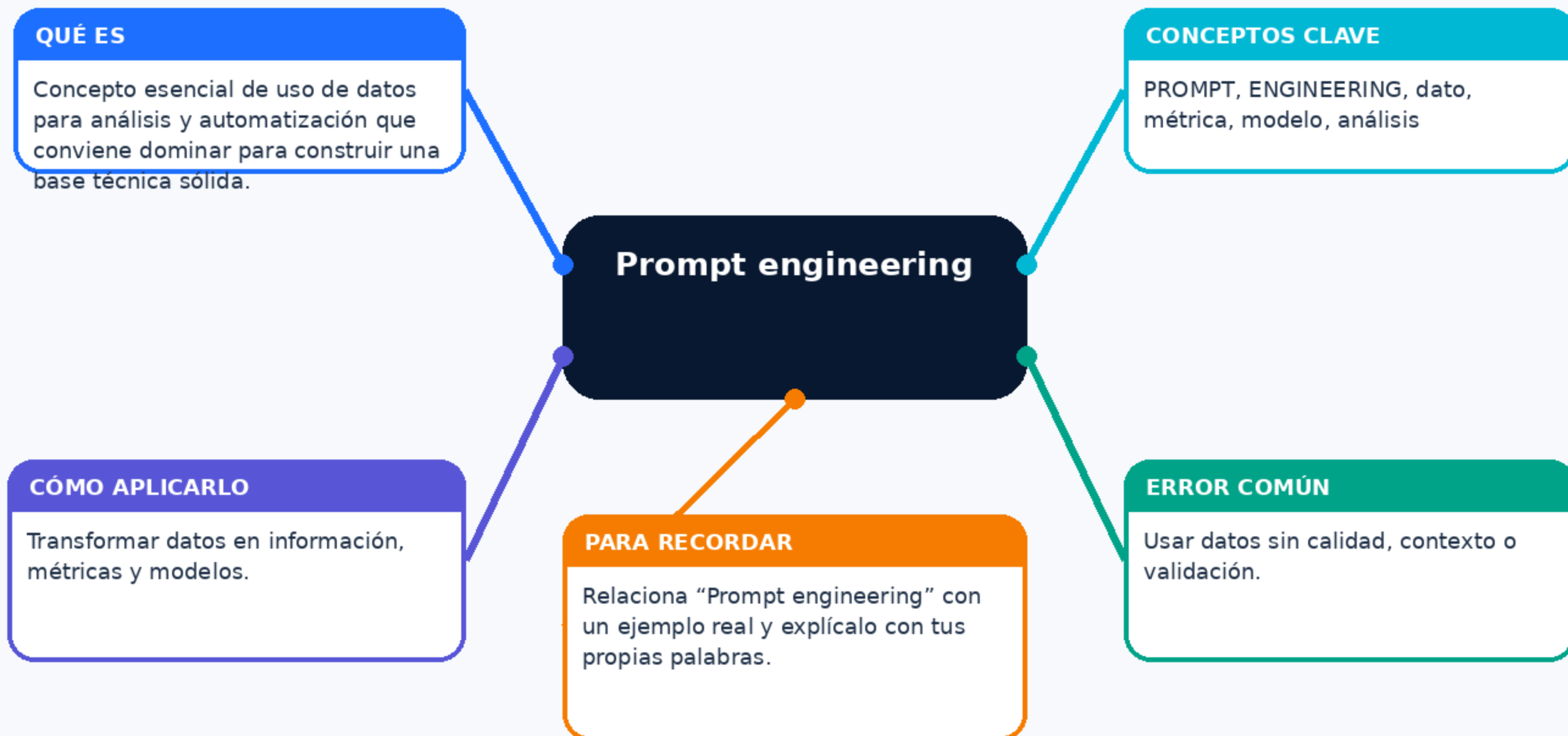
ERROR COMÚN

Usar datos sin calidad, contexto o validación.

Overfitting y underfitting



Prompt engineering



Redes neuronales

QUÉ ES

Concepto esencial de uso de datos para análisis y automatización que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

REDES, NEURONALES, dato, métrica, modelo, análisis

Redes neuronales

CÓMO APLICARLO

Transformar datos en información, métricas y modelos.

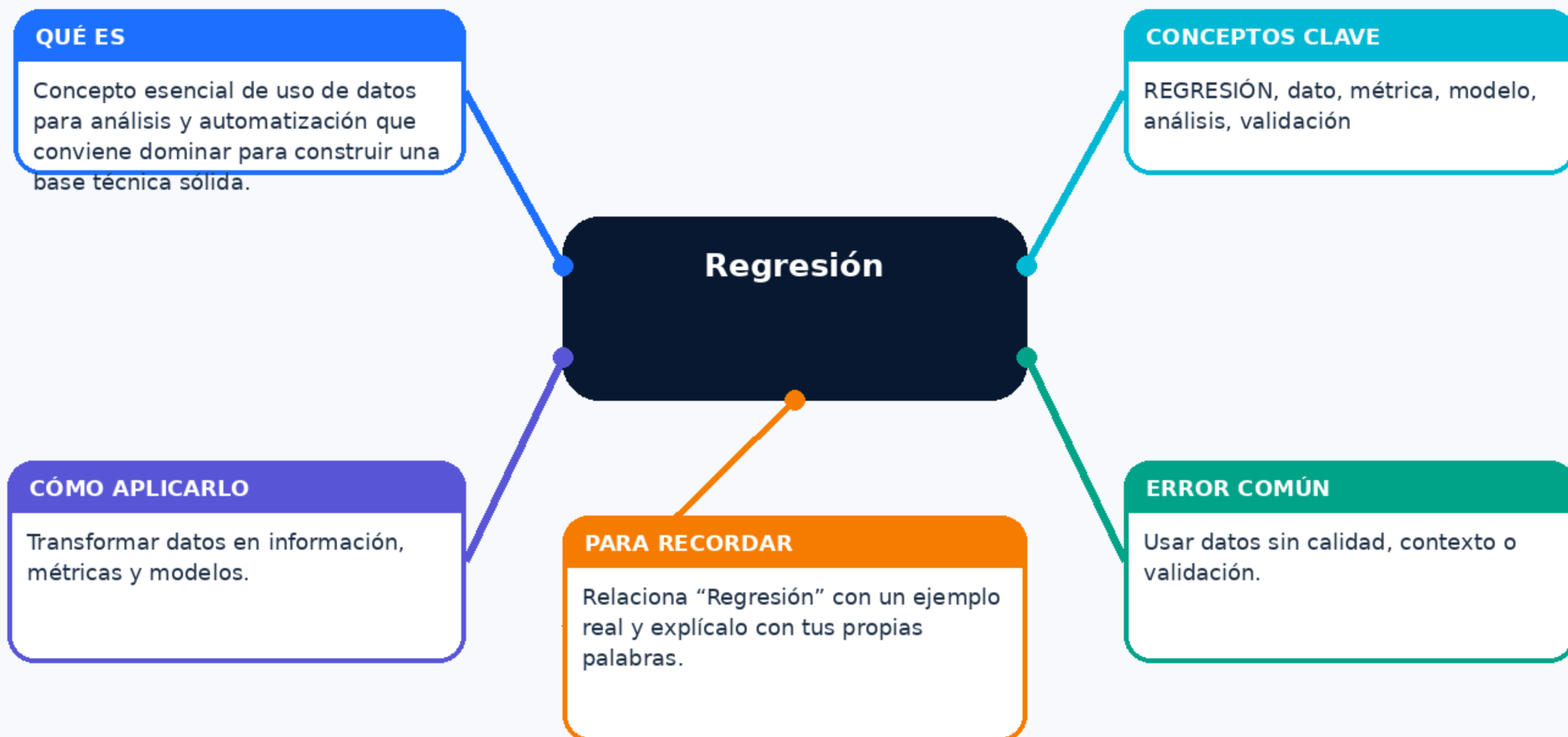
PARA RECORDAR

Relaciona "Redes neuronales" con un ejemplo real y explícalo con tus propias palabras.

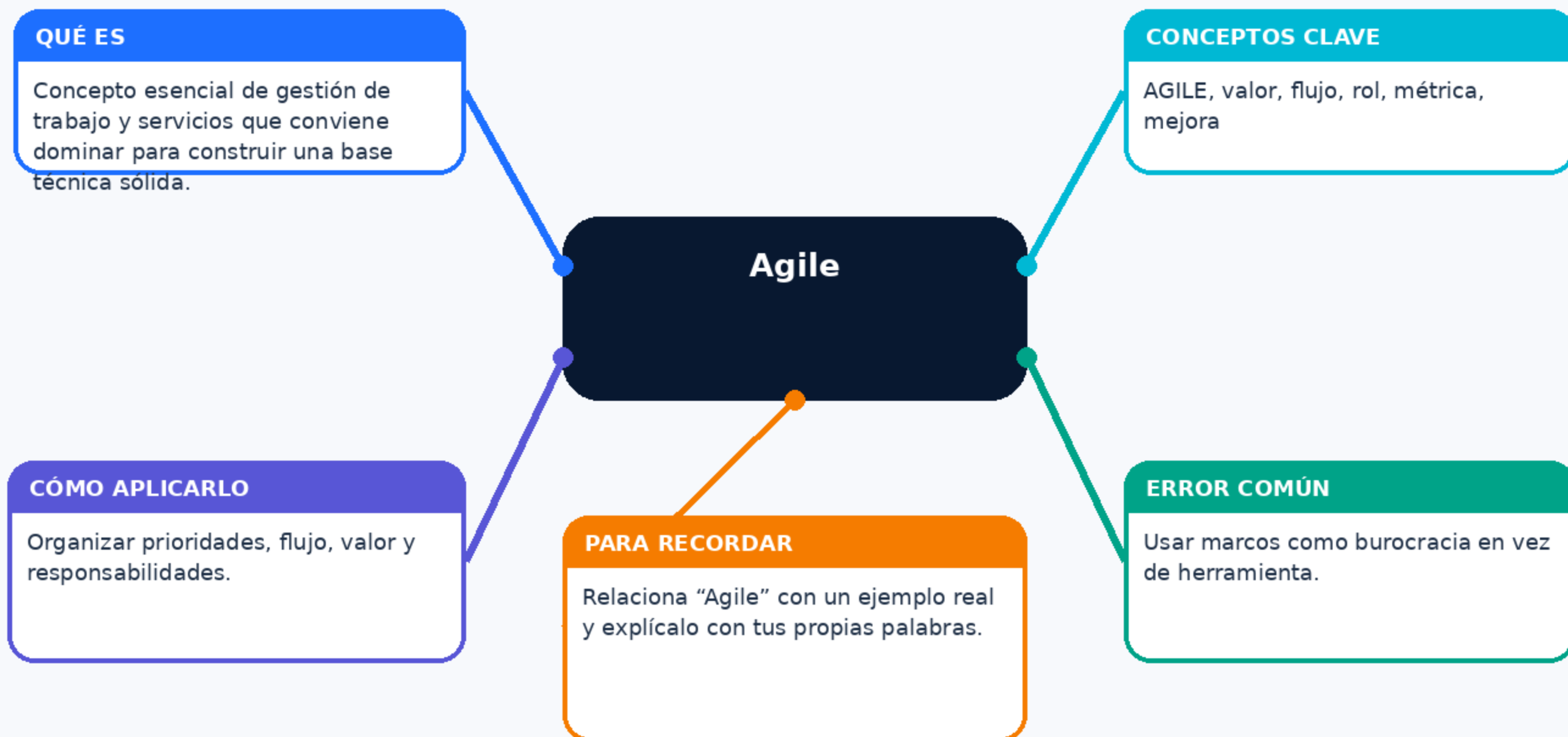
ERROR COMÚN

Usar datos sin calidad, contexto o validación.

Regresión



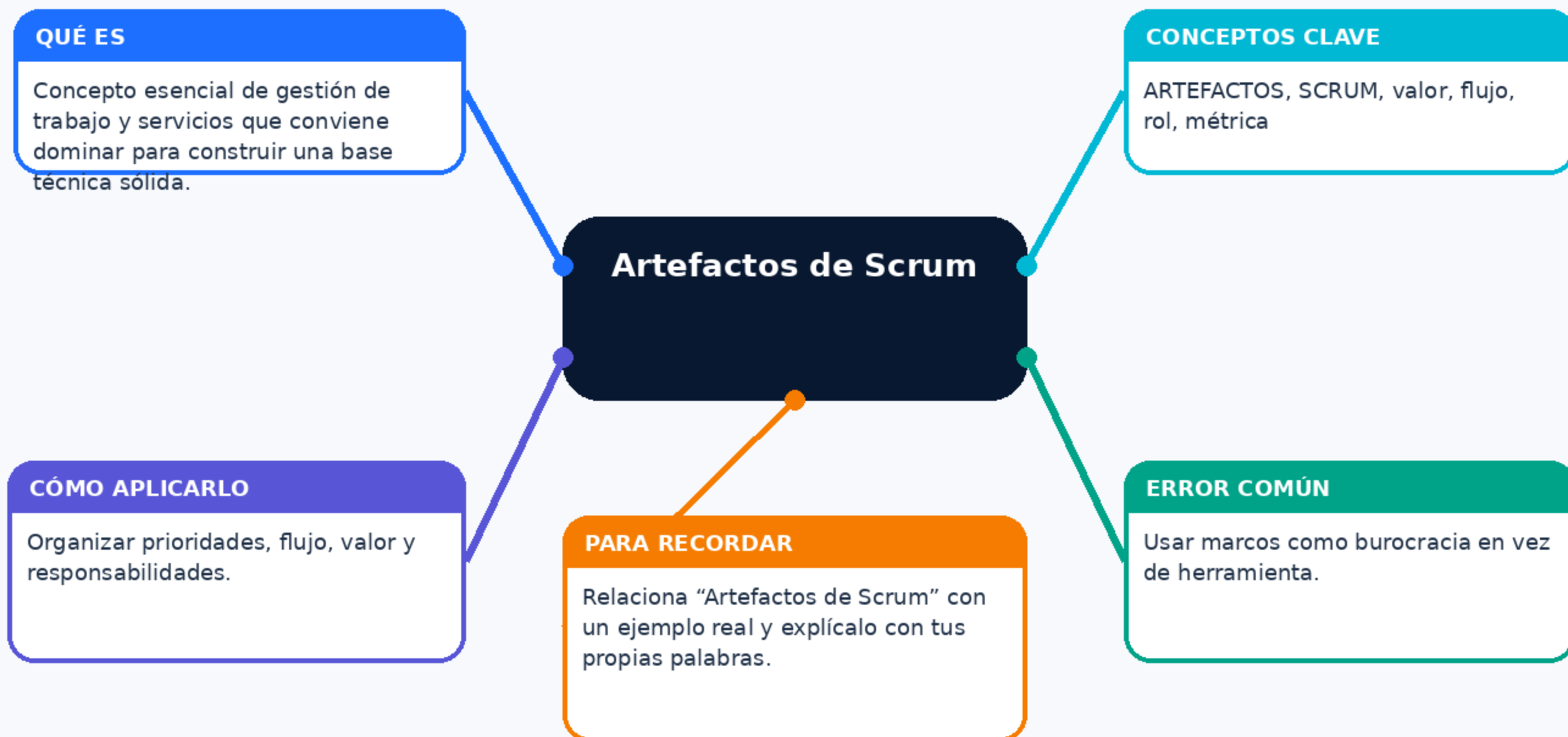
Agile



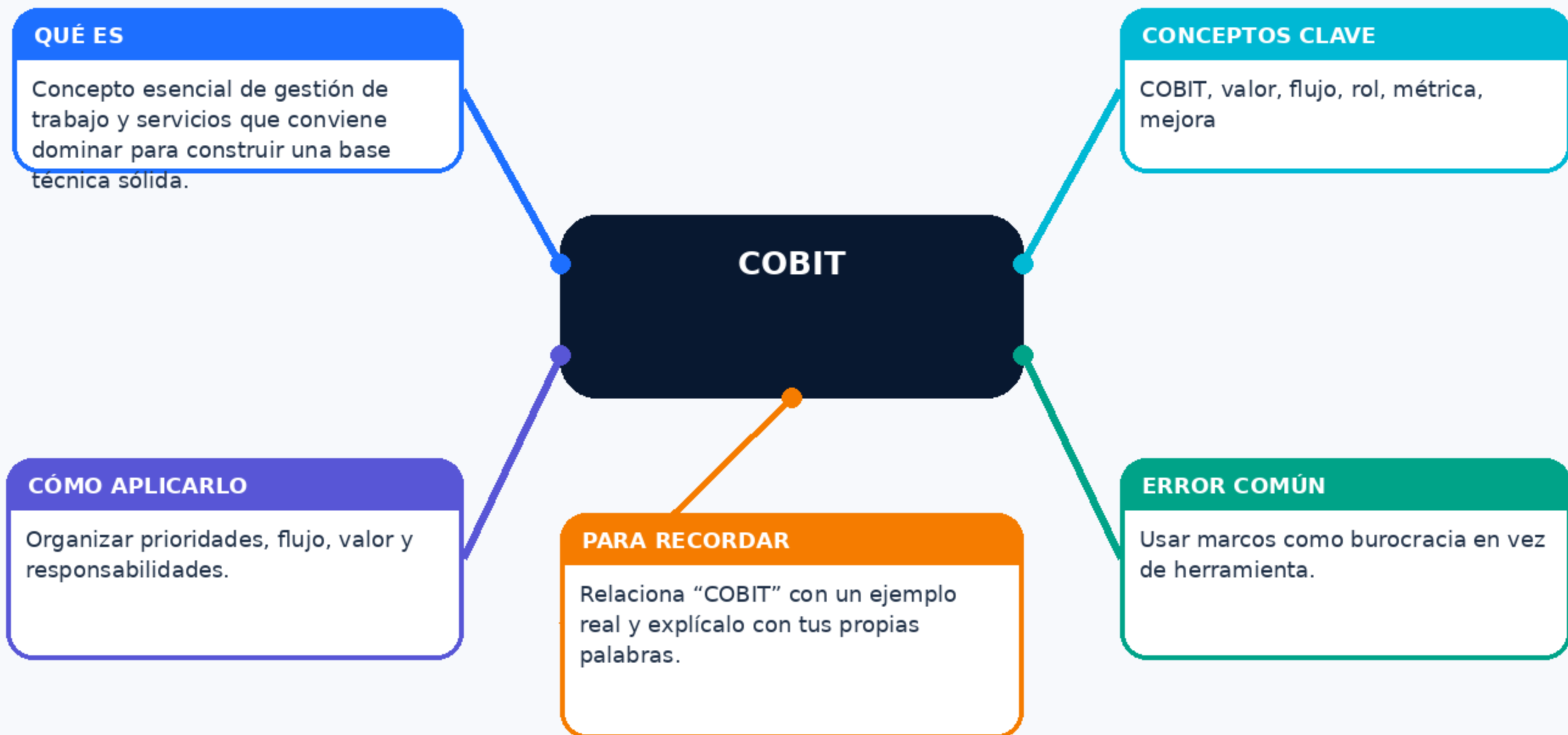
Alcance, tiempo y costo



Artefactos de Scrum



COBIT



Eventos de Scrum



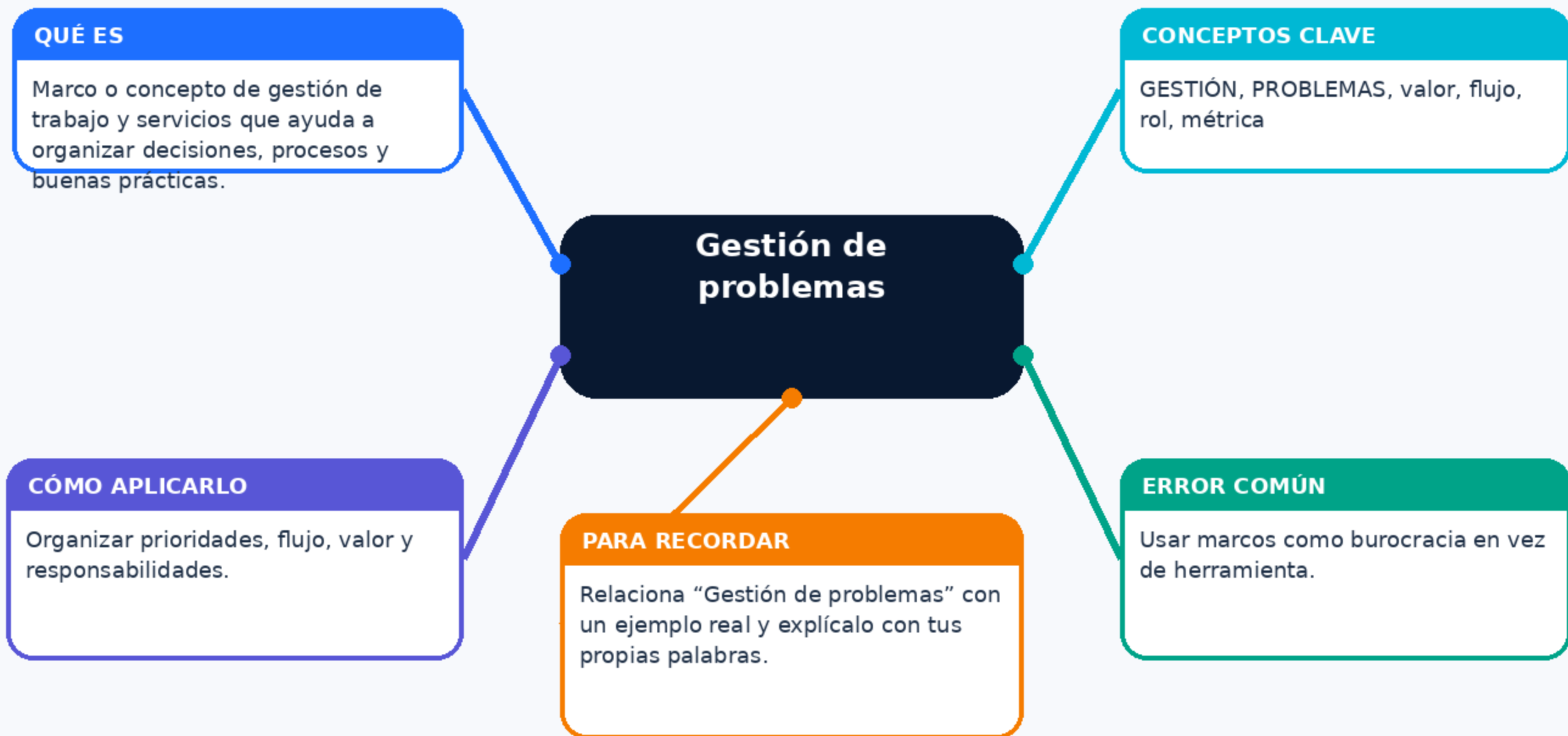
Gestión de cambios



Gestión de incidentes



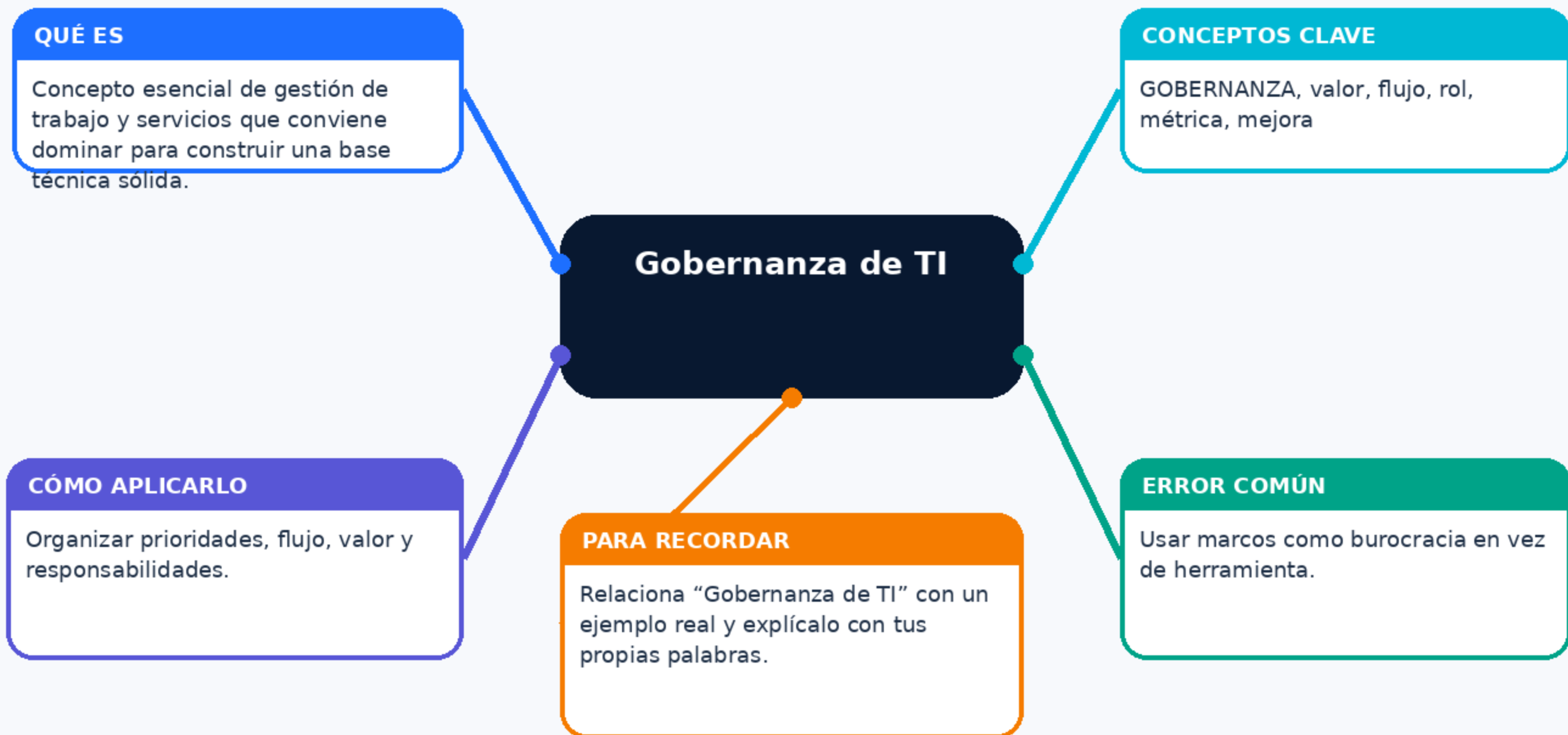
Gestión de problemas



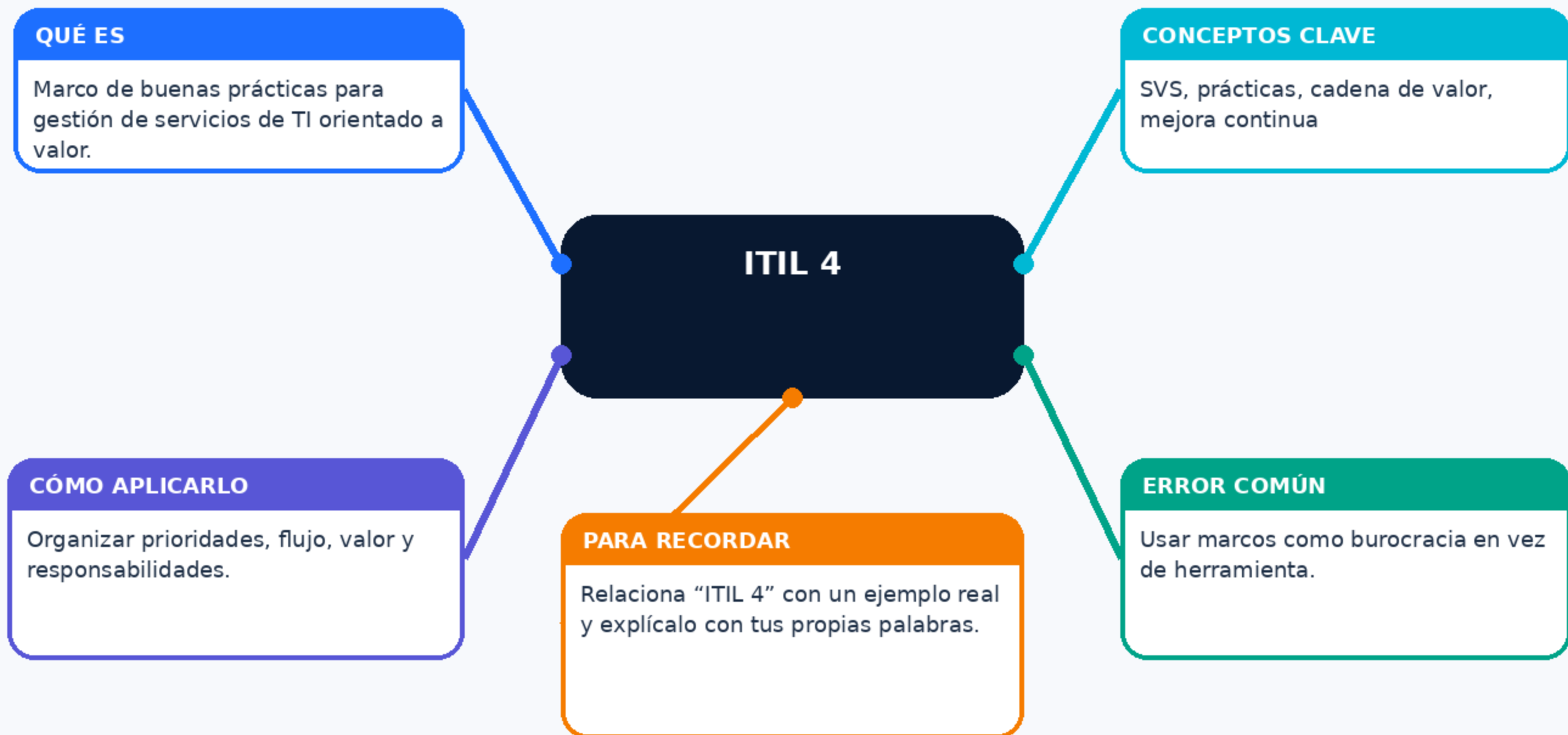
Gestión de proyectos



Gobernanza de TI



ITIL 4



KPI

QUÉ ES

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

KPI, valor, flujo, rol, métrica, mejora

KPI

CÓMO APLICARLO

Organizar prioridades, flujo, valor y responsabilidades.

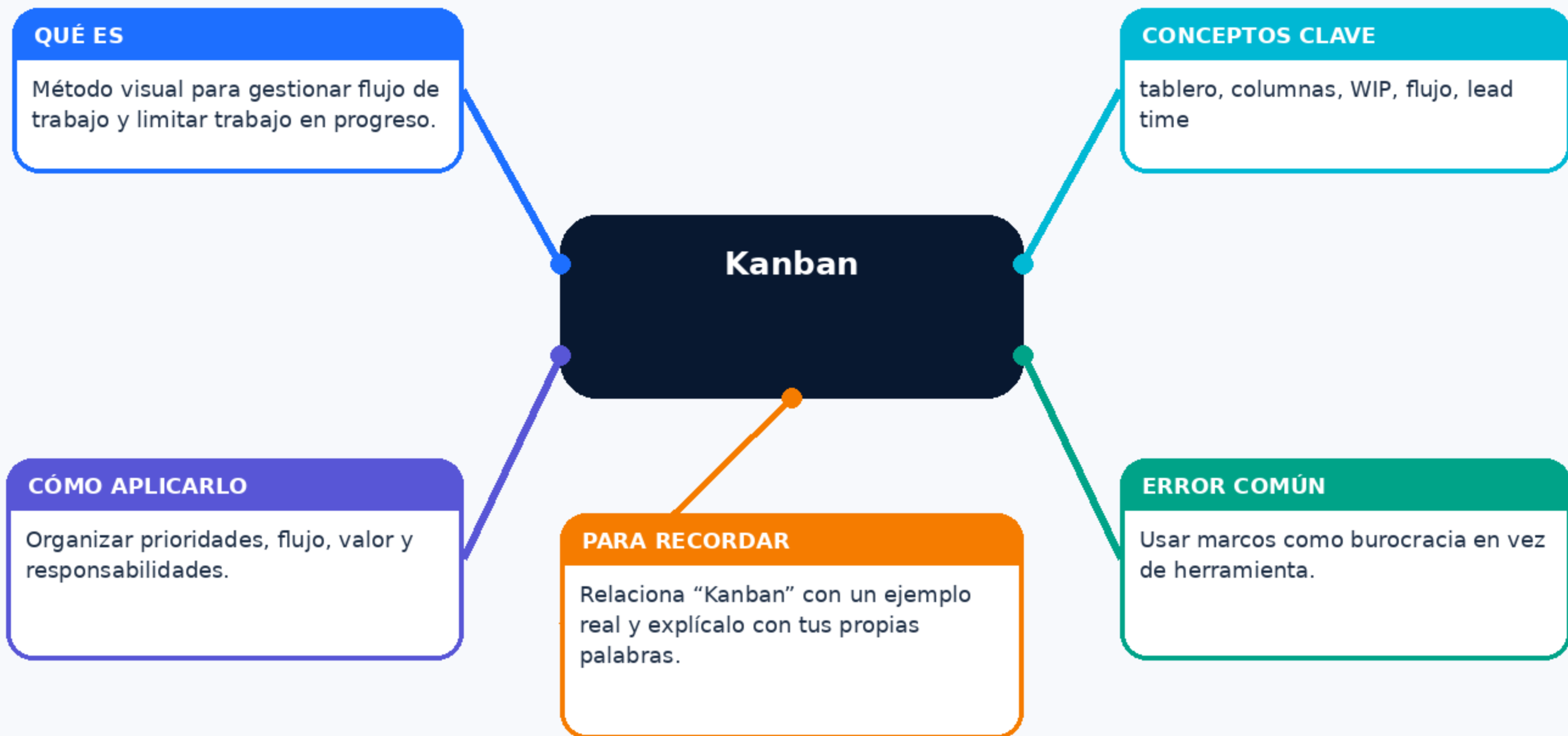
PARA RECORDAR

Relaciona "KPI" con un ejemplo real y explícalo con tus propias palabras.

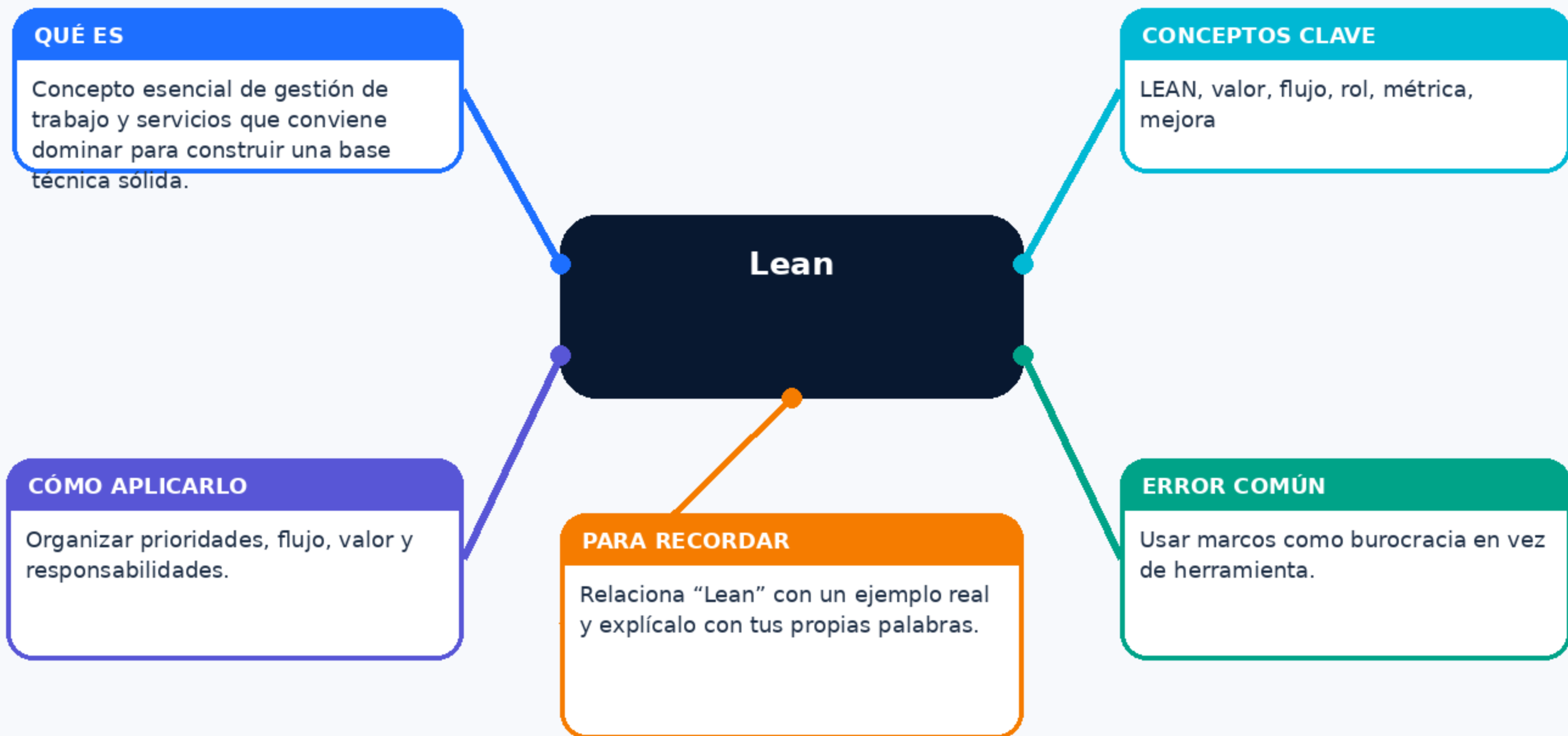
ERROR COMÚN

Usar marcos como burocracia en vez de herramienta.

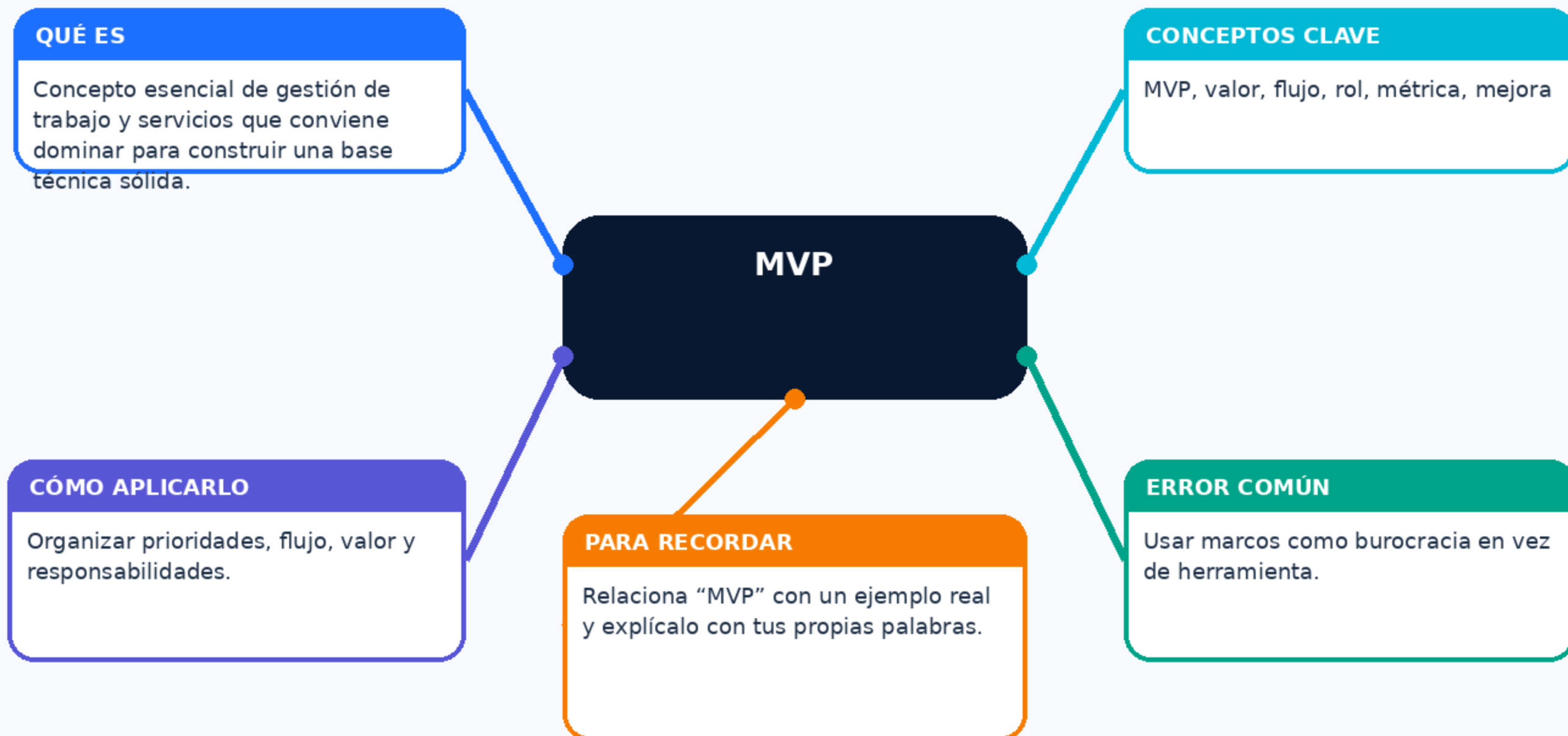
Kanban



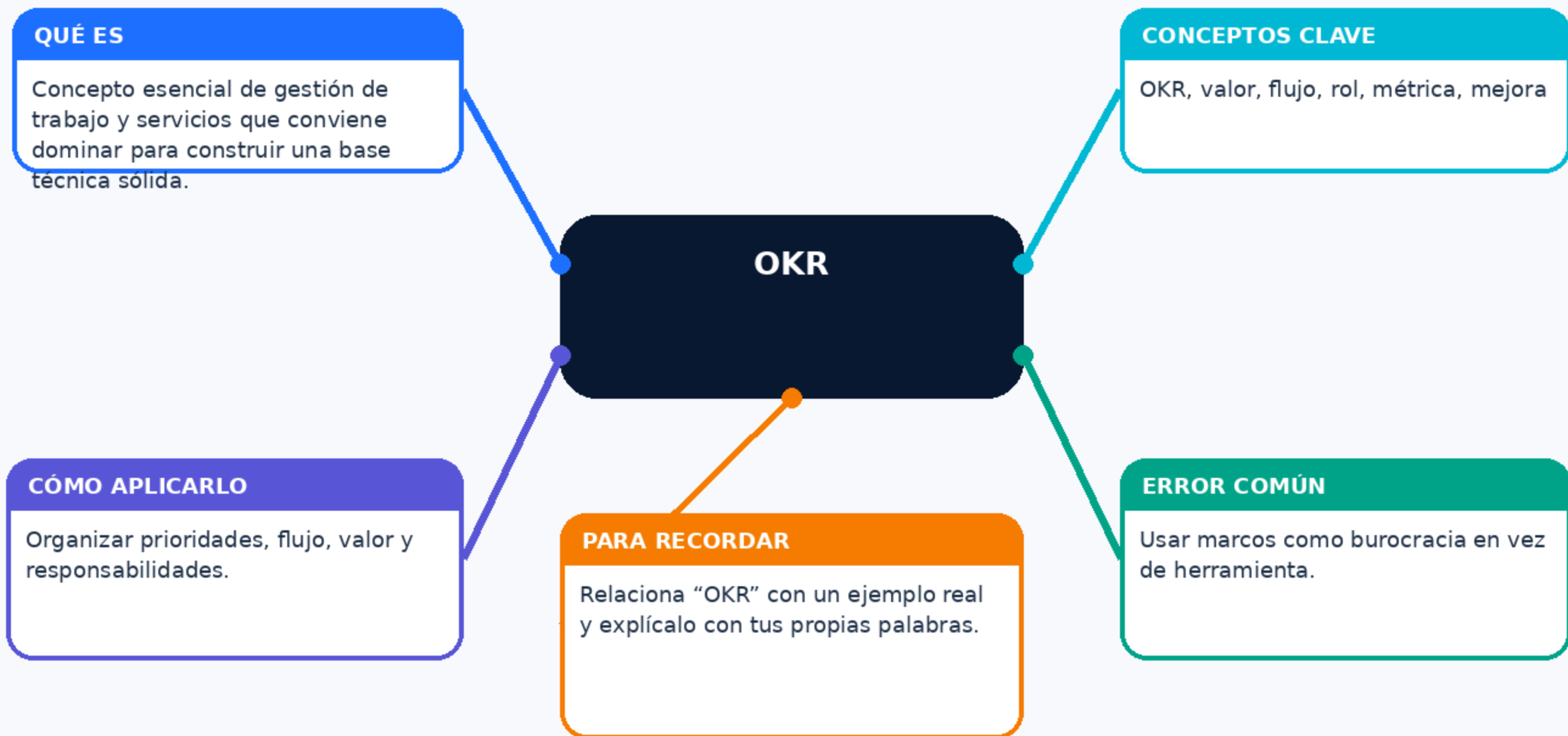
Lean



MVP



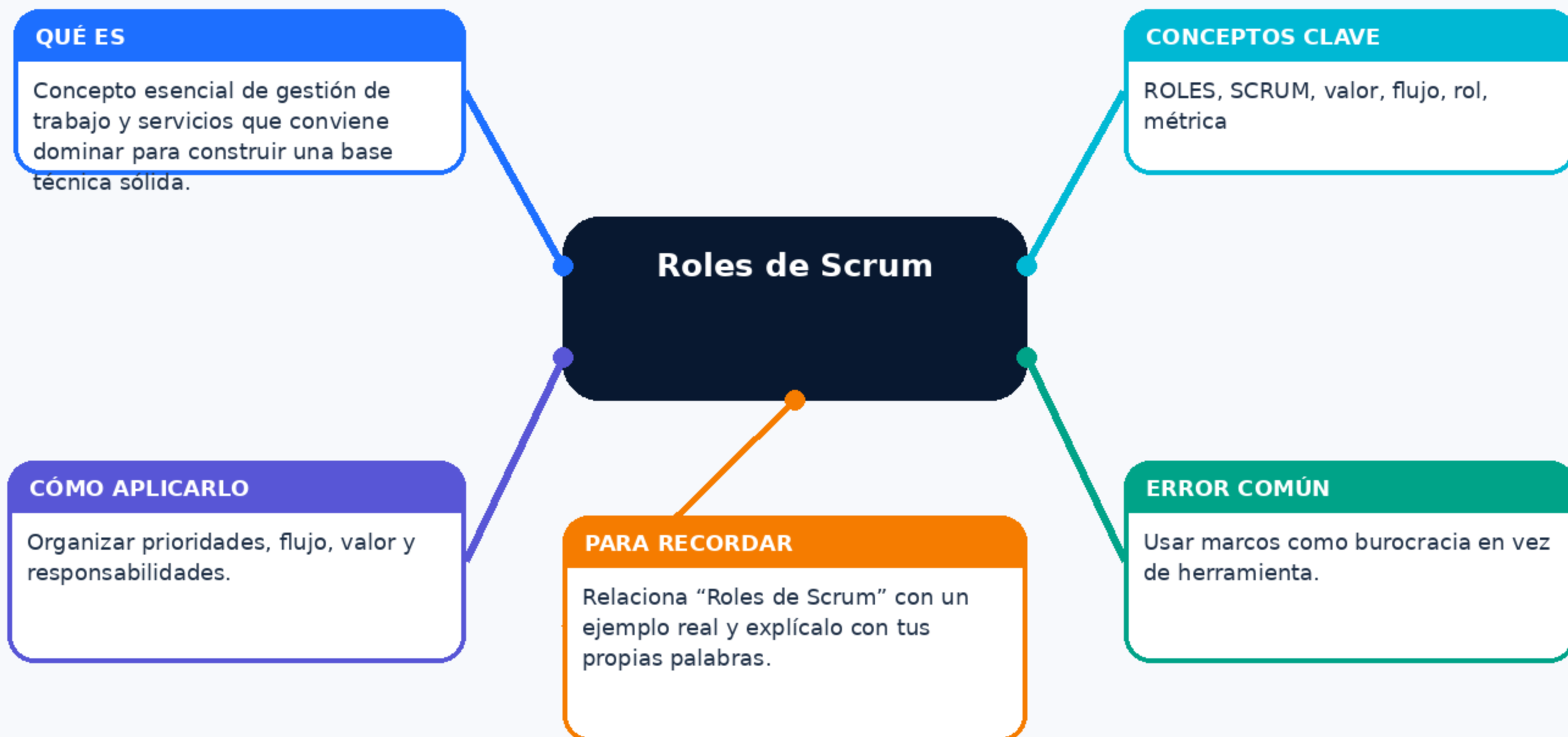
OKR



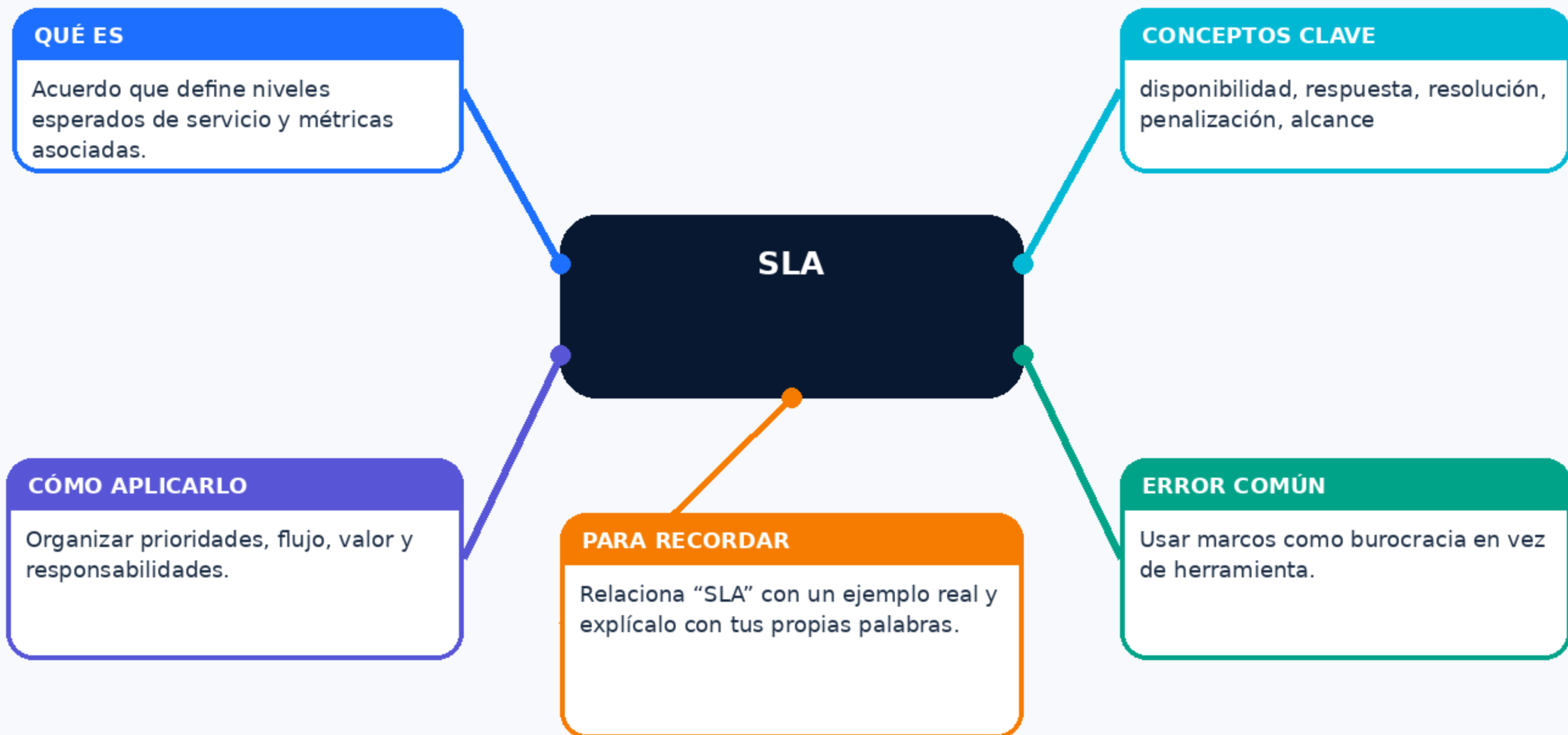
Riesgos de proyecto



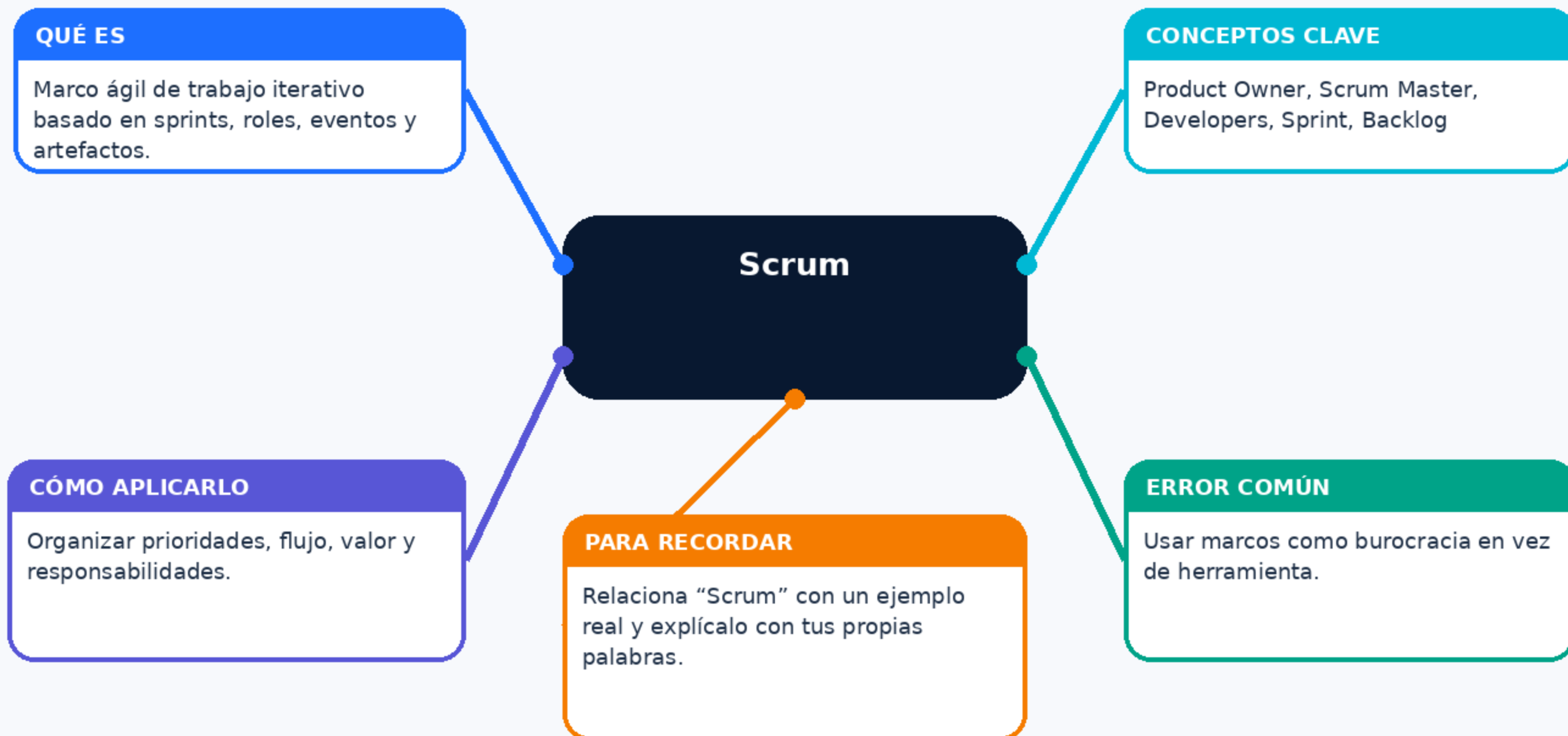
Roles de Scrum



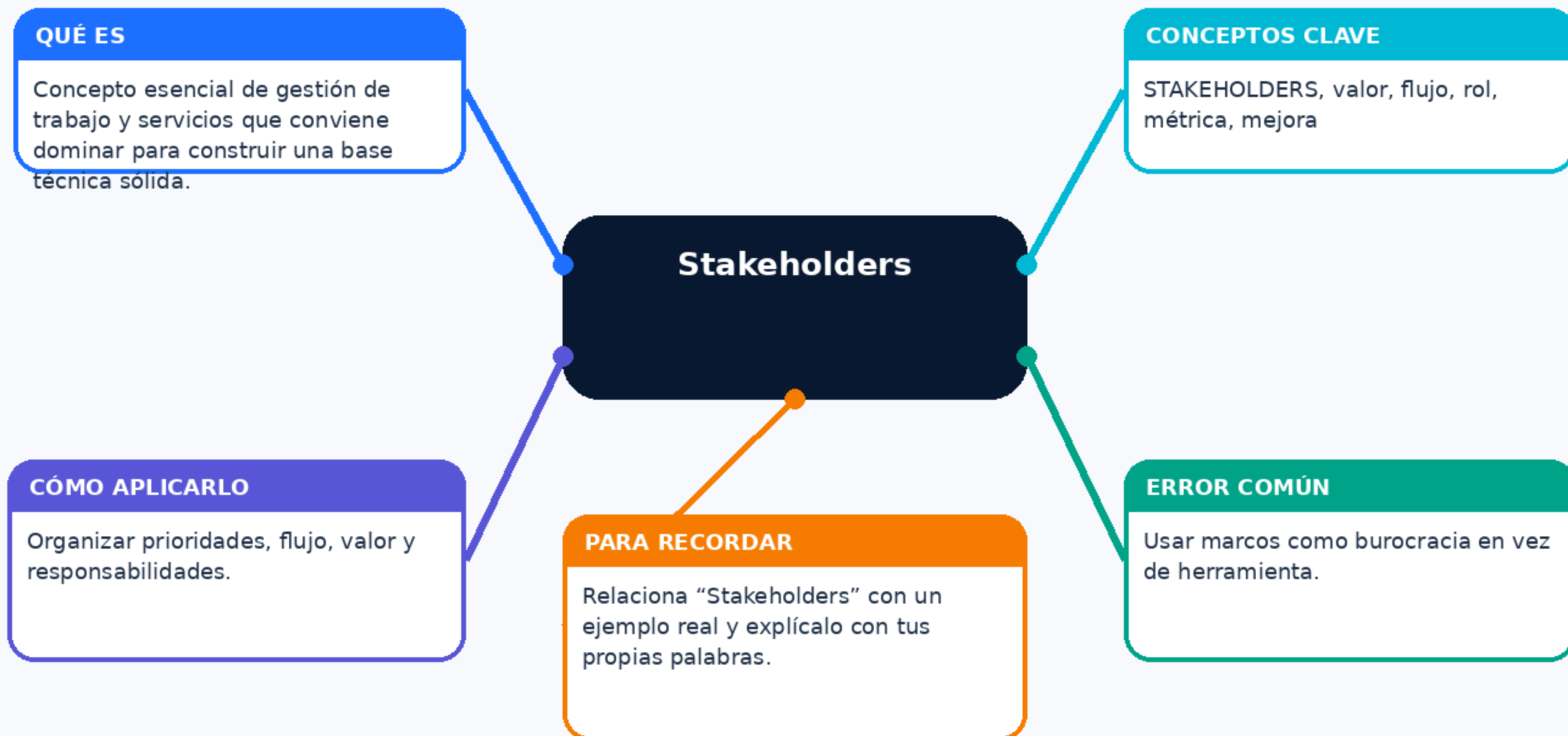
SLA



Scrum



Stakeholders



WIP

QUÉ ES

Concepto esencial de gestión de trabajo y servicios que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

WIP, valor, flujo, rol, métrica, mejora

WIP

CÓMO APLICARLO

Organizar prioridades, flujo, valor y responsabilidades.

PARA RECORDAR

Relaciona "WIP" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Usar marcos como burocracia en vez de herramienta.

Actualizaciones fallidas



Alto uso de CPU

QUÉ ES

Componente o tecnología de diagnóstico y resolución de incidentes con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

ALTO, USO, CPU, síntoma, evidencia, causa

Alto uso de CPU

CÓMO APLICARLO

Aislar causas con evidencia y pasos reproducibles.

PARA RECORDAR

Relaciona "Alto uso de CPU" con un ejemplo real y explícalo con tus propias palabras.

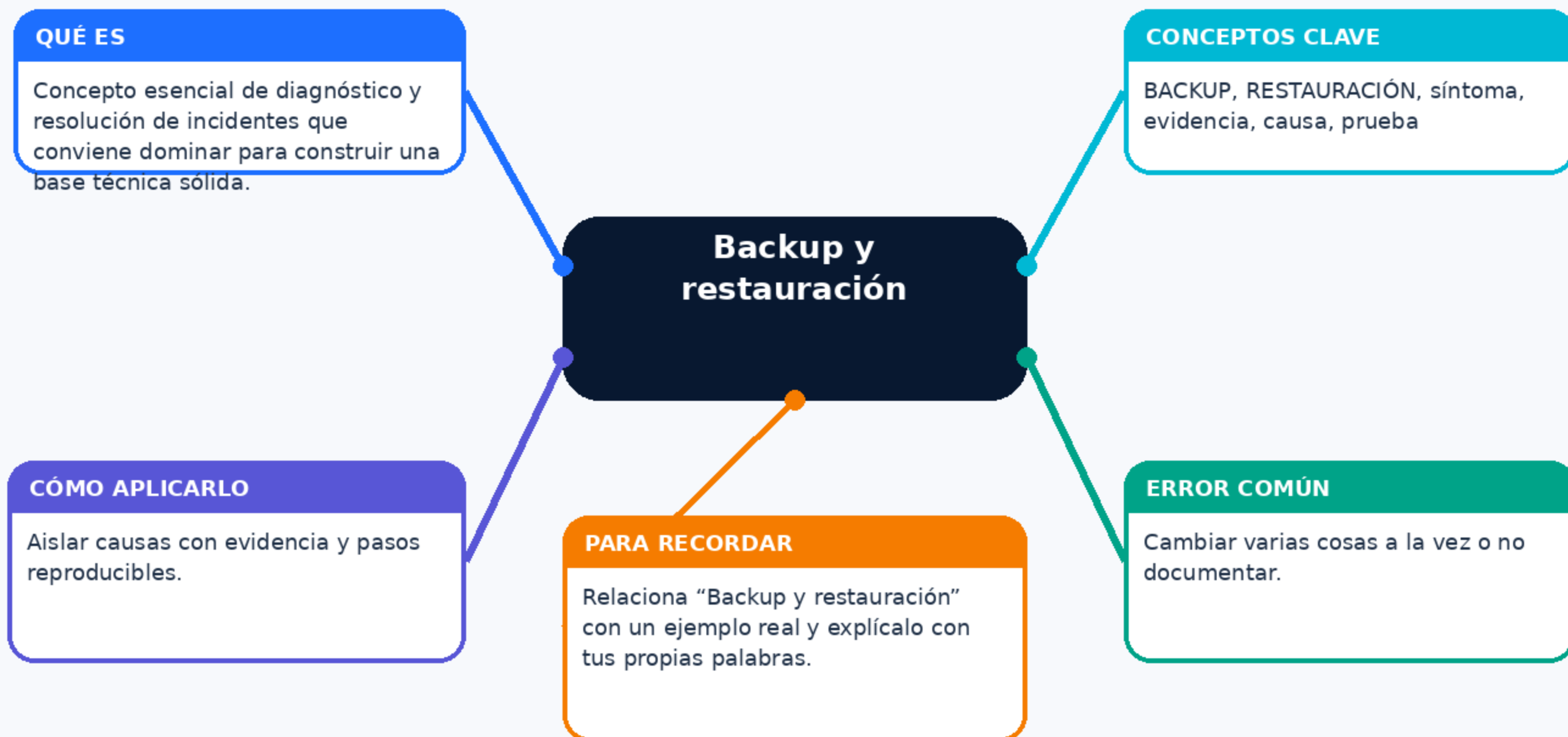
ERROR COMÚN

Cambiar varias cosas a la vez o no documentar.

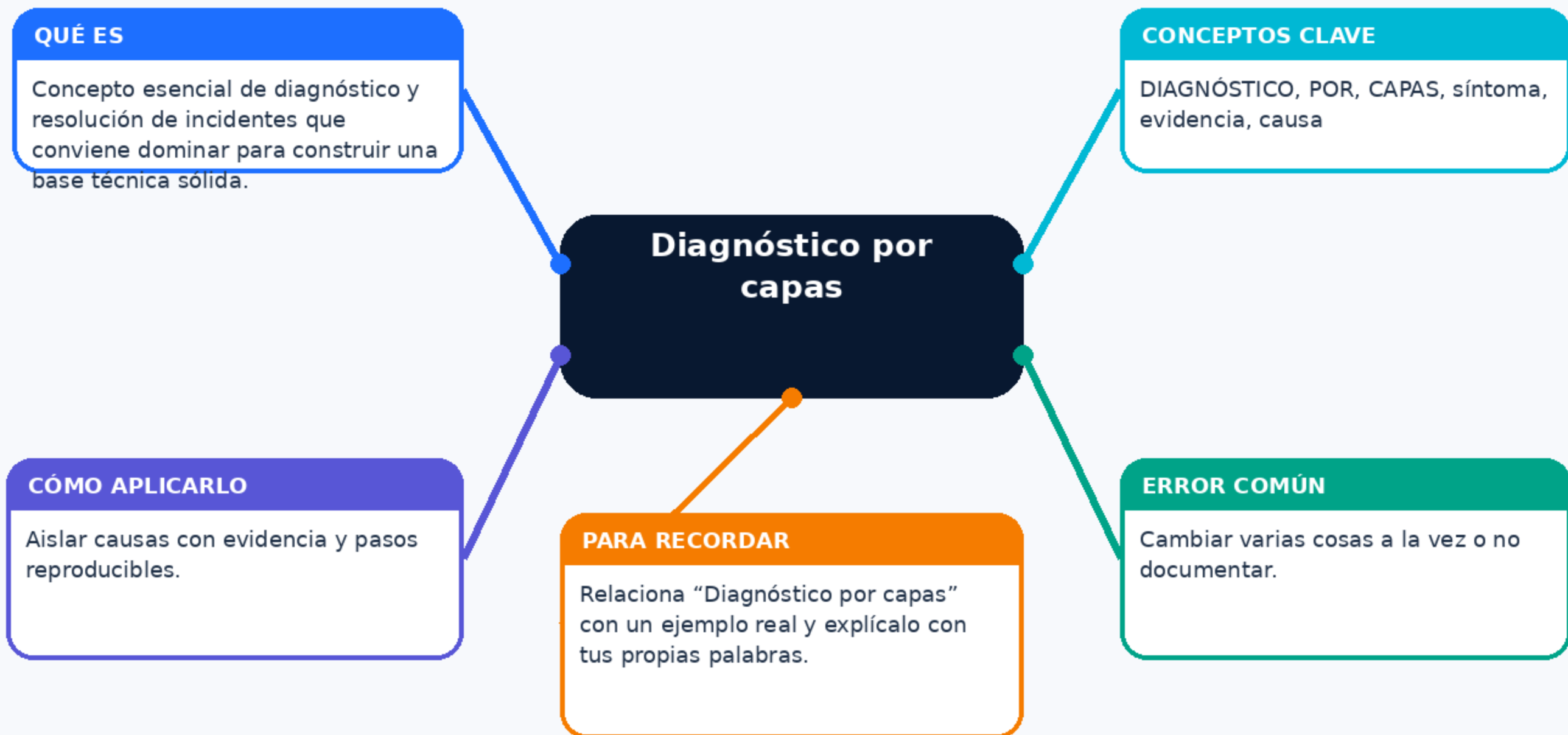
Aplicación no abre



Backup y restauración



Diagnóstico por capas



Disco lleno

QUÉ ES

Componente o tecnología de diagnóstico y resolución de incidentes con impacto directo en capacidad, rendimiento o conectividad.

CONCEPTOS CLAVE

DISCO, LLENO, síntoma, evidencia, causa, prueba

Disco lleno

CÓMO APLICARLO

Aislar causas con evidencia y pasos reproducibles.

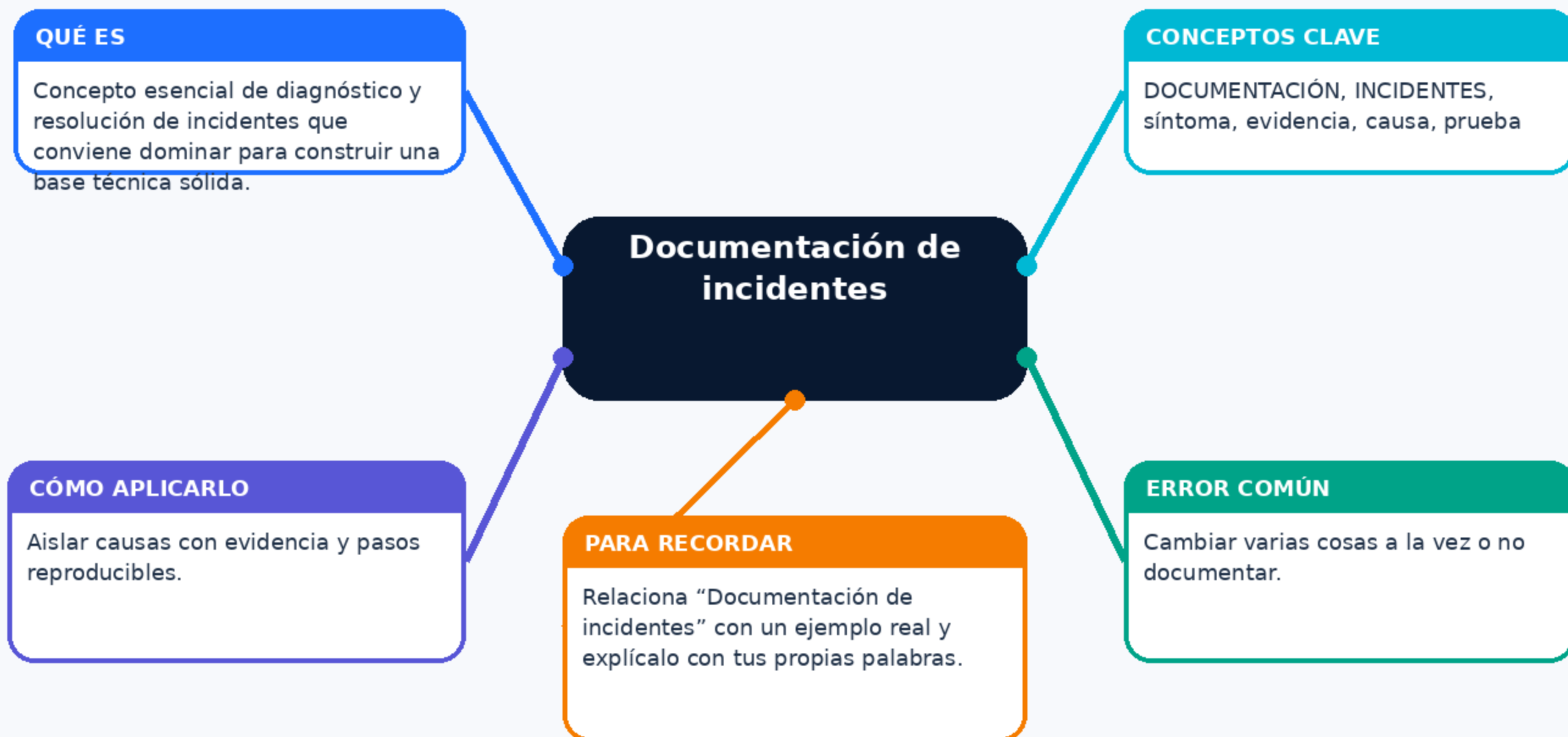
PARA RECORDAR

Relaciona "Disco lleno" con un ejemplo real y explícalo con tus propias palabras.

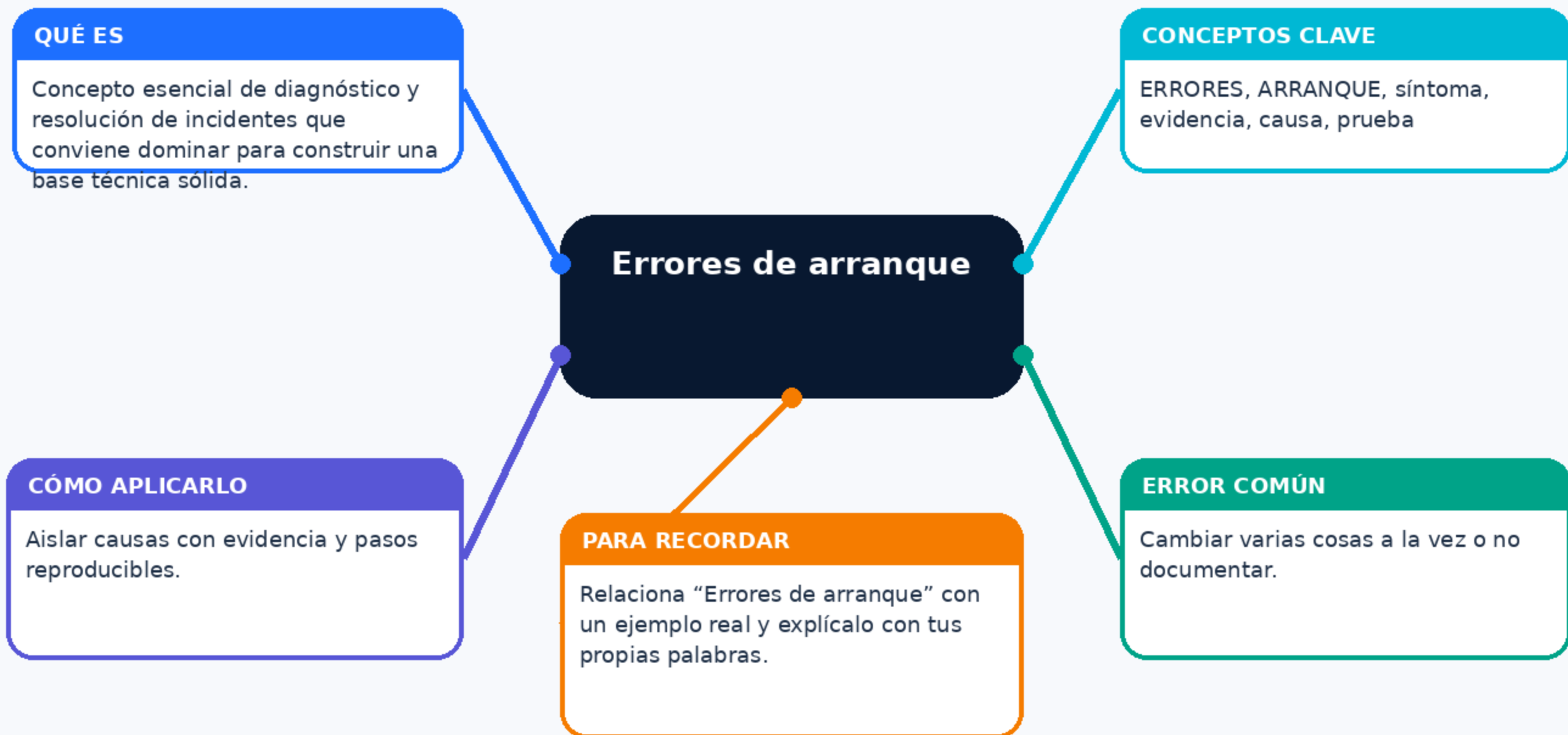
ERROR COMÚN

Cambiar varias cosas a la vez o no documentar.

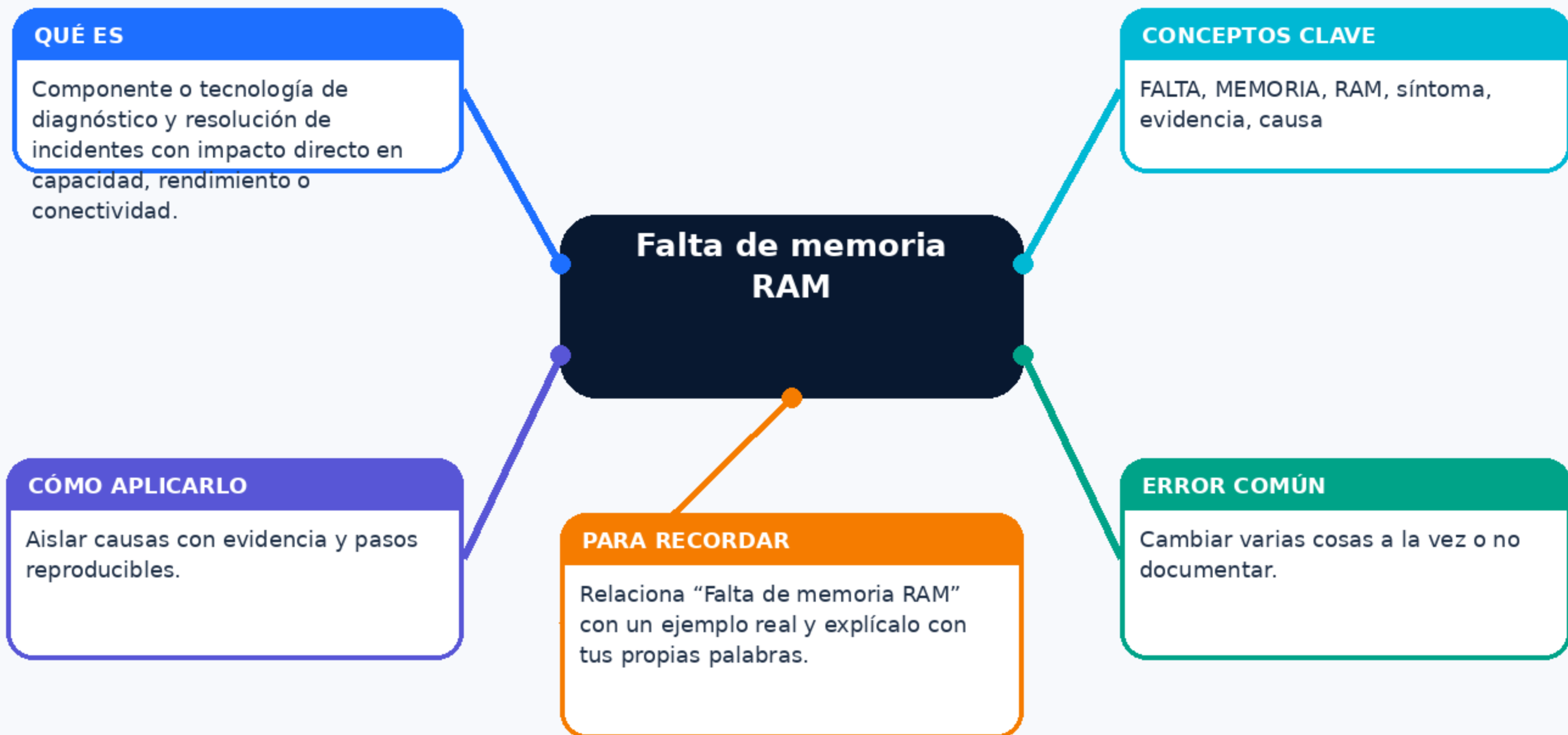
Documentación de incidentes



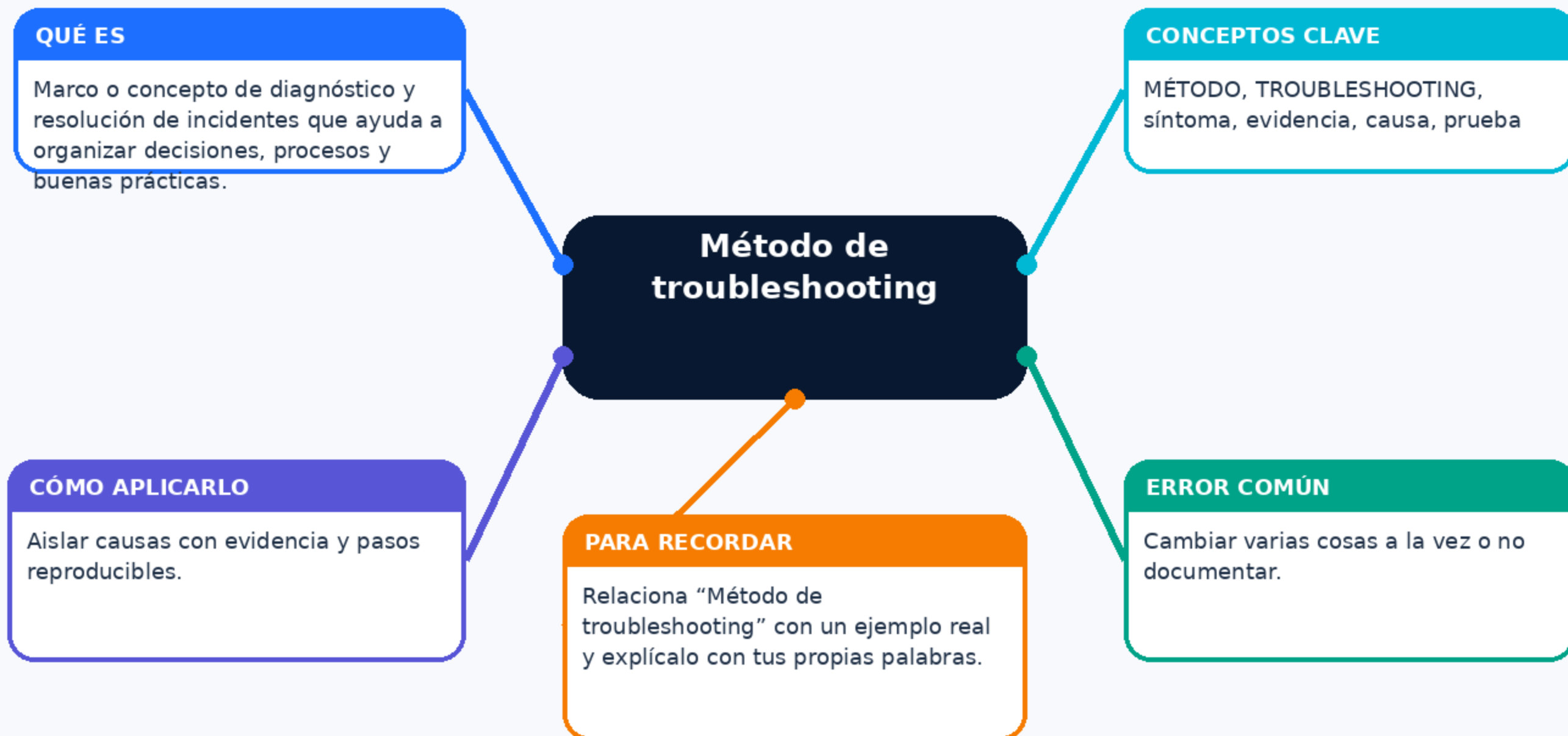
Errores de arranque



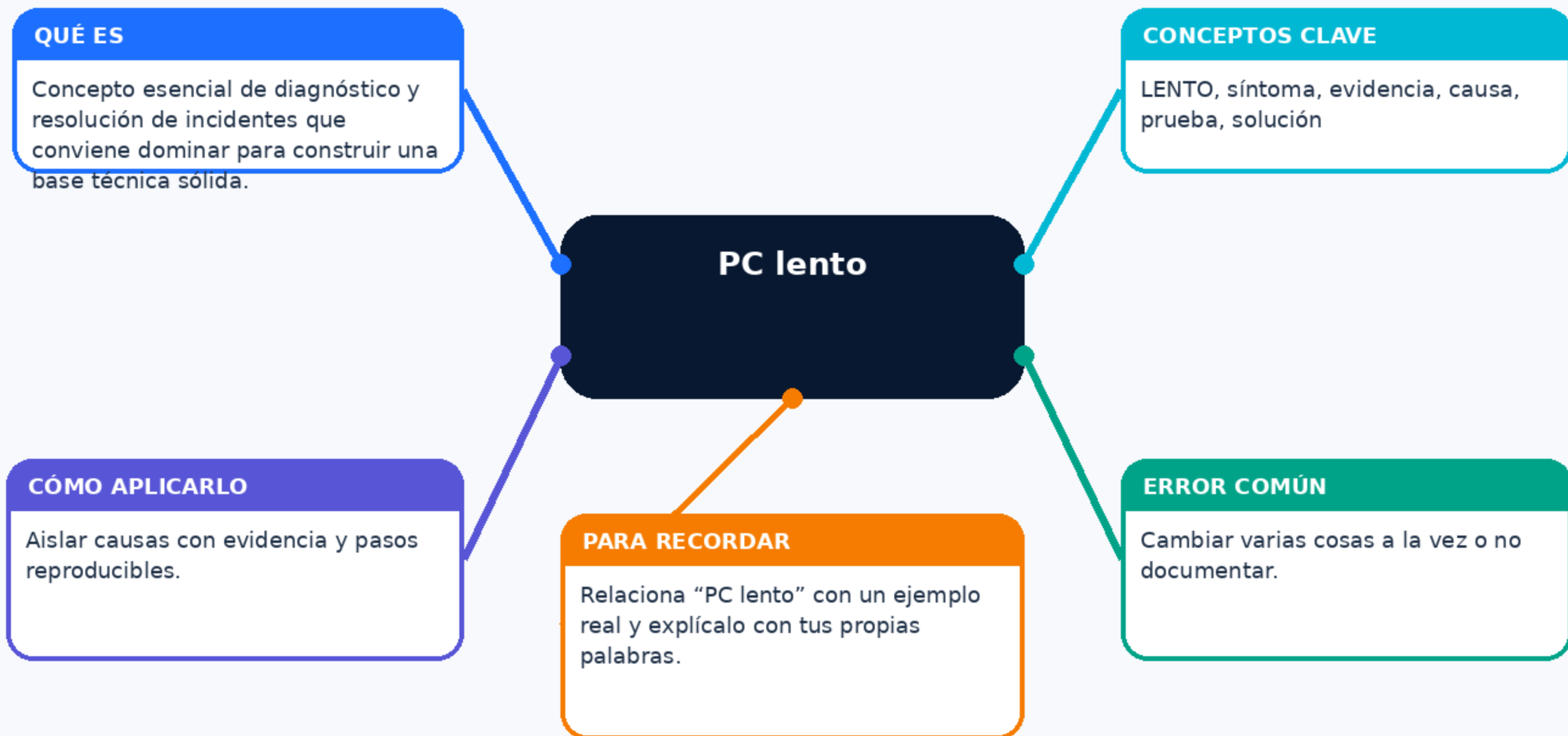
Falta de memoria RAM



Método de troubleshooting



PC lento



Pantalla azul

QUÉ ES

Concepto esencial de diagnóstico y resolución de incidentes que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

PANTALLA, AZUL, síntoma, evidencia, causa, prueba

Pantalla azul

CÓMO APLICARLO

Aislar causas con evidencia y pasos reproducibles.

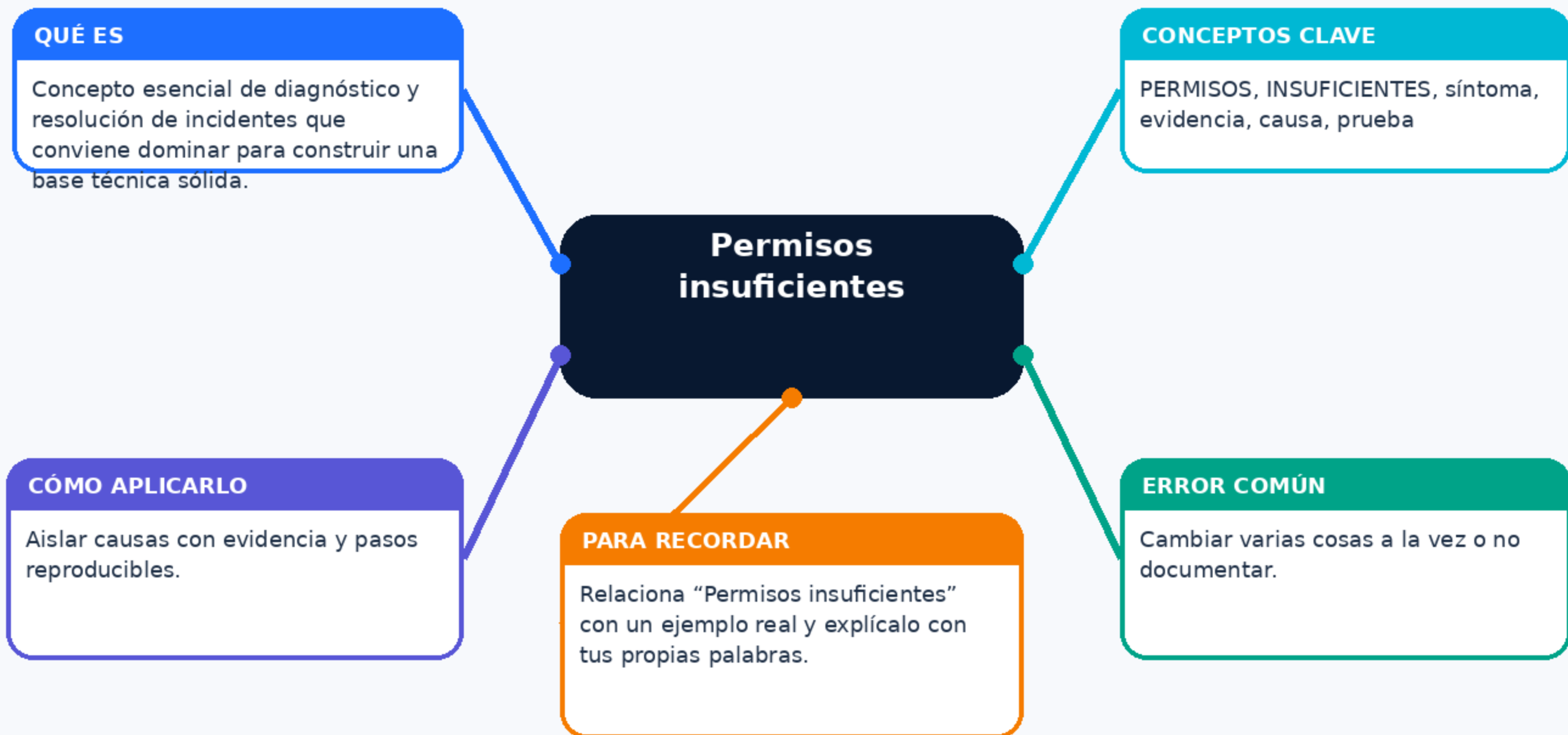
PARA RECORDAR

Relaciona "Pantalla azul" con un ejemplo real y explícalo con tus propias palabras.

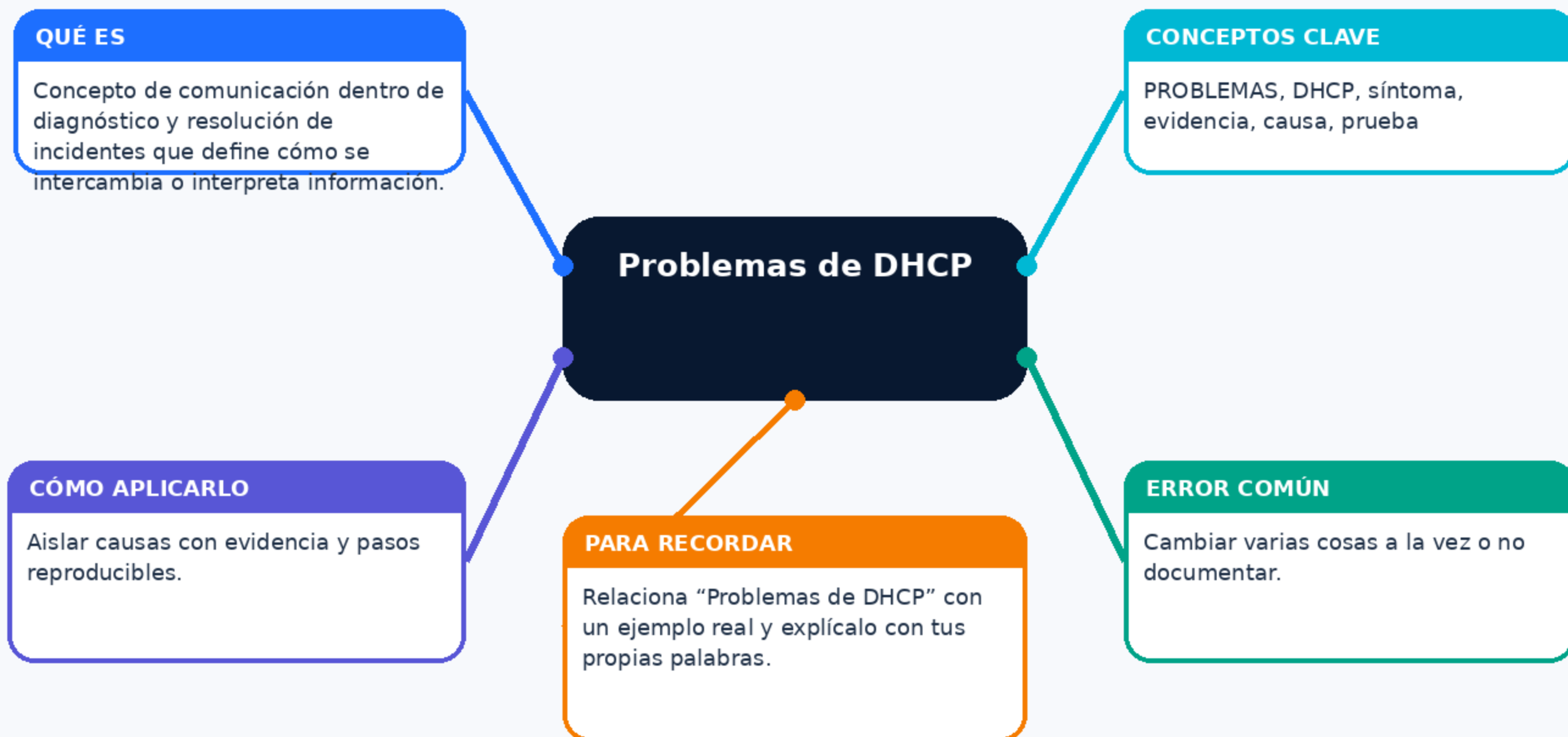
ERROR COMÚN

Cambiar varias cosas a la vez o no documentar.

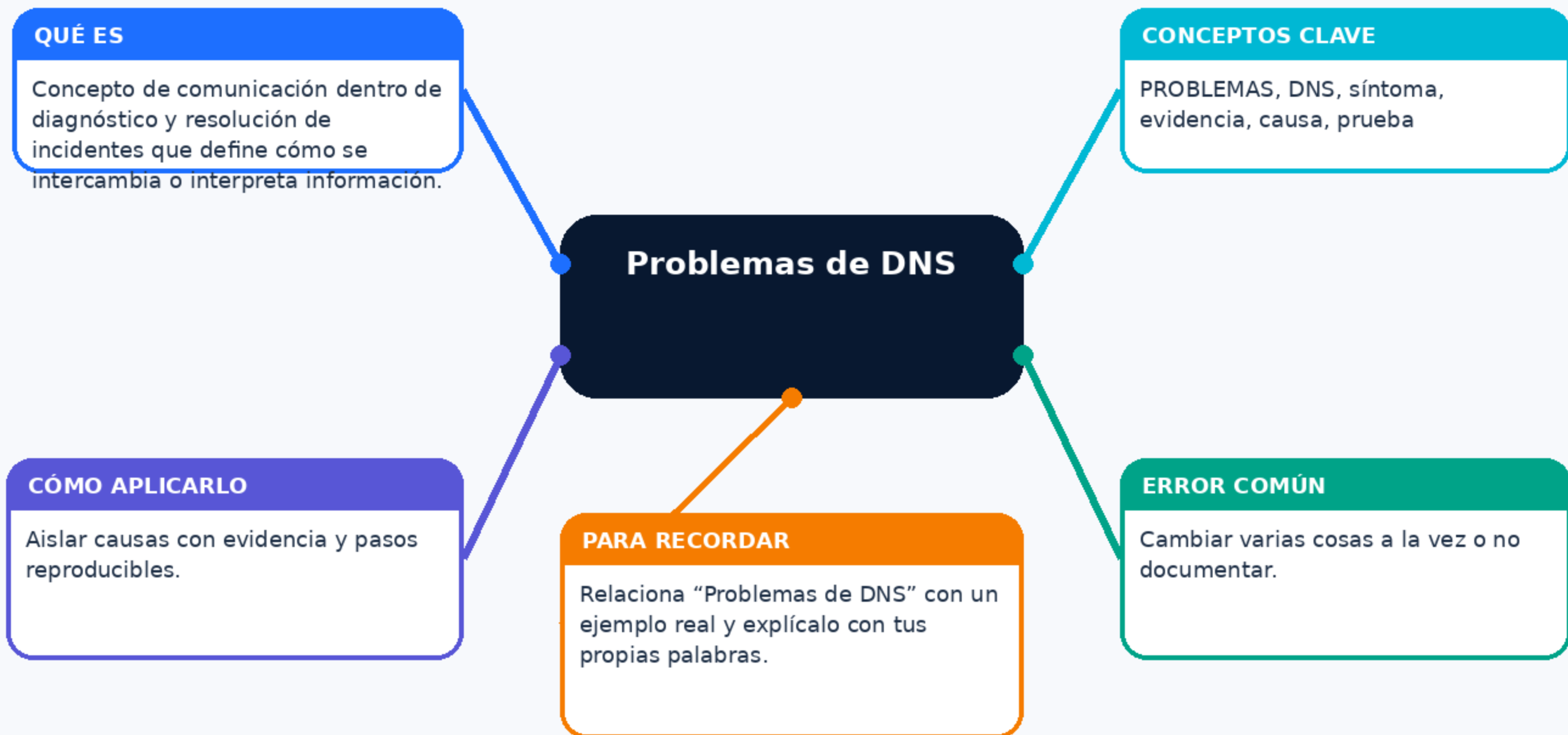
Permisos insuficientes



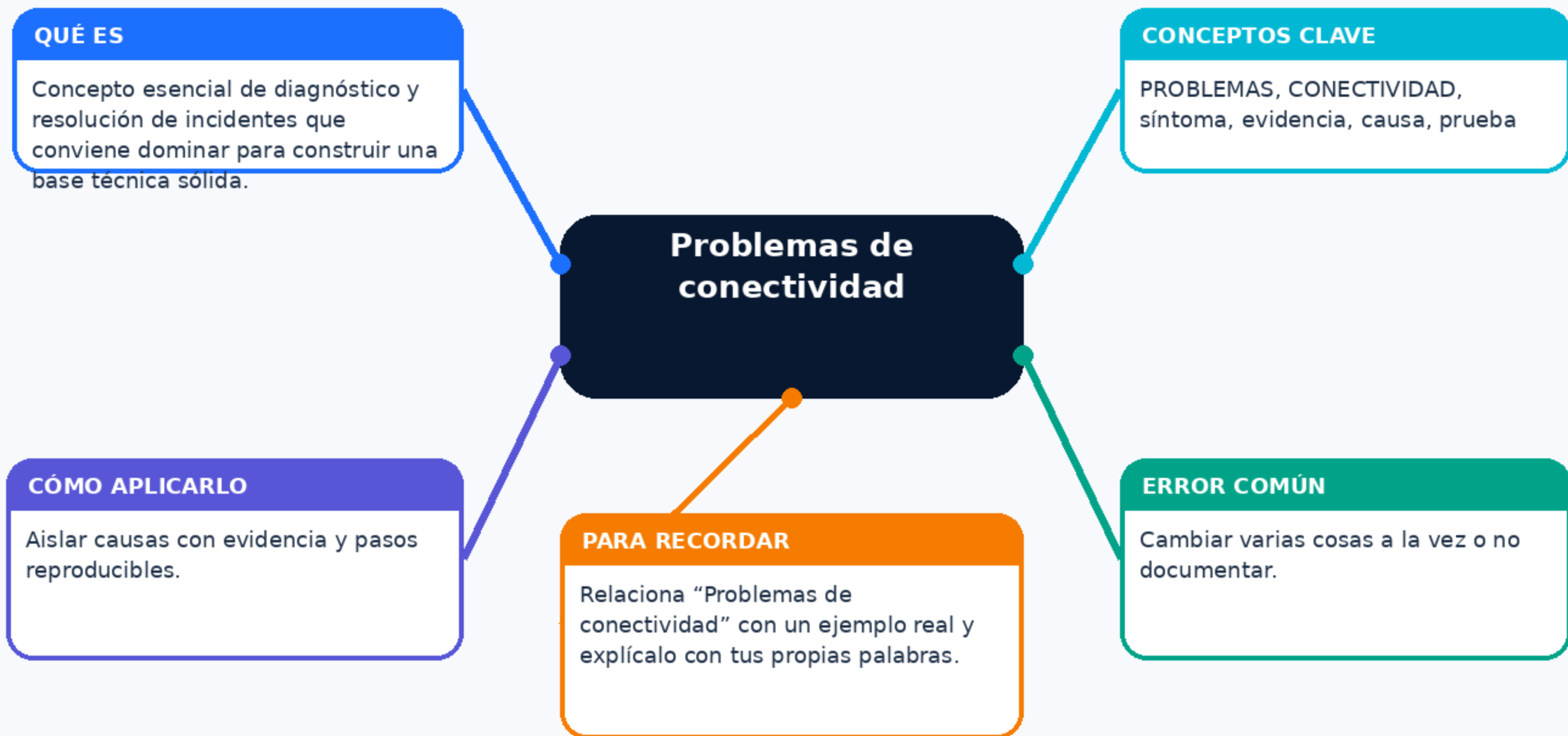
Problemas de DHCP



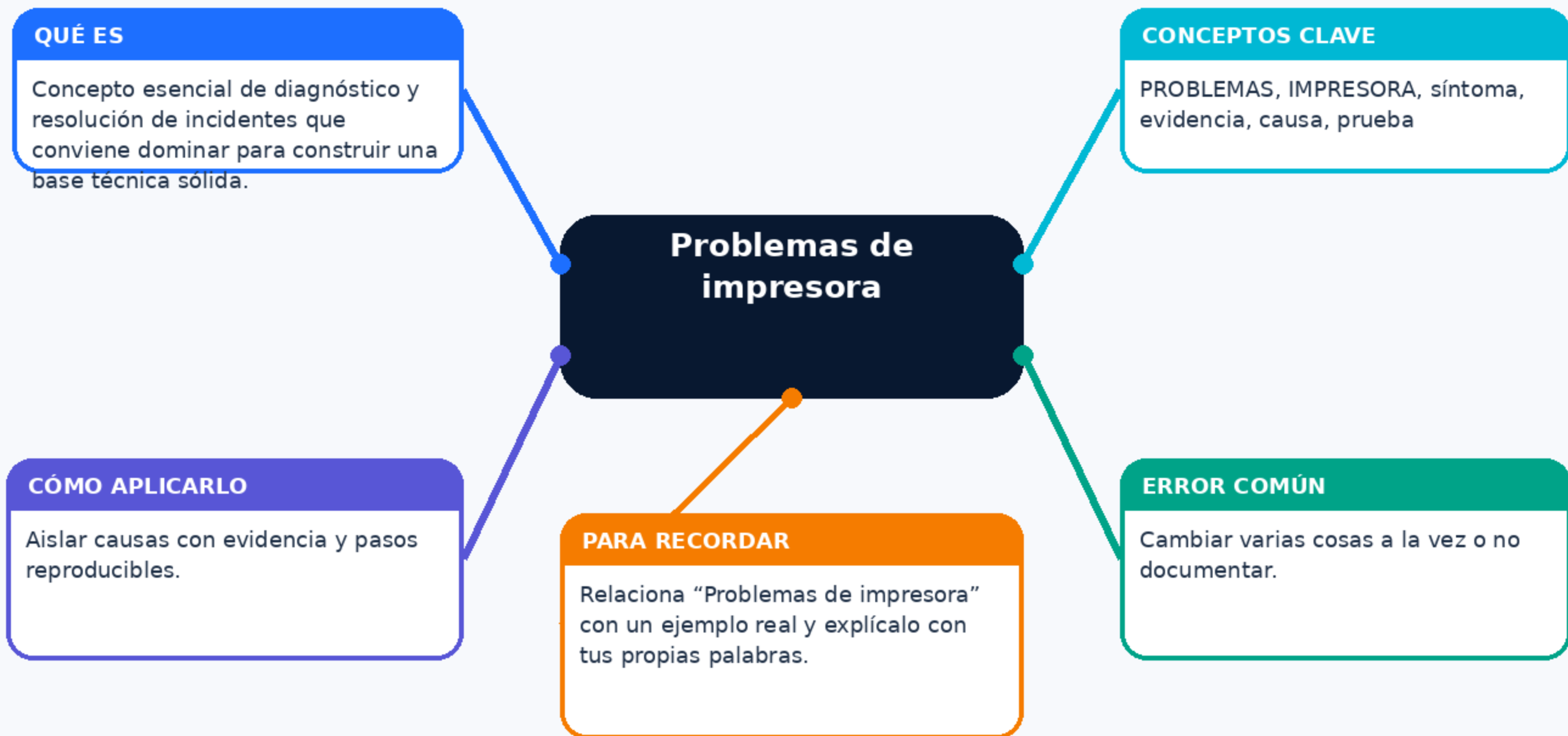
Problemas de DNS



Problemas de conectividad



Problemas de impresora



Recuperación de archivos



Administrador de redes



Administrador de sistemas



Analista de ciberseguridad



Analista de datos



CV para TI



Certificaciones iniciales



Cloud Engineer

QUÉ ES

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

CLOUD, ENGINEER, objetivo, práctica, evidencia, comunicación

Cloud Engineer

CÓMO APLICARLO

Convertir aprendizaje en evidencias, práctica y empleabilidad.

PARA RECORDAR

Relaciona "Cloud Engineer" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Estudiar sin proyecto, práctica o objetivo claro.

Desarrollador backend



Desarrollador frontend

QUÉ ES

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

DESARROLLADOR, FRONTEND, objetivo, práctica, evidencia, comunicación

Desarrollador frontend

CÓMO APLICARLO

Convertir aprendizaje en evidencias, práctica y empleabilidad.

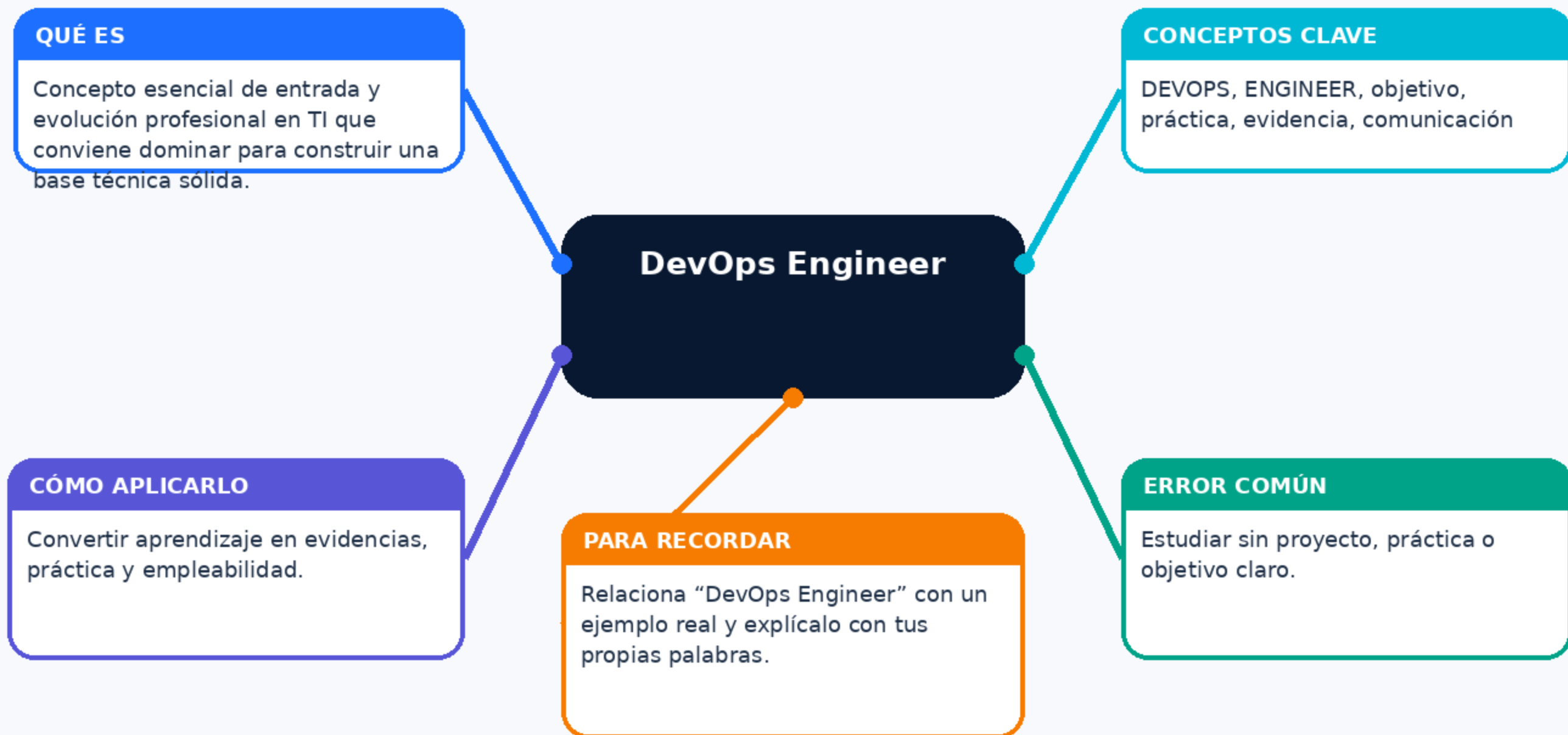
PARA RECORDAR

Relaciona "Desarrollador frontend" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Estudiar sin proyecto, práctica o objetivo claro.

DevOps Engineer



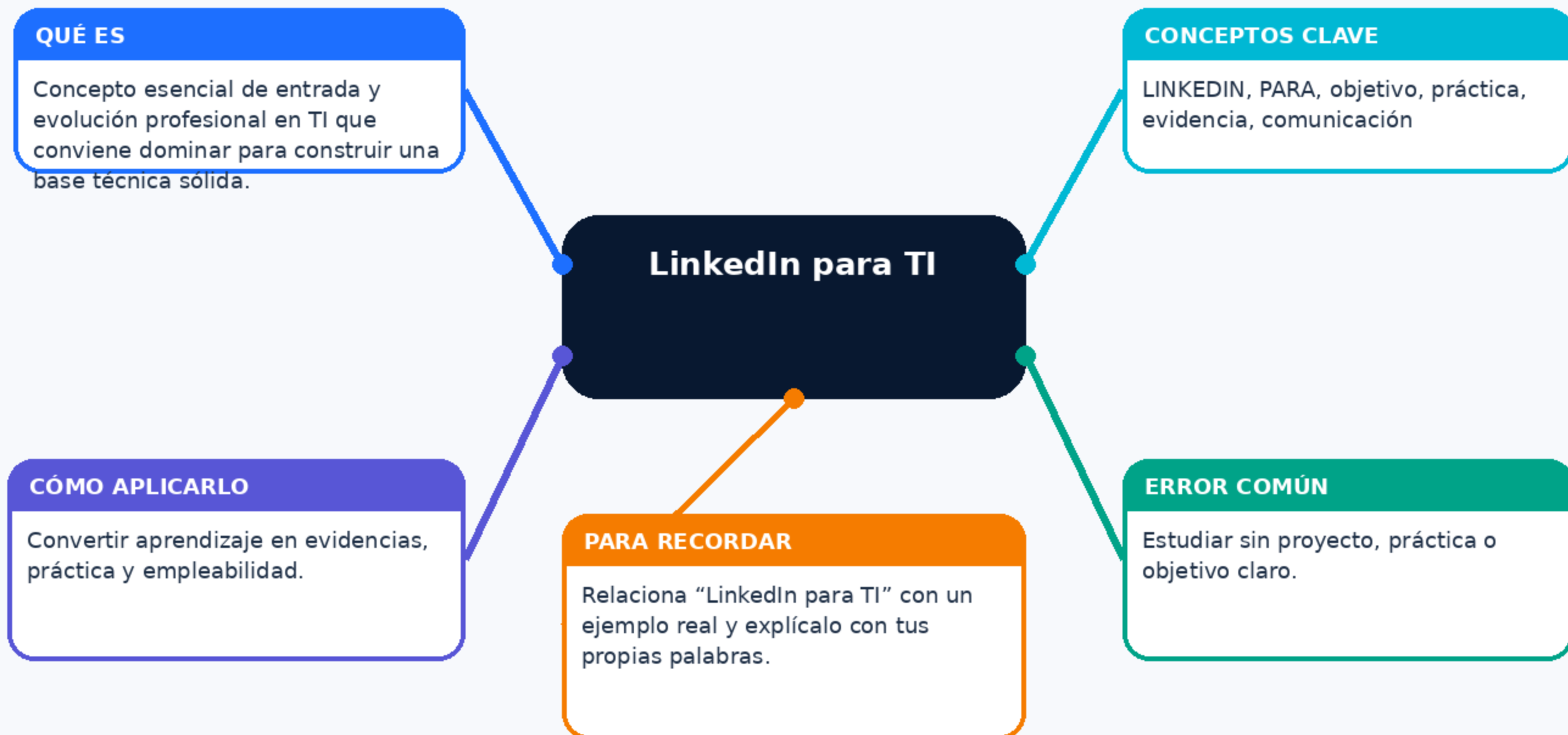
Entrevista técnica



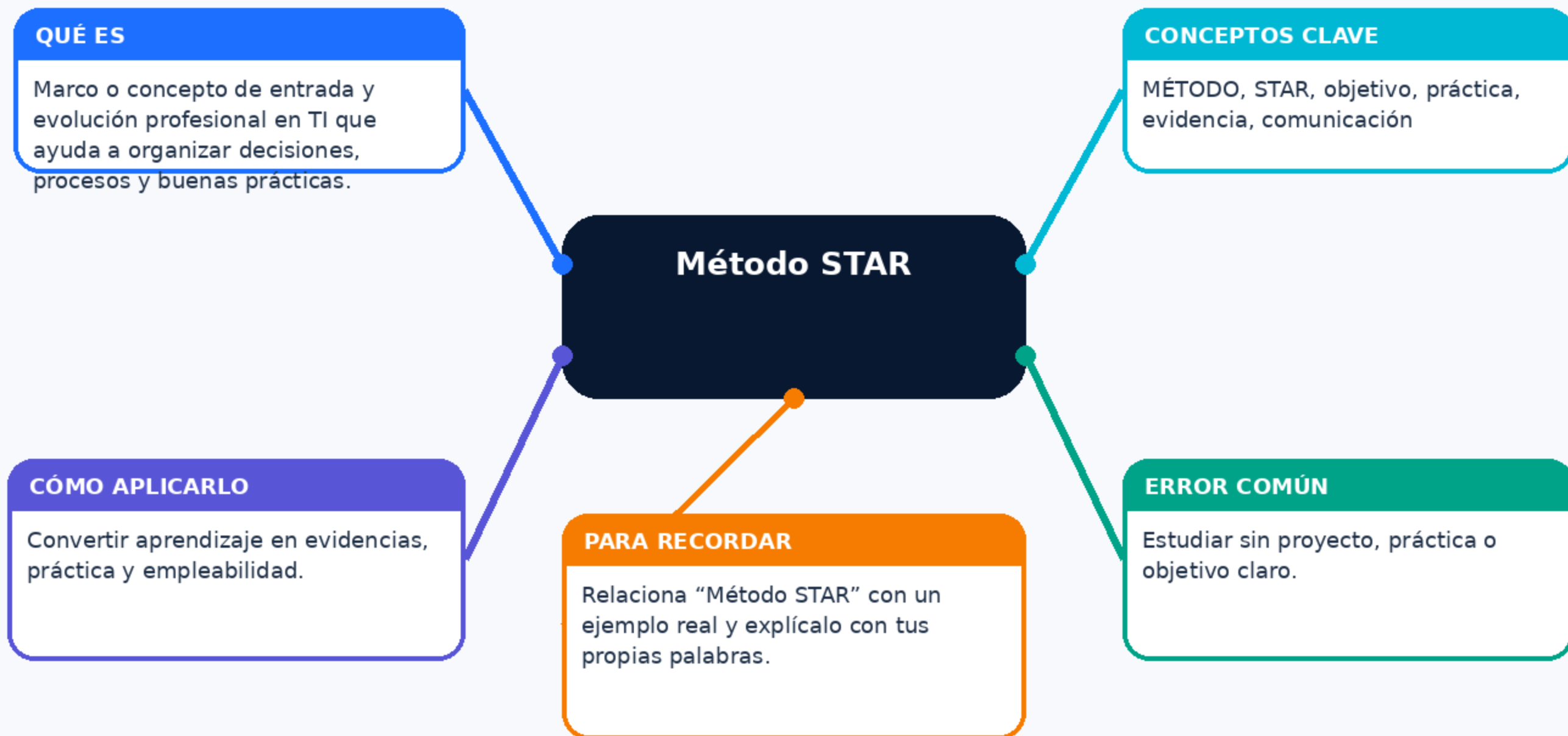
Help Desk



LinkedIn para TI



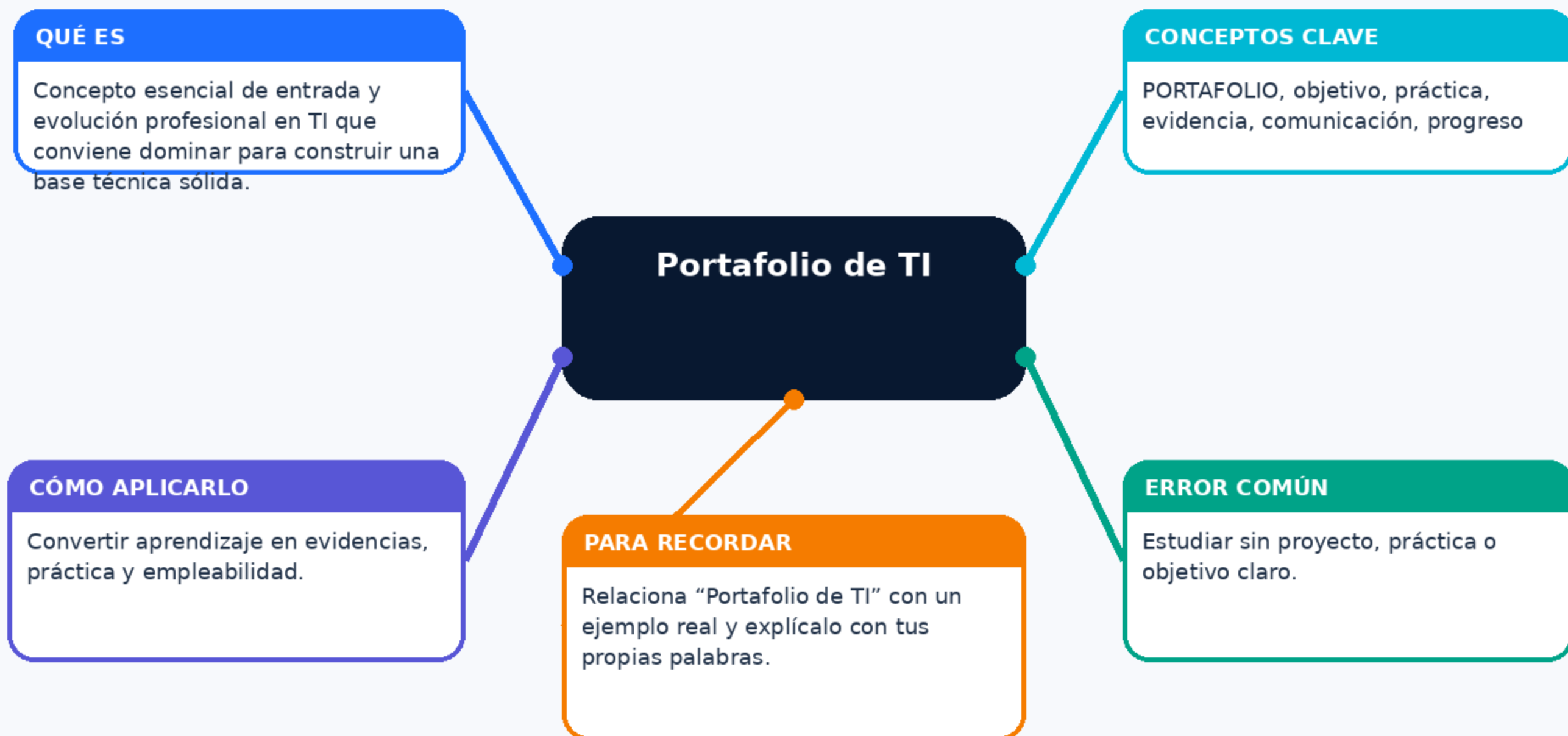
Método STAR



Plan de estudio de 90 días



Portafolio de TI



Ruta de entrada a TI



Soporte técnico

QUÉ ES

Concepto esencial de entrada y evolución profesional en TI que conviene dominar para construir una base técnica sólida.

CONCEPTOS CLAVE

SOPORTE, TÉCNICO, objetivo, práctica, evidencia, comunicación

Soporte técnico

CÓMO APLICARLO

Convertir aprendizaje en evidencias, práctica y empleabilidad.

PARA RECORDAR

Relaciona "Soporte técnico" con un ejemplo real y explícalo con tus propias palabras.

ERROR COMÚN

Estudiar sin proyecto, práctica o objetivo claro.