🗄 **Avenir AI Solutions**

Supabase Schema Reference

Version: 2.0.0

Last Updated: October 17, 2025

Database: PostgreSQL 15 (Supabase)

Status: Production Ready

📊 Schema Overview

Database: public

Total Tables: 8

Table	Purpose	Row Count (Est.)	Primary Use
`clients`	Client accounts & settings	100-1,000	Authentication, API keys
`lead_memory`	Enriched leads	10,000-100,000	Lead storage, AI data
`lead_actions`	Lead history	50,000-500,000	Activity tracking
`prospect_candidates`	Discovered prospects	1,000-10,000	Outreach targets
`growth_brain`	AI learning & analytics	1,000-10,000	Intelligence engine
`prospect_outreach_log`	Outreach tracking	5,000-50,000	Email logs

1. Table: public.clients

Client accounts with personalization and SMTP settings

Key Columns

Column	Type	Nullable	Description
id	UUID	No	Internal database ID
client_id	UUID	No	Unique client identifier (used in all relationships)
business_name	TEXT	No	Company name
email	TEXT	No	Login email (unique)
api_key	TEXT	No	Authentication token for lead API
password_hash	TEXT	No	bcryptjs hashed password
language	TEXT	No	Preferred language (en/fr)
industry_category	TEXT	Yes	Industry classification
email_tone	TEXT	No	Email tone (Friendly, Professional, Formal, Energetic)
is_test	BOOLEAN	No	Auto-detected test data flag
is_internal	BOOLEAN	No	Flag for Avenir's internal client

## 2. Table: public.lead_memory

Core lead storage with AI enrichment data

Key Columns

Column	Type	Description
`id`	UUID	Lead ID (PK)
`client_id`	UUID	→ clients.client_id (FK)
`name`	TEXT	Lead name
`email`	TEXT	Lead email
`intent`	TEXT	AI-detected intent
`tone`	TEXT	Communication tone
`urgency`	TEXT	Urgency level (High/Medium/Low)
`ai_summary`	TEXT	AI-generated summary
`confidence_score`	NUMERIC(3,2)	Lead quality score (0.0-1.0)
`current_tag`	TEXT	Active, Follow-Up, Converted, etc.

Current Tag Values

- **Active** — New or actively being pursued
- **Follow-Up** — Requires follow-up action
- **Converted** — Successfully converted to customer
- **Archived** — Archived for later review
- **Deleted** — Soft-deleted

5. Table: public.growth_brain ⭐

AI learning snapshots, growth analytics, and feedback loop

> 🧠 **Central Intelligence Hub**
>
> This table powers the AI feedback loop by storing learning snapshots, pattern detection, conversion tracking, and predictive analytics. Every significant event and insight flows through growth_brain.

Complete Schema

```
CREATE TABLE public.growth_brain (
  id                 UUID PRIMARY KEY DEFAULT gen_random_uuid(),
  client_id          UUID REFERENCES clients(client_id) ON DELETE CASCADE,
  event_type         TEXT NOT NULL,
  learning_snapshot  JSONB,
  insight_text       TEXT,
  confidence         NUMERIC(3,2) CHECK (confidence >= 0 AND confidence <= 1),
  created_at         TIMESTAMPTZ DEFAULT NOW()
);

-- Indexes
CREATE INDEX idx_growth_brain_client_id ON growth_brain(client_id);
CREATE INDEX idx_growth_brain_event_type ON growth_brain(event_type);
CREATE INDEX idx_growth_brain_created_at ON growth_brain(created_at DESC);
CREATE INDEX idx_growth_brain_confidence ON growth_brain(confidence DESC);
CREATE INDEX idx_growth_brain_snapshot ON growth_brain USING GIN (learning_snapshot);
```

Event Types

Event Type	Description	Triggered By
`conversion`	Lead converted to customer	Tag change to "Converted"
`reversion`	Converted lead reverted	Revert modal action
`pattern_detected`	New behavioral pattern	Intelligence engine
`engagement_score`	Overall engagement level	Daily cron job
`urgency_trend`	Urgency pattern over time	Intelligence engine
`tone_distribution`	Tone analysis	Intelligence engine
`conversion_prediction`	Future conversion likelihood	Predictive model

Learning Snapshot Structure Examples

Conversion Event

```json
{
  "lead_id": "a1b2c3d4-...",
  "lead_name": "Sarah Chen",
  "conversion_value": 5000,
  "days_to_convert": 7,
  "touch_points": 3,
  "initial_urgency": "High",
  "final_tag": "Converted"
}
```

Pattern Detection Event

```json
{
  "pattern_type": "urgency_increasing",
  "affected_leads": 23,
  "time_period": "last_7_days",
  "confidence": 0.89,
  "recommendation": "Increase follow-up frequency",
  "data_points": [
    {"date": "2025-10-10", "avg_urgency": "Medium"},
    {"date": "2025-10-17", "avg_urgency": "High"}
  ]
}
```

Engagement Score Event

```json
{
  "overall_score": 87,
  "trend": "increasing",
  "period": "last_30_days",
  "metrics": {
    "total_leads": 150,
    "active_leads": 120,
    "avg_confidence": 0.85,
    "high_urgency_pct": 0.34
  },
  "top_sources": ["organic_search", "referral"],
  "top_industries": ["Real Estate", "Legal"]
}
```

Conversion Prediction Event

```json
{
  "predicted_conversions": 12,
  "prediction_window": "next_30_days",
  "confidence": 0.78,
  "high_value_leads": [
    {"lead_id": "...", "probability": 0.92, "estimated_value": 3000}
  ],
  "recommended_actions": [
    "Follow up with leads tagged 'Follow-Up'",
    "Focus on high-confidence leads (>0.85)"
  ]
}
```

🔄 AI Feedback Loop Integration

How growth_brain Powers the System

1. Record Events — Every significant action (conversion, pattern detection) is logged with context

2. Analyze Patterns — Intelligence engine queries historical snapshots to identify trends

3. Generate Predictions — Machine learning models use historical data to predict future outcomes

4. Improve Accuracy — System continuously learns from actual outcomes vs predictions

Example: Conversion Tracking Flow

```javascript
// When a lead is converted
async function onLeadConverted(lead, client_id) {
  // 1. Calculate metrics
  const daysToConvert = calculateDays(lead.timestamp, Date.now());
  const touchPoints = await countTouchPoints(lead.id);

  // 2. Record learning snapshot
  await supabase.from('growth_brain').insert({
    client_id,
    event_type: 'conversion',
    learning_snapshot: {
      lead_id: lead.id,
      initial_urgency: lead.urgency,
      days_to_convert: daysToConvert,
      touch_points: touchPoints
    },
    insight_text: `Lead "${lead.name}" converted in ${daysToConvert} days`,
    confidence: 0.95
  });
}
```

Example: Prediction Generation

```javascript
// Generate predictions from historical conversions
async function predictConversions(client_id) {
  // 1. Fetch historical conversions
  const { data: conversions } = await supabase
    .from('growth_brain')
    .select('learning_snapshot')
    .eq('client_id', client_id)
    .eq('event_type', 'conversion')
    .order('created_at', { ascending: false })
    .limit(100);

  // 2. Analyze patterns
  const avgDaysToConvert = calculateAverage(
    conversions.map(c => c.learning_snapshot.days_to_convert)
  );

  // 3. Generate prediction
  const prediction = {
    predicted_conversions: estimateConversions(activeLeads, avgDaysToConvert),
    confidence: 0.78,
    based_on: conversions.length + ' historical conversions'
  };

  // 4. Save prediction to growth_brain
  await supabase.from('growth_brain').insert({
    client_id,
    event_type: 'conversion_prediction',
    learning_snapshot: prediction,
    insight_text: `Predicted ${prediction.predicted_conversions} conversions`,
    confidence: prediction.confidence
  });
}
```
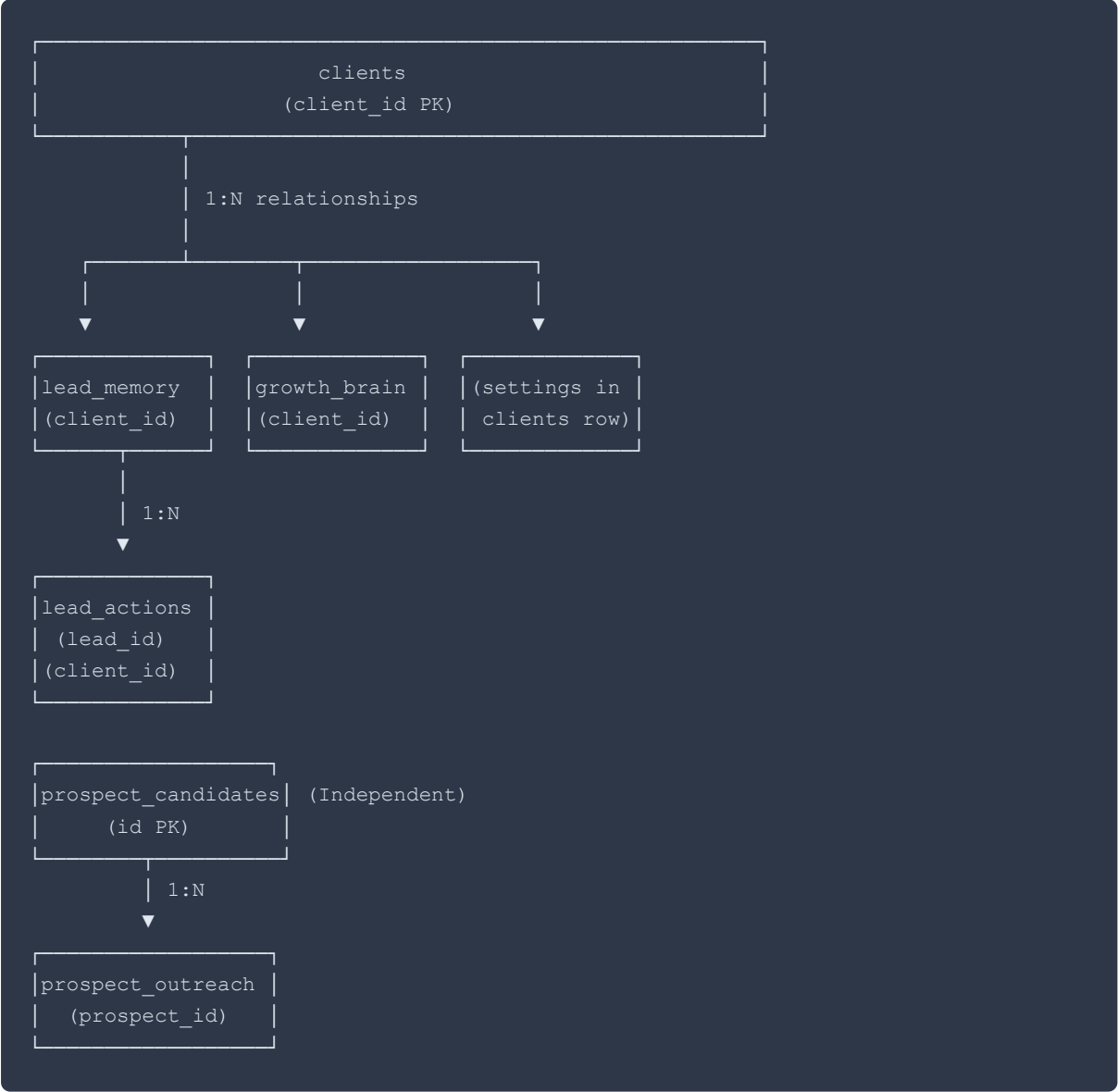
🔗 Relationships & Foreign Keys

Entity Relationship Diagram

```
┌──────────────────────────────────────────┐
│                clients                   │
│              (client_id PK)              │
└──────────────────────────────────────────┘
         │
         │ 1:N relationships
         │
   ┌─────┴───────────────────┐
   │                 │       │
   ▼                 ▼       ▼
┌──────────┐  ┌──────────┐  ┌──────────────┐
│lead_memory│  │growth_brain│ │(settings in  │
│(client_id)│  │(client_id) │ │ clients row) │
└──────────┘  └──────────┘  └──────────────┘
   │
   │ 1:N
   ▼
┌──────────┐
│lead_actions│
│  (lead_id) │
│(client_id) │
└──────────┘


┌──────────────────┐
│prospect_candidates│  (Independent)
│      (id PK)      │
└──────────────────┘
      │ 1:N
      ▼
┌──────────────────┐
│prospect_outreach │
│   (prospect_id)  │
└──────────────────┘
```

Foreign Key Constraints

Table	Column	References	On Delete
lead_memory	client_id	clients(client_id)	CASCADE
lead_actions	lead_id	lead_memory(id)	CASCADE
lead_actions	client_id	clients(client_id)	CASCADE
growth_brain	client_id	clients(client_id)	CASCADE
prospect_outreach_log	prospect_id	prospect_candidates(id)	CASCADE

> ⚠️ **CASCADE Behavior:**
>
> When a client is deleted, all related records are automatically deleted:
>
> - All `lead_memory` rows
> - All `lead_actions` rows
> - All `growth_brain` rows

📊 Query Examples

Fetch Latest Insights for Client

```
SELECT
  event_type,
  insight_text,
  learning_snapshot->'metrics' as metrics,
  confidence,
  created_at
FROM growth_brain
WHERE client_id = '550e8400-e29b-41d4-a716-446655440000'
ORDER BY created_at DESC
LIMIT 10;
```

Find All Conversion Events

```
SELECT
  client_id,
  learning_snapshot->>'lead_name' as lead_name,
  learning_snapshot->>'days_to_convert' as days,
  created_at
FROM growth_brain
WHERE event_type = 'conversion'
  AND (learning_snapshot->>'days_to_convert')::integer <= 7
ORDER BY created_at DESC;
```

Calculate Average Conversion Time

```
SELECT
  AVG((learning_snapshot->>'days_to_convert')::numeric) as avg_days,
  COUNT(*) as total_conversions,
  MIN((learning_snapshot->>'days_to_convert')::numeric) as fastest,
  MAX((learning_snapshot->>'days_to_convert')::numeric) as slowest
FROM growth_brain
WHERE event_type = 'conversion'
  AND client_id = '550e8400-e29b-41d4-a716-446655440000';
```

Find High-Confidence Predictions

```sql
SELECT
  event_type,
  insight_text,
  confidence,
  learning_snapshot->'predicted_conversions' as predictions,
  created_at
FROM growth_brain
WHERE confidence >= 0.8
  AND event_type LIKE '%prediction%'
ORDER BY confidence DESC, created_at DESC
LIMIT 20;
```

Avenir AI Solutions — Supabase Schema Reference

Version 2.0.0 | October 17, 2025

Database: PostgreSQL 15 (Supabase)