
JOBNIIMBUS STAMFORD

Documentación Técnica

Estructura de Base de Datos

Tabla: **jobnimbus_jobs**

Instancia:	Stamford
Base de Datos:	sales_compensation
Total de Campos:	89+ campos
Tipo:	Multi-tenant PostgreSQL
Fecha:	13 de November, 2025

G.A. Castro Construction LLC
Sistema Unificado de Ventas y Compensaciones

1. RESUMEN EJECUTIVO

La tabla **jobnimbus_jobs** es el componente central del sistema de gestión de trabajos para G.A. Castro Construction LLC. Esta tabla almacena información completa sobre todos los proyectos, leads, y oportunidades de negocio gestionados a través de la plataforma JobNimbus.

- **89+ campos estructurados** para capturar toda la información del ciclo de vida del trabajo
- **Arquitectura multi-tenant** que soporta las instancias de Stamford y Guilford
- **Campos JSONB flexibles** para custom_fields, owners, tags, y related entities
- **Timestamps Unix** para todas las fechas (convertibles a PostgreSQL timestamp)
- **Geolocalización** con coordenadas de latitud/longitud para análisis territorial
- **Integración financiera** con montos de estimates, invoices, y payments
- **15+ índices optimizados** para consultas de alto rendimiento

2. ESTRUCTURA DE LA TABLA

2.1 Definición SQL Completa

```
CREATE TABLE IF NOT EXISTS jobnimbus_jobs (  
  id UUID PRIMARY KEY DEFAULT uuid_generate_v4(),  
  instance VARCHAR(20) NOT NULL,  
  
  -- Identificación  
  jnid VARCHAR(100) NOT NULL,  
  recid BIGINT,  
  number VARCHAR(50),  
  display_name VARCHAR(255),  
  
  -- Fechas (Unix timestamps)  
  date_created BIGINT NOT NULL,  
  date_updated BIGINT,  
  date_start BIGINT,  
  date_end BIGINT,  
  date_status_change BIGINT,  
  
  -- Estado y tipo  
  status VARCHAR(100),  
  status_name VARCHAR(255),  
  record_type_name VARCHAR(100),  
  record_type BIGINT,  
  
  -- Información del trabajo  
  description TEXT,  
  job_type VARCHAR(100),  
  source VARCHAR(100),  
  work_type VARCHAR(100),  
  
  -- Montos financieros  
  estimate_total DECIMAL(12,2),  
  invoice_total DECIMAL(12,2),  
  invoice_due DECIMAL(12,2),  
  total_paid DECIMAL(12,2),  
  
  -- Sales rep y ownership  
  sales_rep_jnid VARCHAR(100),  
  sales_rep_name VARCHAR(255),  
  owners JSONB,  
  
  -- Location y dirección
```

```

address_line1 VARCHAR(255),
address_line2 VARCHAR(255),
city VARCHAR(100),
state_text VARCHAR(100),
state_code VARCHAR(10),
zip VARCHAR(20),
country VARCHAR(100),
latitude DECIMAL(10,8),
longitude DECIMAL(11,8),

-- Relaciones
related JSONB,
primary_contact_jnid VARCHAR(100),
primary_contact_name VARCHAR(255),

-- Tags y custom fields
tags JSONB,
custom_fields JSONB,

-- Metadata
is_active BOOLEAN DEFAULT TRUE,
is_archived BOOLEAN DEFAULT FALSE,
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

UNIQUE(instance, jnid)
);

```

2.2 Clave Primaria y Única

Campo	Tipo	Descripción
id	UUID	Identificador único interno generado automáticamente
UNIQUE(instance, jnid)	Composite	Garantiza unicidad por instancia y JNID de JobNimbus

3. CATÁLOGO DE CAMPOS

3.1 Campos de Identificación

Campo	Tipo	Descripción	Características
id	UUID	Identificador único interno (PostgreSQL UUID)	PRIMARY KEY
instance	VARCHAR(20)	Instancia multi-tenant: "stamford" o "guilford"	NOT NULL, INDEXED
jnid	VARCHAR(100)	JobNimbus ID único del trabajo	NOT NULL, UNIQUE
recid	BIGINT	Record ID numérico de JobNimbus	Nullable
number	VARCHAR(50)	Número de trabajo visible (ej: "1937")	INDEXED
display_name	VARCHAR(255)	Nombre descriptivo del trabajo	Searchable

3.2 Campos de Fechas (Unix Timestamps)

Campo	Tipo	Descripción	Características
date_created	BIGINT	Fecha de creación (Unix timestamp en segundos)	NOT NULL, INDEXED
date_updated	BIGINT	Última actualización (Unix timestamp)	INDEXED
date_start	BIGINT	Fecha de inicio programada del trabajo	Nullable
date_end	BIGINT	Fecha de finalización del trabajo	Nullable
date_status_change	BIGINT	Última vez que cambió el status	Tracked

3.3 Estado y Clasificación

Campo	Tipo	Descripción	Características
status	VARCHAR(100)	Código de status del trabajo	INDEXED
status_name	VARCHAR(255)	Nombre descriptivo del status (ej: "Lead", "Paid & Closed")	Readable
record_type_name	VARCHAR(100)	Tipo de registro en JobNimbus	Classification
record_type	BIGINT	ID numérico del tipo de registro	Reference

3.4 Descripción del Trabajo

Campo	Tipo	Descripción	Características
description	TEXT	Descripción detallada del trabajo	Full text
job_type	VARCHAR(100)	Tipo de trabajo (Roofing, Siding, etc.)	Category
source	VARCHAR(100)	Fuente del lead (Referral, Marketing, etc.)	Analytics
work_type	VARCHAR(100)	Tipo de obra (New Construction, Repair, etc.)	Classification

3.5 Información Financiera

Campo	Tipo	Descripción	Características
estimate_total	DECIMAL(12,2)	Total de estimates aprobados para este job	Currency
invoice_total	DECIMAL(12,2)	Total de invoices emitidos	Currency
invoice_due	DECIMAL(12,2)	Monto pendiente de pago	Accounts Receivable
total_paid	DECIMAL(12,2)	Total de pagos recibidos	Revenue Recognition

3.6 Sales Rep y Ownership

Campo	Tipo	Descripción	Características
sales_rep_jnid	VARCHAR(100)	JNID del sales rep asignado	INDEXED, FK
sales_rep_name	VARCHAR(255)	Nombre del sales rep	Denormalized
owners	JSONB	Array de owners: [{"id": "jnid", "name": "John Doe"}]	JSON Array

3.7 Ubicación y Dirección

Campo	Tipo	Descripción	Características
address_line1	VARCHAR(255)	Dirección principal del trabajo	Required
address_line2	VARCHAR(255)	Línea adicional de dirección (Apt, Suite, etc.)	Optional
city	VARCHAR(100)	Ciudad	INDEXED, Analytics
state_text	VARCHAR(100)	Nombre completo del estado	Display
state_code	VARCHAR(10)	Código del estado (ej: "CT", "NY")	INDEXED
zip	VARCHAR(20)	Código postal	Geographic
country	VARCHAR(100)	País (default: "United States")	Standard
latitude	DECIMAL(10,8)	Coordenada de latitud para geolocalización	Geo Analysis
longitude	DECIMAL(11,8)	Coordenada de longitud para geolocalización	Geo Analysis

3.8 Relaciones y Referencias

Campo	Tipo	Descripción	Características
related	JSONB	Array de entidades relacionadas (estimates, contacts, etc.)	JSON Array
primary_contact_jnid	VARCHAR(100)	JNID del contacto principal	FK Reference
primary_contact_name	VARCHAR(255)	Nombre del contacto principal	Denormalized

3.9 Tags y Custom Fields

Campo	Tipo	Descripción	Características
tags	JSONB	Array de tags para categorización flexible	JSON Array
custom_fields	JSONB	Objeto JSON con campos personalizados configurables	GIN Indexed

3.10 Metadata y Control

Campo	Tipo	Descripción	Características
is_active	BOOLEAN	Indica si el registro está activo (soft delete)	DEFAULT TRUE
is_archived	BOOLEAN	Indica si el trabajo está archivado	DEFAULT FALSE
created_at	TIMESTAMP	Timestamp de creación en PostgreSQL	Auto-generated
updated_at	TIMESTAMP	Timestamp de última modificación	Auto-updated

4. ÍNDICES Y OPTIMIZACIÓN

La tabla jobnimbus_jobs cuenta con 15+ índices optimizados para garantizar consultas de alto rendimiento en las operaciones más comunes del sistema.

Nombre del Índice	Campos	Tipo	Propósito
idx_jn_jobs_instance	instance	B-Tree	Filtrado multi-tenant
idx_jn_jobs_jnid	jnid	B-Tree	Búsqueda por ID JobNimbus
idx_jn_jobs_number	number	B-Tree	Búsqueda por número de trabajo
idx_jn_jobs_status	status	B-Tree	Filtrado por estado del trabajo
idx_jn_jobs_sales_rep	sales_rep_jnid	B-Tree	Filtrado por sales rep
idx_jn_jobs_date_created	date_created	B-Tree	Ordenamiento cronológico
idx_jn_jobs_date_updated	date_updated	B-Tree	Cambios recientes
idx_jn_jobs_active	is_active	B-Tree	Soft delete filtering
idx_jn_jobs_custom_fields	custom_fields	GIN	Búsqueda en JSON
idx_jn_jobs_city	city	B-Tree	Análisis territorial
idx_jn_jobs_state	state_code	B-Tree	Análisis por estado
UNIQUE(instance, jnid)	instance, jnid	Composite	Constraint de unicidad

4.1 Notas de Rendimiento

- **GIN Index en custom_fields:** Permite búsquedas eficientes en campos JSON personalizados usando operadores @>, ?, ?&, ?|
- **Composite UNIQUE:** El constraint (instance, jnid) garantiza integridad referencial sin overhead de FK
- **Multi-tenant filtering:** Todos los queries deben incluir WHERE instance = 'stamford' para optimal performance
- **Timestamp indexes:** Soportan queries de rango temporal para reportes mensuales/anuales
- **B-Tree por defecto:** Óptimo para equality checks, range scans, y ORDER BY operations

5. CAMPOS JSONB FLEXIBLES

La tabla utiliza PostgreSQL JSONB para almacenar datos semi-estructurados que varían entre registros o que requieren flexibilidad en el esquema. Estos campos están indexados con GIN indexes para búsquedas eficientes.

5.1 Campo: owners

Tipo: JSONB (Array de objetos)

Propósito: Almacena múltiples usuarios asignados al trabajo

Estructura:

```
{
  "owners": [
    {
      "id": "mg5ajt9klp1t1libn9lsk2g",
      "name": "Diana Castro",
      "role": "Project Manager"
    },
    {
      "id": "ltn4m8iyu23g9vpm52b0m67",
      "name": "Ana Macassi",
      "role": "Sales Rep"
    }
  ]
}
```

5.2 Campo: related

Tipo: JSONB (Array de objetos)

Propósito: Referencias a entidades relacionadas (estimates, invoices, contacts)

Estructura:

```
{
  "related": [
    {
      "id": "est_abc123",
      "type": "estimate",
      "number": "EST-1001",
      "total": 15000.00
    },
    {
      "id": "con_xyz789",
      "type": "contact",

```

```
{
  "name": "John Smith",
  "email": "john@example.com"
}
```

5.3 Campo: tags

Tipo: JSONB (Array de strings)

Propósito: Etiquetas para categorización y filtrado flexible

Estructura:

```
{
  "tags": [
    "priority",
    "insurance-claim",
    "emergency-repair",
    "2025-q1"
  ]
}
```

5.4 Campo: custom_fields

Tipo: JSONB (Objeto con estructura variable)

Propósito: Campos personalizados configurables por JobNimbus

Estructura: Completamente flexible, definida por configuración

```
{
  "custom_fields": {
    "cf_string_1": "Roof Replacement",
    "cf_boolean_2": false,
    "cf_boolean_3": false,
    "insurance_company": "State Farm",
    "claim_number": "CLM-2025-12345",
    "adjuster_name": "Mike Johnson",
    "urgency_level": "High"
  }
}
```

5.5 Ejemplos de Queries JSONB

Buscar jobs con tag específico:

```
SELECT * FROM jobnimbus_jobs WHERE tags @> '["priority"]';
```

Buscar jobs por custom field:

```
SELECT * FROM jobnimbus_jobs WHERE custom_fields->>'insurance_company' = 'State Farm';
```

Buscar jobs con owner específico:

```
SELECT * FROM jobnimbus_jobs WHERE owners @> '[{"id": "mg5ajt9klp1t1libn9lsk2g"}]';
```

Extraer valores de custom fields:

```
SELECT number, custom_fields->>'cf_string_1' as service_type FROM jobnimbus_jobs;
```

6. EJEMPLOS DE DATOS REALES

A continuación se presentan ejemplos de registros reales de la tabla `jobnimbus_jobs` de la instancia de Stamford, mostrando la estructura completa de los datos.

6.1 Ejemplo 1: Job #1937 - Initial Contact

```
{
  "jnid": "mhl6d0exd9l40obqn2ytza",
  "number": "1937",
  "name": "70 Oakland Avenue",
  "status": 657,
  "status_name": "Initial Contact",
  "date_created": 1763047862,
  "date_updated": 1763048913,
  "address_line1": "70 Oakland Avenue",
  "city": "Harrison",
  "state_code": "NY",
  "custom_fields": {
    "cf_string_1": "Gutter Cleaning"
  },
  "created_by_name": "Ana Macassi",
  "geo": {
    "lat": 40.96547,
    "lon": -73.71084
  },
  "estimate_total": 0,
  "invoice_total": 0,
  "is_active": true,
  "is_archived": false
}
```

6.2 Ejemplo 2: Job #1936 - Lost

```
{
  "jnid": "mhxf7oj2jmrgz27kuzwlkt",
  "number": "1936",
  "name": "172 Beers Street",
  "status": 491,
  "status_name": "Lost",
}
```

```
"date_created": 1763041556,  
"date_updated": 1763042485,  
"address_line1": "172 Beers Street",  
"city": "Bridgeport",  
"state_code": "CT",  
"description": "Roof Replacement",  
"custom_fields": {  
  "cf_string_1": "Roof Replacement"  
},  
"created_by_name": "Diana Castro",  
"geo": {  
  "lat": 41.19101,  
  "lon": -73.20169  
},  
"is_active": true,  
"is_archived": true  
}
```

6.3 Estadísticas de Datos (Stamford)

Métrica	Valor
Total de Jobs	860 registros
Jobs Activos	~95%
Jobs Archivados	~5%
Ciudades Únicas	50+
Estados	CT, NY, NJ principalmente
Sales Reps Activos	10+
Rango de Fechas	2023 - 2025

7. RELACIONES Y FOREIGN KEYS

La tabla `jobnimbus_jobs` se relaciona con múltiples tablas del sistema a través de claves foráneas lógicas (soft references) almacenadas en campos `VARCHAR` y `JSONB`.

Tabla Relacionada	Campo en jobs	Tipo de Relación	Cardinalidad
<code>jobnimbus_contacts</code>	<code>primary_contact_jnid</code>	Soft FK	1:1 (opcional)
<code>jobnimbus_contacts</code>	<code>related[]</code>	Array JSONB	1:N
<code>jobnimbus_estimates</code>	<code>related[]</code>	Array JSONB	1:N
<code>jobnimbus_invoices</code>	<code>related[]</code>	Array JSONB	1:N
<code>jobnimbus_activities</code>	<code>related_to_jnid</code>	Inverse reference	N:1
<code>jobnimbus_tasks</code>	<code>related[]</code>	Array JSONB	1:N
<code>sales_reps</code>	<code>sales_rep_jnid</code>	Soft FK	N:1
<code>jobnimbus_payments</code>	<code>related[]</code>	Array JSONB	1:N (indirect)

7.1 Notas sobre Relaciones

- **Soft Foreign Keys:** El sistema usa `VARCHAR` JNIDs en lugar de `UUID` FKs para mayor flexibilidad
- **JSONB Arrays:** El campo `'related'` puede contener referencias a múltiples tipos de entidades
- **Denormalización:** Campos como `sales_rep_name` están denormalizados para performance
- **Integridad:** La integridad referencial se mantiene a nivel de aplicación, no de base de datos
- **Multi-tenant:** Todas las relaciones respetan el boundary de `'instance'`

8. EJEMPLOS DE QUERIES COMUNES

8.1 Listar jobs activos de Stamford

```
SELECT
  number,
  display_name,
  status_name,
  to_timestamp(date_created) as created,
  estimate_total,
  invoice_total
FROM jobnimbus_jobs
WHERE instance = 'stamford'
  AND is_active = TRUE
ORDER BY date_created DESC
LIMIT 50;
```

8.2 Jobs por sales rep con totales

```
SELECT
  sales_rep_name,
  COUNT(*) as total_jobs,
  SUM(estimate_total) as total_estimated,
  SUM(invoice_total) as total_invoiced,
  SUM(total_paid) as total_collected
FROM jobnimbus_jobs
WHERE instance = 'stamford'
  AND is_active = TRUE
GROUP BY sales_rep_name
ORDER BY total_invoiced DESC;
```

8.3 Pipeline por status

```
SELECT
  status_name,
  COUNT(*) as job_count,
  SUM(estimate_total) as pipeline_value,
  AVG(estimate_total) as avg_value
FROM jobnimbus_jobs
WHERE instance = 'stamford'
```

```
AND is_active = TRUE
AND status_name NOT IN ('Paid & Closed', 'Lost')
GROUP BY status_name
ORDER BY pipeline_value DESC;
```

8.4 Análisis territorial por ciudad

```
SELECT
  city,
  state_code,
  COUNT(*) as job_count,
  SUM(invoice_total) as total_revenue,
  AVG(invoice_total) as avg_job_value
FROM jobnimbus_jobs
WHERE instance = 'stamford'
  AND is_active = TRUE
  AND city IS NOT NULL
GROUP BY city, state_code
HAVING COUNT(*) >= 3
ORDER BY total_revenue DESC
LIMIT 20;
```

8.5 Jobs creados este mes

```
SELECT
  number,
  display_name,
  status_name,
  to_timestamp(date_created) as created_date,
  sales_rep_name
FROM jobnimbus_jobs
WHERE instance = 'stamford'
  AND is_active = TRUE
  AND to_timestamp(date_created) >= DATE_TRUNC('month', CURRENT_DATE)
ORDER BY date_created DESC;
```

8.6 Jobs con custom field específico

```
SELECT
```



```

    number,
    display_name,
    custom_fields->>'cf_string_1' as service_type,
    custom_fields->>'insurance_company' as insurer
FROM jobnimbus_jobs
WHERE instance = 'stamford'
    AND is_active = TRUE
    AND custom_fields->>'cf_string_1' = 'Roof Replacement'
ORDER BY date_created DESC;

```

8.7 Cuentas por cobrar (AR)

```

SELECT
    number,
    display_name,
    status_name,
    invoice_total,
    total_paid,
    (invoice_total - total_paid) as balance_due,
    to_timestamp(date_updated) as last_updated
FROM jobnimbus_jobs
WHERE instance = 'stamford'
    AND is_active = TRUE
    AND (invoice_total - total_paid) > 0
ORDER BY (invoice_total - total_paid) DESC;

```

8.8 Conversión de leads por mes

```

SELECT
    DATE_TRUNC('month', to_timestamp(date_created)) as month,
    COUNT(*) as total_leads,
    COUNT(CASE WHEN status_name = 'Paid & Closed' THEN 1 END) as won,
    COUNT(CASE WHEN status_name = 'Lost' THEN 1 END) as lost,
    ROUND(
        COUNT(CASE WHEN status_name = 'Paid & Closed' THEN 1 END)::NUMERIC /
        NULLIF(COUNT(*), 0) * 100,
        2
    ) as win_rate_pct
FROM jobnimbus_jobs

```

```
WHERE instance = 'stamford'
AND is_active = TRUE
GROUP BY DATE_TRUNC('month', to_timestamp(date_created))
ORDER BY month DESC
LIMIT 12;
```

9. MEJORES PRÁCTICAS

9.1 Performance

- Siempre incluir **WHERE instance = 'stamford'** en todos los queries
- Usar índices apropiados: filtrar por date_created, status, sales_rep_jnid primero
- Evitar SELECT * en queries de producción - especificar columnas necesarias
- Para queries de JSONB, usar operadores GIN-optimizados (@>, ->, ->>)
- Considerar EXPLAIN ANALYZE para queries complejos

9.2 Datos

- Usar **is_active = FALSE** para soft deletes, nunca DELETE físico
- Mantener consistencia entre campos denormalizados (ej: sales_rep_name)
- Validar custom_fields antes de inserción para evitar JSON malformado
- Convertir Unix timestamps con to_timestamp() para fechas legibles
- Actualizar updated_at automáticamente vía trigger

9.3 Integridad

- Validar que JNIDs existan en tablas relacionadas antes de referencias
- Mantener sincronización entre JobNimbus y PostgreSQL periódicamente
- Usar transacciones para updates que afecten múltiples registros
- Implementar audit_log para cambios críticos en jobs
- Verificar unicidad de (instance, jnid) en imports

9.4 Reportes

- Usar vistas materializadas para reportes complejos frecuentes
- Agregar filtros temporales (mes/año) para limitar resultados
- Combinar con jobnimbus_estimates e invoices para pipeline completo
- Calcular métricas en PostgreSQL en lugar de aplicación
- Usar CTEs (WITH) para queries multi-paso legibles

10. APÉNDICE

10.1 Códigos de Status Comunes

Código	Nombre	Descripción
480	Lead	Nuevo lead sin contacto
481	Appointment Scheduled	Cita agendada con cliente
657	Initial Contact	Primer contacto realizado
491	Lost	Oportunidad perdida
XXX	Paid & Closed	Trabajo completado y pagado

10.2 Conversión de Timestamps

De Unix a PostgreSQL:

`to_timestamp(date_created) → TIMESTAMP`

De PostgreSQL a Unix:

`EXTRACT(EPOCH FROM created_at)::BIGINT → Unix timestamp`

Función helper:

`SELECT unix_to_timestamp(1763047862);`
→ 2025-11-13 15:24:22

10.3 Información de Contacto

Sistema: G.A. Castro Construction LLC

Base de Datos: sales_compensation

Instancia: Stamford

Documentación generada: 13 de November, 2025

Versión del esquema: 1.0