



■ filial de isa

# Lineamientos para Información y Datos

Dirección de Arquitectura y Operación Tecnológica  
Documento de Lineamientos para Información y Datos para XM  
Junio, 2016

## Control de Cambios

Versión	Fecha	Responsable	Descripción
1.0	2016-06-22	Sandra Betancourt Nataly Zapata	Versión inicial sobre lineamientos de Información y Datos.





■ filial de isa

## Contenido

<b>Introducción</b>	<b>3</b>
<b>1. Estándares Generales Nombramiento de Objetos</b>	<b>3</b>
<b>2. Bases de Datos</b>	<b>3</b>
2.1 Definición de Tablas	5
2.2 Definición de Atributos	6
2.3 Definición de Vistas	6
2.4 Definición de Vistas Materializadas	6
2.5 Definición de Sinónimos	7
2.6 Definición de Índices	7
2.7 Definición de Restricciones (Constraints)	8
2.8 Definición de Procedimientos Almacenados	10
2.9 Definición de Funciones	10
2.10 Definición de Paquetes	10
2.11 Definición de Jobs	11
2.12 Definición de Clúster	11
2.13 Definición de Triggers	11
2.14 Definición de Enlaces a la Base de Datos (Database Links)	12
2.15 Definición de Snapshot	12
2.16 Definición de Secuencias	12
2.17 Sistema de Clases	12
2.17.1 Clases	12
2.17.2 Atributos de clases (parámetros de métodos)	13
2.17.3 Operaciones/métodos	13
2.17.4 C #	13
2.17.5 Diagramas de clases.	13

Lineamientos Información y Datos	2
2.17.6 Asociación/Relaciones	13

CONFIDENCIAL

## Introducción

Este documento contiene la presentación de lineamientos para la gestión de información y datos en XM enmarcados en los estándares generales en el nombramiento de objetos en las bases de datos, con el fin de contar con un estándar práctico para la definición y administración de los diferentes objetos que involucra la base de datos.

## 1. Estándares Generales Nombramiento de Objetos

- Los nombres deben ser claros, nemotécnicos y en español. Deben ser fácilmente comprensibles para su uso posterior.
- Los nombres no deben ser extensos, deben contar máximo con 80 caracteres de longitud.
- Los nombres deben ser fáciles de recordar, se pueden usar abreviaturas con el fin de usar nombres más cortos.
- Al momento de nombrar clases u otros elementos del sistema es importante crear una lista con abreviaturas y palabras a usar en el sistema con el fin de reutilizarlas.

En XM se han adoptado dos (2) diferentes notaciones que se describen a continuación:

### Notación Pascal

Es una convención genérica para la formación de nombres simples o compuestos, según la cual la primera letra de cada palabra se escribe en mayúscula. Ejemplo: *SalarioEmpleado*.

Esto solo aplica para los objetos en SQL SERVER ya que en Oracle no son case sensitive

### Notación Camel

Es una convención genérica para la formación de nombres simples o compuestos, muy similar a Pascal Casing, según la cual la primera letra de cada palabra se escribe en mayúscula, excepto la primera letra de la primera palabra que se escribe en minúscula. Ejemplo: *salarioEmpleado*.

Para la definición de nombre de objetos se debe tener claras las siguientes definiciones:

**Identificador:** Código único, en ningún caso su objetivo es almacenar información del elemento.

**Nombre corto:** Nombre de máximo 20 caracteres, muestra la información básica del elemento y permite, por ejemplo en un reporte identificar fácilmente cual es el elemento.

**Nombre largo:** máximo 80 caracteres, es un nombre más descriptivo y detallado.

Es necesario definir nombre corto y nombre largo en algunos objetos por facilidad en manejo de reportes, despliegues en pantallas o restricciones de otras aplicaciones.

En general, todo objeto tendrá el campo adicional llamado *Descripción* que permitirá ser más específico en la identificación del objeto, pero este será de formato libre.

## 2. Bases de Datos

Actualmente en XM se manejan dos tipos de sistema de gestión de bases de datos de tipo objeto relacional que son Oracle y SQL Server, dónde se almacena la información crítica de la compañía. Por esta razón es necesario especificar la administración de los objetos que se crean,

actualizan y eliminan en dichas bases de datos con el fin de asegurar un gobierno sobre la información y así garantizar la estandarización y calidad de los datos.

Para bases de datos SQL Server se debe tener en cuenta.

- Los objetos creados que pertenezcan al esquema **dbo** son aquellos que son transversales o maestros para diferentes aplicaciones.
- Se debe crear un esquema por aplicativo o por sistema. Para los objetos propios de cada aplicación de debe mantener el nombre del esquema igual que el de la aplicación.
- Es recomendable que los objetos sean lo suficientemente descriptivos de la entidad o función que representan.
- Para la creación de un objeto dentro de la base de datos debe siempre tener definido el esquema asociado a la aplicación. Solo los objetos comunes o transversales se crean en el esquema **dbo**.

Para bases de datos ORACLE se debe tener en cuenta el esquema de licenciamiento, este incide si es una base de datos interna o externa. Cada sistema se identifica con las siguientes siglas:

BD/Sistema	Identificador
<b>PDN1</b>	
Administración de Cuentas	AC
Administración de agentes	AG
Actualización de Tarifas <sup>1</sup>	AT
Balance de Generación	BG
Cargo por Capacidad	CC
Cálculo de Demanda	CD
Cálculo de Energía	CE
Cargar información de Agentes y CND	CI
Demandas	DE
Disponibilidad Comercial	DC
Despacho Ideal	DI
Entrada Manual	EM
Facturación Electrónica	FE
Facturación SIC	FBO
Flujos de Trabajo - Reliflow	FT
Garantías	GA
Transferencia de información	IS
Interfaz para registro de fronteras	IN
Ingreso y compensación (antes CALICOM)	IC
Liquidación de Contratos de Energía	LC
Liquidación y Facturación STN	LF
Liquidación y Administración de cuentas	LA
Limitación de Suministro	LS
Usuarios no Regulados	NR

<sup>1</sup> Hacen parte del modelo de datos de Liquidación de Contratos y algunas son LCT\_ y en la actualidad están como PCT\_ de PUBLICON

Publicación de contratos - Publicon	PC
PM Y MM	PM
Liquidar recargos de Reconciliación, Desviación, restricciones y AGC	RA
Rendimientos Financieros	RE
Registro Fronteras	RF
Solicitud de Información (SICSOLI contadores)	SI
Seguridad MEM	SM
Facturación de Sistema de Transmisión	ST
TIES	TI
<b>DSPCND</b>	
Despacho, Redespacho Programado	DRP
<b>CIMCND</b>	
Neptuno (información hidrológica)	NEP
Arranque y Parada Unidades térmicas	APU
Sistema de Información del SIN	SIO
Copia de datos de programación (Precio oferta, Disponibilidad, Generación y AGC)	WES
<b>APLAOM</b>	
Pronostico de Demanda	DEM
<b>APLCND (se replican objetos de PDN1</b>	
Integración CNDNET	NET

## 2.1 Definición de Tablas

Las tablas son la unidad básica de almacenamiento de datos en una base de datos. Los datos son guardados en filas y columnas y deben cumplir con las siguientes características:

- Los nombres son el singular, ejemplo contacto, compañía,
- Debe ser nemotécnico y en español.
- El nombre debe describir el contenido de la tabla.
- Los nombres de los atributos pueden tener un tamaño hasta de treinta (30) caracteres.

A continuación se especifica la estructura para nombrar el objeto:

	SQL Server		Oracle	
Estructura	YYYYY		XXT_AAAYYY	
Definiciones	YYY	Nombre de la tabla compuesto de un nombre significativo. Debe ser en singular	XX	Indica el identificador del proceso dueño de la tabla. Es en plural
			T	Indica que es un objeto tabla
			AAAYYY	Indica el nombre de la tabla.
Ejemplo	Transformador		ICT_RESUCARG	

Para bases de datos SQL Server se debe tener en cuenta.

- Siempre debe ir precedido por el nombre del esquema.

Nota:

- Si las tablas son temporales debe agregarse el prefijo TMP.

## 2.2 Definición de Atributos

- Se debe utilizar notación Camel.
- Debe ser un nombre significativo con máximo veinticinco (25) caracteres.
- No debe usarse palabras claves como indexes, type, number.
- Cuando un atributo sea clave foránea de otra tabla debe tener el sufijo ID, y un nombre que de claridad de cuál es la tabla de referencia. Ejemplo claselID, dominioID, csgID
- Cuando los nombres sean muy largos, estos se pueden formar con base en abreviaturas previamente definidas, se recomienda de tres o cuatro caracteres así: MaxGenReactiva, maxAbsReactiva. La sugerencia sobre abreviaturas de exactamente tres caracteres, se hace basado en el hecho de que un número fijo establece una regla más estándar y facilita a los programadores recordar los nombres de las variables. No se debe utilizar la convención de unir palabras con el carácter de subrayado.

Para bases de datos ORACLE se debe tener en cuenta:

- Debido a que no es sensible a las mayúsculas y minúsculas, los atributos no pueden nombrarse con notación Camel.

## 2.3 Definición de Vistas

Una vista puede usarse como una presentación adaptada de los datos contenidos en una o más tablas o vistas. Una vista toma la salida de una consulta o query y la trata como tabla.

- Debe ser un nemotécnico con un máximo de 20 caracteres.

A continuación se especifica la estructura para nombrar el objeto:

	SQL Server		Oracle	
Estructura	vYYYYYY		XXV	
Definiciones	v	Prefijo que indica que se trata de una vista	XX	Indica el identificador del proceso dueña de la tabla
	YYY	Nombre de la vista compuesto de un nombre significativo.	V	Indica que es una vista.
Ejemplo	vLocalization		RAV_SBMRESULTADO	

Para bases de datos SQL Server se debe tener en cuenta.

- La vista debe ir precedida por el nombre del esquema.

## 2.4 Definición de Vistas Materializadas

Una vista materializada se define como una vista común que almacena el resultado de una consulta en vez de la definición de la vista. Actualmente en XM se usan como copia de la base



de datos interna PDN1, en la base de datos externa por cuestión de licenciamiento. Normalmente es un subconjunto de la tabla base, y se maneja con filtros.

	ORACLE	
Estructura	YYYXX	
Definiciones	YYY	Indica el nombre de la vista
	XX	Indica el filtro que se utilizó.
Ejemplo	T76SMDEMAND_INTER	

## 2.5 Definición de Sinónimos

Sirve para da un nombre a la tabla o vista sobre el cual se forma el alias. Se utiliza a menudo para enmascarar el nombre y propietario de un objeto, proveedor en cuanto a la ubicación de objetos remotos de una base de datos distribuida y simplificar las declaraciones para los usuarios de la base de datos.

Tiene las siguientes características:

- Lleva el mismo nombre del objeto, salvo en aquellos casos en los cuales se deba tener sinónimos de un objeto con el mismo nombre pero que pertenece a esquemas diferentes por ejemplo el histórico, en este caso se da al sinónimo el mismo nombre del objeto.
- Sólo se debe crear sinónimos privados a objetos que pertenece a otra base de datos como por ejemplo las del CND.

## 2.6 Definición de Índices

Los índices son estructuras opcionales asociadas a las tablas creadas para acelerar la ejecución de sentencias en una tabla.

- Debe contener un máximo de veinte (20) caracteres.

A continuación se especifica la estructura para nombrar el objeto:

	SQL Server		Oracle	
Estructura	IX_NombreCampo		IDX_YYY_ZZZ_V	
Definiciones	IX	Prefijo que indica que se trata de un índice	IDX	Iniciales de la palabra índice.
	Nombre Campo	Nombre del campo que está incluido.	YYY	Nombre de la tabla
			Z	Nombre del primer atributo sobre el cual se formó el índice.
			V	Número que indica cuantos índices se han creado sobre la misma tabla y atributo.
Ejemplo			IDX_SBMATRIBUTOS_FECFINA	

## 2.7 Definición de Restricciones (Constraints)

Las restricciones ayudan a mantener la consistencia semántica de la base de datos. Al nombrarlas es importante colocarle al inicio las iniciales que permitan distinguir el tipo de restricción. Si existe una restricción de integridad referencial entre dos tablas, el nombre de la restricción deberá contener el nombre de las dos tablas, o si es muy largo una abreviatura que permita inferir sus nombres.

### 2.7.1 Claves primarias:

Permite declarar un atributo o conjunto de atributos como la clave primaria de una relación.

- Debe tener máximo 20 caracteres.

A continuación se especifica la estructura para nombrar el objeto:

Estructura	SQL Server		Oracle	
	PK_YYYYYY		PK_AAA_NOMBRECAMPO	
Definiciones	PK	Indica que es una clave primaria se conservará el estándar conocido como Primary Key	PK	Indica que es un Primary Key
	YYY	Es el nombre del índice compuesto de las abreviaturas de las columnas (tres primeros caracteres o más si es necesario) que hacen parte de la clave.	AAAYYY	Seis letras iniciales del nombre de la tabla sobre la cual se define el constraint
			NOMBRE CAMPO	Nombre significativo para el primer campo.
Ejemplo	PK_PlantaGeneracion		PK_A307IDTIPO	

### 2.7.2 Claves Foráneas:

Permite enlazar relaciones de una base de datos.

- Debe tener máximo 20 caracteres.

A continuación se especifica la estructura para nombrar el objeto:

Estructura	SQL Server		Oracle	
	FKYYYYYY_XXXXX		FK_AAAYYY_NOMBRECAMPO	
Definiciones	FK	Indica que es una clave foránea se conservará el estándar conocido con Foreign Key	FK	Indica que es Foreign Key
	YYY	es el nombre del campo que es el Foreign Key	AAAYYY	seis letras iniciales del nombre de la tabla sobre la cual se define el constraint

	XXX	Nombre de la tabla a la que hace referencia.	NOMBRE CAMPO	Nombre significativo para el primer campo
Ejemplo	FK_Causa_MaestroObj		FK_A307IDFTARIFA	

**Nota:** Para el modelo del MID las Foreign Key son al MaestroObj.

### 2.7.3 Claves Únicas:

- Debe tener máximo 20 caracteres.

A continuación se especifica la estructura para nombrar el objeto:

	SQL Server		Oracle	
Estructura	UQXXXX_YYYY		UK_AAAYY_NOMBRECAMPO	
Definiciones	UQ	Indica que es clave única	UK	Indica que es unique key
	XXX	Nombre de la tabla a la cual está asociado la clave única	AAAYY	seis letras iniciales del nombre de la tabla sobre la cual se define el constraint
	YYY	Nombre del campo que se le va a colocar la restricción	NOMBRE CAMPO	Nombre significativo para el primer campo
Ejemplo	uq_NitSiglaVersion			

### 2.7.4 Checks:

- Debe tener máximo 20 caracteres.

A continuación se especifica la estructura para nombrar el objeto:

	SQL Server		Oracle	
Estructura	CKXXXXX_YYY_ZZZ		CK_AAAYY_NOMBRECAMPO	
Definiciones	CK	indica que es una restricción de chequeo de valores para un atributo	CK	Letra que indica que es un check
	XXX	Nombre de la tabla	AAAYY	seis letras iniciales del nombre de la tabla sobre la cual se define el constraint
	YYY	Nombre de la columna a la cual que se le va a colocar la restricción	NOMBRE CAMPO	Nombre significativo para el primer campo
	ZZZ	Tipo de check: NN: Not Null, VE: Valor Especifico (especificado en una cláusula in, o between)		
Ejemplo	CkBahiaEstado_VE		CK_A307DESPACHO	

**Nota:** En el mid los ck se manejan por metadata por medio de los enumerados, sin embargo también se pueden definir por base de datos.

## 2.8 Definición de Procedimientos Almacenados

- En lo posible se debe limitar el uso de procedimientos almacenados debido al impacto que puede tener en la arquitectura del sistema y reglas de negocio definidas.
- Para la aprobación del ingreso de un procedimiento almacenado es necesario el envío de un correo electrónico especificando **XXXX** con un día de anticipación con el fin de realizar su respectiva revisión e impacto dentro de la base de datos.

A continuación se especifica la estructura para nombrar el objeto:

Estructura	SQL Server		Oracle	
	APP_XXXX		SP_YYY	
Definiciones	APP	Nombre de la aplicación que usa el procedimiento	SP	Indica que es un procedimiento almacenado
	XXXX	Nombre del procedimiento que indica si funcionalidad o utilidad.	YYY	Nombre del procedimiento almacenado.
Ejemplo	CNDNET_ConsultarOfertasRDRRecibidasPorAgente		SP_ELIMINAR_FRON_DE_CTRL	

Para bases de datos SQL Server se debe tener en cuenta:

- No se debe utilizar el prefijo "sp\_" debido a que es un prefijo adoptado por Microsoft para los objetos del sistema y puede generar demoras innecesarias del motor al tratar de buscar el objeto en las bases de datos del sistema.

## 2.9 Definición de Funciones

- La vista debe ir precedida por el nombre del esquema.
- Debe iniciar por la letra f.
- Debe ser un mnemotécnico con un máximo de XX caracteres.
- El nombre debe describir claramente la función.

## 2.10 Definición de Paquetes

- Debe ser nemotécnico con un máximo de 20 caracteres.
- Debe ser un nombre descriptivo.
- No se deben usar caracteres especiales, incluyendo guiones y subrayado para separar las palabras.

Dentro del paquete se debe documentar en forma de texto la siguiente información:

<b>Autor del Paquete</b>	Empresa y nombre del analista creador
<b>Descripción</b>	Descripción breve de lo que hace el paquete, preferiblemente fuente Arial tamaño 11.
<b>Bases de datos</b>	Nombre de las bases de datos que intervienen en el proceso.
<b>Entradas/Salidas</b>	Cuáles son las entradas y salidas, ejemplo archivos, queries, tablas, paginas, etc.

<b>Periodicidad</b>	Periodicidad de la ejecución del paquete. Cada cuanto se debe ejecutar en caso de ser necesario especifique su hora.
---------------------	--

- Para los objetos ETL, el proveedor de la aplicación debe entregar un manual de instalación y un archivo ejecutable que realice la instalación automática.
- Se debe asegurar la siguiente estructura de carpetas al momento de almacenar las ETL como se muestra a continuación:

#### Base de Datos\Empresa\Aplicación

Ejemplo: SDB\XM\InformeMensual o MSDB\ISA\Extractos

Para bases de datos SQL Server aplica para ETL desde SQL Server 2014. La estructura del nombre para los paquetes debe ser de la siguiente manera:

**ETLyyybd - Pckyyybd,**

**Comentado [SIBM1]:** Revisar esto en el MID

Dónde se deben nombrar iniciando por los prefijos **Pck o ETL**, **yyy** es la acción a ejecutar (Por ejemplo ValidarEstadoUsuarios, Cargar Estudios).

### 2.11 Definición de Jobs

- Deben reflejar la acción que se desempeña.
- Debe iniciar con un verbo en infinitivo.
- Se debe evitar el uso de caracteres especiales, incluyendo guiones y subrayado para separar las palabras.

Para bases de datos SQL Server se debe tener en cuenta

- La estructura de nombramiento se hace de la siguiente manera **Jobyyyybbb**, dónde **Job** indica que es una tarea tipo Job, **yyy** es la acción a ejecutar y **bbb** es el nombre de la base de datos principal del proceso, preferiblemente el alias del aplicativo.

### 2.12 Definición de Clúster

- Debe identificar las tablas que relaciona y los campos principales de la relación.

Para bases de datos ORACLE se debe tener en cuenta

- La estructura de nombramiento se hace de la siguiente manera **XXC\_AAAYYY\_AAazzz**, dónde **XX** son las iniciales del proceso, **C** indica que es de tipo clúster, **AAAYYY** indica el nombre de la tabla 1 y **AAzzz** indica la tabla 2.

### 2.13 Definición de Triggers

- Debe tener un máximo de veinte (20) caracteres

A continuación se especifica la estructura para nombrar el objeto:

	SQL Server	Oracle
<b>Estructura</b>	Tryyy	XXTR_AAAYY

Definiciones	tr	Letra que indica que es un disparador	XXTR	XX son las iniciales del proceso, TG indica que es de tipo trigger
	YYY	Nombre de la tabla a la que está asociado el disparador	AAAYYY	Nombre del objeto sobre el cual actúa se define Trigger
Ejemplo	trContacto		EMTR_CONCONCEPTOS	

## 2.14 Definición de Enlaces a la Base de Datos (Database Links)

- Debe contar con un máximo de veinte (20) caracteres.

Para bases de datos ORACLE se debe tener en cuenta

- La estructura de nombramiento se hace de la siguiente manera DBL\_ ZZZ\_VVV, donde DBL indica que es un DBLink, ZZZ base de datos destino y VVV Nombre del usuario al que se hace el enlace. Ejemplo DBL\_BASICAS

Comentado [SIBM2]: No es el estándar en Xm

Comentado [SIBM3]: No encontré ningún ejemplo con esto. No se usa

## 2.15 Definición de Snapshot

- Debe contar con un máximo de veinte (20) caracteres.

Para bases de datos ORACLE se debe tener en cuenta

- La estructura de nombramiento se hace de la siguiente manera SN\_ AYYY\_V, donde SN indica que es un snapshot, AYYY indica el nombre del objeto remoto sobre el cual actúa el snapshot y V es el número que indica cuantos snapshot se han creado sobre el mismo objeto.

## 2.16 Definición de Secuencias

- Debe contar con un máximo de veinte (20) caracteres.

Para bases de datos ORACLE se debe tener en cuenta

- La estructura de nombramiento se hace de la siguiente manera XXSQ\_ AYYY\_ZZZ, donde XX indica el identificador del proceso dueño de la tabla, SQ indica que es una secuencia, AYYY indica el nombre de la tabla sobre la cual se creó la secuencia y ZZZ indica el campo sobre el cual se creó la secuencia.

## 2.17 Sistema de Clases

Es necesario al momento de crear bases de datos SQL server tener presente el siguiente dimensionamiento:

### 2.17.1 Clases

Es importante asignar un nombre clave a la clase porque este se convertirá en el nombre de la tabla, el objetivo es que ambos nombres coincidan en lo posible. Cuando se manejan nombres distintos se puede originar un elevado grado de confusión. Los nombres de las clases deben ser claros de modo que no den lugar a ambigüedades. Se utilizará notación Pascal para los nombres de clase

#### 2.17.2 Atributos de clases (parámetros de métodos)

- Si el atributo es “público” (no muy frecuente) se utilizará la notación Pascal, en otro caso se utilizará la notación Camel.
- Para el caso de los parámetros de los métodos siempre se utilizará la notación Camel. Ejemplo voltTrafPrim.

#### 2.17.3 Operaciones/métodos

Hay varios tipos de operaciones o métodos, y estas son las reglas que debe aplicarse a cada tipo:

- El constructor debe tener el mismo nombre de la clase.
- Las propiedades (retornan o modifican el valor de uno de sus atributos) se llamarán igual al atributo correspondiente, usando la notación Pascal.
- Los métodos que son procedimientos deben reflejar una acción (Verbo en infinitivo) y se pueden acompañar con sustantivo y adjetivo (en ese orden) Ejemplo “LeerValorAfectado” o cuando no aplique colocarle un nombre significativo.
- Solo se utilizará la notación “Camel” para nombrar variables privadas y parámetros de los métodos.
- Los nombres de las propiedades adoptan el mismo nombre de la variable correspondiente pero en notación “Pascal”.

#### 2.17.4 C #

Se utilizará la notación Húngara (prefijos) para asignarle nombre a los controles que forman parte de una Interfaz Gráfica de Usuario. Estos prefijos son los que ya están definidos en la documentación estándar de VB6. Así mismo se utilizarán prefijos para los objetos que integren Multimedia, para diferenciar objetos tipo “Video”, Sonido, Texto e Imagen.

#### 2.17.5 Diagramas de clases.

- Se puede tener un diagrama por todo el sistema o incluir uno por cada temática.
- Los nombres de los diagramas deben tener significado.
- Las mismas clases pueden aparecer en varios diagramas.
- Se puede trabajar con paquetes

#### 2.17.6 Asociación/Relaciones

Una asociación es una conexión entre clases que pueden ser asignadas a un Rol.

- Se usarán los estándares para UML, en el cual hay tres elementos: la propia relación y cada uno de los extremos de la misma.
- Las relaciones UML se describen utilizando una frase que constará de un verbo y una preposición y la dirección del nombre de la relación tendrá que ir desde la cardinalidad menor hacia la mayor.