



## Project Overview

**Project Name:**

NetSuite Invoices to Wells Fargo eBill Express Automation

**Industry:**

SaaS / Professional Services

**Role:**

Boomi Integration Developer / Architect

**Project Duration:**

6 Months Schedules / Delivered in 4 Months

## Business Objective

Company has a C# .NET application that aggregates invoice data into two different files: a .csv file which contains data for Invoices that have had some type of payment applied; or a .zip file which contains the .csv listing all new invoices along with their corresponding .pdf.

Support has been difficult as there are many moving pieces, departments involved and points of failure: Boomi Integration populates a SQL table with all Invoice data, including updates; Data Services Team uses that table, applies their own daily processes and creates Views for New and Update Invoices; finally, the Application Development Team's C# application reads those SQL Views to generate the appropriate file.

To further complicate the process, the existing processes run once per night, delaying the Invoices appearing in the eBill Express portal in a timely manner, let alone if there is an error anywhere along the way.

The requested solution is to take the Invoices directly from NetSuite and push to eBill Express, as they are created.

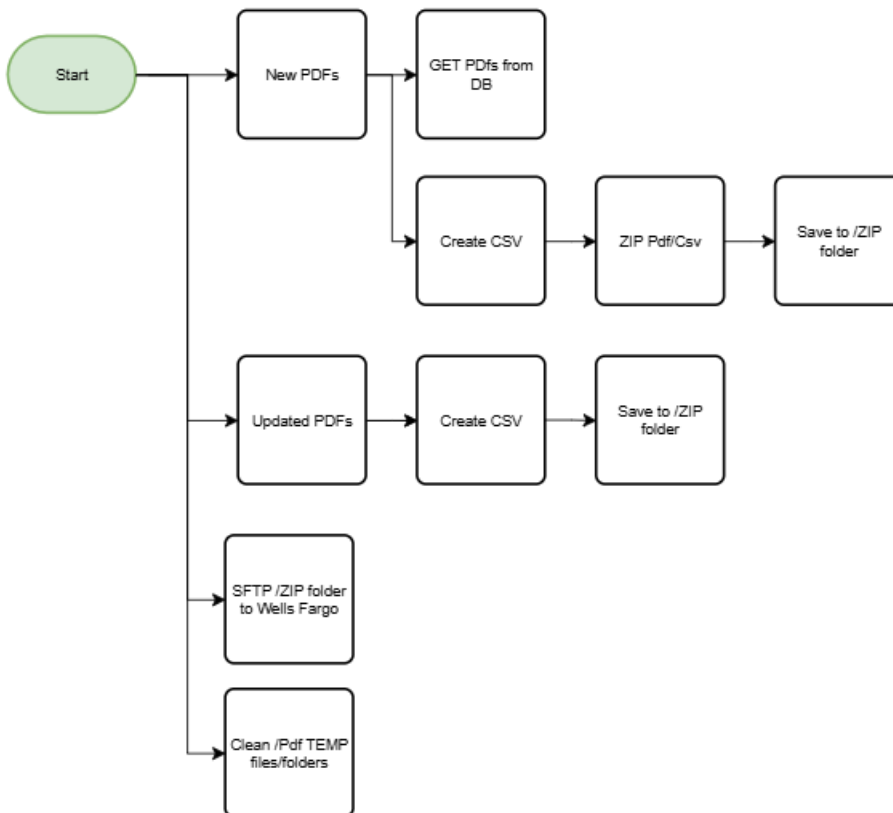
## Systems Integrated

- NetSuite (ERP – Invoicing)
- SQL Server (Reporting / Reference Data)



- WFB eBill Express (Online Banking Payment Portal)

## High-Level Architecture



### Architecture Summary:

Boomi served as the central integration platform, batching invoices simultaneously as they arrive into NetSuite, allowing smaller batches to run in parallel. Manage a local directly to temporarily store .pdf files after they are extracted from the database and prior to adding to .zip file.

## Integration Flows

### 1. New Invoice Flow

- “Lock” batches of New Invoices
- Create /PDF sub-folder using the LockId (i.e. /PDF/500/)
- Pull .pdf files for each Invoice into the sub-folder
- Create .csv of all files and corresponding .pdf



- e. ZIP all files in /PDF/{lockId} folder using existing naming convention + timestamp; store in /ZIP folder.
  - f. Delete /PDF/{lockId} folder
- 2. Updated Invoice**
- a. Generate .csv of all Updated Invoices
  - b. Move to ZIP folder for SFTP
- 3. SFTP File Processing**
- a. Pull all .csv and .zip files
  - b. SFTP to WFB
  - c. Log to DB
  - d. Move to ARCHIVE folder
- 4. Directory / File Cleanup**
- a. Recursively delete all /PDF/{lockId} folders > 5 days (keep it clean!)

## Boomi Capabilities Used

- Native connectors (NetSuite, Disk v2, Database, SFTP)
- Process orchestration and sub-process reuse
- Mapping with validation and default handling
- Groovy scripting (Folder deletion)
- Error handling with notifications and retries
- Environment promotion (DEV → QA → PROD)

## Key Challenges & Solutions

**Challenge:** The End of Month process sends invoices in bulk to NetSuite, so to speed things up, we split eBill Express into smaller parts for simultaneous execution.

**Solution:** Introduce a Staging SQL Table to track "LockId", enabling each batch of records to be assigned and locked to a specific execution process. For example, while Execution #1 retrieves 500 PDFs from the database, Execution #2 can begin by locking the subsequent batch for processing.

## Results & Business Impact

- Reduced processing from 6-8 days, down to 3
- Improved billing payment



- Increased data accuracy between Netsuite and eBill Express
- Scalable foundation for future integrations

## Security & Governance (Optional but Impressive)

- Credential management via Boomi AtomSphere
- Secure SFTP with key-based authentication
- Controlled deployment via environment separation

## Lessons Learned

- Utilizing “Locking” methodology and Simultaneous Executions dramatically increases throughput
- Variable scheduling (Every 15 minutes during their End-of-Month process and then 3-times per day thereafter) is more efficient
- Built reusable Boomi components to accelerate future projects
- Standardized error framework improved support response time

## Tools & Technologies

- Boomi AtomSphere
- NetSuite
- SQL Server
- SFTP