

Configuración de Memoria para Builds

⚠ Actualización Importante

Después de las pruebas, se ha determinado que el proyecto INMOVA requiere **6GB de memoria** para compilar TypeScript exitosamente.

Problema Identificado

Con la configuración anterior de 4GB, TypeScript fallaba con:

```
FATAL ERROR: Ineffective mark-compacts near heap limit  
Allocation failed - JavaScript heap out of memory
```

Solución

Configuración Recomendada

```
NODE_OPTIONS="--max-old-space-size=6144" # 6GB
```

Dónde Configurar

1. Desarrollo Local

Edita `.env.local` :

```
NODE_OPTIONS="--max-old-space-size=6144"
```

2. Scripts de Deployment

Los scripts de deployment deben incluir:

```
NODE_OPTIONS="--max-old-space-size=6144" yarn build
```

3. Variables de Entorno del Servidor

Configura en el servidor de producción:

```
export NODE_OPTIONS="--max-old-space-size=6144"
```

Verificación

Para verificar que TypeScript compila correctamente:

```
cd /home/ubuntu/homming_vidaro/nextjs_space  
NODE_OPTIONS="--max-old-space-size=6144" yarn tsc --noEmit
```



Uso de Memoria por Fase

Fase	Memoria Requerida	Comando
TypeScript Check	6GB	<code>tsc --noEmit</code>
Next.js Build	4-6GB	<code>next build</code>
Development	2-4GB	<code>next dev</code>

Por Qué Se Necesita Más Memoria

El proyecto INMOVA es grande y complejo:

- **70+ rutas** de páginas
- **200+ componentes** TypeScript
- **Múltiples módulos** (STR, Finanzas, Analytics, BI, etc.)
- **Integraciones complejas** (Stripe, DocuSign, Open Banking)
- **Tipos complejos** de Prisma (40+ modelos)

Optimizaciones Implementadas

- ✓ **skipLibCheck: true** - Reduce memoria al no verificar librerías de terceros
- ✓ **Lazy Loading** - Reduce tamaño de bundles, pero no afecta compilación TS
- ✓ **Code Splitting** - Mejora runtime, pero no afecta compilación TS



Alternativas (No Recomendado)

Si 6GB no es suficiente, puedes intentar:

Opción 1: Incrementar a 8GB

```
NODE_OPTIONS="--max-old-space-size=8192"
```

Opción 2: Desactivar verificación de tipos en build

⚠ **No recomendado** - Pierde seguridad de tipos

En `next.config.js`:

```
typescript: {
  ignoreBuildErrors: true, // Ya está en false
}
```

Opción 3: Dividir verificación por módulos

Crear archivos `tsconfig` separados:

- `tsconfig.str.json` - Módulo STR
- `tsconfig.finances.json` - Módulo Finanzas
- `tsconfig.core.json` - Core application

Y verificar cada uno por separado.

Actualizaciones Necesarias

1. Actualizar `.env.example`

✓ Ya actualizado con `NODE_OPTIONS="--max-old-space-size=4096"`

⚠ **TODO:** Actualizar a 6144 si es necesario

2. Actualizar README del Proyecto

Agregar sección sobre requisitos de memoria:

Requisitos del Sistema

- Node.js 18+
- Yarn
- ****Mínimo 6GB de RAM libre**** para builds
- PostgreSQL 14+

3. Actualizar Scripts de CI/CD

Si usas GitHub Actions, Circle CI, etc.:

```
- name: Build
  run: NODE_OPTIONS="--max-old-space-size=6144" yarn build
```



Errores de TypeScript Encontrados

Durante las pruebas con 6GB, se encontraron errores de TypeScript en:

- `lib/str-housekeeping-service.ts`

Estos NO son causados por las optimizaciones, sino que son errores pre-existentes en el código del módulo STR (Short-Term Rental).

Tipos de Errores

1. **Propiedades inexistentes** - `fechaCheckIn`, `fechaCheckOut`, `tipoTurnover`
2. **Enums incorrectos** - `"confirmada"` vs `"CONFIRMADA"`
3. **Campos faltantes** en tipos de Prisma

Recomendación

Estos errores deben ser corregidos por el equipo de desarrollo que mantiene el módulo STR. Probablemente hubo cambios en el schema de Prisma que no se reflejaron en el código.



Checklist de Configuración

- [x] Identificar memoria necesaria (6GB)
- [x] Documentar configuración
- [] Actualizar `.env.example` a 6GB
- [] Actualizar scripts de deployment
- [] Actualizar README con requisitos
- [] Configurar CI/CD con 6GB
- [] Corregir errores de TypeScript en STR module

Contacto

Si tienes problemas con memoria:

1. Verifica que NODE_OPTIONS esté configurado
2. Verifica RAM disponible en tu sistema: `free -h`
3. Considera cerrar otras aplicaciones pesadas
4. Si persiste, incrementa a 8GB