

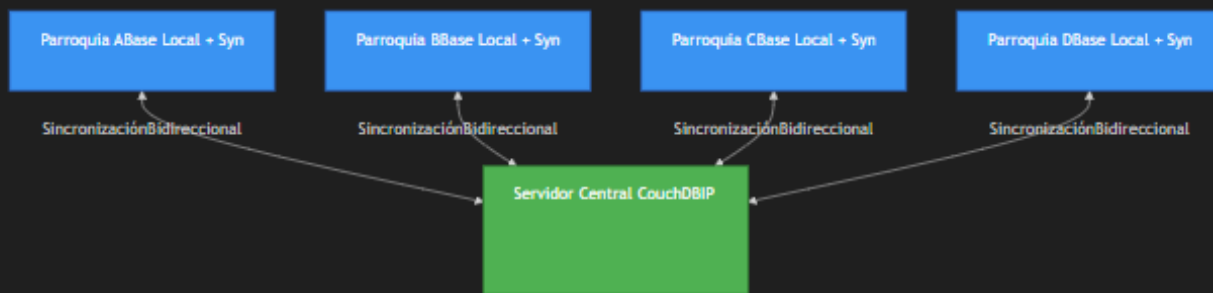
Arquitectura de Sincronización - Sistema de Sacramentos

📄 Resumen Ejecutivo

El sistema está diseñado con una **arquitectura de Nodo Central** donde todas las parroquias se sincronizan con un servidor CouchDB central. Cada instalación funciona de manera **offline-first**, permitiendo trabajo sin conexión y sincronización automática cuando hay internet.

🔧 Arquitectura Recomendada

Opción Implementada: Nodo Central



Ventajas de esta Arquitectura

- ✅ **Simplicidad:** Una sola configuración por parroquia
- ✅ **Centralización:** Todos los datos en un solo lugar para backups
- ✅ **Escalabilidad:** Fácil agregar nuevas parroquias
- ✅ **Offline-First:** Cada parroquia funciona sin internet
- ✅ **Auto-recuperación:** Sincroniza automáticamente al reconectar

Componentes del Sistema

1. Base de Datos Local (PouchDB)

Cada parroquia tiene su propia base de datos local en el navegador:

```
typescript

const localDB = new PouchDB('sacramentos_locales');
```

Características:

- Almacenamiento en IndexedDB del navegador
- Funciona 100% offline
- Búsquedas rápidas con índices
- No requiere servidor

2. Servidor Central (CouchDB)

Un servidor central que todas las parroquias comparten:

```
URL: http://192.168.1.100:5984/sacramentos
Usuario: admin
Contraseña: [configurada en .env]
```

Requisitos:

- IP fija o dominio estático
- CouchDB instalado y configurado
- Puerto 5984 accesible en la red
- Credenciales de administrador

3. Sincronización Bidireccional

typescript



```
localDB.sync(remoteDB, {  
  live: true,    // Sincronización continua  
  retry: true    // Reintentos automáticos  
})
```

Flujo de datos:

1. Usuario crea sacramento en Parroquia A → Se guarda localmente
2. Sistema detecta conexión → Envía al servidor central
3. Servidor central recibe → Notifica a otras parroquias
4. Parroquias B, C, D reciben → Actualizan sus bases locales

Configuración por Instalación

Paso 1: Servidor Central (Una sola vez)

Instalación de CouchDB

En Windows:

powershell



```
# Descargar desde: https://couchdb.apache.org/  
# Instalar y configurar:  
# - Puerto: 5984  
# - Usuario: admin  
# - Contraseña: [segura]  
# - Modo: Single Node
```

En Linux:

bash

```
sudo apt update
sudo apt install couchdb
sudo systemctl enable couchdb
sudo systemctl start couchdb
```

Configuración de CouchDB

1. Acceder a Fauxton (interfaz web): `http://localhost:5984/_utils`
2. Crear base de datos: `sacramentos`
3. Configurar CORS (permitir acceso desde otras máquinas):

bash

```
curl -X PUT http://admin:password@localhost:5984/_node/_local/_config/httpd/enable_cors -d '"true"'
curl -X PUT http://admin:password@localhost:5984/_node/_local/_config/cors/origins -d '"*"'
curl -X PUT http://admin:password@localhost:5984/_node/_local/_config/cors/credentials -d '"true"'
```

4. **Configurar IP fija** en el servidor (importante):

- Windows: Panel de Control → Red → Propiedades IPv4
- Linux: Editar `/etc/netplan/` o `/etc/network/interfaces`

Credenciales

- `admin:password` en el código es solo un ejemplo
- **DEBES cambiar** la contraseña en producción
- Usar contraseñas fuertes (mínimo 16 caracteres)

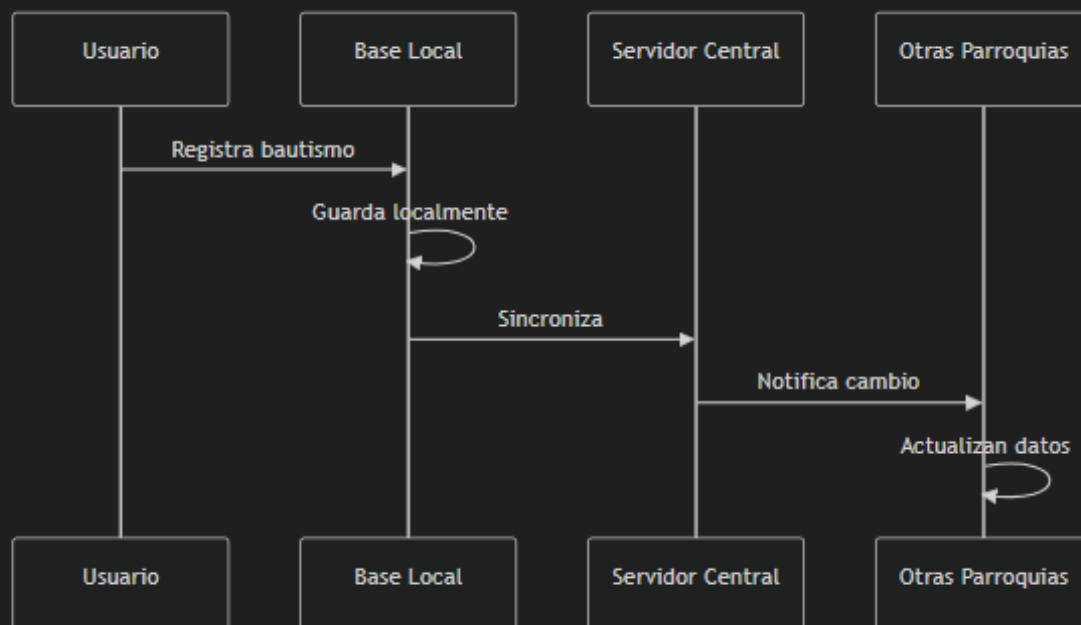
Recomendaciones

1. **No exponer CouchDB a internet** (solo red local)
2. **Usar VPN** si necesitas sincronizar entre ubicaciones remotas
3. **Backups regulares** del servidor central
4. **Firewall** para limitar acceso al puerto 5984

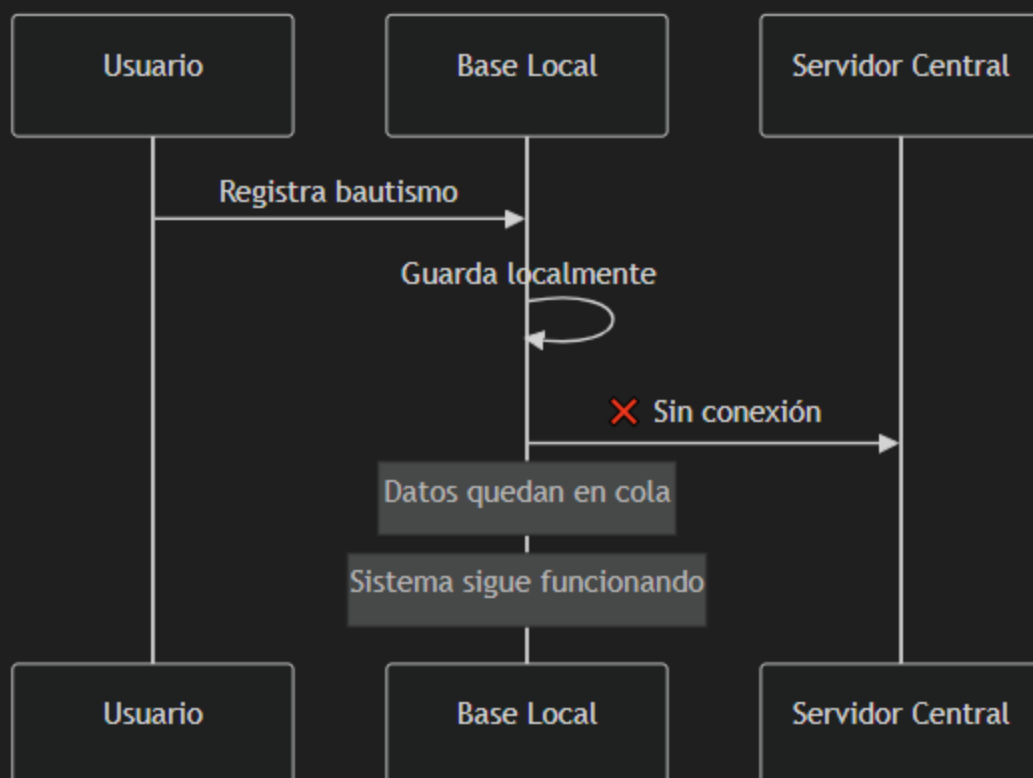


Escenarios de Uso

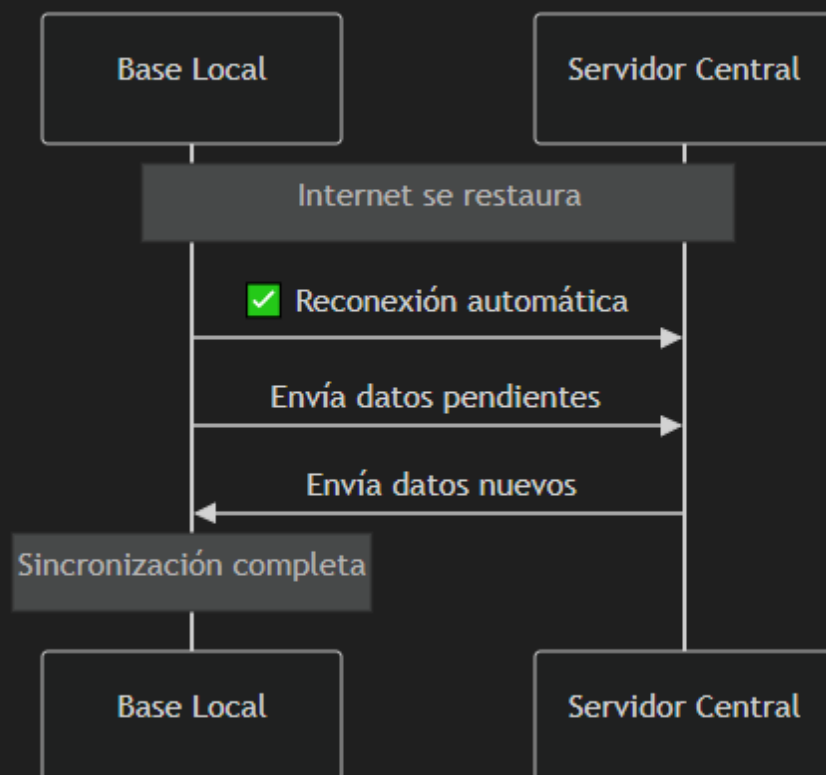
Escenario 1: Trabajo Normal con Internet



Escenario 2: Sin Internet (Offline)



Escenario 3: Reconexión





Instalación Paso a Paso

Para el Administrador de la Diócesis

1. Preparar Servidor Central

```
bash

# Instalar CouchDB
# Configurar IP fija: 192.168.1.100
# Crear base de datos: sacramentos
# Configurar usuario: admin / contraseña segura
```

2. Documentar Configuración

```
IP Servidor: 192.168.1.100
Puerto: 5984
Usuario: admin
Contraseña: [guardar en lugar seguro]
```

3. Preparar Paquete de Instalación

- Copiar sistema completo
- Incluir archivo `.env.template` con instrucciones
- Crear manual de instalación

Para Cada Parroquia

1. Instalar Sistema

```
bash

# Copiar archivos del sistema
# Instalar dependencias
npm install
```

2. Configurar `.env`

```
bash

# Copiar plantilla
cp .env.example .env

# Editar con datos del servidor central
# Cambiar VITE_PARISH_NAME a nombre de esta parroquia
```

3. Iniciar Aplicación

```
bash

npm run dev
```

4. Verificar Sincronización

- Abrir consola del navegador (F12)
- Buscar mensaje: "🔄 Servidor de sincronización configurado"
- Crear un registro de prueba
- Verificar en servidor central

Diagnóstico de Problemas

Problema: No sincroniza

Verificar:

1. Conectividad

```
bash

# Desde la laptop de la parroquia:
ping 192.168.1.100
curl http://192.168.1.100:5984
```



2. Credenciales

```
bash

# Verificar acceso:
curl http://admin:password@192.168.1.100:5984/sacramentos
```



3. Consola del navegador

- Abrir DevTools (F12)
- Ver mensajes de error
- Buscar "✗" en los logs

Problema: Datos duplicados

Causa: Conflictos de sincronización

Solución: PouchDB maneja automáticamente, pero revisar:

typescript

```
// En database.ts, los conflictos se resuelven automáticamente  
// Si persisten, verificar que cada documento tenga _id único
```

Problema: Sincronización lenta

Optimizar:

1. Reducir frecuencia de sincronización
2. Usar índices apropiados
3. Verificar ancho de banda de red



Monitoreo

Logs en Consola del Navegador

```
🕒 Servidor de sincronización configurado: http://192.168.1.100:5984/sacramentos  
🕒 Iniciando sincronización con servidor central...  
▶ Sincronización activa  
✅ Parroquia San José: Sincronización exitosa  
⏸ Sincronización pausada (esperando cambios)
```

Dashboard de CouchDB

Acceder a: `http://192.168.1.100:5984/_utils`

Ver:

- Número de documentos
 - Tamaño de base de datos
 - Actividad de replicación
-

Respuestas a tus Preguntas

¿Cómo funcionaría la instalación?

Respuesta: Cada parroquia instala el sistema y configura su archivo `.env` apuntando al servidor central. No necesitan modificar código, solo configuración.

¿Necesitamos un nodo central?

Respuesta: Sí, es la opción recomendada porque:

- Más simple de administrar
- Backups centralizados
- Fácil agregar parroquias
- Mejor control de datos

¿Qué es `admin:password`?

Respuesta: Son las credenciales del servidor CouchDB central. Todas las parroquias usan las mismas credenciales para conectarse.

¿Qué es `IP_OTRA_PARROQUIA`?

Respuesta: Es la IP del servidor central (no de otra parroquia). Debe ser **fija** o usar un dominio local como `sync.diocesis.local`.

Paso 2: Configuración en Cada Parroquia

Archivo `.env` (copiar de `.env.example`)

```
bash

# En cada laptop de parroquia, crear archivo .env con:

# URL del servidor central (CAMBIAR LA IP)
VITE_COUCHDB_URL=http://192.168.1.100:5984

# Nombre de la base de datos (IGUAL EN TODAS)
VITE_COUCHDB_DB_NAME=sacramentos

# Credenciales del servidor central (IGUALES EN TODAS)
VITE_COUCHDB_USERNAME=admin
VITE_COUCHDB_PASSWORD=TuContraseñaSegura

# Identificación de esta parroquia (ÚNICO POR PARROQUIA)
VITE_PARISH_ID=PARROQUIA_001
VITE_PARISH_NAME=Parroquia San José

# Seguridad (ÚNICO POR PARROQUIA)
VITE_ENCRYPTION_KEY=clave-unica-parroquia-001
```

IMPORTANT

Cada parroquia debe tener:

- Mismo `VITE_COUCHDB_URL` (IP del servidor central)
- Mismo `VITE_COUCHDB_USERNAME` y `VITE_COUCHDB_PASSWORD`
- Diferente `VITE_PARISH_ID` y `VITE_PARISH_NAME`