# **Content Developer Kit**

Preventative Maintenance without Scheduled Service Times Vehicle Key - ACES

V1.1











#### **Table of Contents**

1	Ove	erview	. 3
2	Glo	ossary	. 4
3	Bus	siness Rules	. 4
4	Sch	rema	. 6
	4.1	MOTOR Operation Taxonomy	6
	4.2	MOTOR Qualifier	
	4.3	MOTOR PM	
	4.3.		
	4.3.	2 Header	8
	4.3.		
	4.3.	11	
	4.3.	- 1	
	4.3.		
	4.4	MOTOR SST Note	.12
	4.5	MOTOR Frequency	
	4.6	MOTOR Lubricant	
	4.7	MOTOR Service Type	.13
	4.8	MOTOR Severe Service	
	4.9	MOTOR Warranty	13
5	Dat	ta Dictionary	14
	5.1	XML File: MOTOR_OperationTaxonomy.xml	.14
	5.2	XML File: MOTOR_Frequency.xml	
	5.3	XML File: MOTOR Lubricant.xml	17



# Preventative Maintenance without Schedule Service Times- ACES

	5.4	XML File: MOTOR_ SevereService.xml	18
	5.5	XML File: MOTOR_ Warranty.xml	19
	5.6	XML File: MOTOR_ ServiceType.xml	20
	5.7	XML File: MOTOR_Qualifier.xml	21
	5.8	XML File: MOTOR_PM.xml	
6	San	nple Queries	
	6.1	Maintenance set where time/distance interval values not defined (Frequency F, L, M, N 39	
	6.2	MOTOR_Operation set where Operation is dictated by Indicator Light	40
	6.3	Get MOTOR Qualifiers for MOTOR_Operation set	40
	6.4	Get Replacement Parts for a set of MOTOR_Operation records	41
	6.5	Explode Schedule Service Times to individual mileage intervals for specific vehicle	42
	6.5.		
	6.5.		
	6.5.		
	6.5.4		
	6.5.	·	
	6.5.0		
	6.5.		
7	Dat	ta Usage Tips	
	7.1	Using MOTOR Qualifiers to Extended Vehicle Definitions	
	7.2	Getting the most of the App "ref" Attribute	
o		cumont	47
^	/1/	C111111/111	4/



## 1 Overview

MOTOR "Chek-Chart" has developed the industry-standard Preventive Maintenance database. The **SST** dataset is created by taking each OE-recommended scheduled service procedure, both normal and severe, and associating the required Product Classification Database (PCdb) Replacement Part Type for each procedure.

The data files described in this document comprise the MOTOR Chek-Chart ScheduledServiceTime Delivery database. This database covers OEM scheduled preventive maintenance recommendations for most domestic and imported car and light truck models available in the U.S. from 1985 through the current model year. Low census (low sales volume and exotic) vehicles may be excluded.

Vehicle specific data records in this database are associated to the Autocare Association ACES Vehicle Application Key (VCdb). Replacement parts are listed with the PCdb Part Terminology ID. For more information on the ACES Standard, please see <a href="http://www.autocare.org/what-we-do/technology/technolo

The following special conventions are used in the SST database:

1. The MOTOR Operation Taxonomy presents the MOTOR standard naming convention for operation names in a hierarchical structure. Each LiteralName\_English value represents a unique conceptual operation.



# 2 Glossary

**MOTOR Operations:** MOTOR's standardized Operation naming convention. These operation names are organized into a taxonomy classification by vehicle systems and assemblies.

**MOTOR Qualifier:** MOTOR Qualifiers are notes attached to operations that are used to distinguish between two or more applications that apply to the same vehicle, operation name, and operation position. MOTOR Qualifiers are standardized and organized into a three tier categorization.

**Replacement Part:** A part that is likely required to be replaced while performing an operation. In this dataset, the element <ReplacementPart> contains an attribute of ServiceType. Only records where ServiceType = 2 indicate an actual part that needs to be replaced.

**Service Type:** The Service Type defines the relationship between the Operation in context to a part name specified in the ReplacementPart element. This attribute will indicate whether or not a part is required to be replaced as part of the Operation and if it should be included as part of the estimate. Part Types that are not required for replacement are included to help integrate with other MOTOR products such as Quick Lube.

## 3 Business Rules

- 1) **Business Rule:** (MOTOR\_PM.XML) Each App record will contain 0 or one of each VCdb vehicle attribute. If, for example, a vehicle has three sub models available and a maintenance schedule applies to two of the sub models' that maintenance schedule will be repeated in two different app records, one for each sub model.
- 2) **Business Rule:** (MOTOR\_PM.XML) MOTOR Operations to PCdb parts relationships will be created and maintained on a global level and exploded to each app record. Operation to parts relationships can be adjusted at the app record level only in response to feedback received and on a case by case basis.
- 3) **Business Rule:** (MOTOR\_PM.xml) The standard for the global Operations to ReplacementParts mapping is that each MOTOR Operation is mapped to the parts that are required to be replaced each time the given operation is performed for most vehicles. These are essentially the parts required for an upfront estimate. The technician may find that additional parts require replacement once performing the operation.
- 4) **Business Rule:** (MOTOR\_PM.xml) If more than one note (MOTOR Qualifier) record is present within an app record, then each of the notes must be true for the record to be considered a valid application.

**NOTE:** Many GUI applications may benefit from the approach that multiple note records be concatenated into single strings for the end user to select.



- 5) Business Rule: (MOTOR\_Qualifier.xml) Each Qualifier description will be unique.
- 6) **Business Rule:** (MOTOR\_Qualifier.xml) If an end user indicates that a Qualifier record with a QualifierType of "Vehicle Attribute" is applicable to the vehicle in context, then any data related to a Qualifier record with the same QualifierType and Qualifier Value values as the selected Qualifier can be declared not applicable for the vehicle in context. For example, if the Qualifier "With Air Conditioning" is selected, any records attached to the Qualifier "Without Air Conditioning" can be eliminated from consideration for the end user as both Qualifiers are of the Qualifier Vehicle Type and share the same Qualifier Family value.
- 7) **Business Rule:** (MOTOR\_OperationTaxonomy.xml) Each LiteralName value is unique and can be used to represent the entirety of the given taxonomy path.
- 8) **Business Rule:** (MOTOR\_PM.xml) Each app record and each IncludedOperation will have position values. If the position is not relevant in the given operation, the position value of "N/A" will be used.
- 9) **Business Rule:** <Note> elements with an attribute value of vehicleattribute="yes" pertain to the whole <app> element, including all Included Operations within the app. These are qualifiers that help to describe the vehicle.
- 10) **Business Rule:** <Note> elements with an attribute value of vehicleattribute="no" only pertain to the main MOTOR Operation record in the app, not the Included Operations within the app.
- 11) **Business Rule:** A combination of MOTOR\_OperationTaxonomy, set of concatenated Qualifiers, and VCdb Attributes associated to a MOTOR\_Operation record, Position, and Severe Service should only be included once at most.
- 12) **Business Rule:** (MOTOR\_OperationTaxonomy.xml) While the textual value of a LiteralName associated with an ID may change over time to account for spelling corrections, grammatical changes, and naming standardization, the conceptual meaning of the Operation Taxonomy tied to an ID will not change.
- 13) **Business Rule:** (MOTOR\_Qualifier.xml) While the textual value of a QualifierDescription associated with an ID may change over time to account for spelling corrections, grammatical changes, naming standardization, the conceptual meaning of the MOTOR Qualifier tied to an ID will not change. However, the Qualifier classification (QualifierType and QualifierFamily) may change for a Qualifier ID.
- 14) **Business Rule:** (MOTOR\_PM.xml) If a SST\_Interval has multiple recommendation values, such as Millage and Months, the recommendation is whichever becomes true first.
- 15) **Business Rule:** (MOTOR\_Frequency.xml) MOTOR Frequency data is static. Any required changes would be communicated prior to being implemented.
- 16) **Business Rule:** (MOTOR\_ Lubricant.xml) Existing MOTOR Lubricant records will not change except as needed for spelling, grammatical or similar needs. New records may be introduced with regular deliveries.
- 17) **Business Rule: (MOTOR\_SevereService.xml)** MOTOR Severe Service data is static. Any required changes would be communicated prior to being implemented.

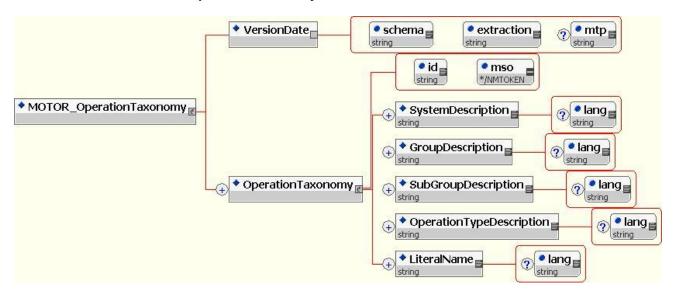


- 18) **Business Rule:** (MOTOR\_Warranty.xml) Existing MOTOR Warranty records will not change except as needed for spelling, grammatical or similar needs. New records may be introduced with regular deliveries.
- 19) **Business Rule:** (MOTOR\_ServiceType.xml) Existing MOTOR Service Type records should not change except as needed for spelling, grammatical or similar needs. Any substantive changes to existing records or the addition of new records will be communicated prior to being implemented.
- 20) **Business Rule:** (MOTOR\_SST\_Note.xml) Existing MOTOR SST Note records may change their description value. While most changes to existing records will be for spelling and grammatical changes, it is possible that substantive changes are made to existing SST Notes with the regular deliveries. Additional SST Note values are often likely to be created as well.
- 21) **Business Rule:** Unless specified otherwise in the SST Notes, the schedule attached to each MOTOR\_Operation instance includes action to all instances of the implied component that relate to the combination of MOTOR\_Operation, Position, and Qualifier in context. For example, if the Operation is Accessory Drive Belt Inspect, the position is N/A, and there are no Qualifiers attached, the schedule is to inspect all of the Accessory Drive Belts on the vehicle.

# 4 Schema

## 4.1 MOTOR Operation Taxonomy

Schema filename: MOTOR\_OperationTaxonomy.xsd

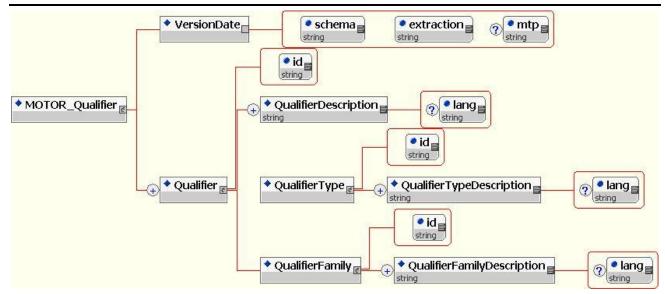


## 4.2 MOTOR Qualifier

Schema filename: MOTOR\_Qualifier.xsd



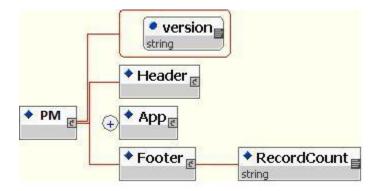
#### **Preventative Maintenance without Schedule Service Times- ACES**



## 4.3 MOTOR\_PM

Schema filename: MOTOR\_PM.xsd

#### 4.3.1 Shell



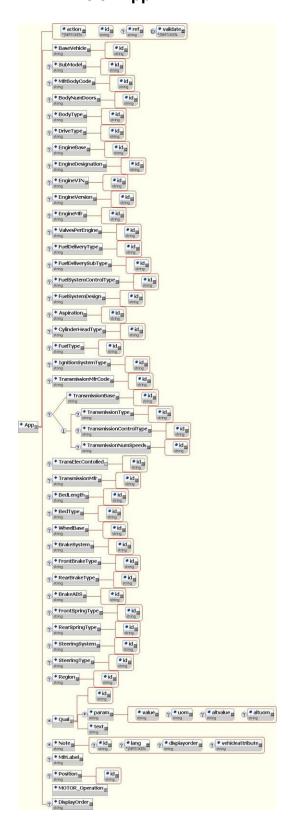


#### 4.3.2 Header



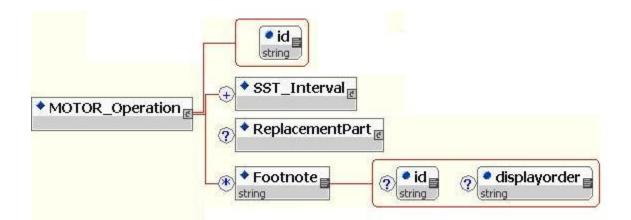


### 4.3.3 App



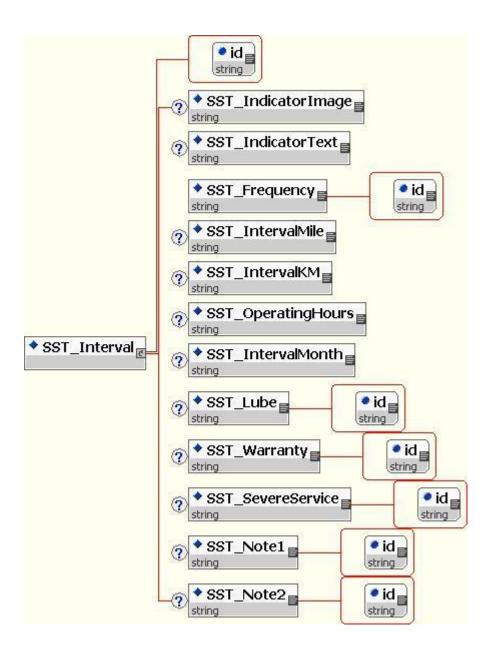


### 4.3.4 MOTOR\_Operation



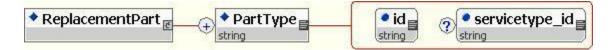


### 4.3.5 SST\_Interval



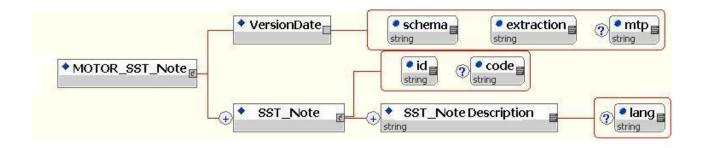


#### 4.3.6 ReplacementPart



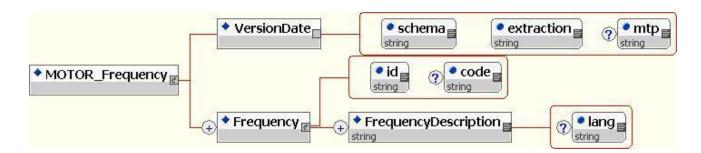
### 4.4 MOTOR SST Note

Schema filename: MOTOR\_SST\_Note.xsd



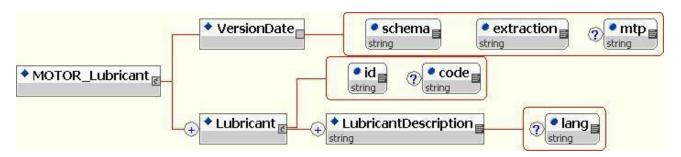
## 4.5 MOTOR Frequency

Schema filename: MOTOR Frequency.xsd



### 4.6 MOTOR Lubricant

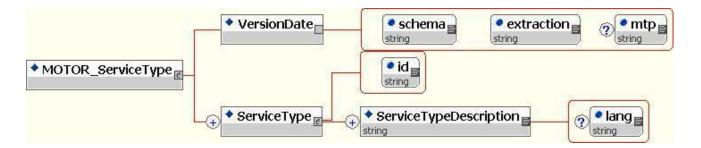
Schema filename: MOTOR\_Lubricant.xsd





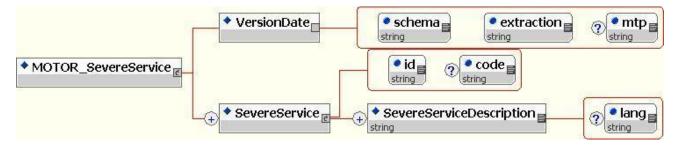
## 4.7 MOTOR Service Type

Schema filename: MOTOR\_ServiceType.xsd



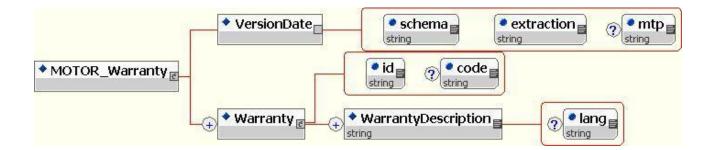
### 4.8 MOTOR Severe Service

Schema filename: MOTOR\_SevereService.xsd



## 4.9 MOTOR Warranty

Schema filename: MOTOR\_Warranty.xsd





# **5 Data Dictionary**

# **5.1 XML File: MOTOR\_OperationTaxonomy.xml**

Standard MOTOR Operation taxonomy definitions

#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
1	MOTOR_OperationTaxonomy	Elements	(VersionDate , OperationTaxonomy+)		Yes		MOTOR Master Operation Taxonomy root element
2	VersionDate	EMPTY		schema	Yes	Yes	Attribute "schema" indicates the schema version date (see date format note below)
				extraction		Yes	Attribute "extraction" indicates the extraction date (see date format note below)
				mtp		No	Attribute "mtp" indicates the MOTOR taxonomy version date, if applicable (see date format note below)
3	OperationTaxonomy	Elements	(SystemDescription , GroupDescription , SubGroupDescription , OperationTypeDescription , LiteralName)	id	Yes	Yes	The attribute "id" is unique and represents MOTOR Operation taxonomy path description, this id is referenced in MOTOR_PM.xml as MOTOR_Operation id
				mso			Yes/No flag to indicate if this taxonomy is MOTOR Standard Operation. This



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							information is used mainly for internal QC purposes.
4	SystemDescription	Туре	String	lang	Yes	No	MOTOR Standard System description The attribute "lang" indicates the language for the description, default language is English. Descriptions can be multiple, each with different "lang" attribute
5	GroupDescription	Туре	String	lang	Yes	No	MOTOR Standard Group description The attribute "lang" indicates the language for the description, default language is English. Descriptions can be multiple, each with different "lang" attribute
6	SubGroupDescription	Туре	String	lang	Yes	No	MOTOR Standard SubGroup description The attribute "lang" indicates the language for the description, default language is English. Descriptions can be multiple, each with different "lang" attribute
7	OperationTypeDescription	Туре	String	Lang	Yes	No	MOTOR Standard Operation type description The attribute "lang" indicates the language for the description, default language is English.



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							Descriptions can be multiple, each with different "lang" attribute
8	LiteralName	Туре	String	lang	Yes	No	MOTOR Standard Operation description The attribute "lang" indicates the language for the description, default language is English. Descriptions can be multiple, each with different "lang" attribute

# 5.2 XML File: MOTOR\_Frequency.xml

Scheduled service frequency descriptions.

#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
1	MOTOR_Frequency	Elements	(VersionDate , Frequency +)		Yes		MOTOR PMSST Frequency root element
2	VersionDate	EMPTY		schema	Yes	Yes	Attribute "schema" indicates the schema version date (see date format note below)
				extraction		Yes	Attribute "extraction" indicates the extraction date (see date format note below)
3	Frequency	Elements	(FrequencyDescription)	id	Yes	Yes	The attribute "id" is unique and represents MOTOR Frequency, this id is referenced in MOTOR_PM.xml as



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							MOTOR_Frequency id
				Code		Yes	Legacy PM Frequency Code value.
4	FrequencyDescription	Туре	String		Yes		MOTOR Frequency description

# **5.3 XML File: MOTOR\_Lubricant.xml**

Scheduled service Lubricant descriptions.

#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
1	MOTOR_ Lubricant	Elements	(VersionDate , Lubricant +)		Yes		MOTOR PMSST Lubricant root element
2	VersionDate	EMPTY		schema	Yes	Yes	Attribute "schema" indicates the schema version date (see date format note below)
				extraction		Yes	Attribute "extraction" indicates the extraction date (see date format note below)
3	Lubricant	Elements	(LubricantDescription)	id	Yes	Yes	The attribute "id" is unique and represents MOTOR Lubricant, this id is referenced in MOTOR_PM.xml as MOTOR_ Lubricant id
				Code		Yes	Legacy PM Lubricant Code value.
4	LubricantDescription	Туре	String		Yes		MOTOR Lubricant description



# **5.4 XML File: MOTOR\_ SevereService.xml**

Scheduled service SevereService descriptions.

#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
1	MOTOR_ SevereService	Elements	(VersionDate , SevereService +)		Yes		MOTOR PMSST SevereService root element
2	VersionDate	EMPTY		schema	Yes	Yes	Attribute "schema" indicates the schema version date (see date format note below)
				extraction		Yes	Attribute "extraction" indicates the extraction date (see date format note below)
3	SevereService	Elements	(SevereServiceDescription)	id	Yes	Yes	The attribute "id" is unique and represents MOTOR SevereService, this id is referenced in MOTOR_PM.xml as MOTOR_ SevereService id
				Code		Yes	Legacy PM SevereService Code value.
4	SevereServiceDescription	Туре	String		Yes		MOTOR SevereService description



# **5.5 XML File: MOTOR\_ Warranty.xml**

Scheduled service Warranty descriptions.

#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
1	MOTOR_ Warranty	Elements	(VersionDate , Warranty+)		Yes		MOTOR PMSST Warranty root element
2	VersionDate	EMPTY		schema	Yes	Yes	Attribute "schema" indicates the schema version date (see date format note below)
				extraction		Yes	Attribute "extraction" indicates the extraction date (see date format note below)
3	Warranty	Elements	(WarrantyDescription)	id	Yes	Yes	The attribute "id" is unique and represents MOTOR Warranty, this id is referenced in MOTOR_PM.xml as MOTOR_ Warranty id
				Code		Yes	Legacy PM Warranty Code value.
4	WarrantyDescription	Туре	String		Yes		MOTOR Warranty description



# 5.6 XML File: MOTOR\_ ServiceType.xml

Service Type descriptions for Replacement Parts. Service Type indicates rather or not the replacement part is required to be replaced for a specific Operation.

#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
1	MOTOR_ ServiceType	Elements	(VersionDate , ServiceType+)		Yes		MOTOR ServiceType root element
2	VersionDate	EMPTY		schema	Yes	Yes	Attribute "schema" indicates the schema version date (see date format note below)
				extraction		Yes	Attribute "extraction" indicates the extraction date (see date format note below)
3	ServiceType	Elements	(ServiceTypeDescription)	id	Yes	Yes	The attribute "id" is unique and represents MOTOR ServiceType, this id is referenced in MOTOR_PM.xml in the ReplacmentPart element as servicetype_ID
4	ServiceTypeDescription	Туре	String		Yes		MOTOR Warranty description



# 5.7 XML File: MOTOR\_Qualifier.xml

MOTOR Qualifier description definition. When there is more than one MOTOR application for the same MOTOR operation description on base vehicle with vehicle attributes, the operation will be qualified with appropriate description, which is not covered by VCDB attributes. For example: With Air condition and Without Air condition.

#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
1	MOTOR_Qualifier	Elements	(VersionDate , Qualifier+)		Yes		MOTOR Qualifier root element
2	VersionDate	EMPTY		schema	Yes	Yes	Atttibute "schema" indicates the schema version date (see date format note below)
				extraction		Yes	Attribute "extraction" indicates the extraction date (see date format note below)
				mtp		No	Attribute "mtp" indicates the MOTOR taxonomy version date, if applicable (see date format note below)
3	Qualifier	Elements	(QualifierDescription+, QualifierType, QualifierFamily)	id	Yes	Yes	Attribute "id" is unique for each MOTOR qualifier description. This id is referenced in MOTOR_PM.xml as Note
4	QualifierDescription	Elements	string	lang	Yes	No	MOTOR Qualifier description. The attribute "lang" indicates the language for the description; default language is English. Descriptions can be multiple, each with a



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							different "lang" attribute
5	QualifierType	Elements	(QualifierTypeDescription+)	id	Yes	Yes	MOTOR Qualifier type
6	QualifierTypeDescription	Elements	string	lang	Yes	No	MOTOR Qualifier type description. The attribute "lang" indicates the language for the description; default language is English. Descriptions can be multiple, each with a different "lang" attribute
7	QualifierFamily	Elements	(QualifierFamilyDescription+)	id	Yes	Yes	MOTOR Qualifier Family
8	QualifierFamilyDescription	Elements	string	lang	Yes	No	MOTOR Qualifier family description. The attribute "lang" indicates the language for the description; default language is English. Descriptions can be multiple, each with a different "lang" attribute

# **5.8 XML File: MOTOR\_PM.xml**

The core SST table. The delivered files will be broken up by AAIA Make and MOTOR Operation name. The character "&" in the Operation name will be replaced by "\_and\_". An example delivery file name is "MOTOR\_SST\_Acura\_Brake\_Line\_R\_and\_R.xml."

#	# Element name Cont Typ		Attributes	Element Reqd.	Attribute Reqd.	Description
,	1 PM Eleme	nts (Header , App+ , Footer)	version	Yes	Yes	MOTOR root element, attribute version indicates the PM specification version number



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
2	Header	Elements	(Company, SenderName, SenderPhone, SenderPhoneExt?, TransferDate, MfrCode?, DocumentTitle, DocFormNumber?, EffectiveDate, ApprovedFor?, SubmissionType, MapperCompany?, MapperCompany?, MapperPhoneExt?, MapperPhoneExt?, MapperEmail?, VcdbVersionDate, QdbVersionDate, PcdbVersionDate, MOTOR_OperationVersionDate, MOTOR_QualifierVersionDate)		Yes		Header section describes data file information such as supplier, effective date, various data elements version dates etc.
3	Company	Туре	String		Yes	N/A	Data supplier company name, MOTOR Information Systems
4	SenderName	Туре	String		Yes	N/A	Data supplier contact person name
5	SenderPhone	Туре	String		Yes	N/A	Data supplier contact person phone number
6	SenderPhoneExt	Туре	String		No	N/A	Data supplier contact person extension phone number
7	TransferDate	Туре	String		Yes	N/A	Data file create date, formatted as "CCYY-MM-DD", where "CC" represents century, "YY" represents two digit year, "MM" represents two digit month and "DD" represents two digit day



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
8	MfrCode	Туре	String		No	N/A	Vehicle manufacturer code
9	DocumentTitle	Type	String		Yes	N/A	Brief description of the contents in the document
10	DocFormNumber	Type	String		No	N/A	Data supplier's document number, if available
11	EffectiveDate	Type	String		Yes	N/A	Date on which the data contents in the file are effective from. Formatted as "CCYY-MM-DD", where "CC" represents century, "YY" represents two digit year, "MM" represents two digit month and "DD" represents two digit day.
12	ApprovedFor	Туре	String		No	N/A	ISO country code for which the data is approved. For US market the code is "US" and for Canada it is "CA"
13	SubmissionType	Туре	String		Yes	N/A	Data submission type, TEST, FULL or UPDATE. If the submission type is TEST or FULL, all applications in the file must have "action" attribute "A" to indicate "add" records. If the submission type is UPDATE, the "action" attribute can be either "A" for "add" records or "D" for "deleted" records. For the updated records, there will be two applications, one with



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							action attribute "D" and other with action attribute "A". By default, MOTOR Information Systems will deliver data in FULL
14	MapperCompany	Туре	String		No	N/A	Name of the company that mapped the data to AAIA standard data
15	MapperContact	Type	String		No	N/A	Contact person from the mapping
16	MapperPhone	Type	String		No	N/A	Mapping contact person's phone number
17	MapperPhoneExt	Type	String		No	N/A	Mapping contact person's extension phone number
18	MapperEmail	Type	String		No	N/A	Mapping contact person's e-mail address
19	VcdbVersionDate	Туре	String		Yes	N/A	Version date from Vcdb database. Formatted as "CCYY-MM-DD", where "CC" represents century, "YY" represents two digit year, "MM" represents two digit month and "DD" represents two digit day
20	QdbVersionDate	Туре	String		No	N/A	Version date from Qdb database, if it is used in this data deliverable. Formatted as "CCYY-MM-DD", where "CC" represents century, "YY" represents two digit year, "MM" represents two digit month and "DD" represents two digit day
21	PcdbVersionDate	Туре	String		Yes	N/A	Version date from Pcdb database. Formatted as "CCYY-MM-DD", where



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							"CC" represents century, "YY" represents two digit year, "MM" represents two digit month and "DD" represents two digit day
22	MOTOR_OperationVersionDat e	Туре	String		Yes	N/A	MOTOR Operation Taxonomy version date. Formatted as "CCYY- MM-DD", where "CC" represents century, "YY" represents two digit year, "MM" represents two digit month and "DD" represents two digit day
22	MOTOR_QualifierVersionDate	Туре	String		Yes	N/A	MOTOR Qualifiers version date. Formatted as "CCYY-MM-DD", where "CC" represents century, "YY" represents two digit year. "MM" represents two digit month and "DD" represents two digit day.
23	Арр	Elements	(BaseVehicle, SubModel?, MfrBodyCode?, BodyNumDoors?, BodyType?, , DriveType?, EngineBase?, EngineDesignation?, EngineVIN?, EngineVersion?, EngineMfr?, ValvesPerEngine?, FuelDeliveryType?, FuelDeliverySubType?, FuelSystemControlType?, FuelSystemDesign?, Aspiration?, CylinderHeadType?,	action id ref validate	Yes	Yes Yes No No	Groups MOTOR Operation data as an application. The values attribute "action" are "A" for "add" and "D" for "delete" applications. The attribute "id" uniquely identifies the application including base vehicle, vehicle attributes, MOTOR Operation and applicable MOTOR Qualifiers.



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
			FuelType?, IgnitionSystemType?, TransmissionMfrCode?, (TransmissionBase   (TransmissionType?, TransmissionControlType?, TransmissionNumSpeeds?))?, TransElecContolled?, TransmissionMfr?, TransferCaseBase?, TransferCaseMfr?, BedLength?, BedType?, WheelBase?, BrakeSystem?, FrontBrakeType?, RearBrakeType?, RearSpringType?, RearSpringType?, SteeringSystem?, SteeringType?, RestraintType?, Region?, Qual*, Note*, MfrLabel?, Position?, MOTOR_Operation, DisplayOrder?)?				The optional "ref" attribute references the source data.  The optional "validate" attribute indicates if the application must be validated against Vcdb data. Possible values are "yes" and "no"
	BaseVehicle	Туре	String	id	Yes	Yes	VCdb Attribute. References the Base Vehicle table in Vcdb database. The attribute "id" indicates the BaseVehicleID
	SubModel	Туре	String	id	No	Yes	VCdb Attribute. References the SubModel table in Vcdb database. The attribute "id" indicates the SubModelID.
	MfrBodyCode	Туре	String	id	No	Yes	VCdb Attribute.



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							References the MfrBodyCode table. The attribute "id" indicates the MfrBodyCodeID
	BodyNumDoors	Туре	String	id	No	Yes	VCdb Attribute. References the BodyNumDoors table. The attribute "id" indicates the BodyNumDoorsID
	BodyType	Туре	String	id	No	Yes	VCdb Attribute. References the BodyType table. The attribute "id" indicates the BodyTypeID
	DriveType	Туре	String	id	No	Yes	VCdb Attribute. References the DriveType table. The attribute "id" indicates the DriveTypeID
	EngineBase	Туре	String	id	No	Yes	VCdb Attribute. References the EngineBase table. The attribute "id" indicates the EngineBaseID
	EngineDesignation	Туре	String	id	No	Yes	VCdb Attribute. References the EngineDesignation table. The attribute "id" indicates the EngineDesignationID
	EngineVIN	Туре	String	id	No	Yes	VCdb Attribute. References the EngineVIN table. The attribute "id" indicates the EngineVINID
	EngineVersion	Type	String	id	No	Yes	VCdb Attribute.



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							References the EngineVersion table. The attribute "id" indicates the EngineVersionID
	EngineMfr	Туре	String	id	No	Yes	VCdb Attribute. The manufacturer that actually built the engine. References the Mfr table. The attribute "id" indicates the MfrID
	ValvesPerEngine	Туре	String	id	No	Yes	VCdb Attribute. References the Valves table. The attribute "id" indicates ValvesID
	FuelDeliveryType	Туре	String	id	No	Yes	VCdb Attribute. References the FuelDeliveryType table. The attribute "id" indicates FuelDeliveryTypeID
	FuelDeliverySubType	Туре	String	id	No	Yes	VCdb Attribute. References the FuelDeliverySubType table. The attribute "id" indicates FuelDeliverySubTypeID
	FuelSystemControlType	Туре	String	id	No	Yes	VCdb Attribute. References the FuelSystemControlType table. The attribute "id" indicates FuelSystemControlTypel D
	FuelSystemDesign	Туре	String	id	No	Yes	VCdb Attribute. References the FuelSystemDesign table. The attribute "id"



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							indicates FuelSystemDesignID
	Aspiration	Type	String	id	No	Yes	VCdb Attribute. References the Aspiration table. The attribute "id" indicates AspirationID
	CylinderHeadType	Туре	String	id	No	Yes	References the CylinderHeadType table. The attribute "id" indicates CylinderHeadTypeID
	FuelType	Туре	String	id	No	Yes	VCdb Attribute. References the FuelType table. The attribute "id" indicates FuelTypeID
	IgnitionSystemType	Туре	String	id	No	Yes	VCdb Attribute. References the IgnitionSystemType table. The attribute "id" indicates IgnitionSystemTypeID
	TransmissionMfrCode	Туре	String	id	No	Yes	VCdb Attribute. References the TransmissionMfrCode table. The attribute "id" indicates TransmissionMfrCodeID
	TransmissionBase	Туре	String	id	No	Yes	VCdb Attribute. References the TransmissionBase table. The attribute "id" indicates TransmissionBaseID
	TransmissionType	Туре	String	id	No	Yes	VCdb Attribute. References the TransmissionType table. The attribute "id"



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							indicates TransmissionTypeID
	TransmissionControlType	Туре	String	id	No	Yes	VCdb Attribute. References the TransmissionControlType table. The attribute "id" indicates TransmissionControlType ID
	TransmissionNumSpeeds	Туре	String	id	No	Yes	VCdb Attribute. References the TransmissionNumSpeeds table. The attribute "id" indicates TransmissionNumSpeeds ID
	TransElecContolled	Empty		id	No	Yes	VCdb Attribute. References the ElecControlled table. The attribute "id" indicates ElecControlledID
	TransmissionMfr	Туре	String	id	No	Yes	VCdb Attribute. The manufacturer that actually built the transmission. References the Mfr table. The attribute "id" indicates MfrID
	TransferCaseBase	Туре	String	id	No	Yes	VCdb Attribute. References the TransferCaseBase table. The attribute "id" indicates TransferCaseBaseID
	TransferCase	Туре	String	id	No	Yes	VCdb Attribute. References the TransferCase table. The attribute "id" indicates



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
	TransferCaseMfr	Туре	String	id	No	Yes	TransferCaseID  VCdb Attribute. The manufacturer that actually built the Transfer Case.  References the Mfr table.
	BedLength	Туре	String	id	No	Yes	The attribute "id" indicates MfrID  VCdb Attribute. References the BedLength table. The attribute "id" indicates
	BedType	Туре	String	id	No	Yes	BedLengthID  VCdb Attribute.  References the BedType table. The attribute "id" indicates BedTypeID
	WheelBase	Туре	String	id	No	Yes	VCdb Attribute. References the WheelBase table. The attribute "id" indicates WheelBaseID
	BrakeSystem	Туре	String	id	No	Yes	VCdb Attribute. References the BrakeSystem table. The attribute "id" indicates BrakeSystemID
	FrontBrakeType	Туре	String	id	No	Yes	VCdb Attribute. The brake type used on the front wheels. References the BrakeType table. The attribute "id" indicates BrakeTypeID
	RearBrakeType	Туре	String	id	No	Yes	VCdb Attribute. The brake type used on the rear wheels. References the BrakeType table. The attribute "id" indicates



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							BrakeTypeID
	BrakeABS	Туре	String	id	No	Yes	VCdb Attribute. References BrakeABS table. The attribute "id" indicates BrakeABSID
	FrontSpringType	Туре	String	id	No	Yes	VCdb Attribute. The basic suspension type used in the front of the vehicle. References the SpringType table. The attribute "id" indicates SpringTypeID
	RearSpringType	Туре	String	id	No	Yes	VCdb Attribute. The basic suspension type used in the rear of the vehicle. References the SpringType table. The attribute "id" indicates SpringTypeID
	SteeringSystem	Туре	String	id	No	Yes	VCdb Attribute. References the SteeringSystem table. The attribute "id" indicates SteeringSystemID
	SteeringType	Туре	String	id	No	Yes	VCdb Attribute. References the SteeringType table. The attribute "id" indicates SteeringTypeID
	RestraintType	Туре	String	id	No	Yes	VCdb Attribute. References the RestraintType table. The attribute "id" indicates RestraintTypeID
	Region	Туре	String	id	No	Yes	VCdb Attribute. Region where sold. References



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							the Region table. The attribute "id" indicates RegionID. This element will be rarely used with PMSST. In most cases, if region distinguishes between parts, a MOTOR qualifier will be used.
	Qual	Elements	(param* , text)	id	No	Yes	This element will be empty in the PMSST product. MOTOR Qualifiers are coded using Note.
	param	Туре	String	value	Yes	Yes	"param" substitutes the
				uom		No	value and " <b>uofm"</b> for Qdb qualifiers
				altvalue		No	
				altuom		No	
	text	Туре	String	id	No	Yes	Additional qualifier text for the coded qualifier
	Note	Туре	String	id	No	No	Note element describes
				lang		No	MOTOR_Qualifiers for
				displayorder		No	the application. The attribute "id" refers to the attribute "Qualifier.id" in MOTOR_Qualifier.xml file "vehicleattribute" indicates if the MOTOR Qualifier is a vehicle attribute not covered by ACES specs
				vehicleattrib ute		No	
	MfrLabel	Туре	String		No	N/A	Manufacturer specific descriptions, if available
	Position	Туре	String	id	No	Yes	PCdb Attribute. References the AAIA Position table. (Part of the PCDB database provided



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							by AAIA. MOTOR does not provide the Positions table). The attribute "id" indicates PositionID
	MOTOR_Operation	Element	( SST_Interval, ReplacementPart?)	id	Yes	Yes	Container for MOTOR application. The attribute "id" refers to OperationTaxonomy "id" in MOTOR_OperationTaxonomy.xml file
	SST_Interval	Elements	(SST_IndicatorImage, SST_IndicatorText, SST_Frequency, SST_IntervalMile, SST_IntervalKM, SST_OperatingHours, SST_IntervalMonth, SST_Lube, SST_Warranty, SST_SevereService, SST_Note1, SST_Note2)	Id	Yes	Yes	
	SST_IndicatorImage	Туре	String		No		Text description of display on vehicle dashboard when indicator light service is required
	SST_IndicatorText	Туре	String		No		Image file name of art showing display on vehicle dashboard when indicator light service is required
	SST_Frequency	Туре	String	id	No	Yes	Frequency in which recommendation Operation should be performed. Relates to Frequency.id in MOTOR_Frequency.xml
	SST_IntervalMile	Туре	String		No		Recommended



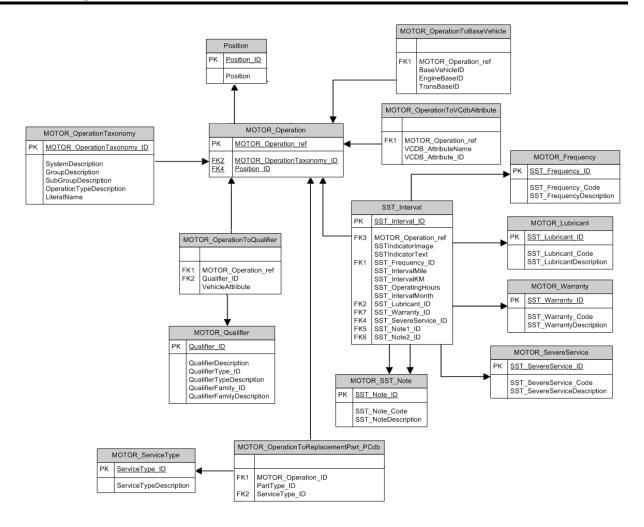
#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							maintenance interval in miles
	SSTInterval_KM	Туре	String		No		Recommended maintenance interval in kilometers
	SST_OperatingHours	Туре	String		No		Recommended maintenance interval in hours of vehicle operation
	SST_IntervalMonth	Туре	String		No		Recommended maintenance interval in months
	SST_Lube	Туре	String	id	No	Yes	Recommended lubricant. Relates to Lubricant.id in MOTOR_Lubricant.xml
	SST_Warranty	Туре	String	id	No	Yes	Warranty protection advisory. Relates to Warranty.id in MOTOR_ Warranty.xml
	SST_SevereService	Type	String	id	Yes	Yes	Severe service interval indicator (Y=yes; N=no) if indicated by the manufacturer. Recommended lubricant. Relates to SevereService.id in MOTOR_ SevereService.xml
	SST_Note1	Туре	String	id	No	Yes	Scheduled Service related notes data. Relates to SST_Note.id in MOTOR_SST_Note .xml
	SST_Note2	Туре	String	id	No	Yes	Scheduled Service related notes data. Relates to SST_Note.id in MOTOR_SST_Note



#	Element name	Content Type	Content Model	Attributes	Element Reqd.	Attribute Reqd.	Description
							.xml
	ReplacementPart	Elements	(PartType+)		No		Container for Parts applicable to the main operation
	PartType T	Туре	String	id	Yes	Yes	PCdb Attribute. Part applicable to operation. The attribute "id" refers to
				servicetype _id		Yes	PCDB Part Terminology ID The attribute servicetype_id references the MOTOR_ServiceType.xm I file,
	DisplayOrder	Type	String		No	Yes	Display order sequence number, when its required to display data in specific order.
	Footer	Туре	String		No	N/A	Container for footer tags, current specs call for Record count, which indicates total number of "App" elements in the file
	RecordCount	Туре	String		No	N/A	Indicates the number of (applications) "App" elements in the file



# 6 Sample Queries



The following sample queries assume that the delivery data has been imported into a relational database with the data model shown in the diagram above. In the above diagram, we did our best to match the field and table names with the elements found in the XML delivery and schema files. The above data model is only concerned with content and does not address other aspects that developers may be concerned with such as version control.

Vehicle configuration handling can vary greatly depending on infrastructure and application requirements. For example, if non-ACES vehicle key is used in the GUI, then you will need to determine how that vehicle key relates to VCdb and how the two should be mapped. Even when using the ACES vehicle key, there are different ways the vehicle configurations can be handled depending on requirements. For example, the application may require that engine is selected by the end user and thus the data would need to be exploded to VCdb Engine to query the correct data based customer input.



The following sample queries are designed as if the only VCdb vehicle coding applied to the data is BaseVehicleID (BaseVehicleID =1). This is the equivalent of Year, Make, Model, and Vehicle Type. The focus of these queries is the extraction of service intervals and gathering the PCdb part IDs that can be used to query a parts catalogue. Please note that these quires were used MS Access SQL syntax.

```
SELECT MOTOR_Operation.MOTOR_Operation_ref, MOTOR_OperationTaxonomy.LiteralName,
       Position. Position, SST_Interval.SST_Interval_ID, SST_Interval.SSTIndicatorImage,
       SST_Interval.SSTIndicatorText, MOTOR_Frequency.SST_Frequency_ID,
       MOTOR_Frequency.SST_Frequency_Code, MOTOR_Frequency.SST_FrequencyDescription,
       MOTOR Lubricant.SST Lubricant Code, MOTOR Lubricant.SST LubricantDescription,
       MOTOR_SST_Note.SST_Note_Code AS SST_Note_Code1,
       MOTOR_SST_Note.SST_NoteDescription AS SST_NoteDescription1,
       MOTOR SST Note 1.SST Note Code AS SST Note Code2,
       MOTOR_SST_Note_1.SST_NoteDescription AS SST_NoteDescription2
FROM (((MOTOR Operation INNER JOIN MOTOR OperationToBaseVehicle ON
       MOTOR_Operation.MOTOR_Operation_ref =
       MOTOR_OperationToBaseVehicle.MOTOR_Operation_ref) LEFT JOIN
       MOTOR OperationTaxonomy ON MOTOR Operation.MOTOR OperationTaxonomy ID =
       MOTOR_OperationTaxonomy.MOTOR_OperationTaxonomy_ID) LEFT JOIN [Position] ON
       MOTOR_Operation.Position_ID = Position.Position_ID) LEFT JOIN ((((SST_Interval LEFT
       JOIN MOTOR SST Note AS MOTOR SST Note 1 ON SST Interval.SST Note2 ID =
       MOTOR_SST_Note_1.SST_Note_ID) LEFT JOIN MOTOR_SST_Note ON
       SST_Interval.SST_Note1_ID = MOTOR_SST_Note.SST_Note_ID) LEFT JOIN
       MOTOR Frequency ON SST Interval.SST Frequency ID =
       MOTOR_Frequency_SST_Frequency_ID) LEFT JOIN MOTOR_Lubricant ON
       SST_Interval.SST_Lubricant_ID = MOTOR_Lubricant.SST_Lubricant_ID) LEFT JOIN
       MOTOR_Warranty ON SST_Interval.SST_Warranty_ID =
       MOTOR_Warranty.SST_Warranty_ID) ON MOTOR_Operation.MOTOR_Operation_ref =
       SST_Interval.MOTOR_Operation_ref
WHERE (((MOTOR_Frequency.SST_Frequency_ID) In (10,2081,4256,6,8,9)) AND
       ((MOTOR OperationToBaseVehicle.BaseVehicleID)=1))
```

# 6.1 Maintenance set where time/distance interval values not defined (Frequency F, L, M, N, P, R)

This sample query shows de-normalized data for maintenance records that do not have specific intervals attached. This query does not bring in MOTOR Qualifiers, Included Operations, VCdb Attributes, Replacement Parts, and distance or time specific interval fields.



# 6.2 MOTOR\_Operation set where Operation is dictated by Indicator Light

This sample query retrieves maintenance records for work that is required when a service indicator light is lit. Some records will have a description of the indicator that is shown on the vehicles dashboard. This field will be more fully populated as the product matures. Records that have a frequency code of "I" or have a value in the indicator text field fall into this category. If a record has a frequency of "I" and also mileage/km or time frequency data, then the operation should be performed at the interval (assumed to be frequency "Every") or when the indicator light is on; whichever comes first.

```
SELECT MOTOR Operation.MOTOR Operation ref, MOTOR OperationTaxonomy.LiteralName,
       Position.Position, SST_Interval.SST_Interval_ID, SST_Interval.SSTIndicatorImage,
       SST_Interval.SSTIndicatorText, MOTOR_Frequency.SST_Frequency_ID,
       MOTOR_Frequency.SST_Frequency_Code, MOTOR_Frequency.SST_FrequencyDescription,
       SST_Interval.SST_IntervalMile, SST_Interval.SST_IntervalKM,
       SST_Interval.SST_OperatingHours, SST_Interval.SST_IntervalMonth,
       MOTOR Lubricant.SST Lubricant Code, MOTOR Lubricant.SST LubricantDescription,
       MOTOR SST Note.SST Note Code AS SST Note Code1,
       MOTOR_SST_Note.SST_NoteDescription AS SST_NoteDescription1,
       MOTOR_SST_Note_1.SST_Note_Code AS SST_Note_Code2,
       MOTOR_SST_Note_1.SST_NoteDescription AS SST_NoteDescription2
FROM (((MOTOR_Operation INNER JOIN MOTOR_OperationToBaseVehicle ON
       MOTOR_Operation.MOTOR_Operation_ref =
       MOTOR_OperationToBaseVehicle.MOTOR_Operation_ref) LEFT JOIN
       MOTOR_OperationTaxonomy ON MOTOR_Operation.MOTOR_OperationTaxonomy_ID =
       MOTOR OperationTaxonomy.MOTOR OperationTaxonomy ID) LEFT JOIN [Position] ON
       MOTOR Operation. Position ID = Position. Position ID) LEFT JOIN (((((SST Interval LEFT
       JOIN MOTOR SST Note AS MOTOR SST Note 1 ON SST Interval.SST Note2 ID =
       MOTOR SST Note 1.SST Note ID) LEFT JOIN MOTOR SST Note ON
       SST_Interval.SST_Note1_ID = MOTOR_SST_Note.SST_Note_ID) LEFT JOIN
       MOTOR_Frequency ON SST_Interval.SST_Frequency_ID =
       MOTOR Frequency.SST Frequency ID) LEFT JOIN MOTOR Lubricant ON
       SST_Interval.SST_Lubricant_ID = MOTOR_Lubricant.SST_Lubricant_ID) LEFT JOIN
       MOTOR_Warranty ON SST_Interval.SST_Warranty_ID =
       MOTOR_Warranty.SST_Warranty_ID) ON MOTOR_Operation.MOTOR_Operation_ref =
       SST_Interval.MOTOR_Operation_ref
WHERE (((MOTOR Frequency.SST Frequency ID)=7) AND
       ((MOTOR_OperationToBaseVehicle.BaseVehicleID)=1)) OR
       (((SST_Interval.SSTIndicatorImage) Is Not Null)) OR (((SST_Interval.SSTIndicatorText) Is Not
       Null))
```

## 6.3 Get MOTOR Qualifiers for MOTOR\_Operation set

Each MOTOR\_Operation record in this data model, or each App element in the XML files, can be related to 0, 1, or more than one MOTOR Qualifier. It is important that an application takes the qualifiers into affect so that the same procedure is not duplicated and incorrect information is not included in a final estimate. If more than one Qualifier is



attached to an Operation, then both Qualifiers must be true for the record to be relevant. A possible strategy for having an end user select the correct qualification is to concatenate multiple qualifiers into a single string. Some MOTOR Qualifiers are of the Qualifier Type "Vehicle Type". These apply to the vehicle as a whole and not just the operation in context. The following sample query retrieves all the qualifiers for Indicator light type MOTOR\_Operation records. The results of this query will have a null Qualifier value if there are no Qualifiers attached and empty string Operation Taxonomy value if the qualifier is a Vehicle Attribute Qualifier.

# 6.4 Get Replacement Parts for a set of MOTOR\_Operation records

The following sample query demonstrates how to retrieve AAIA Part Type ID codes that can be used to retrieve part numbers and related information for a part application database coded to PCdb. This query retrieves part types for all records for a BaseVehicleID. This query uses a table Parts that is not in the above diagram. This table can be found in the AAIA ACES PCdb database that is made available to subscribers.



# 6.5 Explode Schedule Service Times to individual mileage intervals for specific vehicle.

The following steps demonstrate a portion of a potential method for creating an estimate for scheduled service using the SST database. The steps below only consider recommended services based on mileage. It is important that all relevant types of recommendations (including recommendations by months, indicator light, etc.) are also taken into affect. These queries include the gathering of PCdb Part Type IDs but not how to use them to fetch part prices. as that process will vary based on the provider dataset.

# 6.5.1 Append Frequency values of 1 (1st), 2 (2nd) and A (at) to temporary table

The following query will create TempTable1, populated with one time maintenance interval records where the value in the SSTInterval\_Miles value in SST\_Interval equals the mileage on the vehicle's odometer. For simplicity, this query is only exporting the foreign keys to the related look-up tables. Qualifiers are not part of this query but will be addressed in the following steps. Not all of the interval fields are included in the following query. The required fields will vary by application.

# 6.5.2 Append Frequency "E" to temporary table

The Frequency value of "E" indicates that the Operation is performed at every mileage that is a multiple of the SST\_IntervalMile value up until a max interval you choose to present. Generally, OE scheduled maintenance schedules are valid for intervals up to between 150,000 and 250,000 miles. Typically end users will want to see data for intervals within a range greater than and less than the vehicles current mileage.



To append the "E" intervals, cycle through each Interval\_ID for the vehicle in context with an SST\_Frequency\_ID of 5 and append incrementing multiples of the SST\_IntervalMile value until the max value desired has been reached or exceeded. The following is an example append query for an individual multiple based on the table TempTable1 created above. Replace  $\underline{x}$  value for an incrementing integer. This query uses representative SST\_Interval\_ID of 1.

#### 6.5.3 Append Frequency "X" to temporary table

The Frequency value of "X" indicates the interval values are for every multiple of the SST\_IntervalMile value after 1 or 2 initial service instances. This frequency should be handled the same as "E" with the exception that the initial service, 2 if existent or 1 if not, needs to be added to each multiple value. The following sample query shows how to retrieve the initial service value to append to an interval multiple for a unique Vehicle, MOTOR\_Operation\_ref, SST\_SevereService\_ID, Position, and MOTOR\_OperationTaxonomy\_ID combination. After retrieving the initial Mileage and Months, use the logic for the "E" frequency process and add the initial value to each Mileage and Month value.

**Note:** There is a potential that, early in the product's life, there are instances where an interval record relates to a MOTOR\_Operation\_ref record without a corresponding "1" or "2" interval. This can happen if the qualifiers or VCdb Attributes differ from those of the "x" record. In this case you will need to look at the other MOTOR\_Operation records with the same combination of MOTOR\_OperationTaxonomy\_ID and SST\_SevereService\_ID to find the initial mileage record.



#### 6.5.4 Add Qualifiers

Use the following query to retrieve all qualifier values related to the MOTOR\_Operation\_ref values in TempTable1. A single record could have more than one qualifier. For this hypothetical method, cycle through the qualifiers for each MOTOR\_Operation\_ref and concatenate multiple qualifiers for a single MOTOR\_Operation\_ref and delaminate by comma. Update the TempTable1 Qualifier field with these values. This query assumes that vehicle qualifiers and non-vehicle qualifiers are being treated equally and should be presented to end user for selection when creating the estimate.

#### 6.5.5 Create Customer Selection Table

The following two queries create a customer selection table. The first query appends the **maximum** Mileage value that **is less than or equal to** an end user entered vehicle mileage for each combination of MOTOR\_OperationTaxonomy\_ID, SST\_ServerService\_ID and Qualifier. The second query appends the **minimum** Mileage value that **is greater than** an end user entered vehicle mileage for each combination of

MOTOR\_OperationTaxonomy\_ID, SST\_ServerService\_ID, Position\_ID, and Qualifier. This table will also need a True/False column (not created by this query) to represent the end user selection. It is important to understand that often the same combination of MOTOR\_OperationTaxonomy\_ID, SST\_ServerService\_ID, Position\_ID, and Qualifier will appear twice, once for the max mileage before current and once for the min mileage after current. In these cases the end user should not be allowed to select both cases.



```
SELECT TempTable1.* INTO TempTable2
FROM TempTable1 INNER JOIN
       (SELECT TempTable1.MOTOR_OperationTaxonomy_ID, TempTable1.SST_SevereService_ID,
              TempTable1.Qualifier, TempTable1.Position_ID, Max(TempTable1.SST_IntervalMile)
               AS MaxMiles, First(TempTable1.SST_IntervalMonth) AS MaxMonth
       FROM TempTable1
       GROUP BY TempTable1.MOTOR OperationTaxonomy ID,
               TempTable1.SST SevereService ID, TempTable1.Qualifier, TempTable1.Position ID
       HAVING (((Max(TempTable1.SST IntervalMile))<=60000 And
               (Max(TempTable1.SST IntervalMile)) Is Not Null))) AS MaxMilesTable
       ON (TempTable1.SST_IntervalMonth = MaxMilesTable.MaxMonth) AND
       (TempTable1.SST IntervalMile = MaxMilesTable.MaxMiles) AND (TempTable1.Qualifier =
       MaxMilesTable.Qualifier) AND (TempTable1.SST_SevereService_ID =
       MaxMilesTable.SST_SevereService_ID) AND (TempTable1.MOTOR_OperationTaxonomy_ID
       = MaxMilesTable.MOTOR_OperationTaxonomy_ID) AND (TempTable1.Position_ID =
       MaxMilesTable.Position ID)
```

```
INSERT INTO TempTable2
SELECT TempTable1.*
FROM TempTable1 INNER JOIN
       (SELECT TempTable1.MOTOR OperationTaxonomy ID, TempTable1.SST SevereService ID,
               TempTable1.Qualifier, TempTable1.Position_ID, Min(TempTable1.SST_IntervalMile)
               AS MinMiles, First(TempTable1.SST IntervalMonth) AS MinMonth
       FROM TempTable1
       GROUP BY TempTable1.MOTOR_OperationTaxonomy_ID,
               TempTable1.SST_SevereService_ID, TempTable1.Qualifier, TempTable1.Position_ID
       HAVING (((Min(TempTable1.SST_IntervalMile))>60000 And
               (Min(TempTable1.SST_IntervalMile)) Is Not Null))) AS MinMilesTable
       ON (TempTable1.Position ID = MinMilesTable.Position ID) AND
       (TempTable1.MOTOR_OperationTaxonomy_ID =
       MinMilesTable.MOTOR OperationTaxonomy ID) AND (TempTable1.SST SevereService ID
       = MinMilesTable.SST_SevereService_ID) AND (TempTable1.Qualifier =
       MinMilesTable.Qualifier) AND (TempTable1.SST IntervalMile = MinMilesTable.MinMiles)
       AND (TempTable1.SST IntervalMonth = MinMilesTable.MinMonth)
```

#### 6.5.6 Get Replacement Parts

The following sample query shows how to retrieve PCdb IDs of parts that are likely needed to be replaced for the selected Operations.



#### 6.5.1 Get Included Operations

Included Operations are not common in the SST product but may need to be considered based on application requirements. Included Operations will help to identify when major Operations are repeated because two or more selected Operations include the same Operation, or an included Operation on one record is the same as a selected main Operation. The query below retrieves the Included Operations data for the selected Operations. Once this data is collected it can be processed to determine if more than one selected Operation contains the same Included Operation, or if selected Operation is a repeat of an included Operation on another record. This information can then be presented to the end user so the estimate can be adjusted accordingly.

When processing Included Operations, it is important to understand that a Position\_ID of 1 indicates that Position does not matter and if another Included Operation with the same taxonomy is present then there is overlap, no matter the position attached to the second record. Similarly, if an Included Operation does not have a qualifier attached, it overlaps with another Included Operation with any qualifier attached.

TempTable2.MOTOR\_Operation\_ref = IncludedOperation.MOTOR\_Operation\_ref

# 7 Data Usage Tips

## 7.1 Using MOTOR Qualifiers to Extended Vehicle Definitions

The MOTOR Qualifiers dataset can be used to extend vehicle definitions beyond the ACES VCdb standard. All vehicle definition qualifiers are assigned to the Qualifier Type of "Vehicle Attribute." These qualifiers are then further qualified by Qualifier Family. The grouping of the Qualifier family is based on the standard that only one Qualifier within a Qualifier Family can be true for a specific physical vehicle at a time. For example, "With Air Conditioning" and "Without Air Conditioning" are in the same family because both cannot be true about the same vehicle at the same time. By attaching declared vehicle attribute qualifiers to a unique physical vehicle, perhaps represented by VIN, an application can then predetermine that an application is not likely to apply to a vehicle if there is a vehicle attribute qualifier attached that belongs to a family wherein another qualifier from that same family has been declared for that vehicle. However, this qualifier classification is not perfect for all qualifier families. It is suggested that this logic is not used to hide data, but rather to promote specific data to the top of the list.



## 7.2 Getting the most of the App "ref" Attribute

Each application is delivered with an attribute of "ref." This attribute can be utilized to increase performance if parsing the XML output files into a SQL relational database by compressing the data. Each application which shares an equivalent ref have the same MOTOR Operation Name, set of Notes, set of VCdb Attributes (with the exception of EngineBaseID, and TransBaseID), Position, set of SST Intervals, set of Included Operations, and set of Replacement Parts. App records with the same "ref" can have differing BaseVehicleID, EngineBaseID, and TransBaseID.

The ref attribute is also a useful tool when communicating feedback to MOTOR. This ID will allow us to reference exactly the piece of data of concern.

## 8 Document

**Document History** 

Date	Version	Change Reference
1/2011	1.0	Initial release. Adapted from PMSST CDK version 1.1.
8/2015	1.1	Updated to new format.

