Can AI Build Your Next API Integration?

Lessons from Pandium's Experimentation with LLMs

Presented by Shon Urbas

Agenda

1. What is Pandium
2. Our AI Journey: The Early Days
3. The Smallest Problem: JSON Key Path Mapping
4. Building the Foundation
5. A Rekindling: AI in Production July, 2024
6. The Revival: Q4 2024 Success
7. Live Demo
8. Key Lessons Learned
9. The Future of AI in Integration Development
10. Closing: The Real AI Revolution

Shon Urbas

linkedin.com/in/shon-urbas/

- CTO & Co-Founder of Pandium
- Former Platform Enginering Manager at Handshake (Aquired by Shopify)

What is Pandium?

pandium
integrate all the things

What is Pandium?

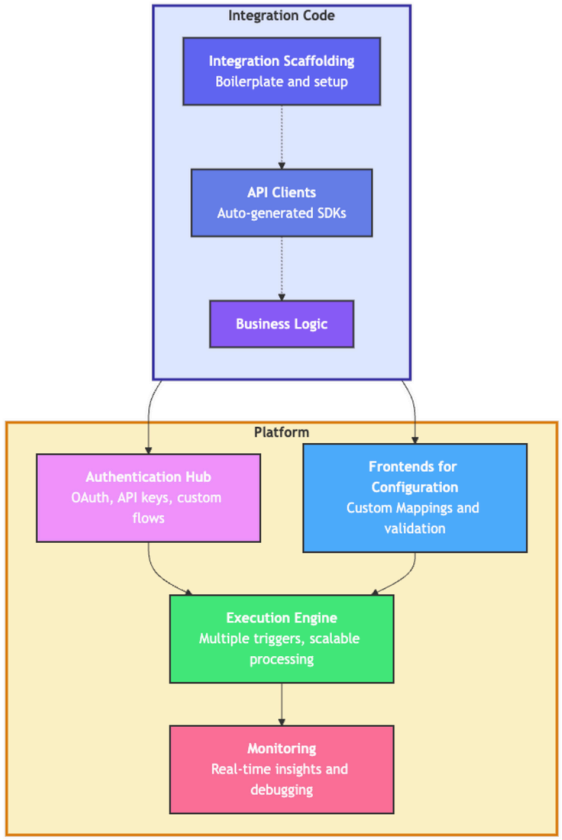
The Integration Challenge

Our typical engineer users face:

- **Time sink:** 60-80% of development time on integrations, not core features
- **Complexity creep:** Each integration touches authentication, data mapping, error handling, rate limiting
- **Maintenance burden:** APIs change, breaking existing integrations

What is Pandium?

Pandium's Solution



Our AI Journey: The Early Days

pandium
integrate all the things

Our AI Journey: The Early Days

Phase 1: The "Whole Shabang" Approach (Birth of ChatGPT Era)

Late 2022: Ambitious but Naive

- **Goal**: Generate entire integrations from natural language prompts
- **Reality**: Models produced syntactically correct but functionally broken code

Example: *"Connect Salesforce to Slack"* → 200 lines of code that didn't actually work, forget about at scale

Our AI Journey: The Early Days

The Expert Partnership Experiment

Mid-2023: Bringing in the Consultants

- Partnered with AI consulting firms
- Promise: "We'll make this production-ready"
- **Results:**
  - 3x longer timelines then quoted
  - Couldn’t even approach doing the whole thing
  - Drove us to think smaller so that's good, but even failed at that
  - **Key insight:** Hype was ahead of practicality

Our AI Journey: The Early Days

The Pivot: From Sky-High to Surgical

Discovery: Most integration work doesn't need AI

- **90%** can be handled deterministically
- **10%** requires intelligent transformation
- **Focus shift:** Target the 90% that we can, and give the AI stuff more time.

The Smallest Problem: JSON Key Path Mapping

pandium
integrate all the things

The Smallest Problem: JSON Key Path Mapping

The "Aha!" Moment

Problem: Mapping arbitrary JSON structures

```
// Source API
{
  "customer_info": {
    "full_name": "John Doe",
    "contact": {"email": "john@example.com"}
  }
}
// Target API expects
{
  "name": "John Doe",
  "email_address": "john@example.com"
}
```

```
$.customer_info.full_name -> $.name
$.customer_info.contact.email -> $.email_address
```

Building the Foundation

pandium
integrate all the things

Building the Foundation

What We Automated Deterministically

Q4 2024: Focus on the 90%

API Client Generation

- Parse OpenAPI specs
- Generate type-safe SDKs
- Handle authentication patterns

Project Scaffolding

- Standard project structure
- Configuration templates
- Testing frameworks

Documentation

- Auto-generated API docs
- Integration guides
- Specification

A Rekindling: AI in Production July, 2024

pandium
integrate all the things

A Rekindling: AI in Production July, 2024

Fresh off our JSON mapping struggles, I was convinced AI was all hype

- **My brilliant idea:** Submit talk proposal "AI Integration Hype vs Reality: Why It Doesn't Work"
- **Conference committee's response:** 🚫 REJECTED 🚫

What I Learned:

- Peoples enthusiasm in exploring the space was infectious
- Not all the things shown were production ready, but that wasn’t really the point
- Lots of clever design patterns.
- Early glimpses of what we'd later call "agentic programming"

Key insight: Reinforced the thought that by working on the deterministic parts first, we could build a foundation for a successful AI model

The Revival: Q4 2024 Success

pandium
integrate all the things

The Revival: Q4 2024 Success

What Changed?

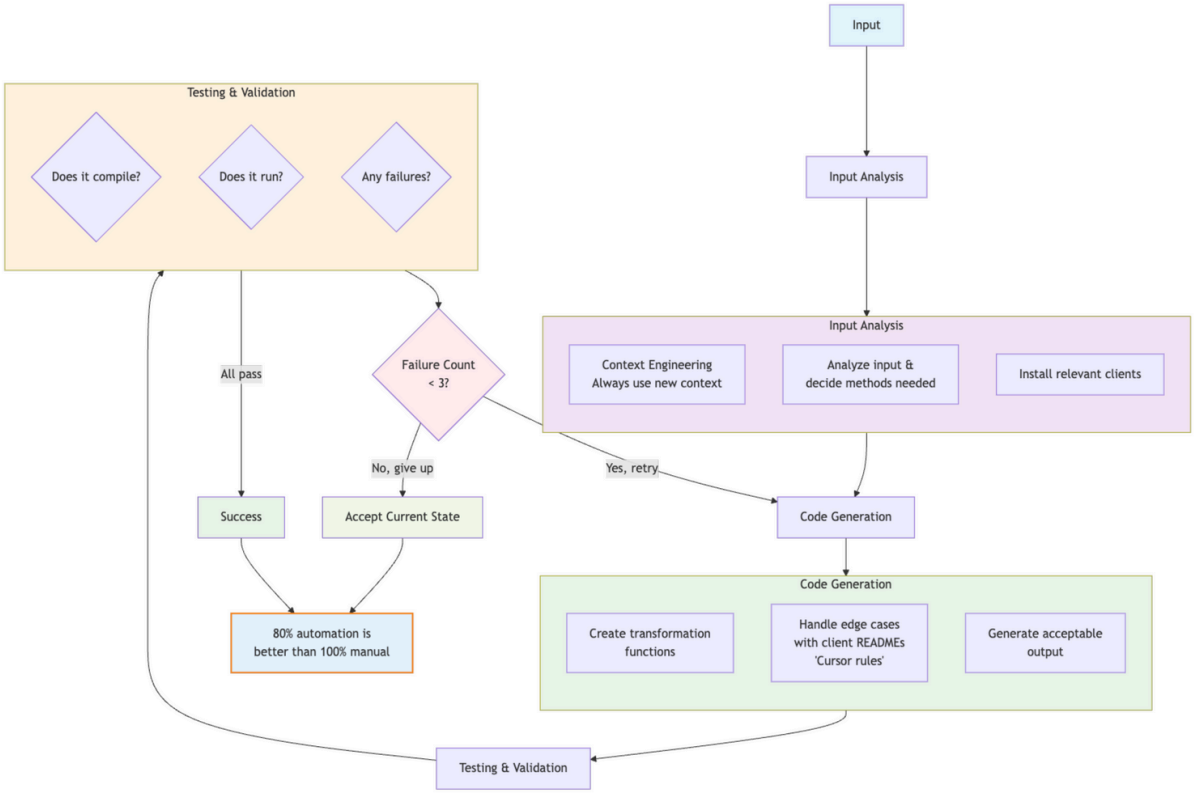
Technology maturation:

- Much larger contexts, but still has constraints.
- “Context Engineering:” i.e. making sure only needed info is in prompt
- More reliable code generation

Our refined “Agentic” approach:

- Narrow AI scope to mapping functions only (a huge upgrade from just JSON PATHs)
- Robust testing and validation
- Iterative failure handling

The Revival: Q4 2024 Success



The Revival: Q4 2024 Success

The "Give Up Early" Principle

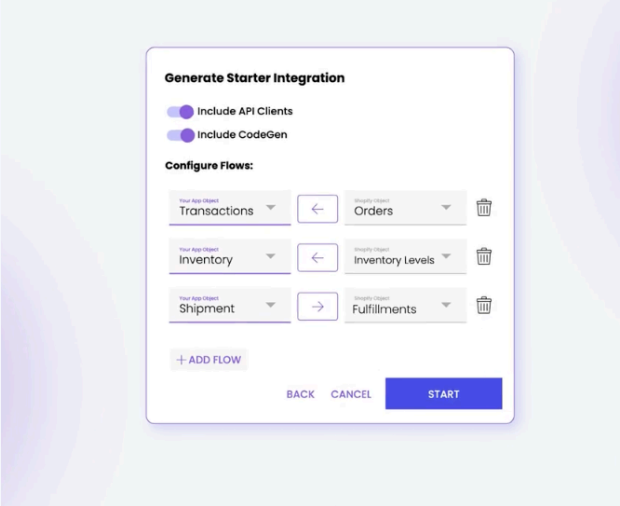
Key insight: Not everything needs to be perfect. i.e. We only try 3 times, for a successful flow before moving on.

- **Deterministic parts:** Keep them deterministic
- **AI parts:** Allow controlled failure
- **Success criteria:** 90% automation is better than 100% manual

Example

pandium
integrate all the things

Example: Inputs



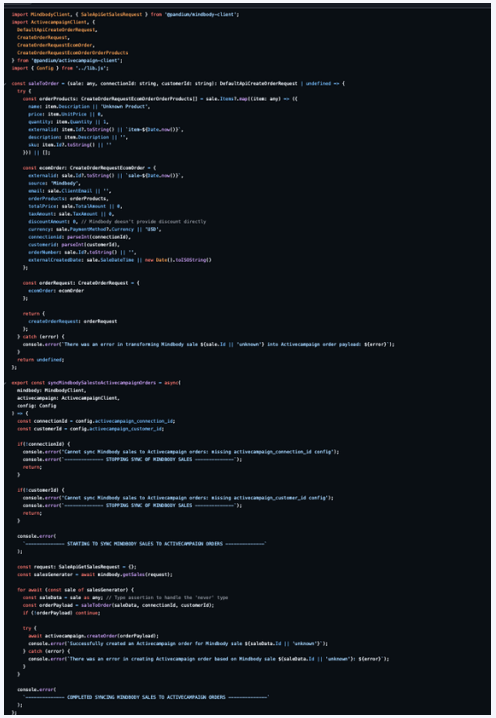
The “Prompt” for user:

Connector A: ActiveCampaign
Resource A: Sales
Connector B: MindBody
Resource B: Order

Example: Method Defs

Example of a “Product”

Example: All + Imports



All the Code Generated

```
import MindbodyClient, { SaleApiGetSalesRequest } from '@pandium/mindbody-client';
import ActivecampaignClient, {
  DefaultApiCreateOrderRequest,
  CreateOrderRequest,
  CreateOrderRequestEcomOrder,
  CreateOrderRequestEcomOrderOrderProducts
} from '@pandium/activecampaign-client';
import { Config } from '../lib.js';
```

Imports

Example: Mapper

```typescript
const saleToOrder = (sale: any, connectionId: string, customerId: string): DefaultApiCreateOrderRequest | undefined => {
  try {
    const orderProducts: CreateOrderRequestEcomOrderOrderProducts[] = sale.Items?.map((item: any) => ({
      name: item.Description || 'Unknown Product',
      price: item.UnitPrice || 0,
      quantity: item.Quantity || 1,
      externalid: item.Id?.toString() || `item-${Date.now()}`,
      description: item.Description || '',
      sku: item.Id?.toString() || ''
    })) || [];

    const ecomOrder: CreateOrderRequestEcomOrder = {
      externalid: sale.Id?.toString() || `sale-${Date.now()}`,
      source: 'Mindbody',
      email: sale.ClientEmail || '',
      orderProducts: orderProducts,
      totalPrice: sale.TotalAmount || 0,
      taxAmount: sale.TaxAmount || 0,
      discountAmount: 0, // Mindbody doesn't provide discount directly
      currency: sale.PaymentMethod?.Currency || 'USD',
      connectionid: parseInt(connectionId),
      customerid: parseInt(customerId),
      orderNumber: sale.Id?.toString() || '',
      externalCreatedDate: sale.SaleDateTime || new Date().toISOString()
    };

    const orderRequest: CreateOrderRequest = {
      ecomOrder: ecomOrder
    };

    return {
      createOrderRequest: orderRequest
    };
  } catch (error) {
    console.error(`There was an error in transforming Mindbody sale ${sale.Id || 'unknown'} into Activecampaign order payload: ${error}`);
  }
  return undefined;
};
```

Example: The loop

```
export const syncMindbodySalestoActivecampaignOrders = async(
  mindbody: MindbodyClient,
  activecampaign: ActivecampaignClient,
  config: Config
) => {
  const connectionId = config.activecampaign_connection_id;
  const customerId = config.activecampaign_customer_id;

  if(!connectionId) {
    console.error('Cannot sync Mindbody sales to Activecampaign orders: missing activecampaign_connection_id config');
    console.error(`-------------- STOPPING SYNC OF MINDBODY SALES --------------`);
    return;
  }

  if(!customerId) {
    console.error('Cannot sync Mindbody sales to Activecampaign orders: missing activecampaign_customer_id config');
    console.error(`-------------- STOPPING SYNC OF MINDBODY SALES --------------`);
    return;
  }

  console.error(
    `-------------- STARTING TO SYNC MINDBODY SALES TO ACTIVECAMPAIGN ORDERS --------------`
  );

  const request: SaleApiGetSalesRequest = {};
  const salesGenerator = await mindbody.getSales(request);

  for await (const sale of salesGenerator) {
    const saleData = sale as any; // Type assertion to handle the 'never' type
    const orderPayload = saleToOrder(saleData, connectionId, customerId);
    if (!orderPayload) continue;

    try {
      await activecampaign.createOrder(orderPayload);
      console.error(`Successfully created an Activecampaign order for Mindbody sale ${saleData.Id || 'unknown'}`);
    } catch (error) {
      console.error(`There was an error in creating Activecampaign order based on Mindbody sale ${saleData.Id || 'unknown'}: ${error}`);
    }
  }

  console.error(
    `-------------- COMPLETED SYNCING MINDBODY SALES TO ACTIVECAMPAIGN ORDERS --------------`
  );
};
```

Example: Result

99.999% deterministic code

User given (including API clients) ~200k of code to start integration

With 96 lines trully generated by LLM. Rest is all normal Code Gen

Key Lessons Learned

pandium
integrate all the things

Key Lessons Learned

Technical Lessons

- Scope Limitation: AI excels at narrow, well-defined tasks
- Hybrid Approach: Combine AI with deterministic systems
- Validation is Critical: AI output must be thoroughly tested
- Failure Recovery: Build systems that handle AI mistakes gracefully

Key Lessons Learned

Business Lessons

- Solution-First Thinking: Find problems that fit AI capabilities
- Avoid "AI for AI's Sake": Value must be demonstrable
- Developer Experience: AI should enhance, not replace, developer control
- Iterative Development: Start small, prove value, scale up

The Future of AI in Integration Development

pandium
integrate all the things

The Future of AI in Integration Development

What's Next for Pandium

Short-term (next 3 months):

- Multiple Objects to Single Objects Flows
- Event Based
- CLI Access

Medium-term (3+ months):

- Forking and Build (i.e. I have a Xero Integration... now make a QBO one.)
- AI Assisted API Client generation
- Automated performance tuning

The Future of AI in Integration Development

Industry Predictions

What will happen:

- AI will augment, not replace, developers
- Hybrid human-AI workflows will become standard
- Domain-specific AI tools will outperform general-purpose ones (just like real life)

What won't happen:

- Complete automation of complex integrations
- AI replacing the need for technical expertise
- One-size-fits-all AI solutions

Closing: The Real AI Revolution

pandium
integrate all the things

Closing: The Real AI Revolution

The Bottom Line

AI's true value in integration development:

- Not: Building entire systems from scratch
- But: Accelerating the tedious, repetitive work
- Result: Developers focus on business logic

Questions?

pandium
integrate all the things

pandium
integrate all the things

Shon Urbas

Linkedin.com/in/shon-urbas/

Subscribe to the podcast to hear conversations about integrations and ecosystems

Thank You! Any Questions?