Budget Rollup Report

1. Contextualisation

1.1 Problem statement

- PMs track expenses and budgets on excel spreadsheets on their personal computer
- Finance has static forecasts which are set at max twice a year

1.2 How I am addressing this problem

- Making PO's and invoices visible in Sitetracker for all PM's
- Making budgets visible for everyone
- Making budgets dynamic (Only changed by Rob when a site has a budget item that deviates from the template amount)
- Boiling forecasting to a simple formula which is calculated with a combination of live data from Xero (PO's, invoices) in Sitetracker and the dynamic budgets set by the team for the respective site
  - **Addressed further in the next section below**

2. Forecast To Complete (FTC) Logic

As a requirement for the forecast report, finance needed an automated calculation of Forecasting numbers. We were able to come up with the logic below which would suit our purpose.

2.1 Representations of FTC Formula

2.1.1 Salesforce formula field

See here for the implementation. Also displayed below.

```
IF (
  AND (
    Billed_Po_Count__c == All_PO_Count__c,
    OR (
      ISBLANK(Activity__c),
      AND (
         NOT(ISBLANK(Activity__c)),
         NOT(ISBLANK(Actual_Date__c))
      )
    )
  ),
  0,
  IF (
    MAX(Calculated_Budget_Amount__c,  PO_Actual__c) - Invoice_Actual__c
< 0,
    0,
    MAX(Calculated_Budget_Amount__c,  PO_Actual__c) - Invoice_Actual__c
  )
)
```

2.2 Worked example of forecast logic

Stepping through every possible scenario and representing the FTC in the bottom swimlane at the end of each phase.

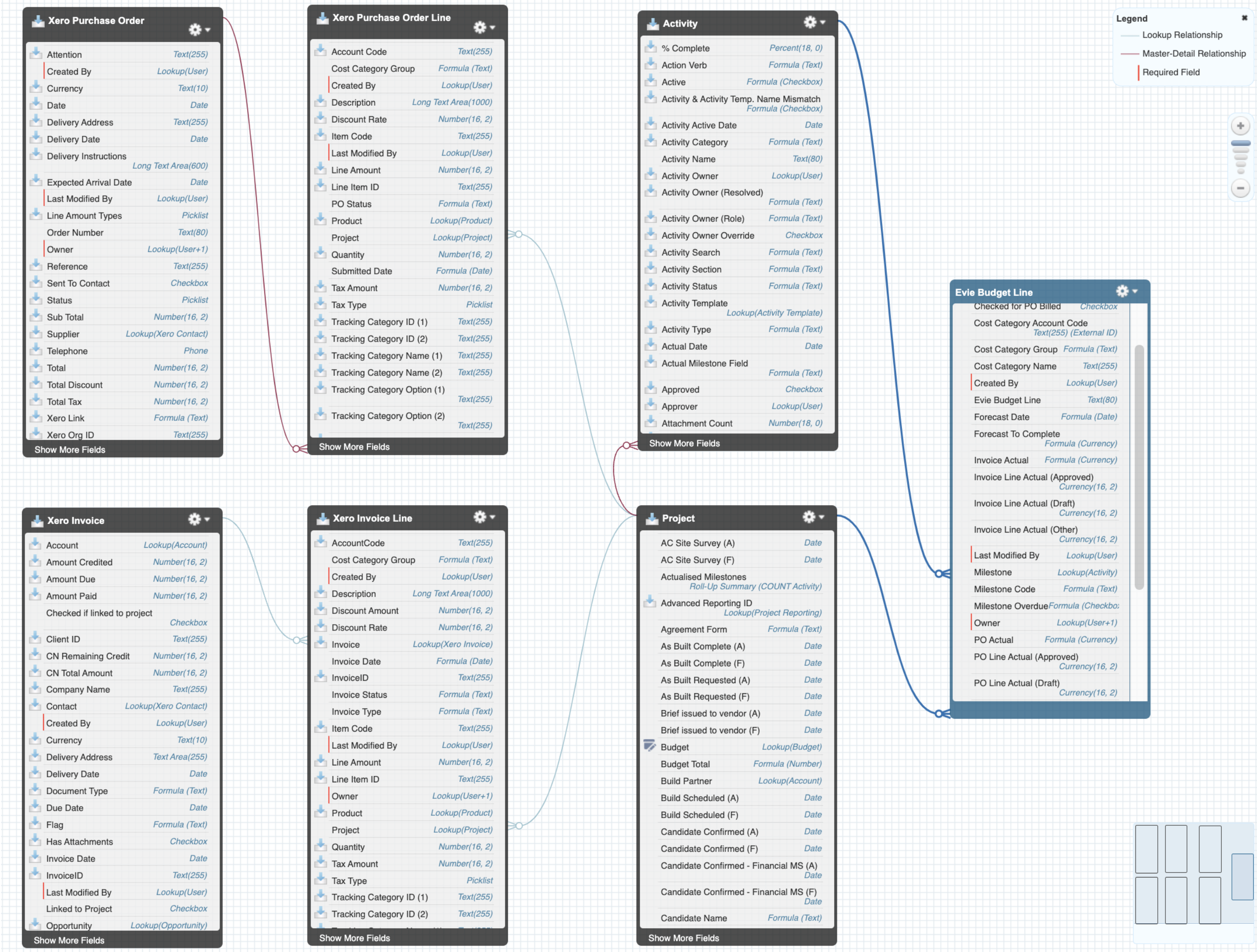
3. V1 Budget

The V1 BRR has 3 intended functions:

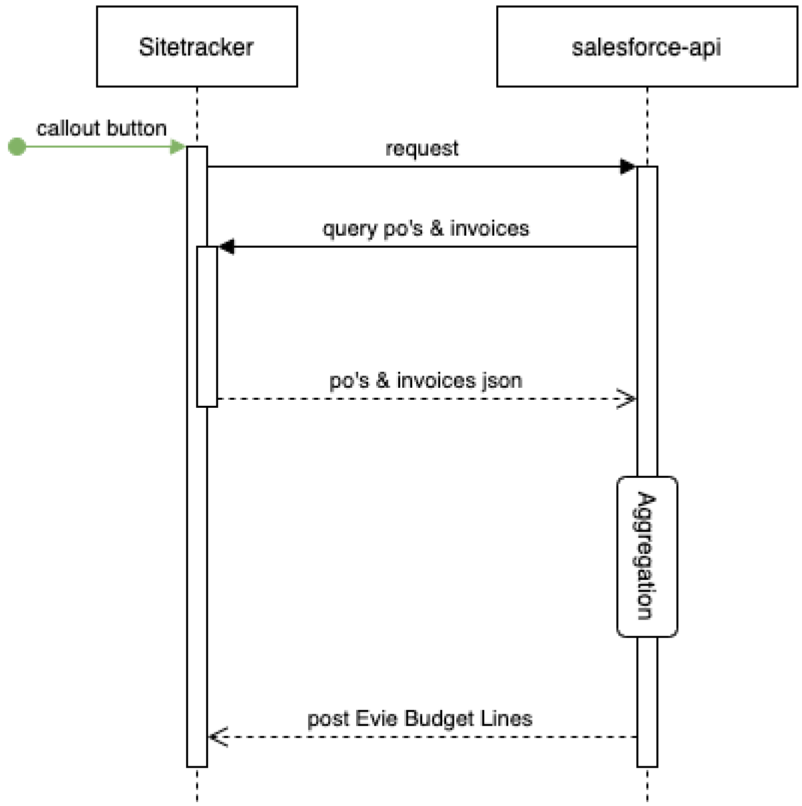
1. Expose financial data at a high level to Project Managers
2. Create relationships between physical milestones and budget lines on a Project
3. Make a FTC calculation based on the inputs pushed into Sitetracker by this service

See here for an overview of the salesforce callout logic Exploratory Apex Class Implementation

3.1 Entity Relationships

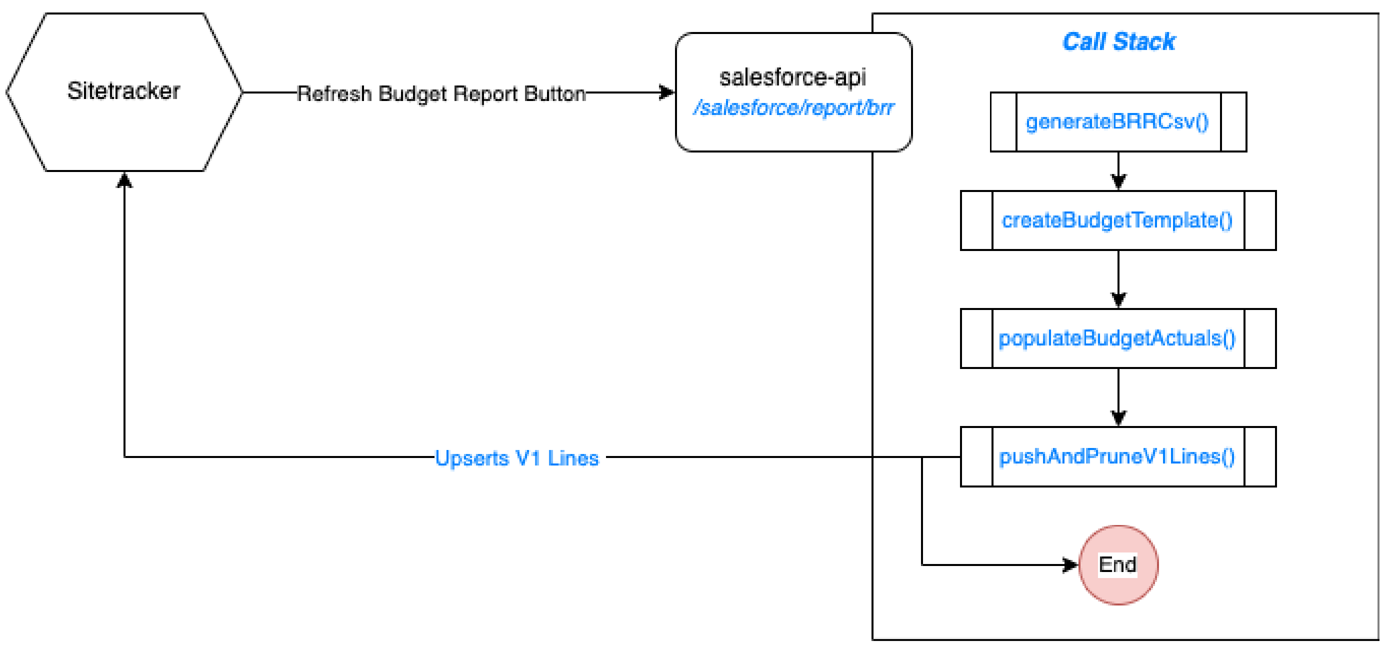


3.2 High Level Process Flow



V1 Process Flow

3.3 V1 Architecture



V1 Service Architecture

3.4 Tax Calculations

We have to adjust the the line amounts used as inputs into the BRR due to a bug in Xero not displaying the Tax Types of invoice lines correctly. We **always** want the amounts to be used in the BRR as **Tax Exclusive**.

The logic in the service is as follows (in psuedocode):

```
// if there is tax applied
if(taxType == "INPUT") {
  // calculate the current tax rate %
  calculatedTaxAmountPreAdjustment = round((CGCB__Quantity__c *
CGCB__Unit_Amount__c)/CGCB_Tax_Amount__c);
  // if tax rate is 11 we need to reduce the line amount to be tax
EXCLUSIVE
  if(calculatedTaxAmountPreAdjustment == 11) {
    lineAmountType = "Inclusive"; // add flag to show which ones have
been changed
    // subtract the calculated tax amount
    CGCB__Line_Amount__c = line.CGCB__Tax_Amount__c * 10;
    calculatedTaxAmountPostAdjustment = Math.round(CGCB__Line_Amount__c
/ CGCB__Tax_Amount__c);
  } else {
    // tax is correct and we dont need to adjust
    lineAmountType = "Exclusive"
  }
} else {
  // there is not tax applied (tax exempt invoice lines)
  calculatedTaxAmount = line.CGCB__Tax_Type__c;
  lineAmountType = "NoTax";
}
```

3.5 I/O

3.5.1 Service inputs

```
// to post internally
curl -X POST -H "content-type:application/json" "http://salesforce-api:
3000/salesforce/report/brr" -d @testPost.json
```

```
// to post externally (with authentication)
curl -X POST -H "content-type:application/json" "https://api.goevie.com.
au/api/v1/site_tracker"
```

```
// testPost.json
{ "projectId": "a1H5g000000plfmEAA", "upstream": "false" }
```

3.5.2 Service outputs

Below is the structure of a budgetLine that gets upserted into Sitetracker. Note that the *Forecast_Date__c* and *Forecast_To_Complete__c* is optional as it does not get set in the service, but by salesforce via a flow, and a formula respectively.

```
export interface budgetLine {
  Cost_Category_Name__c: string;
  Cost_Category_Account_Code__c: string;
  Template_Budget_Amount__c: number;
  Revised_Budget_Amount__c: number;
  Calculated_Budget_Amount__c: number;
  PO_Line_Actual_Approved__c: number;
  PO_Line_Actual_Submitted__c: number;
  PO_Line_Actual_Draft__c: number;
  PO_Line_Actual_Other__c: number;
  Forecast_To_Complete__c?: number;
  Invoice_Line_Actual_Approved__c: number;
  Invoice_Line_Actual_Draft__c: number;
  Invoice_Line_Actual_Other__c: number;
  Unique_Identifier__c: string;
  Forecast_Date__c?: string;
  Project__c: string;
}
```

3.5 Amending Budgets

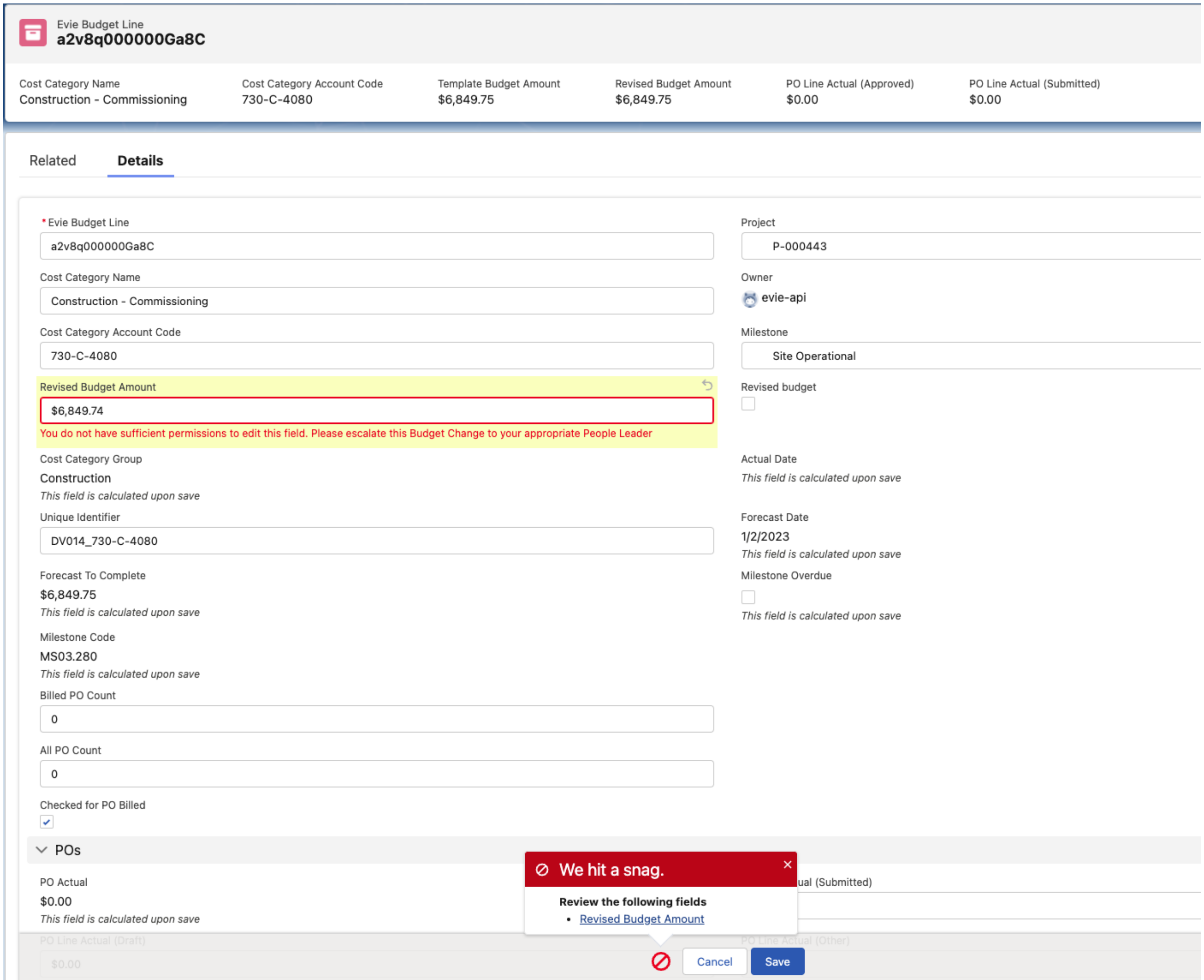
For sites where the budget deviates from the template board approved amount, the budget should be updated accordingly in Sitetracker.

> For now, only 4 accounts can edit the *Revised Budget Amount* field:
>
> 1. evie-api
> 2. Joe Oppenheimer
> 3. Karleen McKenna
> 4. Rob Glover

The intended process is as follows:

- PM requests overspend Rob approves change Rob updates *Revised Budget Amount*

If a Project Manager attempts to change the *Revised Budget Amount* field they will receive a form validation error like to one below.



The logic for these validation rules live here:

- Validation Rules
    - Lock_Field_from_API_Editing
        - Prevents API from overwriting budget amounts when they have been manually overwritten

```
• AND(
    $User.Email = "admin@goevie.com.au",
    Budget__c=TRUE,
    ISCHANGED( Revised_Budget_Amount__c )
  )
```

    - Lock_field_from_PM_Editing
        - Prevents those without the appropriate permissions from updating the budget

```
• AND(
    NOT(
      OR(
        $User.Email = "rob.glover@goevie.com.au",
        $User.Email = "admin@goevie.com.au",
        $User.Email = "karleen.mckenna@goevie.com.au"
        )
      ),
    ISCHANGED(Revised_Budget_Amount__c)
  )
```

Once the Revised Budget field has been manually modified, a flow will trigger to ensure that the 'Is Budget Revised' checkbox is ticked. The flow is labelled 'Mark budget line as revised'. When this checkbox is checked (TRUE), the Calculated_Budget_Amount__c field will reflect the revised budget amount. When the checkbox is NOT checked (FALSE), the Calculated_Budget_Amount__c field will reflect the template budget amount.

The formula for the Calculated Budget Amount field is as below:

```
IF( Is_Budget_Revised__c , Revised_Budget_Amount__c,
Template_Budget_Amount__c)
```

For a full breakdown of changes made on the 8/06, see https://docs.google.com/document/d/10JxfijGGrrRujrle3Fmy3E6MdKUemxqd-zTkaUjfCQI/edit?usp=sharing

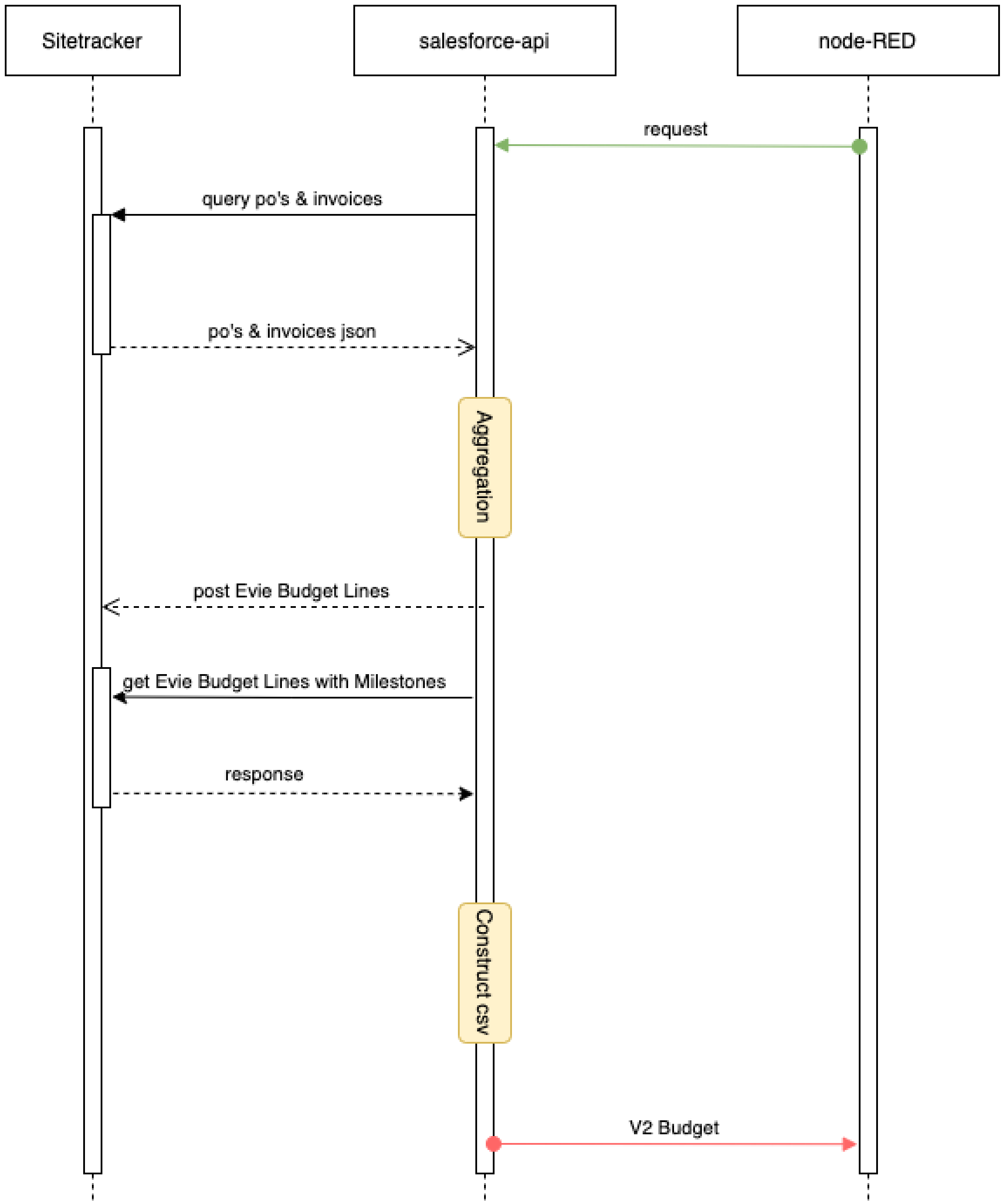
4. V2 Budget

The V2 Budget module is intended to satisfy the requirements as prescribed by Alex and the finance team.
In order to have a detailed awareness of cash flow and capital spent/forecasted capital spend, we needed to display the financials that are stored in Twix Riva on a time scale series.

There are 4 types of rows.

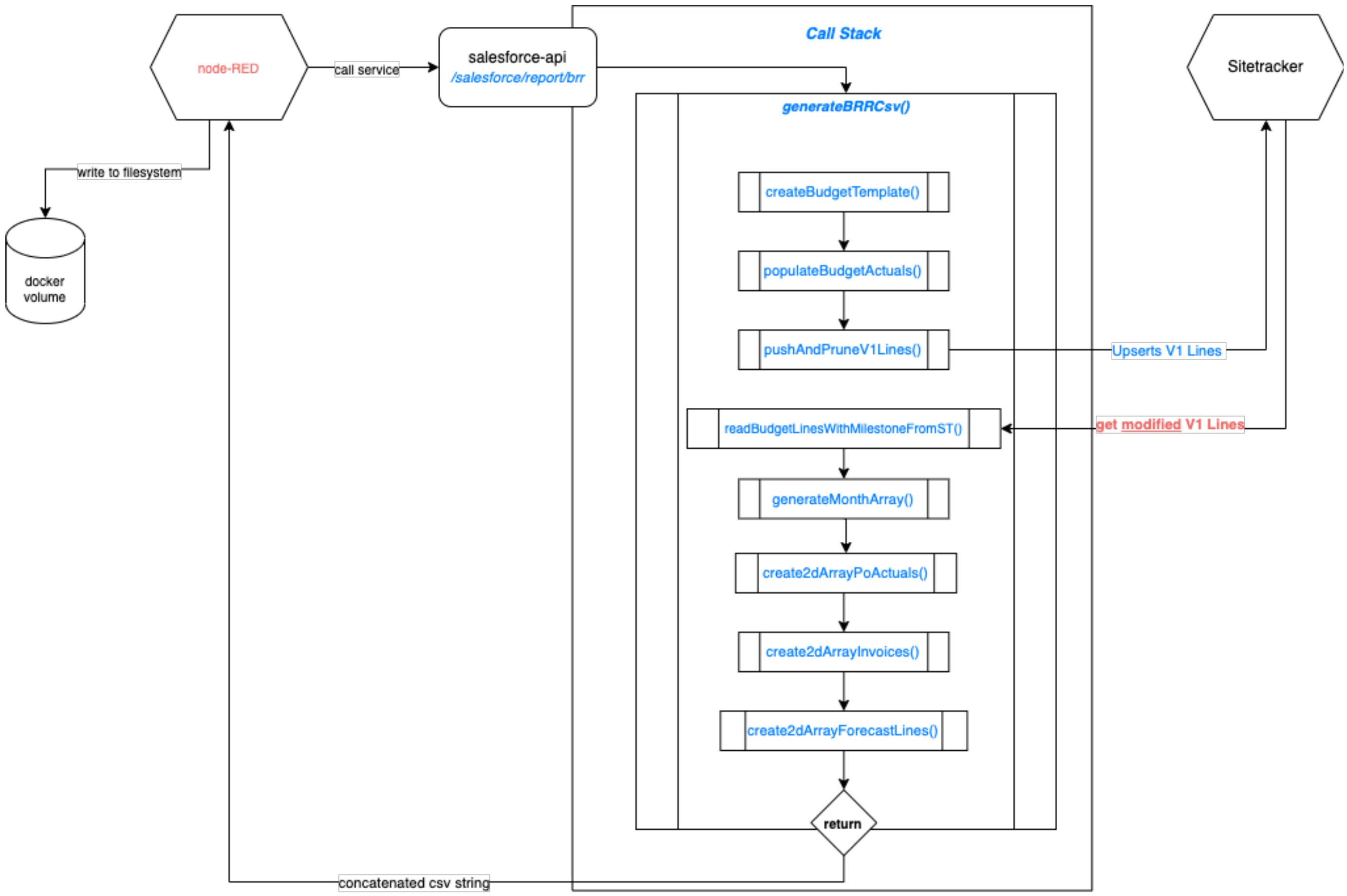
1. BUDGET
    a. This is the budget set as at the month prescribed by the forecast date on the Budget Line
2. INVOICES
    a. The aggregated Invoice amounts bucketed into months
3. PO's
    a. The aggregated PO amounts bucketed into months
4. FORECAST
    a. The calculated forecast logic as detailed above in section 2.

4.2 High Level Process Flow



V2 Process Flow

4.3 V2 Architecture



V2 Service Architecture

4.4 I/O

4.4.1 Service inputs

```
// to post internally
curl -X POST -H "content-type:application/json" "http://salesforce-api:
3000/salesforce/report/brr" -d @testPost.json
```

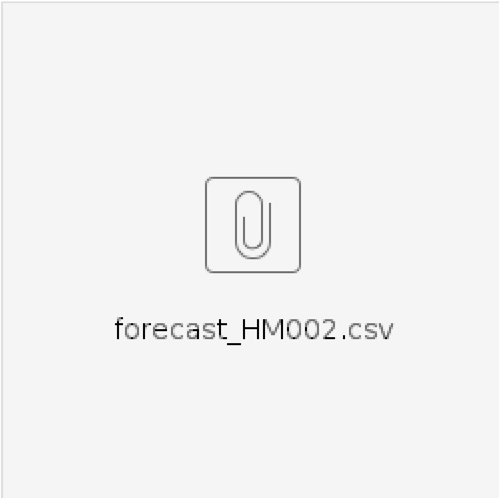
```
// to post externally (with authentication)
curl -X POST -H "content-type:application/json" "https://api.goevie.com.
au/api/v1/site_tracker"
```

```
// testPost.json
{ "projectId": "a1H5g000000plfmEAA", "upstream": "false" }
```

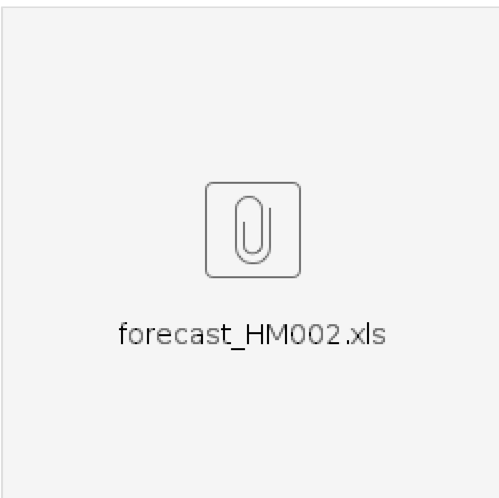
4.4.2 Service outputs

Files are written to the local filesystem and then are emailed to the software email group.

1. Raw CSV



2. xls



4.4.3 Error Handling

If any amount satisfies one of the following defined conditions, they get placed in that appropriate error column ***n*** columns from the end as denoted by the number in the enum directly below.

```
export enum milestoneErrorType {
  NoError = 0,
  ForecastInPast = 4,
  MilestoneNotExists = 3,
  ForecastDateIsNull = 2,
  OldAccountCode = 1,
  ActualDateMissing = 5,
}
```

```
// Error conditions
if (bl.Cost_Category_Account_Code__c.length < 10) {
      result.errorType = milestoneErrorType.OldAccountCode;
    } else {
      result.errorType = milestoneErrorType.MilestoneNotExists;
    }
    result.forecastDate = NULL_DATE;
  } else if (bl.Activity__c && isDateNull(bl.Forecast_Date__c)) {
    result.forecastDate = NULL_DATE;
    result.errorType = bl.Actual_Date__c
      ? milestoneErrorType.ForecastDateIsNull
      : milestoneErrorType.ActualDateMissing;
  } else {
    result.forecastDate = add(new Date(bl.Forecast_Date__c), {
      months: 1,
    });

    if (bl.forecastToComplete > 0 && result.forecastDate < new Date()) {
      result.errorType = milestoneErrorType.ForecastInPast;
      result.forecastDate = NULL_DATE;
    }
```

5. Dependencies

Name	Link to Dependency	Function	Risk Factor	Triggering action	Notes
Map Xero PO's to project	https://sitetracker-evie.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=3018q000000057iAAA	If an PO is allocated to a site, it links the object to the related project	**LOW**	When a Xero Purchase Order PO record is created or updated	
Map Xero Invoice to project	https://sitetracker-evie.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=3018q000000057jAAA	If an Invoice is allocated to a site, it links the object to the related project	**LOW**	When a Xero Invoice record is created or updated	
Map Milestones to Evie Budget Lines	https://sitetracker-evie.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=3018q00000006d0AAA	Populates the Milestone lookup field in the Evie Budget Lines	**MEDIUM** If mappings are not correct, forecasting gets thrown off	When an Evie Budget Line is created or updated	Needs to be maintained if mappings change. Not particularly elegant, but functional
Mark budget line as revised	https://sitetracker-evie.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=3018q00000006VpAAI	Ticks *'revised budget'* checkbox	**LOW**	When Revised Budget Amount is changed **&** user is **NOT** evie-api	

Linked to project Mapping for Xero Invoices	https://sitetracker-evie.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=3018q00000005ybAAA	Checks box '*linked to project'* to expose Invoices that are allocated to a site in the Xero Invoice Line Report on the *'Budget Vs Expense'* section of the Project page in Sitetracker	LOW	When a Xero Invoice is created or updated	

6. Embedded Business Logic

Business Rule	Why?	Where?	Risk	Notes
Adjust all line amounts to tax exclusive	Xero does not deliver line amounts in the back end as inclusive/exclusive, so we have to determine it by using the Tax amount and the total amount	filterInvoiceLines.ts line 86 to 104 filterPoLines.ts line 55 to 71	**MEDIUM** Difficult to validate	
Do not account for deleted /voided objects	These invoices/PO's were never actioned/paid so therefore they are as if they do not exist	filterInvoiceLines.ts line 40 to 45 filterPoLines.ts line 27 to 30	**LOW**	
Do not insert V1 Lines if they are not in template AND they are not generated by the service	We only add ad hoc likes when they are: **1.** Not in the template (no budget assigned anyway) **2.** Old cost codes which will disappear as they are reallocated	pushAndPruneV1Lines.ts line 26 to 54	**LOW**	
Error handling	So we can easily identify data quality issues	generateBudget.ts line 248 to 266	**LOW**	
Move all forecasts with valid milestones forward one month	To account for the approval process delay	generateBudget.ts Line 260	**LOW**	
Mapping of milestones to Evie Budget Lines	As per Alex/Joe/Rob/Karleens specifications	Map milestones to Evie Budget Lines [flow]	**LOW**	