

REPORTE ENVIOS COMERCIAL CORPORATIVO

Fuente: /var/www/html/qareportes/app/Http/Controllers/Cloud/CorporativoController.php

1) Nombre del reporte

- **Nombre:** Reporte Envios Comercial Corporativo
- **Código / Alias:** report_envios_comercial_corporativo_
- **Responsable:** Equipo de Reportes Cloud
- **Propósito (1 línea):**

2) Alcance temporal

- **Campo(s) de fecha utilizados:**
 - `ose_fhpreferenciapartida` (fecha de partida de orden de servicio)
 - `inc_fhincidencia` (fecha de incidencia)
 - `fecha_preferencial` (fecha referencial alternativa)
 - `created_date` (fecha de registro en validación)
- **Tipo de rango (diario / mensual / personalizado):**
 - Calculado automáticamente entre `$first_date` y `$second_date` por mes, con lógica de retroceso de hasta 3 meses según mes actual.
 - En casos especiales (enero a marzo) incluye meses del año anterior.
- **Inclusión de horas (sí/no):** Sí [00:00:00 - 23:59:59]
- **Zona horaria:** La del servidor

3) Origen de datos (tablas/APIs)

? 3.1 Conexión / Base:

- **Base principal:** `$this->empresarial` (ERP Empresarial)

- **Base secundaria:** `$this->centos` (ERP Centos)
- **Versiónado:** Se elige conexión antigua o actual según el mes/año:
 - Si `año == 2023` y `mes <= 9` → usa `empresarial_old` y `centosold`
 - Caso contrario → usa `empresarial` y `centos`

Observación general:

El proceso recorre un rango de fechas generado por `obtainRangeDaysByYear()`, ejecutando iterativamente la función `report()` para cada rango mensual.

Los resultados se registran en la tabla de control `emp_reportes_validation`.

? 3.2 Tabla / Recurso: `emp_empresa`

- **Alias:** `emp`
- **Campos utilizados:**
 - `id` → Identificador de la empresa
 - `ruc` → Número de documento (clave de relación)
 - `grupo` → Clasificación de empresa (`COMERCIAL` o `CORPORATIVO`)
 - `gestor` → Usuario responsable o gestor asociado
- **Condiciones (WHERE):**
 - `grupo` **no nulo**
 - `grupo` **en** ('COMERCIAL', 'CORPORATIVO')
 - `ruc` **no en** ('20563023016', '20512528458')
- **Tipo de unión:**
 - `LEFT JOIN` con `emp_persona` (`emp.ruc = per.per_nrodocumento`)
- **Observaciones:**
 - Esta tabla define el universo de empresas que serán evaluadas.
 - El resultado alimenta los arreglos `$companys_group` y `$persons`.

? 3.3 Tabla / Recurso: `emp_persona`

- **Alias:** `per`
- **Campos utilizados:** `per_nrodocumento`
- **Unión:** `LEFT JOIN` con `emp_empresa` (`emp.ruc = per.per_nrodocumento`)
- **Observaciones:**
 - Se usa solo como apoyo para validar la existencia del RUC en personas.
 - Sin filtros directos, dependiente del JOIN principal.

? 3.4 Tabla / Recurso: `emp_ordenservicio`

- **Alias:** No declarado
- **Campos utilizados (clave → descripción):**
 - `ose_id` → ID único de orden

- `ose_termorigenatencion` → Origen
- `ose_termdestinoentrega` → Destino
- `ose_montofinal` → Monto del servicio
- `ose_fhpreferenciapartida` / `ose_fhpreferencial` → Fechas de referencia
- `ose_estadoentregado`, `ose_tipoanulado`, `ose_estadoPago`, `eliminado`, `ose_estado`
- `ose_remiteempresa`, `ose_destinaempresa` → Empresa remitente y destinataria

- **Condiciones (WHERE):**

- `ose_estadoPago NOT IN ('DA', 'AN')`
- `ose_tipoanulado = ''`
- `eliminado = '0'`
- `ose_estado = '1'`
- `ose_fhpreferenciapartida BETWEEN [first_date 00:00:00, second_date 23:59:59]`
- `ose_remiteempresa` O `ose_destinaempresa` **pertenecen** a `$companys_group`

- **Observaciones:**

- Base principal para obtener **todas las órdenes de servicio** por empresa.
- Se filtran órdenes válidas, sin anulación, dentro del rango de fechas.

? 3.5 Tabla / Recurso: `emp_detalle_comppago`

- **Alias:** `dcp`

- **Unión:** `LEFT JOIN` con `emp_comppago` (`cp.cop_id = dcp.dcp_cop_id`)

- **Campos utilizados:** `dcp_cop_id`, `dcp_ose_id`

- **Condiciones:**

- `dcp_ose_id IN (ordenes válidas)`
- `cp.cop_estado_pago IS NULL` O `cp.cop_estado_pago != 'AN'`

- **Observaciones:**

- Permite identificar **órdenes sin pago registrado**, que luego se agrupan como **pendientes de pago** (`$pending_payments`).

? 3.6 Tabla / Recurso: `emp_os_detalle`

- **Campos utilizados:**

- `osd_peso` → Peso de la carga
- `osd_osid` → ID de la orden de servicio

- **Condiciones:**

- `osd_osid IN (ordenes válidas)`
- `osd_eliminado = '0'`

- **Observaciones:**

- Fuente de los valores de **kilogramos transportados**.
- Se agrupan por OS (`$details_orders_group`).

? 3.7 Tabla / Recurso:

emp_procesos_historial_app

- **Conexión:** `$this->centos`
 - **Campos utilizados:** `id`, `idose`, `proceso`, `nombre_metodo`, `status`
 - **Condiciones:**
 - `idose IN (ordenes válidas)`
 - `proceso IN ('embarque', 'desembarque', 'inventario')`
 - `nombre_metodo IN (lista de métodos permitidos)`
 - `status = 1`
 - **Observaciones:**
 - Permite validar qué órdenes pasaron por **embarque, desembarque o inventario**.
 - Las faltantes se marcan como **no procesadas** en esas etapas.
-

? 3.8 Tabla / Recurso: emp_incidencias

- **Alias:** `inc`
 - **Unión:** `LEFT JOIN emp_ordenservicio os ON os.ose_id = inc.inc_idos`
 - **Campos utilizados:**
 - `inc_id`, `inc_tipo`, `inc_fhincidencia`, `inc_idos`, `inc_descripcion`, `inc_responsable_terminal`
 - `ose_remiteempresa`, `ose_destinaempresa`
 - **Condiciones:**
 - `inc.inc_tipo = 'REVISAR'`
 - `inc_fhincidencia BETWEEN rango de fechas`
 - `ose_estadoPago NOT IN ('DA', 'AN')`
 - `ose_tipoanulado = ''`
 - `ose_estado = 1`
 - `inc.eliminado IS NULL OR inc.eliminado = 0`
 - `inc_descripcion NOT IN (lista de desechos)`
 - **Observaciones:**
 - Registra **incidencias** relevantes (excluye desechos).
 - Genera agrupación por empresa y día (`$incidencias_group`).
-

? 3.9 Tabla / Recurso: emp_incidencias (SUNAT)

- **Campos utilizados:** `inc_id`, `inc_idos`
- **Condiciones:**
 - `inc_idos IN (ordenes válidas)`

- `inc_descripcion` IN ('3.1 M. DECOMISADA POR SUNAT.', etc.)
- `inc_fhincidencia` BETWEEN rango de fechas
- `eliminado` IS NULL OR `eliminado` = 0

- **Observaciones:**

- Subconsulta específica para **incidencias SUNAT**, agrupadas por `inc_idos`.
-

? 3.10 Tabla / Recurso: `emp_entrega`

- **Campos utilizados:** `ent_id`, `ent_osid`, `ent_persona`, `ent_eliminado`

- **Condiciones:**

- `ent_osid` IN (ordenes válidas)
- `ent_persona` IN ('67676767', '87878787', '98989898', '35353535')
- `ent_eliminado` = '0'

- **Observaciones:**

- Identifica entregas específicas a personas clave.
 - Se utiliza para descartar órdenes que ya fueron entregadas.
-

? 3.11 Tabla / Recurso: `emp_adicionales`

- **Campos utilizados:** `adc_id`, `adc_tipo`, `adc_ordenservicioid`, `eliminado`

- **Condiciones:**

- `adc_ordenservicioid` IN (ordenes válidas)
- `eliminado` = '1'

- **Observaciones:**

- Separa las guías con **servicios adicionales** (montacarga "M" o embalaje "K").
 - Las OS con múltiples adicionales son excluidas del conteo final.
-

? 3.12 Procedimiento interno:

`obtainRangeDaysByYear()`

- **Propósito:** Genera las fechas `first_day` y `last_day` por mes entre `first_month` y `last_month`.
 - **Uso:** Iterar mensualmente en `update()` para llamar a `report($first_date, $second_date)`.
-

? 3.13 Tabla de control:

`emp_reportes_validation`

- **Campos:** `report`, `first_date`, `second_date`, `created_date`, `status`
 - **Condiciones:** Inserta un registro por ejecución mensual del reporte.
 - **Observaciones:**
 - Registra el resultado de cada iteración (`status = 1` éxito, `0` error).
 - Facilita trazabilidad de generación por rango de fechas.
-

? 3.14 Función auxiliar:

```
generateArrayOfDays($first, $second)
```

- **Propósito:** Devuelve arreglo con todas las fechas entre dos días.
 - **Uso:** Agrupar y acumular métricas por día en `$group`.
-

? 3.15 Observaciones globales

- El código combina **consultas cruzadas entre bases** (ERP empresarial y Centos).
- Gran parte del análisis se realiza **en memoria**, agrupando por `ruc` y `fecha`.
- Los índices principales usados:
 - `ose_id`, `ruc`, `inc_idos`, `ent_osid`, `adc_ordenservicioid`.
- No usa vistas materializadas, pero genera una **tabla lógica mensual consolidada** en memoria para análisis.

4) Filtros globales del reporte

- **Inclusiones:**
 - Empresas con grupo "COMERCIAL" o "CORPORATIVO".
 - Órdenes en estado activo (`ose_estado = 1`).
 - Procesos con `status = 1`.
- **Exclusiones:**
 - Empresas con RUC en lista de exclusión.
 - Estados de pago 'DA', 'AN'.
 - Tipos anulados.
 - Incidencias de tipo "DESECHO" o "FALTANTE" definitivo.
- **Reglas por estado / habilitado:**
 - Solo registros habilitados (`eliminado = 0` o nulo).
- **Parámetros externos:**
 - Rango de fechas (`first_date`, `second_date`).

5) Transformaciones y Reglas de negocio

- **Derivaciones:**

- `kilogramo` = suma(`osd_peso`) por orden.
- `monto_facturacion_pendiente` = total de `monto` en órdenes sin pago.
- `cantidad_facturacion_pendiente` = conteo de esas mismas órdenes.

- **Mapeos de estado:**

- Órdenes no embarcadas, no desembarcadas, no inventariadas se cuentan en grupos `embarque`, `desembarque`, `inventario`.

- **Validaciones:**

- Ignora incidencias sin descripción válida.
- Excluye entregas ya completadas.

- **Condicionales:**

- Si `fecha_preferencial` <= `fecha_consult` y está en rango, se usa como fecha principal.
- Si existe incidencia por RUC y fecha, incrementa campo `incidencia`.

6) Estructura de salida

- **Tabla/archivo destino:** `emp_reportes_validation`

- **Particionado / sufijo:**

- Por mes (fecha inicial y final del rango procesado)

- **Clave primaria:** `report`, `first_date`, `second_date`

- **Columnas del reporte(nombre → tipo → descripción):**

- `report` → VARCHAR → Código o identificador del reporte ejecutado
- `first_date` → DATE → Fecha inicial del rango consultado
- `second_date` → DATE → Fecha final del rango consultado
- `created_date` → DATETIME → Fecha y hora de generación del registro
- `status` → INT → Resultado del procesamiento (1=éxito, 0=error)

- **Ordenamiento por defecto:**

- Por `created_date DESC`

7) Frecuencia y operación

- **Frecuencia de actualización:** PENDIENTE (Esperando info de Eduardo)
- **Ventana recalculada:** Hasta 3 meses previos.
- **Tamaño de lote:** Por rango mensual (`obtainRangeDaysByYear`).
- **Tolerancia a fallos:**
 - Inserta registro en `emp_reportes_validation` con `status = 0` en caso de excepción.
- **Tiempos esperados:** PENDIENTE (Esperando info de Eduardo)

8) Calidad y controles

- **Validaciones previas:**
 - Verificación de rango de fechas válido.
 - Exclusión de registros eliminados o anulados.
- **Post validaciones:**
 - Inserción de registro en tabla de validación (`status`).
- **Controles de duplicado:**
 - Evita múltiples inserciones del mismo mes mediante llaves.
- **Métricas de completitud:**
 - Conteo de órdenes procesadas vs. esperadas por rango.

9) Dependencias externas

- **APIs / Servicios:** No aplica.
- **Autenticación / Headers:** No requerido.
- **Mapeos / Contratos:**
 - Depende del contrato interno entre bases `empresarial` y `centos`.

10) Historial de cambios

2025-10-29 17:00 - Elian Franco Arroyo Rodas - Documentación inicial

Revisión #1

Creado 2025-10-29 16:30:24 -05 por Elian

Actualizado 2025-10-29 17:00:26 -05 por Elian