

Supplement Seven

Child Care DM Data Dictionary

Child Care Data Definitions

Tables

Table	
Name	Comment
CHILD_KEY	
DM_ACTIVITY_TYPE	
DM_AGE_BAND	
DM_AGE_GROUP	
DM_AGE_MTHS	The chronological age of an individual.
DM_ASSESSMENT	
DM_BODYPART	
DM_BUILDING_TYPE	Building Codes.
DM_CENTER_PRG	TSPC Reference Table has the primary program status of the center.
DM_CENTER_STAT	TSTC - Center Status
DM_CENTER_TYPE	
DM_CHILD	
DM_CHILD_CARE_TYPE	
DM_CHILD_PROVIDER	
DM_CHLD_CR_REASON	
DM_CHLDCR_WORKER	Dimensional Table contains worker information. NS14WRKR IMS database.
DM_CMPLNCE_STATUS	Dimensional Table contains a list of compliance status ato fulfill the official requirements.
DM_CNTY_DISTRICT	Dimensional Table - association between county and district. TDNC
DM_COMMUNICATION	
DM_COUNTY	Identifies the specific county associated with the processing of the case.
DM_CR_TKR_PARENT	
DM_CURRICULUM	
DM_DISTRICT_GROUP	Dimensional Table contains OHIO district information. TDIS
DM_ELIG_SOURCE	
DM_ENFRMNT_STAT	Dimensional Table contains a list of enforcement actions pursuant to Chapter 119 of the ORC that could potentially close the program should the center be unable to achieve compliance.
DM_FIELD_OFFICE	
DM_FNDNG	Dimensional Table contains a list of inspection rule parts that will be used to group inspection findings. From TPART Reference Table.
DM_FNDNG_GRP	Dimensional Table contains a list of inspection rule parts that will be used to group inspection findings. From TPART Reference Table.
DM_GENDER	Identifies a gender of a person.
DM_ILLNESS_TYPE	
DM_INCDNT	
DM_INCIDENT_GROUP	
DM_INCIDENT_LOC	
DM_INCIDENT_RESULT	
DM_INCIDENT_TYPE	
DM_INCOME_LEVEL	
DM_INJURY_TYPE	
DM_INSPECTION_NOTICE	TCIN Reference Table has the type of inspection notification given

Table	
Name	Comment
	to the center.
DM_INSPECTION_SCOPE	TCIS Reference Table
DM_INSPECTION_TYPE	TCIT - Inspection Type
DM_INVOICE_STAT	
DM_INVSTGTRY_PLAN	Table represents the method used to investigate the complaint.
DM_JOBPOS	Job Descriptions.
DM_MED_PLAN	
DM_MONTH	One of the twelve divisions of the calendar year.
DM_PARENT_DEMO_PRF	
DM_PAYMENT_SRC	
DM_PERSON	A Party that is a human individual.
DM_PRIORITY_RANK	
DM_PROVIDER	This database comprises all information related to child care centers licensed by the State of Ohio.
DM_PROVIDER_GROUP	
DM_PROVIDER_TYPE	
DM_QUALIFICATION	
DM_QUALITY_RTNG	TSQR Reference Table has quality rating information.
DM_QUARTER	Subset of Month to facilitate manipulation of quarterly tables like FACT_PERSON_WAGE, FACT_PERSON_JOBENT, etc.
DM_RACE	Identifies a racial group of a person.
DM_RATING_STATUS	
DM_REFERRAL	
DM_REFERRAL_SPECIALIST	
DM_RISK_STAT	Dimensional Table contains severity indicator that will be used to group noncompliant inspections under proper group. Manually loaded.
DM_ROOM	
DM_RULE	Dimensional Table contains complaint related rule and rule parts. TRUEL and TPART
DM_RULE_CHAPTER	Dimensional Table contains the center types by ORC chapter
DM_SCREENING_TOOL	
DM_SRVC_CONTR	Type of licensed center.
DM_SULS_WORKER	
DM_SUTQ_ADMIN	
DM_SUTQAPPL_STATUS	
DM_SUTQAPPL_TYPE	
DM_SUTQLETTER	
DM_TEACHER	
DM_TME_SPAN	
FACT_AWARD_PROGRAM	
FACT_CHILD	
FACT_CHLDCR_INCDNT	
FACT_EARLY_LEARNING	
FACT_INCDNT_INJURY	
FACT_INSP_COMPLAINT	Fact Table has the inspection related complaints information.
FACT_LICENSE_CENTER	This database comprises all information related to licensed child care centers inspection information.
FACT_LICENSE_INSP	This database comprises all information related to licensed child care centers inspection information and inspection findings in case of noncompliance.
FACT_SUTQ_APPLICATION	
FACT_SUTQ_CENTER	
FACT_SUTQ_LETTER_SENT	
FACT_SUTQ_PROGRAM	This database comprises all information related to licensed child care centers inspection information.
FACT_SUTQ_RATING	
FACT_SUTQ_VISIT	
FACT_SUTQ_VISITROLE	
PARENT_KEY	
PARTY_KEY_XREF	A cross reference between the timestamp person ID and the integer

Table	
Name	Comment
	person surrogate key. TRIGGER: This table is populated by triggers in the PROVIDER and PERSON tables whenever a row is inserted into either of those tables.
STG_CHLD_CR_CSLOAD	
STG_SUTQ_ADMIN_EMAIL	

Column(s) of "CHILD_KEY" Table		
Name	Datatype	Comment
CHILD_KEY	INTEGER	Surrogate Key
CNTY_CD	CHAR(2)	
CHILD_ID	CHAR(11)	

Column(s) of "DM_ACTIVITY_TYPE" Table		
Name	Datatype	Comment
ACTIVITY_CD	CHAR(3)	Denotes the activity(s) the child was engaged in the incident/injury. Values are: D1 75 76 77 78 80 81 82 83
ACTIVITY_DESC	CHAR(75)	Description of the activity the child was engaged in the incident. Values: CODE DESC D1 DIAPER CHANGE 75 ARRIVAL/DEPARTURE 76 NAPTIME/REST PERIOD 77 CLASSROOM ACTIVITY 78 MEALS/SNACK 80 INDOOR PLAY/GROUP ACTIVITIES/FREE PLAY 81 OUTDOOR PLAY 82 TRANSITION BETWEEN ACTIVITIES 83 BUS/VEHICLE/DURING TRANSPORTATION XX NO VALUE

Column(s) of "DM_AGE_BAND" Table		
Name	Datatype	Comment
AGE_BAND_CD	CHAR(2)	This is a code to identify a specific age range.
AGE_BAND_DESC	CHAR(20)	This is the description of the age range.

Column(s) of "DM_AGE_GROUP" Table		
Name	Datatype	Comment
AGE_GRP_CD	CHAR(1)	Denotes the assigned age group associated with the child for the incident/injury on the associated segment. Table-driven values are: I - Infant P - Preschool S - School age T - Toddler Y - Young Infant X NO VALUE
AGE_GRP_DESC	CHAR(30)	Description of the age groups. CODE DESC I INFANT P PRESCHOOL S SCHOOL AGE T TODDLER Y YOUNG INFANT

Column(s) of "DM_AGE_GROUP" Table		
Name	Datatype	Comment
		X NO VALUE

Column(s) of "DM_AGE_MTHS" Table		
Name	Datatype	Comment
AGE_MONTHS	SMALLINT	The age of the individual in months at the time of service.
AGE_BAND_CD	CHAR(2)	This is a code assigned to the age in months to determine which age range.

Column(s) of "DM_ASSESSMENT" Table		
Name	Datatype	Comment
ASSMNT_ID	SMALLINT	An indicator for the Assessment tool used by the Program to assess children enrolled.
ASSMNT_DESC	CHAR(50)	Description of the Assessment tool used by the Program to assess children enrolled. Values: 1. NO VALUE 2. High Scope - COR 3. Self Created 4. Galileo 5. Creative Curriculum Developmental Continuum 6. Other

Column(s) of "DM_BODYPART" Table		
Name	Datatype	Comment
BODY_PART_CD	CHAR(3)	Denotes the body part(s) for the incident/injury. Values are from TBOD: E1 E2 66A 66B 66C 66D 66E 66F 66G 67 68 69 70 71A 71B
BODY_PART_DESC	CHAR(75)	Description of the body parts for the incident: Values : CODE DESC E1 LUNGS/DIFFICULTY BREATHING E2 WHOLE BODY 66A HEAD 66B FACE 66C EYE 66D CHIN 66E EAR 66F NOSE 66G THROAT 67 MOUTH/TEETH 68 BACK 69 GENITALS/BUTTOCKS 70 NECK 71A ARM 71B FINGERS 71C HAND 71D SHOULDER/COLLAR BONE 72 FRONT OF TRUNK/STOMACH 73A LEG 73B KNEE 73C TOE 73D FOOT XX NO VALUE

Column(s) of "DM_BUILDING_TYPE" Table		
Name	Datatype	Comment

Column(s) of "DM_BUILDING_TYPE" Table		
Name	Datatype	Comment
BLDNG_CD	CHAR(1)	The building type of the center
BLDNG_DESC	CHAR(30)	Description of the building type: Values: Code Description 1 E 2 E with I-2 3 A-4 4 BB-303 5 BB-57 6 BB-61 7 B (OCC.-49) 8 I-2 9 I-4

Column(s) of "DM_CENTER_PRG" Table		
Name	Datatype	Comment
CENTER_PRG_CD	CHAR(2)	Indicates the primary program status of the center. Values are from (TSPC ref tbl): A: APPLICATION C: CLOSED F: FISCAL L: LICENSE O: NO APPLICATION R: RENEWAL T: TRANSFERRED
CENTER_PRG_DESC	CHAR(30)	Description of the program: CENTER_PRG_CD CENTER_PRG_DESC A APPLICATION C CLOSED F FISCAL L LICENSE O NO APPLICATION R RENEWAL T TRANSFERRED

Column(s) of "DM_CENTER_STAT" Table		
Name	Datatype	Comment
CENTER_STAT_CD	CHAR(2)	Indicates the status of license program. Values are (TSTC ref table): A: FILING DEADLINE MISSED B: CENTER RELOCATED C: OWNERSHIP CHANGED D: LICENSE REVOKED E: VOLUNTARY CLOSURE F: FACILITY TYPE CHANGE G: CLOSED VIA SETTLEMENT H: CLOSED FOR FINANCIAL REASONS I: TIME LIMIT EXCEEDED O: DEFAULT VALUE P: PROVISIONAL (SIX-MONTH)
CENTER_STAT_DESC	CHAR(30)	Description of the type of license program. Values are (TSTC ref table): CENTER_STAT_CD CENTER_STAT_DESC A FILING DEADLINE MISSED B CENTER RELOCATED C OWNERSHIP CHANGED D LICENSE REVOKED

Column(s) of "DM_CENTER_STAT" Table		
Name	Datatype	Comment
		E VOLUNTARY CLOSURE F FACILITY TYPE CHANGE G CLOSED VIA SETTLEMENT H CLOSED FOR FINANCIAL REASONS I TIME LIMIT EXCEEDED O DEFAULT VALUE P PROVISIONAL (SIX-MONTH)

Column(s) of "DM_CENTER_TYPE" Table		
Name	Datatype	Comment
CENTER_TYP_CD	CHAR(1)	Center Types: 1,4,7
CENTER_TYP_DESC	CHAR(30)	Center Type description: 1 CENTER 4 TYPE A HOME 7 DAY CAMP

Column(s) of "DM_CHILD" Table		
Name	Datatype	Comment
CHILD_KEY	INTEGER	Surrogate Key
CNTY_CD	CHAR(2)	A unique code that identifies a specific county. (e.g. 87, 88 etc.)
SSN_NBR	INTEGER	9-digit SSN of the child.
CHILD_ID	CHAR(11)	11-digit CDJFS-assigned unique identified for the child on the associated segment. Should contain the child's SSN with leading zeros; however, many counties use caretaker parent's SSN with either a 2-digit prefix or suffix.
BIRTH_DTE	DATE	Date of birth of the child. The child cannot be over 13 years of age for regular child care services and 18 years of age for special needs service.
LAST_NAME	CHAR(30)	Last name of the child.
FIRST_NAME	CHAR(20)	First name of the child.
RACE_TYP_CD	CHAR(2)	A code identifying a person's race. (e.g. A, B, W etc.)
GNDR_TYP_CD	CHAR(2)	A code that identifies the sex of a person (e.g M, F, U)
DW_PARTY_KEY	INTEGER	%AttDef

Column(s) of "DM_CHILD_CARE_TYPE" Table		
Name	Datatype	Comment
FT_PT_CD	CHAR(1)	Indicates the type of roster record or intended action. Values are: Roster Record Types F - Full-time child care roster P - Part-time child care roster O - Overpayment child care roster U - Underpayment child care roster A - Advance child care roster D - Deposit child care roster Intended Action C - Copy existing roster line R - Remove existing roster line
FT_PT_DESC	CHAR(45)	The child care type text description. The values are: Roster Record Types

Column(s) of "DM_CHILD_CARE_TYPE" Table		
Name	Datatype	Comment
		F - Full-time child care roster P - Part-time child care roster O - Overpayment child care roster U - Underpayment child care roster A - Advance child care roster D - Deposit child care roster Intended Action C - Copy existing roster line R - Remove existing roster line

Column(s) of "DM_CHILD_PROVIDER" Table		
Name	Datatype	Comment
PROVIDER_ID	CHAR(15)	The Provider ID for which the child is authorized.
APPROVAL_DTE	DATE	The approval date for services from the provider.
CNTY_CD	CHAR(2)	A unique code that identifies a specific county. (e.g. 87, 88 etc.)
PROVIDER_TYP	CHAR(2)	Code associated with the type of the provider on the associated segment. Values are: Licensed Centers: A - Type A Homes C - Child Care Center D - Day Camp H - Head Start S - School Affiliated Provider Certified Providers: B - Type B Homes I - In-Home Aide L - Limited Provider
SSN_LIC_NBR	CHAR(15)	The provider's ssn or license number.
ADDR_LN_1	CHAR(45)	The provider's street address.
CITY_NM	CHAR(25)	The provider's city name.
ADDR_STATE_CD	CHAR(2)	The provider's state.
ADDR_ZIP_CD_5	INTEGER	The provider's 5 digit zip code.
ADDR_ZIP_CD_4	SMALLINT	The provider's 4 digit zip code.
PHONE_NBR	CHAR(10)	The provider's phone number.
FED_TAX_ID	CHAR(9)	The provider's federal tax id.
JHS_IND	CHAR(1)	The provider's JHS.

Column(s) of "DM_CHLD_CR_REASON" Table		
Name	Datatype	Comment
EMP_TRNG_CD	CHAR(1)	Indicates the reason the caretaker parent needs child care services. Values are: E - Employment T - Training B - Both O - Other
EMP_TRNG_DESC	CHAR(15)	Child care reason description. Values are: E - Employment T - Training B - Both O - Other

Column(s) of "DM_CHLD_CR_WORKER" Table		
Name	Datatype	Comment

Column(s) of "DM_CHLDCR_WORKER" Table		
Name	Datatype	Comment
WRKR_CD	CHAR(6)	WORK14-CDE-USERID Six character worker ID that meets the specific criteria entered on the Licensing Personnel Menu
WRKR_FIRST_NM	CHAR(15)	WORK14-NAM-FIRST The first name of the worker.
WRKR_MI	CHAR(1)	WORK14-NAM-MIDDLE The middle initial of the worker.
WRKR_LAST_NM	CHAR(15)	WORK14-NAM-LAST The last name of the worker.
WRKR_JOBPOS_CD	CHAR(1)	One-character code that indicates the specialist's job position in the child care system. L S O
DISTRICT_CD	CHAR(1)	Official ODJFS district office number.
WRKR_BGN_DTE	DATE	WORK14-DTE-BEGIN The first day the specialist has access to the CCIDS/Licensing subsystem.
WRKR_END_DTE	DATE	WORK14-DTE-END The ending date that the specialist is authorized to use the CCIDS/Licensing subsystem.

Column(s) of "DM_CMLNCE_STATUS" Table		
Name	Datatype	Comment
CMLNCE_STAT_CD	CHAR(1)	The current compliance status of the specific rule. Values: A D I O P S U
CMLNCE_STAT_DESC	CHAR(30)	The description of the current compliance status of the specific rule: Complaints actually use table TCRS. A ALLEGED D NOT DETERMINED O ADDITIONAL OUT OF COMPLIANCE S SUBSTANTIATED U UNSUBSTANTIATED Inspections use table TCPL I IN COMPLIANCE O OUT OF COMPLIANCE P IN APPEAL S SUBSTANTIATED

Column(s) of "DM_CNTY_DISTRICT" Table		
Name	Datatype	Comment
CNTY_CD	CHAR(2)	County Code
DISTRICT_CD	CHAR(1)	District codes of the counties: 0 : 89, 90 1 : 07, 10, 15, 34, 41, 50, 67, 76, 77, 78, 79, 85, 91 2 : 01, 08, 09, 13, 14, 31, 36, 57, 68, 83, 92

Column(s) of "DM_CNTY_DISTRICT" Table		
Name	Datatype	Comment
		3: 04 ,18 ,28 ,43,47 ,52 ,93 4: 03, 05, 11, 12, 16, 17, 19, 21,23,24,25,27,29,30,33, 37,38,40,42,44,45,46,49,51,53,54,55,56,58, 59,60,61,64, 65,66,70,71,73,75,80,82,84,88,94 5: 02,06,20,22,26,32,35,39,48,62,63,69,72,74, 81,86,87,95

Column(s) of "DM_COMMUNICATION" Table		
Name	Datatype	Comment
CMNCTN_ID	INTEGER	Indicates if the Program uses a communication tool to parents/guardians regarding their infants and toddlers.
CMNCTN_DESC	CHAR(50)	Indicates if the Program uses a communication tool to parents/guardians regarding their infants and toddlers.

Column(s) of "DM_COUNTY" Table		
Name	Datatype	Comment
CNTY_CD	CHAR(2)	A unique code that identifies a specific county. (e.g. 87, 88 etc.)
CNTY_DESC	CHAR(40)	A text description of County Code. (e.g. Wood, Wyandot etc.)
FAMILY_CNT	INTEGER	The number of families in the county according to the 2000 Census.
POVERTY_100_CNT	INTEGER	The number of families in the county who are at or below the 100% Federal Poverty Level according to the 2000 Census.
POVERTY_130_CNT	INTEGER	The number of families in the county who are above the 100% but at or below the 135% Federal Poverty Level according to the 2000 Census.
POVERTY_185_CNT	INTEGER	The number of families in the county who are above the 135% but at or below the 185% Federal Poverty Level according to the 2000 Census.

Column(s) of "DM_CR_TKR_PARENT" Table		
Name	Datatype	Comment
PARENT_KEY	INTEGER	Surrogate Key that is generated based on a new Parent Key and Service Date.
CNTY_CD	CHAR(2)	A unique code that identifies a specific county. (e.g. 87, 88 etc.)
SSN_NBR	INTEGER	9-digit SSN of the individual identified on the child care application as the primary caretaker parent of the child.
LAST_NAME	CHAR(30)	Last name of the caretaker parent.
FIRST_NAME	CHAR(20)	First name of the caretaker parent.
BIRTH_DTE	DATE	Date of birth of the caretaker parent. The caretaker parent should be at least 11 years old, based on date at time of entry.
ADDR_LN_1	CHAR(45)	Street address for the application caretaker parent.
CITY_NM	CHAR(25)	City for the application caretaker parent.
ADDR_STATE_CD	CHAR(2)	USPS-standard state abbreviation for the physical location for the application caretaker parent. Values are: IN - Indiana

Column(s) of "DM_CR_TKR_PARENT" Table		
Name	Datatype	Comment
		KY - Kentucky MI - Michigan OH - Ohio PA - Pennsylvania WV - West Virginia.
ADDR_ZIP_CD_5	INTEGER	First 5-digits of the zip code for the application caretaker parent.
ADDR_ZIP_CD_4	SMALLINT	Last 4-digits of the zip code for the application caretaker parent.
NBR_CR_TKR_PARENTS	SMALLINT	For the household applying for child care services, the total number of caretaker parents.
EMP_TRNG_CD	CHAR(1)	Indicates the reason the caretaker parent needs child care services. Values are: E - Employment T - Training B - Both O - Other
DW_PARTY_KEY	INTEGER	The Party Key of the individual from the Data Warehouse.

Column(s) of "DM_CURRICULUM" Table		
Name	Datatype	Comment
CRCLM_ID	SMALLINT	The Curriculum used by the Program to assess children. Values: 1 to 8
CRCLM_DESC	CHAR(50)	Description of the curriculum to assess children. Values: 1. NO VALUE 2. Creative Curriculum 3. Innovations 4. Reggio 5. Montessori 6. High Scope 7. Self Created 8. Other

Column(s) of "DM_DISTRICT_GROUP" Table		
Name	Datatype	Comment
DISTRICT_CD	CHAR(1)	Official ODJFS district office number of the district for the alleged operation on the associated segment. Values are: 0 General 1 Canton 2 Cincinnati 3 Cleveland 4 Columbus 5 Toledo
DISTRICT_DESC	CHAR(30)	Description of the districts: Values are: 0 General 1 Canton 2 Cincinnati 3 Cleveland 4 Columbus 5 Toledo

Column(s) of "DM_ELIG_SOURCE" Table		
Name	Datatype	Comment

Column(s) of "DM_ELIG_SOURCE" Table

Name	Datatype	Comment
ELIG_SRC_CD	CHAR(2)	<p>Two-digit status code which correlates to the pay source for the authorized provider/child on the associated segment. Values are:</p> <p>01 - SSI Child care services are provided to an SSI recipient because of SSI eligibility. Valid pay sources are: 320, 899</p> <p>02 - MEDICAID Child care services are provided to a Medicaid recipient because of Medicaid eligibility. Valid pay sources are: 320, 899</p> <p>04 - OWF Ohio Works First . Valid pay sources are: 312, 314, 320, 322, 324, 326, 342, 899</p> <p>06 - INCOME ELIGIBLE Child care services are provided free of charge because the family's only sources of non-exempt gross monthly income is OWF. Valid pay sources are: 320, 321, 322, 324, 342, 899</p> <p>07 - WITHOUT REGARD TO INCOME Protective child care services are provided free of charge without regard to family income. Valid pay sources are: 323, 325, 329, 899</p> <p>08 - FEE Non-guaranteed child care services are provided for current recipients or new applicants whose adjusted monthly income is at or below 150% (as of 6/9/03) of the current federal poverty level. Also included are families receiving transitional child care services. Families are required to pay a a fee for child care services. Valid pay sources are: 312, 313, 314, 320, 321, 322, 323, 324, 325, 326, 327, 329, 342, 899</p> <p>10 - OTHER Child care services are provided for an individual who does not meet any of the preceding eligibility status codes. Valid pay sources are: 899</p>
ELIG_SRC_DESC	CHAR(25)	<p>The eligible source text description. The values are:</p> <p>SSI Child care services are provided to an SSI recipient because of SSI eligibility. Valid pay sources are: 320, 899</p> <p>MEDICAID Child care services are provided to a Medicaid recipient because of Medicaid eligibility. Valid pay sources are: 320, 899</p> <p>OWF Ohio Works First . Valid pay sources are: 312, 314, 320, 322, 324, 326, 342, 899</p> <p>INCOME ELIGIBLE Child care services are provided free of charge because the family's only sources of non-exempt gross monthly income is OWF. Valid pay sources are: 320, 321, 322, 324, 342, 899</p> <p>WITHOUT REGARD TO INCOME Protective child care services are provided</p>

Column(s) of "DM_ELIG_SOURCE" Table		
Name	Datatype	Comment
		<p>free of charge without regard to family income. Valid pay sources are: 323, 325, 329, 899</p> <p>FEE</p> <p>Non-guaranteed child care services are provided for current recipients or new applicants whose adjusted monthly income is at or below 150% (as of 6/9/03) of the current federal poverty level. Also included are families receiving transitional child care services. Families are required to pay a fee for child care services. Valid pay sources are: 312, 313, 314, 320, 321, 322, 323, 324, 325, 326, 327, 329, 342, 899</p> <p>OTHER</p> <p>Child care services are provided for an individual who does not meet any of the preceding eligibility status codes. Valid pay sources are: 899</p>

Column(s) of "DM_ENFRMNT_STAT" Table		
Name	Datatype	Comment
ENFRMNT_CD	CHAR(2)	The current enforcement status of the license program.
ENFRMNT_DESC	CHAR(30)	<p>The description of the current enforcement status of the license program.</p> <p>Values:</p> <p>ENFRMNT_CD ENFRMNT_DESC</p> <p>O PENDING</p> <p>X INJUNCTION</p> <p>Y CHAPTER 119</p> <p>Z COMBINATION</p>

Column(s) of "DM_FIELD_OFFICE" Table		
Name	Datatype	Comment
CNTY_NUM	SMALLINT	County number.
FIELD_OFFICE_DESC	CHAR(25)	<p>Description of the field office:</p> <p>Cincinnati</p> <p>Cleveland/Canton</p> <p>Columbus</p> <p>Toledo</p>
CNTY_NM	CHAR(12)	County Name.

Column(s) of "DM_FNDNG" Table		
Name	Datatype	Comment
FNDNG_GRP_KEY	INTEGER	<p>Rule Part Surrogate Key:</p> <p>GRP_EFF_DTE,</p> <p>CHAPTER_CD,</p> <p>GRP_RULE_CD,</p> <p>GRP_PART_CD,</p> <p>1 and 2 : RESERVED FOR N/A - no findings and corrections</p> <p>1=0001-01-01, 212,000,0, XX</p> <p>2=0001-01-01, 213,000,0,XX</p>
FNDNG_CD	CHAR(5)	The 5-char licensing rule finding code. An observation made during an inspection that substantiates or not a facilities adherence to the rules outlined in the Ohio Administrative Code.
FNDNG_EFF_DTE	DATE	Findings effective date.
CHAPTER_CD	CHAR(3)	Code indicates the version of the Ohio

Column(s) of "DM_FNDNG" Table		
Name	Datatype	Comment
		Revised Code (ORC) or Ohio Administrative Code (OAC). Values are: 212 213 214 218
FNDNG_RULE_CD	CHAR(3)	Findings rule code.
FNDNG_PRT_CD	CHAR(1)	Findings rule part code.
FNDNG_SVRTY_NBR	SMALLINT	Severity Indicator that make a non-compliance serious risk.
FNDNG_DESC	CHAR(75)	Text description of the inspection finding rule parts.

Column(s) of "DM_FNDNG_GRP" Table		
Name	Datatype	Comment
FNDNG_GRP_KEY	INTEGER	Rule Part Surrogate Key: GRP_EFF_DTE, CHAPTER_CD, GRP_RULE_CD, GRP_PART_CD, 1 and 2 : RESERVED FOR N/A - no findings and corrections 1=0001-01-01, 212,000,0, XX 2=0001-01-01, 213,000,0,XX
GRP_EFF_DTE	DATE	Inspection Rule Effective Date.
CHAPTER_CD	CHAR(3)	Code indicates the version of the Ohio Revised Code (ORC) or Ohio Administrative Code (OAC). Values are: 212, 213, 214, 218
GRP_RULE_CD	CHAR(3)	The number given to a regulation or bylaw governing procedures for licensed facilities as defined in the Ohio Administrative Code Revised. High level three digit code that groups the finding rules.
GRP_PRT_CD	CHAR(1)	Code indicates the rule part level of the inspection rules.
GRP_DESC	CHAR(75)	Text description of the inspection finding rule parts.

Column(s) of "DM_GENDER" Table		
Name	Datatype	Comment
GNDR_TYP_CD	CHAR(2)	A code that identifies the sex of a person (e.g M, F, U)
GNDR_DESC	CHAR(20)	A text description of Gender Code. (e.g Male, Female, Unknown)

Column(s) of "DM_ILLNESS_TYPE" Table		
Name	Datatype	Comment
ILLNESS_CD	CHAR(2)	Code indicates the Illness.
ILLNESS_DESC	CHAR(75)	The description of the Illness Values (TILL ref tbl): B1 ALLERGIC REACTION/ASTHMA B2 COLLAPSE/FAINT B3 STOMACHACHE/VOMITING/DIARRHEA A B4 OTHER ILLNESS (SPECIFY IN SUMMARY SECTION) B5 N/A-INJURY/INCIDENT B6 FEVER

Column(s) of "DM_ILLNESS_TYPE" Table		
Name	Datatype	Comment
		B7 DIAPER RASH 32 SEIZURE 36 NO PULSE/BREATHING XX NO VALUE

Column(s) of "DM_INCDNT" Table		
Name	Datatype	Comment
INCIDENT_KEY	INTEGER	Surrogate Key
LAST_NAME	CHAR(30)	Injured Childs Last Name.
FIRST_NAME	CHAR(30)	Injured Childs First name
BIRTH_DTE	DATE	Injured Childs Birth Date
INCIDENT_DTE	DATE	Indicates the date the incident/injury occurred on the associated segment. Format: CCYYMMDD
INCIDENT_TIME	SMALLINT	Indicates the time of day the incident/injury occurred. Format: HHMM

Column(s) of "DM_INCIDENT_GROUP" Table		
Name	Datatype	Comment
INJRY_TYP_CD	CHAR(2)	Code identifies the type of the injury. Values are (TIH ref tbl): D: Incident E: Illness F: Injury X: NO VALUE
INJRY_TYP_DESC	CHAR(30)	Group description of the type of the injury. Values are (TINJ): D: Incident E: Illness F: Injury X: NO VALUE

Column(s) of "DM_INCIDENT_LOC" Table		
Name	Datatype	Comment
INCDNT_LOC_CD	CHAR(2)	Denotes the location(s) of the incident. Values are: F1 F2 F3 F4 F5 57 58 59 60 61 62 63 64 79
INCDNT_LOC_DESC	CHAR(75)	Description of the location where the incident happened. Values are from TWHR: CODE DESC F1 HIGH CHAIR F2 CRIB F3 IN VEHICLE F4 ON FIELD TRIP/ROUTINE TRIP F5 POOL 57 CLASSROOM 58 HALL/DOORWAY 59 BATHROOM 60 PARKING AREA/DRIVEWAY 61 KITCHEN/EATING AREA 62 STAIRWAY 63 OUTDOOR PLAY AREA 64 INSIDE PLAY AREA/LARGE MUSCLE AREA 79 CHANGING TABLE XX NO VALUE

Column(s) of "DM_INCIDENT_RESULT" Table		
Name	Datatype	Comment

Column(s) of "DM_INCIDENT_RESULT" Table		
Name	Datatype	Comment
RESULT_CD	CHAR(2)	Denotes the result(s) of the incident for the incident/injury on the associated segment. Values: G1 G2 G3 G4 46 47 48 49 50 51 52 53 54 55
RESULT_DESC	CHAR(75)	Description of the result of the incident. Values are from TRST table: CODE DESC G1 BODY PART ELEVATED G2 PRESSURE APPLIED G3 RESTED ON COT G4 CONTACTED CHILDRENS PROTECTIVE SERVICES 46 RETURNED TO NORMAL ACTIVITY 47 WASHED/SOAP 48 ICE 49 HUG/PAT 50 BANDAGE 51 CONTACTED POISON CONTROL 52 SENT HOME EARLY/PICKED UP EARLY 53 EMERGENCY SERVICES CALLED 54 EMERGENCY SERVICES TRANSPORTED CHILD 55 REFERRED FOR FURTHER MEDICAL CARE XX NO VALUE

Column(s) of "DM_INCIDENT_TYPE" Table		
Name	Datatype	Comment
INCIDENT_CD	CHAR(2)	Code indicates the type of the incident.
INCIDENT_DESC	CHAR(75)	Incident desc From TIIO: C1 BLOOD/BRUISE ON CHLD C2 N/A-INJURY/ILLNESS C3 FALL TO SURFACE C4 MEDICATION ERROR 27 FALL-WALK/RUN/TRIP 28 FIGHTING 29 COLLISION W/OBJECT 30 COLLISION W/PERSON 37 DEATH 39 INTRUDER 40 CHILD RAN AWAY 41 MISSING CHILD 42 CHILD UNATTENDED 44 CORPORAL PUNISHMENT 95 VEHICLE ACCIDENT 96 INAPP TOUCH/SEX PLAY 97 ANTHR ADULT FND CHLD 98 WEAPON FOUND 99 BABY FED WRONG BTL XX NO VALUE

Column(s) of "DM_INCOME_LEVEL" Table		
Name	Datatype	Comment
POVERTY_RNGE_CD	CHAR(2)	A code associated with the percent of poverty of the family.
POVERTY_HIGH	SMALLINT	The high value of the poverty range.
POVERTY_LOW	SMALLINT	The low value of the poverty range.
POVERTY_RNGE_DESC	CHAR(20)	The poverty range description.

Column(s) of "DM_INJURY_TYPE" Table		
Name	Datatype	Comment

Column(s) of "DM_INJURY_TYPE" Table

Name	Datatype	Comment
INJURY_CD	CHAR(2)	Code indicates the injury.
INJURY_DESC	CHAR(75)	Description of injury: From TINJ: 19 SCRAPE/SCRATCH 20 BUMP/BRUISE 21 BURN 22 CUT 23 DIFFICULTY BREATHING 24 BITE-HUMAN 25 BITE/STING-ANML/INS 26 SOMETHING IN EYE 31 SUNBURN 33 CHOKING 34 POISONING 35 NOSEBLEED 86 BLOW TO HEAD 87 BROKEN BONE 88 TOOTH(CHP/KO'D/LOOSE 89 BIT TONGUE/CHEEK/LIP 90 PUNCTURE WOUND 91 STUBBED FINGER/TOE 92 SWELLING/REDNESS 93 OBJ INSRT IN BODY PT 94 N/A-INCIDENT/ILLNESS

Column(s) of "DM_INSPECTION_NOTICE" Table

Name	Datatype	Comment
INSP_NOTICE_CD	CHAR(2)	Indicates the type of inspection notification given to the center on the associated segment. Values are: A - Announced U - Unannounced X - NO VALUE
INSP_NOTICE_DESC	CHAR(30)	The description of the type of inspection notification given to the center on the associated segment. Values are (TCIN): A - Announced U - Unannounced

Column(s) of "DM_INSPECTION_SCOPE" Table

Name	Datatype	Comment
INSP_SCOPE_CD	CHAR(2)	Indicates the scope of the inspection performed at the center on the associated segment. Values are (TCIS ref tbl): A: FULL B: PARTIAL C: FOCUSED D: FOC - FULL E: CPL / FULL F: TELEPHONE X: NO VALUE
INSP_SCOPE_DESC	CHAR(30)	Describes the scope of the inspection performed at the center on the associated segment. Values are (TCIS ref tbl): A: FULL B: PARTIAL C: FOCUSED D: FOC - FULL

Column(s) of "DM_INSPECTION_SCOPE" Table		
Name	Datatype	Comment
		E: CPL / FULL F: TELEPHONE

Column(s) of "DM_INSPECTION_TYPE" Table		
Name	Datatype	Comment
INSP_TYP_CD	CHAR(2)	Indicates the type of inspection performed at the center on the associated segment. Values are (TCIT ref table): A: PRE-LICENSURE B: PROVISIONAL C: 1ST YEAR D: 2ND YEAR E: RENEWAL F: FOLLOW-UP G: COMPLAINT H: AMENDMENT I: CAMP MONITOR X: NO INSPECTION
INSP_TYP_DESC	CHAR(30)	The description of inspection performed at the center on the associated segment. Values are (TCIT ref table): A: PRE-LICENSURE B: PROVISIONAL C: 1ST YEAR D: 2ND YEAR E: RENEWAL F: FOLLOW-UP G: COMPLAINT H: AMENDMENT I: CAMP MONITOR X: NO VALUE

Column(s) of "DM_INVOICE_STAT" Table		
Name	Datatype	Comment
INVC_STAT_CD	CHAR(2)	Current status of the invoice (Open, Completed, Approved, Paid, Terminated, or Waiting for Payment) O, P, C, A, T, W, R, X
INVC_STAT_DESC	CHAR(45)	O : Open, C: Completed, A: Approved, P: Paid, T: Terminated, or W: Waiting for Payment R: Reissue X: Epic

Column(s) of "DM_INVSTGTRY_PLAN" Table		
Name	Datatype	Comment
INVSTGTRY_PLN_CD	CHAR(1)	The method used to investigate the complaint.
INVSTGTRY_PLN_DESC	CHAR(30)	Description of the method used to investigate the complaint. Values 1UNANNOUNCED INSP 2TELEPHONE CONTACT 3ANNOUNCED INSP 4OTHER 5JOINT/PCSA

Column(s) of "DM_JOBPOS" Table		
Name	Datatype	Comment
WRKR_JOBPOS_CD	CHAR(1)	One-character code that indicates the specialist's job position in the child care system. L S O
WRKR_JOBPOS_DESC	CHAR(30)	Description that indicates the specialist's job position in the child care system. A-SUPERVISORY SUPPORT STAFF B-BUREAU STAFF F-FISCAL STAFF L-LICENSING SPECIALIST M-SYSTEM MAINTENANCE S-SUPERVISOR U-SECURITY ADMINISTRATOR O-OTHER

Column(s) of "DM_MED_PLAN" Table		
Name	Datatype	Comment
MED_PLAN_IND	CHAR(1)	Indicates whether a medical/physical care plan is on file at the center where the incident/injury occurred.
MED_PLAN_DESC	CHAR(30)	Indicates whether a medical/physical care plan is on file at the center where the incident/injury occurred on the associated segment. Values are: N no plan on file Y plan on file X NO VALUE

Column(s) of "DM_MONTH" Table		
Name	Datatype	Comment
YEAR_MTH_ID	CHAR(6)	% AttDef
YEAR_MTH_NM	CHAR(15)	Text description for Year Month ID. e.g. Jan, 1999
CURR_MTH_IND	CHAR(1)	An indicator to identify current month. This will be set by the ETL process at the end of each extract. e.g. Y, N
CALNDR_YEAR_ID	CHAR(4)	% AttDef
ST_FSCL_YEAR_ID	CHAR(4)	% AttDef
FED_FSCL_YEAR_ID	CHAR(4)	
CALNDR_QRTR_ID	CHAR(5)	
ST_FSCL_QTR_ID	CHAR(5)	
FED_FSCL_QTR_ID	CHAR(5)	
CALNDR_QTR_NM	CHARACTER(10)	The text, FIRST,SECOND,THIRD, OR FOURTH describng the quarter
ST_FSCL_QRTR_NM	CHARACTER(10)	The text, FIRST,SECOND,THIRD, OR FOURTH describng the quarter
FED_FSCL_QTR_NM	CHAR(10)	The text, FIRST,SECOND,THIRD, OR FOURTH describng the quarter
PART_NBR	SMALLINT	
PHRM_INV_PART_NBR	SMALLINT	
PHRM_CLM_PART_NBR	SMALLINT	
Child_Care_Part_Number	SMALLINT	
FIRST_DY_MTH	DATE	
CHLD_SPT_PART_NBR	SMALLINT	
LAST_DY_MTH	DATE	
CREATED_YR_MTH_ID	CHAR(6)	
CREATED_YR_MTH_NM	CHAR(15)	

Column(s) of "DM_MONTH" Table		
Name	Datatype	Comment
PHRM_INV_CMTH_IND	CHAR(1)	
PHRM_CLM_CMTH_IND	CHAR(1)	
CHLD_CR_CMTH_IND	CHAR(1)	
CHLD_SPT_CMTH_IND	CHAR(1)	
WRK_ACTV_PART_NBR	INTEGER	
WRK_ACTV_CMTH_IND	CHAR(1)	
EMPLYMT_CMTH_IND	CHAR(1)	
ELIG_CMTH_IND	CHAR(1)	
ST_FSCL_YEAR	INTEGER	
FED_FSCL_YEAR	INTEGER	
CALNDR_YEAR	INTEGER	
APPLICN_PART_NBR	INTEGER	
APPLICN_CMTH_IND	CHAR(1)	

Column(s) of "DM_PARENT_DEMO_PRF" Table		
Name	Datatype	Comment
PARENT_DEMO_PRF_ID	INTEGER	
FAMILY_SIZE	SMALLINT	For the household applying for child care services, the number of caretaker parents plus all children related by blood or law under the age of 18.
PERCENT_POVERTY	SMALLINT	Calculated field in the source system.
POVERTY_RNGE_CD	CHAR(2)	A code to identify the range of poverty.
EARNED_AMT	DECIMAL(5,0)	Total monthly non-exempt income including self-employment earnings for the household applying for child care services. Includes OWF if OWF is the only source of income
ADC_OWF_AMT	DECIMAL(5,0)	Total monthly TANF (Temporary Aid to Needy Families) amount for the household applying for child care services.
FOOD_STAMPS_AMT	DECIMAL(5,0)	Total monthly Food Stamps amount for the household applying for child care services. This amount is exempt from the earned income total.
HOUSE_ASST_AMT	DECIMAL(5,0)	Total monthly cash housing subsidy amount for the household applying for child care services.
CHILD_SUPPRT_AMT	DECIMAL(5,0)	Total monthly court-ordered child support amount for the household applying for child care services.
OTHER_UNEARN_AMT	DECIMAL(5,0)	Total monthly unearned income amount not specified by the categories listed for the household applying for child care services.
SOC_SEC_AMT	DECIMAL(5,0)	Total monthly social security benefits amount for the household applying for child care services.
UNEMPLYMNT_AMT	DECIMAL(5,0)	Total monthly unemployment compensation benefits amount for the household applying for child care services.
WORKERS_COMP_AMT	DECIMAL(5,0)	Total monthly worker's compensation benefits amount for the household applying for child care services.
MONTH_INCOME_AMT	DECIMAL(5,0)	Total monthly non-exempt income of the household applying for child care services as entered by the worker as "adjusted gross" on the Family Income Screen (NCC029)

Column(s) of "DM_PAYMENT_SRC" Table		
Name	Datatype	Comment
PYMT_SRC_CD	CHAR(3)	Child Care program code for which the child is authorized on the associated segment. Values are: 312 - Job Child Care OWF 313 - Transitional Child Care 314 - Leap Child Care 320 - GA/FSET Child Care 321 - Employment and Training Child Care 322 - Education and Training Child Care 323 - Protective Child Care 324 - Special Needs Child Care 325 - Homeless Child Care 326 - JOBS/Leap/Headstart 327 - Transitional/Headstart 329 - Non-guaranteed/Headstart 342 - Continuous Headstart 899 - Other (Non-reimbursable)
PYMT_SRC_DESC	CHAR(45)	The payment source text description. The values are: 312 - Job Child Care OWF 313 - Transitional Child Care 314 - Leap Child Care 320 - GA/FSET Child Care 321 - Employment and Training Child Care 322 - Education and Training Child Care 323 - Protective Child Care 324 - Special Needs Child Care 325 - Homeless Child Care 326 - JOBS/Leap/Headstart 327 - Transitional/Headstart 329 - Non-guaranteed/Headstart 342 - Continuous Headstart 899 - Other (Non-reimbursable)

Column(s) of "DM_PERSON" Table		
Name	Datatype	Comment
PERSON_KEY	INTEGER	% AttDef
ETHNICITY_CD	CHAR(2)	A code identifying a person's ethnicity. (e.g. Hispanic or non-Hispanic)
FULL_NAME	CHAR(60)	% AttDef
SSN_NBR	INTEGER	A nine digit number issued to a Person by the Social Security Administration.
CRISE_RCPNT_ID	CHAR(12)	A twelve digit number that is assigned by the CRIS-E system to each recipient for unique identification purposes.
MCAID_RCPNT_ID	CHAR(12)	The unique identifier assigned by MMIS to identify an individual. This number will generally match the CRIS-E Recipient Number however if the person was known to MMIS prior to CRIS-E (e.g. converted from CRIS) or not known to CRIS-E, a different unique identifying number would be used.
BIRTH_DTE	DATE	Birth date of a person.
DEATH_DTE	DATE	Death date of a person
GNDR_TYP_CD	CHAR(2)	A code that identifies the sex of a person (e.g M, F, U)
CTZN_CNTRY_CD	SMALLINT	A three digit code identifying a specific political state or nation.(e.g. 165, 169 etc.) The code which is found on table TCOU, is used by ODJFS to track the participant's citizenship. Note that code 999 = United States of America.

Column(s) of "DM_PERSON" Table		
Name	Datatype	Comment
ORGN_CNTRY_CD	SMALLINT	A unique code that identifies a specific political state or nation used by ODJFS to track the participant's original country of citizenship.
RACE_TYP_CD	CHAR(2)	A code identifying a person's race. (e.g. A, B, W etc.)
First_Name	CHAR(20)	First name of the person
Last_Name	CHAR(30)	% AttDef
Middle_Name	CHAR(20)	Middle name of the person
ATD_SCHL_DSTRCT_CD	CHAR(4)	School District the individual currently attends, if applicable.
PRIM_LANG_CD	CHAR(3)	A code identifying the language with which a person is most comfortable and likely to understand. A person's mother tongue.

Column(s) of "DM_PRIORITY_RANK" Table		
Name	Datatype	Comment
PRIORITY_RANK_CD	CHAR(2)	Family priority ranking based on the family eligibility determination date and type of pay source for the authorized provider/child on the associated segment. Values are: A - First Priority; all families eligible before 6/29/93 0 - Second Priority, all guaranteed families eligible after 6/29/93 1 - Third Priority, all non-guaranteed families eligible after 6/29/93
PRIORITY_RNK_DESC	CHAR(60)	The priority rank text description. The values are: First Priority; all families eligible before 6/29/93 Second Priority, all guaranteed families eligible after 6/29/93 Third Priority, all non-guaranteed families eligible after 6/29/93

Column(s) of "DM_PROVIDER" Table		
Name	Datatype	Comment
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
CENTER_NAME	CHAR(52)	The name of the licensed center.
ADDRESS_LN1	CHAR(30)	First address line for the physical location of the center.
ADDRESS_LN2	CHAR(30)	Second address line for the physical location of the center.
CITY_NM	CHAR(15)	City for the physical location of the center.
ADDR_STATE_CD	CHAR(2)	State for the physical location of the center.
ADDR_ZIP_CD	INTEGER	Zip Code for the physical location of the center.
PHONE_NBR	CHAR(10)	Center phone number.
OWNER_NAME	CHAR(65)	Owner of the center
FULLTM_IND	CHAR(1)	Facility provides full time service.
PARTTM_IND	CHAR(1)	Facility provides part time service.
HEADSTR_IND	CHAR(1)	Facility has head start service.
SCHLAGE_IND	CHAR(1)	Facility has school age service.
FULLYR_IND	CHAR(1)	Facility has full year service.
PARTYR_IND	CHAR(1)	Facility has part year service.

Column(s) of "DM_PROVIDER_GROUP" Table		
Name	Datatype	Comment
PROVIDER_GRP_CD	CHAR(2)	
PROVIDER_GRP_DESC	CHAR(20)	

Column(s) of "DM_PROVIDER_TYPE" Table		
Name	Datatype	Comment
PROVIDER_TYP_CD	CHAR(2)	Code associated with the type of the provider on the associated segment. Values are: Licensed Centers: A - Type A Homes C - Child Care Center D - Day Camp H - Head Start S - School Affiliated Provider Certified Providers: B - Type B Homes I - In-Home Aide L - Limited Provider There are 2 type L providers: Limited AI Has agency type = Limited PPI Has agency type =
PROVIDER_GRP_CD	CHAR(2)	
PROVIDER_TYP_DESC	CHAR(25)	The provider type text description. The values are: Type A Homes Child Care Center Day Camp Head Start School Affiliated Provider Type B Homes In-Home Aide Limited Provider

Column(s) of "DM_QUALIFICATION" Table		
Name	Datatype	Comment
QUAL_ID	INTEGER	Qualification Key
QUAL_DESC	CHAR(50)	The job position of the teacher within than particular classroom.
PREREQUISIT_DESC	CHAR(50)	The prerequisite position of the teacher within that classroom.

Column(s) of "DM_QUALITY_RTNG" Table		
Name	Datatype	Comment
SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
SUTQ_DESC	CHAR(30)	Describes the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE

Column(s) of "DM_QUARTER" Table		
Name	Datatype	Comment
CALNDR_QRTR_ID	CHAR(5)	
CALNDR_YEAR_ID	CHAR(4)	
EMPLY_CQTR_IND	CHAR(1)	
ST_FSCL_YEAR_ID	CHAR(4)	
FED_FSCL_YEAR_ID	CHAR(4)	
ST_FSCL_QTR_ID	CHAR(5)	
FED_FSCL_QTR_ID	CHAR(5)	
CALNDR_QTR_NM	CHARACTER(10)	The text, FIRST,SECOND,THIRD, OR FOURTH describing the quarter
ST_FSCL_QRTR_NM	CHARACTER(10)	The text, FIRST,SECOND,THIRD, OR FOURTH describing the quarter
FED_FSCL_QTR_NM	CHAR(10)	The text, FIRST,SECOND,THIRD, OR FOURTH describing the quarter
EMPLY_QPART_NBR	SMALLINT	
EMPLY_CQTR_PARTNBR	SMALLINT	
FIRST_DY_QTR	DATE	
LAST_DY_QTR	DATE	

Column(s) of "DM_RACE" Table		
Name	Datatype	Comment
RACE_TYP_CD	CHAR(2)	A code identifying a person's race. (e.g. A, B, W etc.)
RACE_DESC	CHAR(120)	This field contains an expanded text description of Race. (Asian or Pacific Islander, African-American, White etc.).

Column(s) of "DM_RATING_STATUS" Table		
Name	Datatype	Comment
RTNG_STAT_ID	SMALLINT	Rating Status Code (Surrogate Key: RTNG_STAT_DESC and STAT_RSN_DESC) VALUES: RTNG_STAT_ID,RTNG_STAT_DESC,STAT_RSN_DESC 1, 0, Supervision 2, 0, NULL 3, 0-Getting Ready, NULL 4, 1, NULL 5, 2, NULL 6, 3, NULL 7, Closed, Relocated 8, Closed, ELI Cutoff 9, Closed, Financial Reasons 10, Closed, NULL 11, Closed, ODJFS did not Renew 12, Closed, Unknown 13, Closed, Other 14, Closed, Program let Rating Expire 15, ELES, NULL 16, ELSR, NULL 17, ES, NULL 18, NEES, NULL 19, NESR, NULL 20, Removed, Enviro. Hazards 21, Removed, NULL 22, Removed, Supervision 23, Removed, Admin. Negligence 24, Suspended, Supervision 25, Suspended, Admin. Negligence 26, Suspended, Enviro. Hazards 27, Suspended, NULL 28, Warning, NULL

Column(s) of "DM_RATING_STATUS" Table

Name	Datatype	Comment
		29, Warning, Enviro. Hazards 30, Warning, Supervision 31, Warning, Admin. Negligence 32, Withdrew, NULL
RTNG_STAT_DESC	CHAR(30)	Rating status description. Values: 0 0-Getting Ready 1 2 3 Closed ELES ELSR ES NEES NESR Removed Suspended Warning Withdrew
STAT_RSN_DESC	CHAR(25)	Rating reson description Values: RTNG_STAT_ID,RTNG_STAT_DESC,STAT_RSN_DESC 1, 0, Supervision 2, 0, NULL 3, 0-Getting Ready, NULL 4, 1, NULL 5, 2, NULL 6, 3, NULL 7, Closed, Relocated 8, Closed, ELI Cutoff 9, Closed, Financial Reasons 10, Closed, NULL 11, Closed, ODJFS did not Renew 12, Closed, Unknown 13, Closed, Other 14, Closed, Program let Rating Expire 15, ELES, NULL 16, ELSR, NULL 17, ES, NULL 18, NEES, NULL 19, NESR, NULL 20, Removed, Enviro. Hazards 21, Removed, NULL 22, Removed, Supervision 23, Removed, Admin. Negligence 24, Suspended, Supervision 25, Suspended, Admin. Negligence 26, Suspended, Enviro. Hazards 27, Suspended, NULL 28, Warning, NULL 29, Warning, Enviro. Hazards 30, Warning, Supervision 31, Warning, Admin. Negligence 32, Withdrew, NULL
STAT_ID	SMALLINT	For future use.

Column(s) of "DM_REFERRAL" Table

Name	Datatype	Comment
REFERRAL_ID	SMALLINT	Indicates who the Infant/Toddlers in the program are referred to when needed.
REFERRAL_DESC	CHAR(50)	Description of the referral.

Column(s) of "DM_REFERRAL_SPECIALIST" Table

Name	Datatype	Comment
RFRL_SPECLST_ID	INTEGER	SUTQ Referral Specialist Id (Surrogate Key).
RFRL_SPECLST_PHN	CHAR(15)	SUTQ Referral Specialist's phone number.
RFRL_SPECLST_NM	CHAR(50)	SUTQ Referral Specialist's first and last name.
RFRL_SPECLST_EMAIL	CHAR(70)	SUTQ Referral Specialist's email address.
RFRL_AGY_NM	CHAR(50)	Referral Specialist's Agency.

Column(s) of "DM_RISK_STAT" Table

Name	Datatype	Comment
RISK_CD	CHAR(1)	Code indicates the non compliance is serious risk or not.
RISK_DESC	CHAR(30)	Description of the severity: 0 - NO COMPLIANCE VIOLATIONS 1 - SERIOUS COMPLIANCE VIOLATIONS

Column(s) of "DM_ROOM" Table

Name	Datatype	Comment
ROOM_ID	INTEGER	The unique ID for each of the classrooms within a program
ROOM_AGE_GRP	CHAR(50)	The age group served by the classroom
TEACHER_CNT	SMALLINT	The number of teachers in that classroom
CHLD_CNT	SMALLINT	The number of Children in that classroom
ROOM_STEP_LVL	SMALLINT	The Step/Rating Level for that Classroom. The Rating Level for the Ratio Benchmark is the minimum of the classroom's rating.

Column(s) of "DM_RULE" Table

Name	Datatype	Comment
RULE_EFF_DTE	DATE	Rule Effective Date
CHAPTER_CD	CHAR()	Code indicates the version of the Ohio Revised Code (ORC) or Ohio Administrative Code (OAC). Values are: 212 213 214 218
RULE_CD	CHAR(3)	Field indicates a new alleged rule or additional out of compliance associated with the complaint.
RULE_PART_CD	CHAR(1)	Rule part number.
RULE_DESC	CHAR(75)	Rule Description.

Column(s) of "DM_RULE_CHAPTER" Table

Name	Datatype	Comment
CHAPTER_CD	CHAR(3)	Code indicates the version of the Ohio Revised Code (ORC) or Ohio Administrative Code (OAC).

Column(s) of "DM_RULE_CHAPTER" Table		
Name	Datatype	Comment
		Values are: 212 213 214 218
CHAPTER_DESC	CHAR(30)	Description of the ORC or OAC codes. Values are: 212LICENSED CHILD CARE CENTERS 213LICENSED TYPE A FAMILY CARE HOMES 214CERTIFIED TYPE B FAMILY CARE HOMES 218CHILD DAY CAMPS

Column(s) of "DM_SCREENING_TOOL" Table		
Name	Datatype	Comment
SCRNG_ID	SMALLINT	An Id identifies the screening tool used by the Program for Infant and Toddlers and Preschool age.
SCRNG_DESC	CHAR(50)	Definition of the screening tools used by the Program for Infant and Toddlers and Preschool age. Values: 1. NO VALUE 2. LAP-D 3. ASQ 4. DECA 5. Denver 6. Other

Column(s) of "DM_SRVC_CONTR" Table		
Name	Datatype	Comment
SRVC_CONTR_CD	CHAR(1)	Indicates the type of the service contract (both certified and licensed center). Table-driven values derived from superset of existing license types and certified provider types.
SRVC_CONTR_DESC	CHAR(30)	Type Description A TYPE A HOME B TYPE B HOME C CHILD CARE CENTER D DAY CAMPS E JURISDICTIONAL CASES H HEAD START I IN-HOME AIDE L LIMITED PROVIDER S SCHOOL AFFILIATED 1 FULL-TIME 2 PART-TIME 3 COMBINATION 4 TYPE A HOME 5 SCHOOL AGE 6 HEAD START 7 DAY CAMP

Column(s) of "DM_SULS_WORKER" Table		
Name	Datatype	Comment
SULS_WRKR_ID	SMALLINT	Step Up to Licensing Specialist Id. (Surrogate Key)
SULS_NM	CHAR(50)	Step Up to Licensing Specialist's first and last name.

Column(s) of "DM_SULS_WORKER" Table		
Name	Datatype	Comment
SULS_PHN	CHAR(15)	Step Up to Licensing Specialist's phone number.
SULS_EMAIL	CHAR(70)	Step Up to Licensing Specialist's email address.

Column(s) of "DM_SUTQ_ADMIN" Table		
Name	Datatype	Comment
ADMIN_ID	INTEGER	SUTQ Administrator Id. (Surrogate Key)
ADMIN_PHN	CHAR(15)	SUTQ Administrator's phone number.
ADMIN_NM	CHAR(50)	SUTQ Administrator's first and last name.
ADMIN_EMAIL	CHAR(70)	SUTQ Administrator's email address.

Column(s) of "DM_SUTQAPPL_STATUS" Table		
Name	Datatype	Comment
APPL_STAT_ID	SMALLINT	Status Id indicating the state of the Application.
APPL_STAT_DESC	CHAR(30)	Description of the application status: Values: <ol style="list-style-type: none"> 1. Visit Deferred 2. Visit Scheduled 3. SULS Assigned 4. Onsite Visit Completed 5. Ready for Review 6. Eligibility Determination 7. Application Received 8. Application Completed 9. Application Modified/Error 10. Application Withdrawn 11. Registry Incomplete

Column(s) of "DM_SUTQAPPL_TYPE" Table		
Name	Datatype	Comment
APPL_TYP	CHAR(1)	Application Types. Values: I, R, C
APPL_TYP_DESC	CHAR(25)	Application Type description. Values: Initials (I), Renewals (R), and Change (C)

Column(s) of "DM_SUTQLETTER" Table		
Name	Datatype	Comment
LETTER_TYP_ID	INTEGER	Unique Letter Type Id.
LETTER_NM	CHAR(35)	The name of the letters generated for the program, including eligibility letters.

Column(s) of "DM_TEACHER" Table		
Name	Datatype	Comment
TEACHER_ID	INTEGER	Unique ID for Each Teacher entered into the Visit Tool. It is not universal. So a teacher entered in one tool is only unique only for that Visit.
TEACHER_NM	CHAR(50)	tEACHER nAME
TEACHER_STEP_LVL	SMALLINT	Step Level Associated with that Teacher.
EDUC_QUAL	CHAR(50)	Required Educational Qualifications of the Teacher for the visit.

Column(s) of "DM_TME_SPAN" Table		
Name	Datatype	Comment

Column(s) of "DM_TME_SPAN" Table		
Name	Datatype	Comment
TME_SPAN_CD	CHAR(2)	Denotes the number of hours of care per week associated with the child for the incident/injury .
TME_SPAN_DESC	CHAR(30)	Description of the time span code.

Column(s) of "FACT_AWARD_PROGRAM" Table		
Name	Datatype	Comment
AWARD_PGM_KEY	INTEGER	Surrogate Key: Provider_id, Award_Year_Month_Id,Sutq_Cd
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
YEAR_MTH_ID	CHAR(6)	Year Month Id of the rating awarded to the program.
SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
CPCT_TOT	SMALLINT	Total capacity of the center reported.
ENRLMNT_TOT	SMALLINT	Total enrollment of the center reported.
SUBDZ_TOT	SMALLINT	Total number of subsidized children of the center reported.
SUBDZ_ELI_TOT	SMALLINT	Total (early learning initiative) ELI enrollment number.
AWARD_AMT	DECIMAL(19,4)	Total quality achievement award amount.
AWARD_RATE_DTE	TIMESTAMP	Date of award rating.
AWARD_SUBMT_DTE	TIMESTAMP	The date the Quality Achievement Award to be submitted to OCCRRA
CENTER_TYP_CD	CHAR(1)	Center Types: 1,4,7
CNTY_NUM	SMALLINT	County number.
RCD_CNT	SMALLINT	Default to 1. Used by Cognos.
LOAD_TS	TIMESTAMP	The timestamp in which the table is loaded.
PART_NBR	SMALLINT	The DB2 partition number.

Column(s) of "FACT_CHILD" Table		
Name	Datatype	Comment
CHILD_KEY	INTEGER	Surrogate Key
PARENT_KEY	INTEGER	Surrogate Key that is generated based on a new Parent Key and Service Date.
PROVIDER_ID	CHAR(15)	The Provider ID for which the child is authorized.
CNTY_CD	CHAR(2)	A unique code that identifies a specific county. (e.g. 87, 88 etc.)
SERVICE_DTE	DATE	The month in which the child received the services.
APPROVAL_DTE	DATE	The date in which the child receives approval.
PAYMENT_DTE	DATE	Denotes the month CDJFS issued payment to the provider.
PYMT_SRC_CD	CHAR(3)	Child Care program code for which the child is authorized on the associated segment. Values are: 312 - Job Child Care OWF 313 - Transitional Child Care 314 - Leap Child Care 320 - GA/FSET Child Care

Column(s) of "FACT_CHILD" Table		
Name	Datatype	Comment
		321 - Employment and Training Child Care 322 - Education and Training Child Care 323 - Protective Child Care 324 - Special Needs Child Care 325 - Homeless Child Care 326 - JOBS/Leap/Headstart 327 - Transitional/Headstart 329 - Non-guaranteed/Headstart 342 - Continuous Headstart 899 - Other (Non-reimbursable)
FT_PT_CD	CHAR(1)	Indicates the type of roster record or intended action. Values are: Roster Record Types F - Full-time child care roster P - Part-time child care roster O - Overpayment child care roster U - Underpayment child care roster A - Advance child care roster D - Deposit child care roster Intended Action C - Copy existing roster line R - Remove existing roster line
PARENT_DEMO_PRF_ID	INTEGER	Surrogate Key
PART_NBR	SMALLINT	The DB2 partition number.
YEAR_MTH_ID	CHAR(6)	The year and month the data is partitioned on.
CREATED_YR_MTH_ID	CHAR(6)	The year and month in which the table is loaded.
PROVIDER_TYP_CD	CHAR(2)	Code associated with the type of the provider on the associated segment. Values are: Licensed Centers: A - Type A Homes C - Child Care Center D - Day Camp H - Head Start S - School Affiliated Provider Certified Providers: B - Type B Homes I - In-Home Aide L - Limited Provider
AGE_MONTHS	SMALLINT	The age of the individual in months at the time of service.
ELIG_SRC_CD	CHAR(2)	Two-digit status code which correlates to the pay source for the authorized provider/child on the associated segment. Values are: 01 - SSI Child care services are provided to an SSI recipient because of SSI eligibility. Valid pay sources are: 320, 899 02 - MEDICAID Child care services are provided to a Medicaid recipient because of Medicaid eligibility. Valid pay sources are: 320, 899 04 - OWF Ohio Works First . Valid pay sources are: 312, 314, 320, 322, 324, 326, 342, 899 06 - INCOME ELIGIBLE Child care services are provided free of charge because the family's only sources of non-exempt gross monthly income is OWF.

Column(s) of "FACT_CHILD" Table		
Name	Datatype	Comment
		<p>Valid pay sources are: 320, 321, 322, 324, 342, 899</p> <p>07 - WITHOUT REGARD TO INCOME Protective child care services are provided free of charge without regard to family income. Valid pay sources are: 323, 325, 329, 899</p> <p>08 - FEE Non-guaranteed child care services are provided for current recipients or new applicants whose adjusted monthly income is at or below 150% (as of 6/9/03) of the current federal poverty level. Also included are families receiving transitional child care services. Families are required to pay a fee for child care services. Valid pay sources are: 312, 313, 314, 320, 321, 322, 323, 324, 325, 326, 327, 329, 342, 899</p> <p>10 - OTHER Child care services are provided for an individual who does not meet any of the preceding eligibility status codes. Valid pay sources are: 899</p>
PRIORITY_RANK_CD	CHAR(2)	<p>Family priority ranking based on the family eligibility determination date and type of pay source for the authorized provider/child on the associated segment. Values are:</p> <p>A - First Priority; all families eligible before 6/29/93 0 - Second Priority, all guaranteed families eligible after 6/29/93 1 - Third Priority, all non-guaranteed families eligible after 6/29/93</p>
AT_RISK_IND	CHAR(1)	<p>Indicates whether the authorized child's family is eligible for At Risk child care benefits, currently applicable for pay sources 321, 324, and 329. Values are:</p> <p>N - not at risk Y - at risk.</p>
REL_NONREL_IND	CHAR(1)	<p>Indicates whether the authorized provider is a relative of the child on the associated segment. Values are:</p> <p>N - non-relative R - relative</p>
NBR_DAYS_SERVICE	SMALLINT	The total number of days the child received child care services based on the entered service month.
NBR_HRS_SERVICE	SMALLINT	The total number of hours the child received child care services based on the entered service month.
NBR_DAYS_RPT_ABS	DECIMAL(3,1)	The total number of days the child was absent from child care services based on the entered service month.. The CDFJS cannot pay more than 10 absent days per six-month period (Jan - June, July - Dec.). Once the child's limit of 10 has been reached, additional absences by the child are to be coded with pay source 899 (non-reimbursable) to facility accurate absentee tracking. (As of 1/1/2002)
NBR_HRS_RPT_ABS	SMALLINT	The total number of hours the child was absent from child care services based on the

Column(s) of "FACT_CHILD" Table

Name	Datatype	Comment
		entered service month.. The CDFJS cannot pay more than 10 absent days per six-month period (Jan - June, July - Dec.). Once the child's limit of 10 has been reached, additional absences by the child are to be coded with pay source 899 (non-reimbursable) to facility accurate absentee tracking. (As of 1/1/2002)
CLIENT_PAID_AMT	DECIMAL(9,2)	The amount of money paid by the family (commonly referred to as the family fee or family copay) for services.
COUNTY_PYMT_AMT	DECIMAL(9,2)	The amount of money paid by the county for services.
AUTH_BGN_DTE	DATE	The beginning date of the authorization period for the child.
AUTH_END_DTE	DATE	The ending date of the authorization period for the child.
ISSUANCE_NBR	INTEGER	Generated number associated with the payment sent to OAKS. In HD504 (FLAT-FILE-OUT-NBR-ISSUED)
ISSUANCE_DTE	DATE	Date in which the issuance was sent. full date - in HDE504 FLAT-FILE-OUT-DTE-ISSUED
INVOICE_NBR	DECIMAL(10)	Invoice Number. INVC10-NBR-INVC
INVC_STAT_CD	CHAR(2)	Current status of the invoice (Open, Completed, Approved, Paid, Terminated, or Waiting for Payment)
BASE_AMT	DECIMAL(9,2)	Base amount cost of services before co-pay is applied.
WAIVER_AMT	DECIMAL(9,2)	Amount added when child has a Special Needs Waiver. {PT = Ceil rate} {FT = Ceil rate}
SPCL_NEED_AMT	DECIMAL(9,2)	Amount added when child has a Special Needs indicator.
NONTRAD_AMT	DECIMAL(9,2)	Amount added when child has received care outside defined hours.
SUTQ_ACCR_AMT	DECIMAL(9,2)	Amount added when child has received care at a provider that is either accredited or has any SUTQ rating.
STAR_TWO_AMT	DECIMAL(9,2)	Amount added when child has received care at a provider that has a '2' SUTQ rating.
STAR_THREE_AMT	DECIMAL(9,2)	Amount added when child has received care at a provider that has a '3' SUTQ rating.
REGSTRN_FEE_AMT	DECIMAL(9,2)	Amount of registration fees entered via Provider Website.
TRANSPORTN_FEE_AMT	DECIMAL(9,2)	Amount of Transportation fees entered via Provider Website.
ACTIVITY_FEE_AMT	DECIMAL(9,2)	Amount of activity fees entered via Provider Website

Column(s) of "FACT_CHLDCR_INCDNT" Table

Name	Datatype	Comment
INCIDENT_KEY	INTEGER	Surrogate Key
YEAR_MTH_ID	CHAR(6)	% AttDef
CNTY_CD	CHAR(2)	Standard ODJFS 2-digit county of the center where there the incident/injury occurred.
PROVIDER_ID	CHAR(15)	Indicates the licensed center or certified provider where the incident/injury occurred.
MED_PLAN_IND	CHAR(1)	Indicates whether a medical/physical care plan is on file at the center where the incident/injury occurred on the associated

Column(s) of "FACT_CHLDCR_INCDNT" Table		
Name	Datatype	Comment
		segment. Values are: N - no plan on file Y - plan on file
SRVC_CONTR_CD	CHAR(1)	Indicates the type of the service contract (both certified and licensed center). Table-driven values derived from superset of existing license types and certified provider types.
INJRY_TYP_CD	CHAR(2)	Code identifies the type of the injury. Values are (TIH ref tbl): D: Incident E: Illness F: Injury X: NO VALUE
AGE_GRP_CD	CHAR(1)	Denotes the assigned age group associated with the child for the incident/injury on the associated segment. Table-driven values are: I - Infant P - Preschool S - School age T - Toddler Y - Young Infant X NO VALUE
GNDR_TYP_CD	CHAR(2)	A code that identifies the sex of a person (e.g M, F, U)
TME_SPAN_CD	CHAR(2)	The number of hours of care per week associated with the child for the incident/injury .

Column(s) of "FACT_EARLY_LEARNING" Table		
Name	Datatype	Comment
ELL_KEY	INTEGER	Early Learning Key.
ELL_SEQ	SMALLINT	Curriculum used by the Program for Infants and Toddlers, Preschoolers and school aged children. Values: 1: Infants and Toddlers 2: Preschool 3: School Aged
VISIT_KEY	INTEGER	Visit Key
YEAR_MTH_ID	CHAR(6)	% AttDef
VISIT_BGN_DTE	DATE	Visit begin date.
CRCLM_ID	SMALLINT	The Curriculum used by the Program to assess children. Values: 1 to 8
SCRNG_ID	SMALLINT	An Id identifies the screening tool used by the Program for Infant and Toddlers and Preschool age.
SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
ASSMNT_ID	SMALLINT	An indicator for the Assessment tool used by the Program to assess children enrolled.

Column(s) of "FACT_EARLY_LEARNING" Table		
Name	Datatype	Comment
CMNCTN_ID	INTEGER	Indicates if the Program uses a communication tool to parents/guardians regarding their infants and toddlers.
REFERRAL_ID	SMALLINT	Indicates who the Infant/Toddlers in the program are referred to when needed.
PART_NBR	SMALLINT	Part Number.
RCD_CNT	SMALLINT	Cognos uses.
LOAD_TS	TIMESTAMP	The timestamp in which the table is loaded.
OTHR_SCRNG_ID	SMALLINT	An Id identifies the screening tool used by the Program for Infant and Toddlers and Preschool age.

Column(s) of "FACT_INCDNT_INJURY" Table		
Name	Datatype	Comment
INCIDENT_KEY	INTEGER	Surrogate Key
INCIDENT_SEQ	SMALLINT	
YEAR_MTH_ID	CHAR(6)	% AttDef
ILLNESS_CD	CHAR(2)	Code indicates the Illness.
INJURY_CD	CHAR(2)	Code indicates the injury.
INCIDENT_CD	CHAR(2)	Code indicates the type of the incident.
RESULT_CD	CHAR(2)	Denotes the result(s) of the incident for the incident/injury on the associated segment. Values: G1 G2 G3 G4 46 47 48 49 50 51 52 53 54 55
BODY_PART_CD	CHAR(3)	Denotes the body part(s) for the incident/injury. Values are from TBOD: E1 E2 66A 66B 66C 66D 66E 66F 66G 67 68 69 70 71A 71B
ACTIVITY_CD	CHAR(3)	Denotes the activity(s) the child was engaged in the incident/injury. Values are: D1 75 76 77 78 80 81 82 83
INCDNT_LOC_CD	CHAR(2)	Denotes the location(s) of the incident. Values are: F1 F2 F3 F4 F5 57 58 59 60 61 62 63 64 79

Column(s) of "FACT_INSP_COMPLAINT" Table		
Name	Datatype	Comment
PROVIDER_ID	CHAR(15)	
CMPLNT_RCVD_DTE	DATE	
CMPLNT_SERIAL_NBR	SMALLINT	
RULE_EFF_DTE	DATE	Rule Effective Date
CHAPTER_CD	CHAR()	Code indicates the version of the Ohio Revised Code (ORC) or Ohio Administrative Code (OAC). Values are: 212 213 214 218
RULE_CD	CHAR(3)	Field indicates a new alleged rule or additional out of compliance associated with the complaint.
RULE_PART_CD	CHAR(1)	Rule part number.
YEAR_MTH_ID	CHAR(6)	
CMPLNCE_STAT_CD	CHAR(1)	The current compliance status of the specific rule.

Column(s) of "FACT_INSP_COMPLAINT" Table		
Name	Datatype	Comment
		Values: A D I O P S U
COMPLNCE_IND	CHAR(1)	
SRVC_CONTR_CD	CHAR(1)	
INVSTGTRY_PLN_CD	CHAR(1)	The method used to investigate the complaint.
SMRY_LETTER_DTE	DATE	
NEGLECT_IND	CHAR(1)	
INJURY_DEATH_IND	CHAR(1)	
PART_NBR	SMALLINT	
CNTY_CD	CHAR(2)	
SMRYLTR_YR_MTH_ID	CHAR(6)	

Column(s) of "FACT_LICENSE_CENTER" Table		
Name	Datatype	Comment
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
YEAR_MTH_ID	CHAR(6)	% AttDef
CNTY_CD	CHAR(2)	County Code
CLOSED_DTE	DATE	Indicates when the center on the associated segment was closed. Closed centers will be accessible through the online system for up to two years following the close date.
CPCT_EFF_FROM_DTE	DATE	Denotes the effective begin date of the subsequent capacity information regarding the center.
CPCT_EFF_THRU_DTE	DATE	Denotes the effective end date of the subsequent capacity information regarding the center.
CPCT_INFANT_NBR	SMALLINT	The infant licensed capacity level for the center.
CPCT_TODLR_NBR	SMALLINT	The toddler licensed capacity level for the center .
CPCT_PRESCHL_NBR	SMALLINT	The preschool capacity level for the center .
CPCT_SCHLAGE_NBR	SMALLINT	The school age capacity level for the center .
ENRLMNT_QTY	SMALLINT	Total enrollment amount.
ENRLMNT_INFANT	SMALLINT	Infant enrollment amount.
ENRLMNT_TODLR	SMALLINT	Toddler enrollment amount.
ENRLMNT_PRESCH	SMALLINT	Pre school enrollment amount.
ENRLMNT_SCHLAGE	SMALLINT	School age enrollment amount.
ENRLMNT_EFF_DTE	DATE	The effective date for the enrollment data.
LICNSE_START_DTE	DATE	The begin date for the licensure period of the center.
LICNSE_END_DTE	DATE	The end date for the licensure period of the center
FIRE_INSP_DTE	DATE	The most recent fire inspection date for the center
LICNSE_RNWLDUE_DTE	DATE	Indicates the date the renewal application is due for the center
APPL_DTE	DATE	The license application was received
SPECLST_CD	CHAR(6)	The assigned licensing specialist for the center
BLDNG_CD	CHAR(1)	The building type of the center.
PART_NBR	SMALLINT	The DB2 partition number.
SRVC_CONTR_CD	CHAR(1)	Indicates the type of the service contract

Column(s) of "FACT_LICENSE_CENTER" Table		
Name	Datatype	Comment
		(both certified and licensed center). Table-driven values derived from superset of existing license types and certified provider types.

Column(s) of "FACT_LICENSE_INSP" Table		
Name	Datatype	Comment
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
INSP_DTE	DATE	Indicates the date the inspection was performed at the center
INSP_TYP_CD	CHAR(2)	Indicates the type of inspection performed at the center on the associated segment. Values are (TCIT ref table): A: PRE-LICENSURE B: PROVISIONAL C: 1ST YEAR D: 2ND YEAR E: RENEWAL F: FOLLOW-UP G: COMPLAINT H: AMENDMENT I: CAMP MONITOR X: NO INSPECTION
YEAR_MTH_ID	CHAR(6)	AttDef
FNDNG_GRP_KEY	INTEGER	Rule Part Surrogate Key: GRP_EFF_DTE, CHAPTER_CD, GRP_RULE_CD, GRP_PART_CD, 1 and 2 : RESERVED FOR N/A - no findings and corrections 1=0001-01-01, 212,000,0, XX 2=0001-01-01, 213,000,0,XX
FNDNG_CD	CHAR(5)	The 5-char licensing rule finding code. An observation made during an inspection that substantiates or not a facilities adherence to the rules outlined in the Ohio Administrative Code.
FNDNG_RNKNG	SMALLINT	The four digit sequence number that represents unique inspection rules.
CNTY_CD	CHAR(2)	County Code.
INSP_SCOPE_CD	CHAR(2)	Indicates the scope of the inspection performed at the center on the associated segment. Values are (TCIS ref tbl): A: FULL B: PARTIAL C: FOCUSED D: FOC - FULL E: CPL / FULL F: TELEPHONE X: NO VALUE
INSP_NOTICE_CD	CHAR(2)	Indicates the type of inspection notification given to the center on the associated segment. Values are: A - Announced U - Unannounced X - NO VALUE
SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are

Column(s) of "FACT_LICENSE_INSP" Table		
Name	Datatype	Comment
		(TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
CENTER_STAT_CD	CHAR(2)	Indicates the status of license program. Values are (TSTC ref table): A: FILING DEADLINE MISSED B: CENTER RELOCATED C: OWNERSHIP CHANGED D: LICENSE REVOKED E: VOLUNTARY CLOSURE F: FACILITY TYPE CHANGE G: CLOSED VIA SETTLEMENT H: CLOSED FOR FINANCIAL REASONS I: TIME LIMIT EXCEEDED O: DEFAULT VALUE P: PROVISIONAL (SIX-MONTH)
CENTER_PRG_CD	CHAR(2)	Indicates the primary program status of the center. Values are from (TSPC ref tbl): A: APPLICATION C: CLOSED F: FISCAL L: LICENSE O: NO APPLICATION R: RENEWAL T: TRANSFERRED
SRVC_CONTR_CD	CHAR(1)	Indicates the type of the service contract (both certified and licensed center). Table-driven values derived from superset of existing license types and certified provider types.
NAEYC_ACCR_IND	CHAR(1)	Denotes whether the center s accredited by the NAEYC. Values are: Y - Accredited N- No accreditation
ENFRMNT_CD	CHAR(2)	The current enforcement status of the license program.
PART_NBR	SMALLINT	
ELI_IND	CHAR(1)	An indicator if The Early Learning Initiative (ELI) is a collaborative partnership between the Ohio Department of Education and the Ohio Department of Job and Family Services. Values: Y: ELI PROVIDER N: NOT ELI PROVIDER
INSP_DURATION	SMALLINT	The amount of time in minutes to perform the inspection.
WRKR_CD	CHAR(6)	INSP11-CDE-USERID Inspection Specialist user id.
RISK_CD	CHAR(1)	Code indicates the non compliance is serious risk or not.
SVRTY_PNT	SMALLINT	Severity Points.
ORIG_CMLNCE_CD	CHAR(1)	The current compliance status of the specific rule. Values: A

Column(s) of "FACT_LICENSE_INSP" Table		
Name	Datatype	Comment
		D I O P S U
CMPLNCE_STAT_CD	CHAR(1)	The current compliance status of the specific rule. Values: A D I O P S U
CMPLNCE_PRCES_DTE	DATE	The date of the most recent non-compliance status effective date provided by the specialist.
CMPLNCE_DEADLN_DTE	DATE	The date the specific non-compliant rule <u>finding</u> must be resolved by the facility.
CMPLNCE_EFF_DTE	DATE	The date the non-compliance status is updated.

Column(s) of "FACT_SUTQ_APPLICATION" Table		
Name	Datatype	Comment
APPL_KEY	INTEGER	Unique Step Up to Quality Application Id
YEAR_MTH_ID	CHAR(6)	Year and Month id of the the date the application was submitted.
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
APPL_DTE	TIMESTAMP	The date the application was submitted. (Yearly process)
SUTQ_CD	CHAR(2)	The Rating that the program applied for. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
APPL_TYP	CHAR(1)	Type of the application. Values: Initials (I), Renewals (R), and Change (C)
LEAD_TEACHER_TOT	SMALLINT	Number of lead teachers in the Center as reported.
ASST_TEACHER_TOT	SMALLINT	Number of assistant teachers in the Center as reported.
OTHR_MBR_TOT	SMALLINT	Number of other staff members in the Center as reported.
INFANT_CLSRM_TOT	SMALLINT	Number of classrooms serving infants as reported.
TODLR_CLSRM_TOT	SMALLINT	Number of classrooms serving toddlers as reported.
PRESCHL_CLSRM_TOT	SMALLINT	Number of classrooms serving preschoolers as reported.

Column(s) of "FACT_SUTQ_APPLICATION" Table		
Name	Datatype	Comment
SCHLAGE_CLSRM_TOT	SMALLINT	Number of classrooms serving school age children as reported.
NAEYC_ACCR_IND	CHAR(1)	An indicator if the program is Nationally Accredited.
APPL_STAT_ID	SMALLINT	Status indicating the state of the Application.
CNTY_NUM	SMALLINT	County number.
RCD_CNT	SMALLINT	Default to 1. Used by Cognos.
LOAD_TS	TIMESTAMP	The timestamp in which the table is loaded.
PART_NBR	SMALLINT	The DB2 partition number.
CENTER_TYP_CD	CHAR(1)	Center Types: 1,4,7

Column(s) of "FACT_SUTQ_CENTER" Table		
Name	Datatype	Comment
SUTQ_CNTR_KEY	INTEGER	Surrogate Key PROVIDER_ID AND CURR_RTNG_DTE
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
CURR_RTNG_DTE	DATE	
CUR_RTNG_YR_MTH_ID	CHAR(6)	% AttDef
SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
CNTY_NUM	SMALLINT	County number.
SULS_WRKR_ID	SMALLINT	Step Up to Licensing Specialist Id. (Surrogate Key)
RFRL_SPECLST_ID	INTEGER	SUTQ Referral Specialist Id (Surrogate Key).
PRIM_ADMIN_ID	INTEGER	Primary SUTQ Administrator Id.
SECD_ADMIN_ID	INTEGER	Secondary SUTQ Administrator Id.
RCD_CNT	SMALLINT	Cognos Reports uses.
LOAD_TS	TIMESTAMP	Table load timestamp.
PART_NBR	SMALLINT	

Column(s) of "FACT_SUTQ_LETTER_SENT" Table		
Name	Datatype	Comment
LETTER_KEY	INTEGER	Unique Key.
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
LETTER_SENT_DTE	DATE	The date letter sent.
LETTER_TYP_ID	INTEGER	Unique Letter Type Id. (FK)
LETR_YEAR_MTH_ID	CHAR(6)	% AttDef
SULS_WRKR_ID	SMALLINT	Step Up to Licensing Specialist Id. (Surrogate Key)
RCD_CNT	SMALLINT	For Cognos Use.
LOAD_TS	TIMESTAMP	Table Load timestamp
PART_NBR	SMALLINT	Part Number.

Column(s) of "FACT_SUTQ_PROGRAM" Table

Name	Datatype	Comment
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
YEAR_MTH_ID	CHAR(6)	%AttDef
CPCT_INFANT_NBR	SMALLINT	The infant licensed capacity level for the center.
CPCT_TODLR_NBR	SMALLINT	The toddler licensed capacity level for the center .
CPCT_PRESCHL_NBR	SMALLINT	The preschool capacity level for the center .
CPCT_SCHLAGE_NBR	SMALLINT	The school age capacity level for the center .
ENRLMNT_INFANT	SMALLINT	Number of Infant enrollment in the center.
ENRLMNT_TODLR	SMALLINT	Number of Toddler enrollment in the center.
ENRLMNT_PRESCH	SMALLINT	Number of Pre school enrollment in the center.
ENRLMNT_SCHLAGE	SMALLINT	Number of School age enrollment in the center.
LICNSE_START_DTE	DATE	The begin date for the licensure period of the center.
LICNSE_END_DTE	DATE	The end date for the licensure period of the center
LIC_APPL_DTE	DATE	The license application was received
WRKR_CD	CHAR(6)	The assigned licensing specialist for the center
PART_NBR	SMALLINT	The DB2 partition number.
SUBDZ_INFANT	SMALLINT	Number of subsidized infant in the center.
SUBDZ_TODLR	SMALLINT	Number of subsidized toddler in the center.
SUBDZ_PRESCH	SMALLINT	Number of subsidized pre school children in the center.
SUBDZ_SCHLAGE	SMALLINT	Number of subsidized school aged children in the center.
SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
RFRL_SPECLST_ID	INTEGER	Referral Specialist Id
PRIM_ADMIN_ID	INTEGER	Secondary Administrator Id.
CENTER_TYP_CD	CHAR(1)	Center Types: 1,4,7
CNTY_NUM	SMALLINT	County Number.
SULS_WRKR_ID	SMALLINT	Step Up to Licensing Specialist Worker Id.
RCD_CNT	SMALLINT	Default to 1. Used by Cognos.
LOAD_TS	TIMESTAMP	The timestamp in which the table is loaded.

Column(s) of "FACT_SUTQ_RATING" Table

Name	Datatype	Comment
RTNG_KEY	INTEGER	Surrogate Key: Provider Id Appl Year Mth Quality Rating Stat Id Rating Effective Date
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
YEAR_MTH_ID	CHAR(6)	Year and Month of the date associated with the History Rating action. (Effective date)
RTNG_STAT_ID	SMALLINT	Rating Status Id.
RTNG_EFF_DTE	DATE	The date associated with the History Rating

Column(s) of "FACT_SUTQ_RATING" Table		
Name	Datatype	Comment
		action. (Effective date)
APPL_DTE	DATE	The Application Date that resulted in the Rating.
CAP_RCVD_IND	CHAR(1)	Field indicates if the Corrective Action Plan was received and deals with SUTQ Suspensions. Basically the CAP has been received in writing before the Suspension is ended.
SUSP_END_DTE	DATE	Date when the suspension ends.
CENTER_TYP_CD	CHAR(1)	Center Types: 1,4,7
CNTY_NUM	SMALLINT	County Number grouped under SUTQ field office.
RCD_CNT	SMALLINT	Default to 1. Used by Cognos.
LOAD_TS	TIMESTAMP	The timestamp in which the table is loaded.
PART_NBR	SMALLINT	The DB2 partition number.

Column(s) of "FACT_SUTQ_VISIT" Table		
Name	Datatype	Comment
VISIT_KEY	INTEGER	Visit Key.
PROVIDER_ID	CHAR(15)	License Number (Six-digit number left-justified with zeros) and Certified Provider Number.
YEAR_MTH_ID	CHAR(6)	% AttDef
VISIT_BGN_DTE	DATE	Date of the First Visit to the Program.
VISIT_END_DTE	DATE	Date of the Last Visit to the Program
VISIT_DURATION	DECIMAL(6,4)	Total Duration of visit in hours:minutes.
VISIT_CNT	SMALLINT	Total number of visits to complete the visit.
ALT_PATHWY_IND	CHAR(1)	Indicates if the Program will choose to use the Alternate Pathway for attaining the Ratio Related Step Level using their Accreditations.
APLD_SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
AWRD_SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
RATIO_STEP_LVL	CHAR(4)	Rating/Step Level for the Ratio benchmark.
EDUC_STEP_LVL	CHAR(4)	Rating/Step Level for the Education benchmark.
TRAIN_STEP_LVL	CHAR(4)	Rating/Step Level for the Specialized Training Benchmark.
ENVRNMNT_SCORE	DECIMAL(6,4)	Environmental Rating Scale Score. Factored in when the program chooses the Alternate Pathway.
SULS_WRKR_ID	SMALLINT	Step Up to Licensing Specialist Id. (Surrogate Key)
SUMMER_SCHL_IND	CHAR(1)	Indicates if the Program has a summer school where the enrollment of kids and teachers are going to be significantly higher warranting a summer visit.

Column(s) of "FACT_SUTQ_VISIT" Table		
Name	Datatype	Comment
CENTER_TYP_CD	CHAR(1)	Center Types: 1,4,7
SUPV_VISIT_FLG	CHAR(1)	Indicates if supervisor reviewed the visit.
SUPV_VISIT_DTE	DATE	The Date when the supervisor reviewed the Visit.
DEFERRAL_FLG	CHAR(1)	Indicates if the program received a Deferral in submitting their documents needed for a rating award.
DEFERRAL_DTE	DATE	Date of the deferral.
DOCUMENT_DTE	DATE	Indicates the date when the program mailed in the necessary documents for rating award.
RCMND_SUTQ_CD	CHAR(2)	Indicates the Quality Rating. Values are (TSQR) 0: NO QUALITY RATING 1: ONE-STAR RATING 2: TWO-STAR RATING 3: THREE-STAR RATING X: NO VALUE
PART_NBR	SMALLINT	Part number.
RCD_CNT	SMALLINT	Cognos uses this field.
LOAD_TS	TIMESTAMP	The timestamp in which the table is loaded.
CNTY_NUM	SMALLINT	County number.

Column(s) of "FACT_SUTQ_VISITROLE" Table		
Name	Datatype	Comment
TEACHER_ID	INTEGER	Unique ID for Each Teacher entered into the Visit Tool. It is not universal. So a teacher entered in one tool is only unique only for that Visit.
ROOM_ID	INTEGER	The unique ID for each of the classrooms within a program
VISIT_KEY	INTEGER	Visit Key.
YEAR_MTH_ID	CHAR(6)	%AttDef
VISIT_BGN_DTE	DATE	Date of the First Visit to the Program.
PREYR_TRAIN_HRS	DECIMAL(5,1)	Number of hours of training completed by the teacher in that job position the previous year.
CURRYR_TRAIN_HRS	DECIMAL(5,1)	Number of hours of training completed by the teacher in that job position the current year
QUAL_ID	INTEGER	Qualification Key
PART_NBR	SMALLINT	Part Number.
RCD_CNT	SMALLINT	Cognos uses this field.
LOAD_TS	TIMESTAMP	The timestamp in which the table is loaded.

Column(s) of "PARENT_KEY" Table		
Name	Datatype	Comment
PARENT_KEY	INTEGER	Surrogate Key
CNTY_CD	CHAR(2)	
SSN	INTEGER	

Column(s) of "PARTY_KEY_XREF" Table		
Name	Datatype	Comment
PARTY_KEY	INTEGER	A system generated integer used to uniquely identify a row in the person table. (Surrogate key) This key can be substituted for the person ID when passing data to a data mart. The intention is eventually to replace the person ID timestamp wherever it occurs in the DW with this integer key.

Column(s) of "PARTY_KEY_XREF" Table		
Name	Datatype	Comment
PARTY_ID	TIMESTAMP	% AttDef
PARTY_TYP_CD	CHAR(2)	A unique code that identifies a party as a person or an organization.
INSERT_TS	TIMESTAMP	The system timestamp obtained when the row was inserted.

Column(s) of "STG_CHLD_CR_CSLOAD" Table		
Name	Datatype	Comment
APR_DTE	DATE	
PAY_SRCE	VARCHAR(17)	
AGE_GRP	VARCHAR(10)	
CHLD_100	INTEGER	
CST_100	DECIMAL(31,2)	
CPAY_100	DECIMAL(31,2)	
CHLD_125	INTEGER	
CST_125	DECIMAL(31,2)	
CPAY_125	DECIMAL(31,2)	
CHLD_150	INTEGER	
CST_150	DECIMAL(31,2)	
CPAY_150	DECIMAL(31,2)	
CHLD_165	INTEGER	
CST_165	DECIMAL(31,2)	
CPAY_165	DECIMAL(31,2)	
CHLD_185	INTEGER	
CST_185	DECIMAL(31,2)	
CPAY_185	DECIMAL(31,2)	
CHLD_186	INTEGER	
CST_186	DECIMAL(31,2)	
CPAY_186	DECIMAL(31,2)	

Column(s) of "STG_SUTQ_ADMIN_EMAIL" Table		
Name	Datatype	Comment
ADMIN_EMAIL	CHAR(70)	Administrator's email addresses (master list). Every month it will be updated with new admin emails.
LOAD_TS	TIMESTAMP	Table Load/Update Timestamp

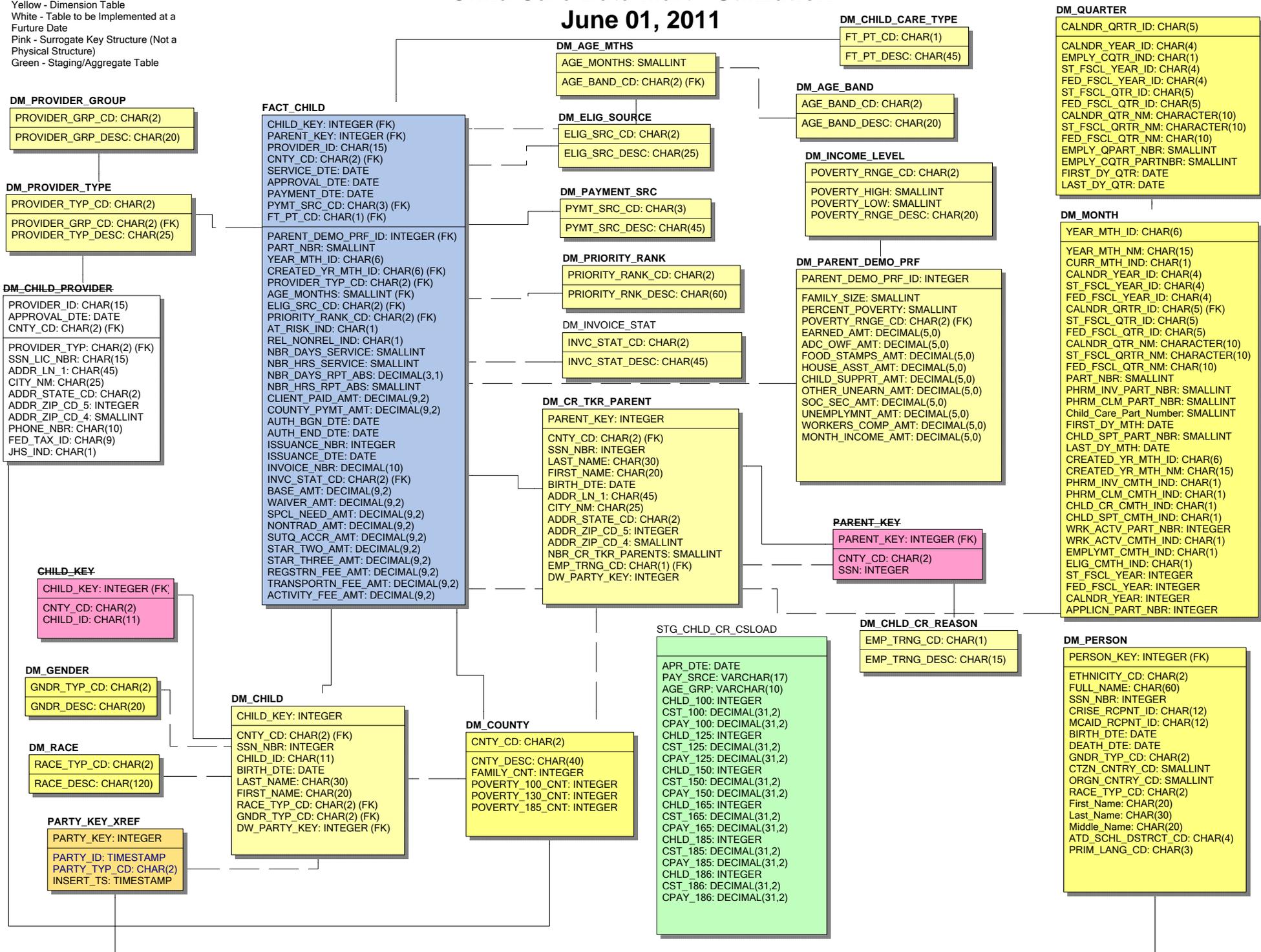
Supplement Eight

Child Care Utilization Data Model

Child Care Data Mart - Utilization

June 01, 2011

Blue - Fact Table
 Yellow - Dimension Table
 White - Table to be Implemented at a Future Date
 Pink - Surrogate Key Structure (Not a Physical Structure)
 Green - Staging/Aggregate Table



Supplement Nine

Child Care Step Up To Quality Data Model

Supplement Ten

Child Care License Center & Inspections Data Model

Supplement Eleven

Child Care Incident Injury Data Model

Supplement Twelve

**JFS Informatica
Development Standards**



Department of
Job and Family Services

John R. Kasich, Governor
Michael B. Colbert, Director

OHIO DEPARTMENT OF JOB & FAMILY SERVICES OFFICE OF INFORMATION SERVICES

APPLICATION DEVELOPMENT – EWBSS BUSINESS INTELLIGENCE SECTION



INFORMATICA NAMING STANDARDS AND DEVELOPMENT BEST PRACTICES

Bill Ennis,
Business Intelligence Development Manager
August 23, 2011

ODJFS Informatica Naming Standards and Best Practices

Naming Standards

SOURCE DEFINITIONS					
<p>When defining the sources, the Database name entered for the IMS and RDBMS sources will have the following format: <db_type>_<db_abbr>_<db_name> e.g. IMS_GS05_CASELOAD or DB2_DW01_ODJFS_DATA_WAREHOUSE</p> <p>Where <db_type> may be IMS, DB2, ORA, SYB, etc. <db_abbr> is the abbreviated database name, e.g. GS05, GS07, GS17, DW01, DW50, etc. <db_name> is the business name for the database, e.g. "ELIGIBILITY_DATA_MART" or "CASELOAD_DATABASE"</p> <p>IMS Data structures will all be created in the Shared Source Folder. If other data sources will have Database name entered as FLAT_FILES_<orig-machine> where <orig-machine> is the host name of the machine that created the file. e.g. FLAT_FILES_RHOSP055</p>					
Source Type	Source Description	Prefix	Name Format	Explanation	Example
VSAM	All IMS data source structures.	IMS_	IMS_<seg-name>	IMS segment name prefixed by IMS.	IMS_GS05PRT1 IMS_GS07ALRT IMS_GS17UNIT
	All other datasets that are extracted from the mainframe.	DSN_	DSN_<dataset-name>	A name that uniquely identifies the dataset on the mainframe, prefixed by DSN. The name may also show reference to the JCL that created the dataset.	DSN_FDW123FA
FLAT	ASCII files received from either Windows or UNIX platforms.	ASCII_	ASCII_<file-name>	A name that uniquely identifies the dataset on the Windows or UNIX platform, prefixed by ASCII.	ASCII_TIME_DIMN
RDBMS	All DB2 tables contained in the various databases.		<table_name>	<table_name> is the actual RDBMS table name, with no reference to the creator at this stage.	PERSON
FLAT or RDBMS	All temporary tables or files that are used during the transformation process.	TEMP_	TEMP_<dev-group-code>_<tab_name>	<dev-group-code> is the Development Group. <tab_name> is the descriptive name for the table/file.	TEMP_DW_WORKER_NAME TEMP_DE_TEST_DATA

TARGET DEFINITIONS					
Target Type	Target Description	Prefix	Name Format	Explanation	Example
RDBMS	All DB2 tables contained in the various databases.		<table_name>	<table_name> is the exact DB2 table name, with no reference to the creator at this stage.	PERSON
FLAT or RDBMS	All temporary tables that are used during the transformation process.	TEMP_	TEMP_ <dev-group-code> _ <tab_name>	<dev-group-code> is the Development Group Code. <tab_name> is the descriptive name for the table/file.	TEMP_DW_WORKER_NAME TEMP_DE_TEST_DATA
FLAT	ASCII files that are written to Windows or UNIX platforms. These files will be used for loading the target environment.	ASCII_	ASCII_ <file-name>	A name that uniquely identifies the file on the Windows or UNIX platform, prefixed by ASCII.	ASCII_TIME_DIMN

MAPPING/TRANSFORMATION DEFINITIONS					
Trans. Type	Transformation Description	Prefix	Name Format	Explanation	Example
Source Definition	This is the definition of the source object(s) appearing in the mapping.	SRC_	SRC_<db-id>_<tab-name>	<db-id> is derived from the middle four chars of the DB2 Creator Id or the ORACLE database name. <tab-name> is the table name. This name may be abbreviated by removing “_” and table prefix.	SRC_DW01_PERSON SRC_DW05_PERSON
Target Definition	This is the definition of the target object(s) appearing in the mappings.	TGT_	TGT_<db-id>_<tab-name>	<db-id> is derived from the middle four chars of the DB2 Creator Id or the ORACLE database name. <tab-name> is the table name. This name may be abbreviated by removing “_” and table prefix.	TGT_DW01_PERSON TGT_DW05_PERSON
Source Qualifier	Transformation that determines the records that the Informatica Server reads when it runs a session.	SQ_ SQL_	SQ_<sel-criteria>	<sel-criteria> summarizes the source records selection criteria.	SQ_ALL_RECS SQ_BDATE_OVER_20
Expression	Transformation that is used to calculate values in a single or multiple rows before passing them to the next transformations and/or targets.	CALC_ EXP_	CALC_<exp-descr> EXP_<exp-descr>	<exp-descr> summarizes the calculations or transformation rules that are made.	CALC_PERSON_AGE exp_PREP_PERSON
Filter	Transformation that provides the means for filtering records in a mapping.	FLT_	FLT_<sel-criteria>	<sel-criteria> describes the selection criteria applied to the data.	FLT_PERSON_AGE_GT_40
Sort	Sorts the data more efficiently	SRT_	SRT_ <sort descr>	<srt desc> A description about the sort.	

Aggregation	Transformation that allows aggregate calculations, such as averages and sums.	AGG_	AGG_<aggr-descr>	<aggr-descr> defines the aggregation applied to the data.	AGG_SUM_SALARY_BONUS
Lookup	With this transformation, you can access data from relational tables that are not sources in your mapping and/or find values in a table. The table can be persistent or non persistent.	LKP_	LKP_<db-id>_<tab-name>_<sel-purpose>	<db-id> is derived from the middle four chars of the DB2 Creator Id or the ORACLE database name. <tab-name> is the table name. This name may be abbreviated by removing “_” and table prefix. <sel-purpose> summarizes the purpose of reading the table.	LKP_DB01_PERSON_CHECK_CODE LKP_DB01_PERSN_GET_SSN
Sequence Generator	This transformation generates numeric values based on the previous/current execution of the transformation in the mapping.	SEQ_	SEQ_<seq-gen-name>	<seq-gen-name> defines the usage of the number generated.	SEQ_GEN_PRSN_SURR_KEY
Stored Procedure	This transformation executes procedures that are stored on the database to automate time-consuming tasks that are too complicated for standard SQL statements	SP_	SP_<db-id>_<proc-name>	<db-id> is derived from the middle four chars of the DB2 Creator Id or the ORACLE database name. <proc-name> is the stored procedure name.	SP_DW01_ACCUM_MTH_SALARY
Normalizer	Transformation normalizes records from COBOL which allows you to import flat files from the mainframe using COBOL copybook.	NORM_	NORM_<source-name>	<source-name> is the name of the source definition to be normalized. All mainframe flat files need Normalizer if a flat file will be used as a source.	NORM_IMS_GS07CTWK

Joiner	Transformation joins two related heterogeneous sources residing in different locations or file systems	JOIN_	JOIN_<source-1>_<source-2>	<source-n> defines the names of the sources to be joined.	JOIN_IMS_GS07CTWK_DW01_PERSON
Mapplet	A single transformation that connects multiple transformations to create the logic that is flexible enough to use in different mappings.	MPL_	MPL_<mpl-descr>	<mpl-descr> a name that describe the usage of the mapplet.	MPL_CONVERT_PERSON_READ_SSN
Mapping	Set of transformations that define the data flow from source to target.	MAP_	MAP_<mapping-descr>	<mapping-descr> describes the purpose intended by the set of transformations.	MAP_CREATE_PERSON_TABLE

BATCHES AND SESSIONS					
Type	Description	Prefix	Name Format	Explanation	Example
Session	Defines the mapping that must be executed by the PowerMart Server, and the required information about the sources, targets and shared memory allocations.	S_	S_<mapping-name>	<mapping-name> is the name of the mapping that the session will execute.	S_MAP_CREATE_PERSON_TABLE
Workflow	Sessions grouped together to improve performance and/or to ensure targets load in a specific order. Each batch can contain any number of sessions or other batches.	B_ or W_ or WF_	B_<batch_descr>	<batch_descr> describes the intended purpose of grouping the sessions/batches together.	B_LOAD_ELIGIBILITY_DIMENSIONS Wf_POST_MART

Best Practices - General Use

No Hard-Coding

- There should be no hard-coding of values in the mapping code, use variables and/or parameter to manage these values.

No Views in SQL

- Do not use Views in the SQL for source qualifier or lookup transformation. This produces undesirable effects in several ways:
- Business logic tends to be embedded in the view instead of the mapping.
 - Increased complexity when debugging the mapping.
 - Lost metadata about the business logic
 - Poorer performance.

Use few transformations

- There is always overhead involved in moving data between transformations.
- Consider more shared memory for large number of transformations. Session shared memory between 12MB and 40MB should suffice.
- Keep mappings as simple as possible. The smaller the better in terms of performance and tuning.
- Never have more than five (5) targets per mapping. This will slow down the mapping exponentially. Complex maps usually demand multiple targets, but the more targets you have, the poorer the performance.

Use flat files

- Using flat files located on the server machine loads faster than a database located on the server machine.
- Fixed-width files are faster to load than delimited files because delimited files require extra parsing.

Calculate once, use many times

- Avoid calculating or testing the same value over and over.
- Calculate it once in an expression, and set a True/False flag.
- Within an expression, use variable ports to calculate a value that can be used multiple times within that transformation.

Only connect what is used

- Delete unnecessary links between transformations to minimize the amount of data moved, particularly in the Source Qualifier.
- This is also helpful for maintenance. If a transformation needs to be reconnected, it is best to only have necessary ports set as input and output to reconnect.
- Avoid crossing port lines between transformations. Any time these "fields" move from object to object, they will be shuffled in memory. By keeping the field lines as straight as possible, you give the server internals a chance at "copying" chunks of memory, rather than field by field data movement.

Watch the data types

- Data types are automatically converted by the PowerCenter engine when types are different between connected ports. These can be resource intensive.
- Minimize data type changes between transformations by planning data flow prior to developing the mapping.

Facilitate reuse

- Plan for reusable transformations upfront.
- Use variables. Use both mapping variables as well as ports that are variables. Variable ports are especially beneficial when they can be used to calculate a complex expression or perform a disconnected lookup call only once instead of multiple.

Manipulate necessary data only

- Reduce the number of non-essential records that are passed through the entire mapping.
- Use active transformations that reduce the number of records as early in the mapping as possible (i.e., placing filters, aggregators as close to source as possible).

Utilize single-pass reads

- Remove or reduce field-level stored procedures.
- If you use field-level stored procedures, the PowerCenter server has to make a call to that stored procedure for every row, slowing performance.

Surrogate Key Generation (No partitioning)

- Use an unconnected lookup to determine the last key value used.
- Define an expression variable (v_max_key) to read the last key used when processing the first record.
- Define an expression variable (v_new_rec) to determine if the record already exists or not.
- Define an expression variable to accumulate number of new keys to be created. For example, v_cnt = iif(v_new_rec, v_cnt + 1, v_cnt).
- Define output port to assign a surrogate key to the record as s_key = iif(v_new_rec, v_max_key + v_cnt, old_key_value)
- Perform an "Update else Insert" or filter new records only and perform "Insert" on the target.

Data Quality and Cleansing

- When processing string fields, check for special characters such as non-printable characters.
- Always apply LTRIM/RTRIM, LPAD/RPAD and LJUST/RJUST functions to strings.
- Assign default values for NOT NULL columns.
- Build in data type conversions if source and target data types are different.
- Always validate the dates when processing data.

"Update else Insert" logic

- Always determine logic in the mapping instead of setting in the session. If you use the session setting, the server will send two statements (update & insert stmt) to the database for every record processed.
- Use Update Strategy transformation and increase performance.

Sort Keys

- When using a Sort, Aggregator, Joiner, or Lookup transformation keep the keys as "small" as possible (measured in precision). Much of the same mathematics that play in computing relational database indexes also play in computing the "indexed" fields that perform the operations listed above.
- Numeric key values process faster than alphanumeric, therefore try some temporary conversion strategies before performing the sort.

Best Practices for Transformations

Lookup Transformation

- Avoid date comparisons in lookup; replace with string.
- Change unused ports to be neither input nor output. This makes the generated SQL override as small as possible, which cuts down on the amount of cache necessary and thereby improves performance.
- Indexes on the fields required in the lookup will improve loading of the lookup data.
- Verify that the data types of the input column and the connected port match (string length, trim, justified – identical).
- When your source is large, cache lookup table columns for those lookup tables of 500,000 rows or less. This typically improves performance by 10 to 20 percent.
- The rule of thumb is not to cache any table over 500,000 rows. This is only true if the standard record length is 1,024 or less. If the record length is more than 1,024, then the 500k rows will have to be adjusted down as the number of bytes increase (i.e., a 2,048 byte row can drop the cache row count to between 250K and 300K, so the lookup table should not be cached in this case).
- Place all conditions that use the equality operator '=' first in the list of conditions under the condition tab.
- Cache lookup tables only if the number of lookup calls is more than 10 to 20 percent of the lookup table

rows. For fewer number of lookup calls, do not cache if the number of lookup table rows is large. For small lookup tables (i.e., less than 5,000 rows), cache for more than 5 to 10 lookup calls.

- If caching lookups and performance is poor, consider replacing with an unconnected, un-cached lookup.
- For overly large lookup tables, use dynamic caching along with a persistent cache. Cache the entire table to a persistent file on the first run, enable the update else insert option on the dynamic cache and the engine will never have to go back to the database to read data from this table.

Expression Transformation

- Numeric operations are faster than string operations.
- Optimize char-varchar comparisons (i.e., trim spaces before comparing).
- Operators are faster than functions (i.e., || vs. CONCAT)
- Disable decimal arithmetic, if possible.
- Use indents and white spaces in the expression building to facilitate readability and ease of maintenance.
- Test expression timing by replacing with constant.
- Always use expression variables to simplify complex transformations:
 - Variables should be defined after all input ports.
 - Variables using other variables in an expression should be defined in the order of dependency.
 - Variables are initialized to zero (0) for numeric and “empty space” for alphanumeric.
 - Variables can retain values across rows until the next evaluation of the expression.
 - Use meaningful variable names in expressions. For example, IIF(item_id=1000, DD_DELETE, DD_INSERT) rather than IIF(item_id=1000, 2, 0).

Router Transformation

- Use to separate data flows instead of multiple Filter Transformations.
- Keep filter conditions simple, move the complex condition expressions into expression objects. When the filter runs slowly it's usually because of a complex condition.

Normalizer Transformation

- Use to pivot rows rather than multiple instances of the same target.

Joiner Transformation

- Be sure to make the source with the smallest amount of data the Master source.

Aggregator Transformation

- Always use sorted data for very large data aggregations.
- Use a Sorter Transformation or hash-auto keys partitioning before an Aggregator Transformation to optimize the aggregate. With a Sorter Transformation, the Sorted Ports option can be used, even if the original source cannot be ordered.
- Replace Aggregate Transformation object with an Expression Transformation object and an Update Strategy Transformation for certain types of Aggregations.
- If the input is sorted into the Aggregator, then check the box for sorted input. This will prevent Informatica from doing a sort before aggregating the data.
- Aggregator transformations do not sort data. The aggregator uses a clustering algorithm, not a sort algorithm. When there are duplicate rows the aggregator may put data out in a seemingly sorted order but it does not guarantee it.

Update Strategy

- Rejected rows from an update strategy are logged to the bad file. Consider filtering before the update strategy if retaining these rows is not critical because logging causes extra overhead on the engine. Choose the option in the update strategy to discard rejected rows.
- If an update override is necessary in a load, consider using a Lookup transformation just in front of the target to retrieve the primary key. The primary key update will be much faster than the non-indexed lookup override.

Best Practices for Processing Data

PowerCenter Server

- Restart the server every 3-4 weeks to avoid memory leaks.

Sessions and Workflows

- ALWAYS design the workflows and sessions/mappings with the capability to restart with ease, in case there is a failure for whatever reason. Using function specific workflows usually works well, or even defining one session per workflow can provide better restart capabilities, especially when using Enterprise Control-M (ME) scheduler to manage the run schedule.
- All sessions and workflows MUST use environment and session parameters.
- When bulk loading, the PowerCenter Server uses the native database drivers and invokes the database bulk utility and bypasses the database log, which increases performance. Without writing to the database log, however, the target database cannot perform rollback. As a result, you may not be able to perform recovery. Therefore, you must weigh the importance of improved session performance against the ability to recover an incomplete session. This option can possibly be used with a pre-session SQL that will delete any data already loaded to the table to avoid duplicates on the table.
- When using an external loader, the PowerCenter session does not connect to the target database directly to load data but instead invokes the external application for the specific database (*autoloader* for DB2 EEE, *sqlldr* for Oracle, etc.) which loads the rows into the target table. This is a much more reliable option for loading huge volumes of data into databases.
- All sessions should have the "Fail Parent if Task Fails" checked on the session level. This is really a **must** if you are executing a starttask command.
- Override Tracing should ALWAYS be set to "NONE", unless you are performing a small data volume test.
- If a file is to be staged during FTP, a "rm -f filename" command is needed in the pre- AND post- session command.
- The "Stop on Error" option should never be set to 0. This will allow unlimited errors during the session and the log file becomes very large.
- All connections between two sessions should have at least a condition that the previous task is "SUCCEEDED", otherwise the workflow will continue processing even when there are "session aborts".
- All sessions that perform inserts only to the database should have commit intervals of 50,000 or more when the session runs.

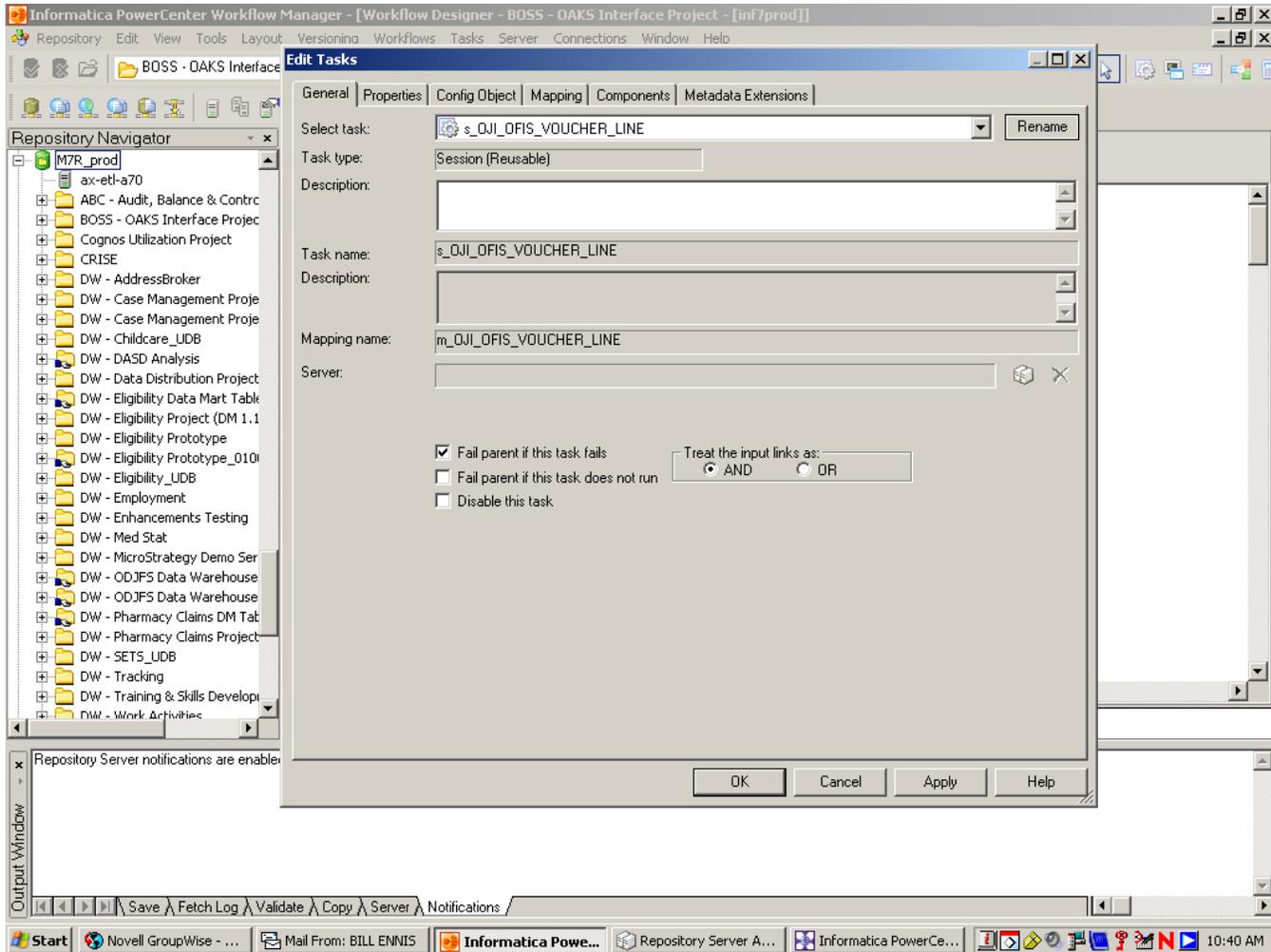
Additional Guidelines for Informatica developers:

1. When testing all target connection should point to the test database even if no rows will be updated or inserted for this connection. In some cases, the actual SQL to update the table is stored in a UPDATE OVERRIDE.
2. On a normal workflow, all sessions should have the Fail Parent if Task Fails. Unless the task is the last session within the workflow, the box "Fail Parent if the Task Does Not Run" should NOT be checked.
3. On SQL select, you should add WITH UR to the end of the SELECT statement especially for large reads or tables with large amount of users.
4. The "On Pre - Post SQL Error" field be set to STOP.
5. In production, Override Tracing should be set to NONE. However, in a small volume test, you might want to change this value.
6. All sessions that do inserts only to the database should have commit intervals 50,000 or more when the job runs in a production or full volume test environment.
7. Lookup cache should be checked as enable
8. For production workflow (batches), the standard email for both success and failures should be used. The standard email should be a reusable email within each production folder.
9. All Event Waits should have a name describing what the purpose for the wait like WT_DEMO_PROF_END. This name can be easily seen in the Workflow Monitor and Workflow Manager.
10. If a file is to be FTP staged, an rm -f FILE NAME is needed in the pre session command. A bug with Informatica 6 that does not properly delete the file before the actual file transfer starts. If the file is large and not needed for future process, then an rm FILE NAME command should be done before the batch ends.
11. All triggers should go to the /home/ftpmart/dev or /home/ftpmart/prod directory.
12. The "stop on errors" should NEVER be set to zero in production or during large volume testing. This zero

- allows for unlimited errors during the session, and sometime can cause problems with giant log files.
13. All UNIX scripts should be stored under the directory /mart_install/info/scripts
 14. The main directory for the TgtFiles and SrcFiles starts with /mar_install/info/mart_dev.
 15. All connections between two sessions should have at least a condition that the previous task is a SUCCEEDED. If a post session fails, then the workflow will continue unless the link has the link has the previous task = SUCCEEDED coded.
 16. If a Workflow will not be used for an extended period, then the Workflow should be marked as Disable. This will prevent batches being scheduled or submitted accidentally.
 17. Each production Workflow should have a description that describes the workflow and any special restart notes. Any special restart instruction should also be placed on the task sessions.
 18. Enable Recovery should not be checked unless the session fails in the middle of loading or updating records. The flag will need to unchecked before the session runs again. This might require that the workflow be unscheduled and rescheduled again to have the removal of the check to take effect.
 19. The server connection should only be updated or modified by the Informatica Administrator. In addition, the Repository Server Administrator tool should only be used by the Informatica Administrator.
 20. For performance, if you know that the input is sorted into the Aggregator, then check the box for sorted input. This will prevent Informatica from doing a sort before aggregating the data.
 21. All new production scheduling will go through Enterprise Control M (EM). The Informatica Administrator will help in creating the required scripts that the tool needs to run successfully. EM allows the scheduler to communicate between various servers and the mainframe for scheduling purpose. However workflows are still needed for Informatica use and special request or test runs.
-
22. For future upgrades and portability, all UNIX file names should use PowerCenter relative address method compared to using absolute file addressing. The relative variable names can be found under the Server properties.
 23. When inserting rows to mainframe DB2, the Normal mode should be used instead of the Bulk load.
 24. Eliminate duplicate rows, use the sort with distinct checked instead of using the aggregator or the ranker.
 25. When running against a file of more than 1000 rows, make sure that verbose is off. Verbose can create a huge log files, and the actual problem can get lost within the log.
 26. For large flat files, you should increase the Default Buffer Block size to at least 96000.
 27. For large maps or large lookup cache due to a join, sort, lookup, or aggregator, you should increase the default sizes and increase the DTM Buffer Size. If you have question about DTM buffer pool or cache size, then contact your Informatica administrator.
 28. When your target is UDB, and you two targets pointing to the same table. You **CANNOT** use BULK mode, because UDB will have internal contention. The NORMAL mode does not have this problem. Bulk mode should also not be used as a target if the session is somehow both reading and writing to the same table.
 29. You can use reusable emails and tasks. In production, all failures should use a standard e-mail that contains the log as an attachment.
 30. A session can write to both a file on the Unix machine and to a relational database. In addition, one session can write to multiple target connections.
 31. The log now contains the number of records read.
 32. Pre and Post SQL can be executed without executing a pre or post session script.
 33. Control rules can now be placed in the links between sessions
 34. A new feature allows a person to compare the difference between 2 maps.
 35. Below are some things I learned during the conversion and upgrade that I felt that might be helpful. These are not new standards or improvements but deals with hints on using the tool.
 36. To stop a Workflow or a session, the Workflow Monitor should be used. However to start a new workflow, the Workflow manager can be used.
 37. In the Workflow Designer, if a larger Workflow looks scramble, the **Layout** followed by **Arrange** will fix the problem.
 38. The only way to change the Update Override for the target is using the Designer tool. Also the next number sequencer is stored on the map and not part of the Workflow Manager. The value can be changed using the Designer tool.
 39. For DB2 mainframe relational connection, the Code Page needs to be Western European. For Oracle UNIX connections, the Code Page should be MS Window Latin 1 (ANSI).
 40. The code page for FTP files can be found in **Set file Properties** followed by **Advance**. The mainframe connection still needs to be EBCDIC US English.
 41. In the previous version, the null character input feature (on the same screen as the Source FTP file code page) was not working, and now this feature is working correctly. This might cause problem if you are not

aware of this feature.

42. Sometimes, a connection of off line and on line is needed to see some historical sessions. To see the log files in workflow monitor, you need to be connected on line. The log file shows the most recent log file with that name, not the log associated with the time that the log ran. For instance, the Workflow Monitor will show the time of the failure and the success, but the log file for both sessions will be the log for the most recent run.
43. If a session fails in the initialization phase, the error might be shown in the workflow log instead of the session log.
44. Always check Fail Parent option, especially with the starttask command. When checked, workflows fail if the session fails. Without this a session can fail, but the workflow will succeed.



ODJFS ETL Architecture – Surrogate Key Strategy & Best Practices

A surrogate key is a unique, meaningless identifier, usually a sequential numeric value, that will be used as the primary key for a table, in place of a natural key. The concept is similar to creating a Customer ID for processing sales transactions, instead of using the Social Security Number or Customer Last Name as the primary key of the Customer table. When designing business intelligence databases a new key should be developed instead of using the source system Customer ID. This document outlines the reasons for implementing a Surrogate Key Strategy.

The Surrogate Key Strategy should be implemented early in the data warehouse physical design stage. No meaning is implied by a surrogate key value. For example, you can not look at a month identifier of '1' and assume that it represents January. A surrogate key allows for more efficient navigation across data structures during the ETL loading process, in business intelligence report queries and facilitates integration across applications. When developing a Surrogate Key Strategy it is important to understand the following trade-offs: it can take more time and effort to navigate between data structures as well as add more complexity to the development phase when surrogate keys are used. The payback in implementing surrogate keys comes in the testing, performance tuning and implementation stages of the project, when the system performance meets or exceeds the customers expectations. The following guidelines should be used in determining where it makes sense to create a surrogate key.

Create a surrogate key if the entity has a natural key and any of the following conditions hold:

- There are multiple sources;
- The natural key is inefficient; or
- The natural key is recycled.

Surrogate versus Virtual

A *natural* key is ideally how the business would identify an entity instance, and a surrogate key is a substitute for this natural key. For example, a natural key for an organization could be taxpayer identifier or DUNS (D&B) Number. If no natural key exists, we can create a virtual key. A *virtual* key is a unique numeric counter used when it is not possible to identify an entity instance. For example, a manufacturing company has the concept of an adjustment. An adjustment is when a pallet is moved between shelves in a warehouse. The business has no way of identifying this kind of transaction, so the virtual key adjustment identifier is created.

Create a Surrogate if Multiple Sources Exist

If the entity you are modeling contains data from multiple sources, there is usually value in creating a surrogate key. For example, assume you learn that both Henry Winkler the student and Hank Winkler the instructor are really the same person. You can add this person as a unique row to the person entity identified by a person identifier surrogate key. This person can then play the two roles of instructor and student.

Create a Surrogate if the Natural Key is Inefficient

If the natural key is inefficient, create a surrogate key. Inefficient means that the data elements which make up the natural key have performance or space issues when joining across tables. This surfaces most frequently when joining a reference to a transaction table, such as a dimension to a fact table. For example, the business might identify a promotion by a promotion code and promotion start date. Although the code and start date have little impact on a promotion table containing only a few thousand rows, there will be an impact on the multimillion row order and credit tables that reference promotion.

Create a Surrogate if the Natural Key Values are Recycled

While working with a manufacturing plant, I learned that one of their key business concepts is identified by a code whose values repeat every 18 months. We decided to use a surrogate key as a buffer layer and create an alternate key on this natural key code plus a date. As an aside, it is a very good practice to always define an alternate key on the natural key when using a surrogate.

Every set of guidelines has exceptions, and for surrogate keys a very real exception is the limitation of application development tools. There is technology in use that requires surrogates while other tools prohibit them.

Best Practice Recommendations

There are different algorithms for surrogate key generation. These can be grouped into four general approaches:

1. **Series Generation** : Keys that are progressive in the collating sequence. There are various methods that can be used to produce keys of this kind, namely:
 - **Next Key Table** - a table used to keep the next key value that is available as a surrogate key.
 - Informatica Sequence Generator facilitates keeping track of the max value.
 - **Max Plus One** - determine the next key value by selecting the max from the data table and adding 1 to the value.
 - Use Informatica Lookup Transformation to cache the max value when the session starts.
 - Use Informatica Source Qualifier, Joiner transformations together with new record count to determine surrogate key.
2. **Pseudo-random Numbers** : Surrogate keys are generated by a database stored procedure applying seeding algorithm to randomly determine a new key.
3. **Composite Key** : Surrogate keys are generated by combining multiple parts to build a unique new value. For example, the first part of the key could be used to control the spread of the inserts throughout the data table and the second part is used to ensure uniqueness.
4. **Database Triggers** : Surrogate keys are generated using a database trigger defined on the data table to determine next value only when an insert statement is executed on the table.

The ODJFS ETL Best Practices recommend that the "Max Plus One" method defined above be used in all processes that generate surrogate keys. The enhanced version of this method is recommended for processes that require parallel (partitioned) processing to improve load process performance.

CREATING SURROGATE KEYS ON DB2

When designing DB2 databases a frequently heard request is for a column that contains sequentially generated numbers. For example, each row has a counter associated with it. When a new row is inserted, the counter should be incremented by one for the new row. This way, each new DB2 row has a unique "row number" associated with it.

A common technique is to maintain a one-row table that contains the sequence number. Each transaction locks that table, increments the number, and then commits the change to unlock the table. In this scenario only one transaction at a time can increment the sequence number. A variation uses something like this

```
SELECT MAX()+ 1
FROM ONEROW_TABLE
WITH RR;
```

The result is the next highest number to be used. This value is used by the application and ONEROW_TABLE must be updated with the incremented value. Performance bottlenecks will occur with this method when a lot of concurrent usage is required.

Identity Columns

Identity columns were formally added to DB2 as of Version 7, but were actually available as of the DB2 Version 6 refresh. The identity property is applied to a DB2 column using the IDENTITY parameter. A column thusly defined will cause DB2 to automatically generate a sequential value for that column when a row is added to the table. For example, identity columns might be used to generate primary key values or a value that somewhat mimics Oracle's row number capability. Using identity columns helps to avoid some of the concurrency and performance problems that can occur when application programs are used to populate sequential values for a "counter" column.

When inserting data into a table that uses an identity column, the program or user will not provide a value for the identity column. Instead, DB2 automatically generates the appropriate value to be inserted.

Only one identity column can be defined per DB2 table. Additionally, the data type of the column must be SMALLINT, INTEGER, or DECIMAL with a zero scale, that is DECIMAL(*n*,0). The data type also can be a user-

defined DISTINCT type based on one of these numeric data types. The designer has control over the starting point for the generated sequential values, and the number by which the count is incremented.

An example creating a table with an identity column follows:

```
CREATE TABLE EXAMPLE
(ID_COL INTEGER NOT NULL
GENERATED ALWAYS AS IDENTITY
START WITH 100
INCREMENT BY 10
...);
```

In this example, the identity column is named ID_COL. The first value stored in the column will be 100 and subsequent INSERTs will add 10 to the last value. So the identity column values generated will be 100, 110, 120, 130, and so on.

Note, too, that each identity column has a property associated with it assigned using the GENERATED parameter. This parameter indicates how DB2 generates values for the column. You must specify GENERATED if the column is to be considered an identity column or the data type of the column is a ROWID. This means that DB2 must be permitted to generate values for all identity columns. There are two options for the GENERATED parameter: ALWAYS and BY DEFAULT.

- **GENERATED ALWAYS** indicates that DB2 will always generate a value for the column when a row is inserted into the table. We will usually specify ALWAYS for your identity columns unless we are using data propagation.
- **GENERATED BY DEFAULT** indicates that DB2 will generate a value for the column when a row is inserted into the table unless a value is specified. So, if we want to be able to insert an explicit value into an identity column we must specify GENERATED BY DEFAULT.

Additionally, we can specify what to do when the maximum value is hit. Specifying the CYCLE keyword will cause DB2 to begin generating values from the minimum value all over again. Of course, this can cause duplicate values to be generated and should only be used when uniqueness is not a requirement.

Actually, the only way to ensure uniqueness of your identity columns is to create a unique index on the column. **The IDENTITY property alone will not guarantee uniqueness.**

Sometimes it is necessary to retrieve the value of an identity column immediately after it is inserted. For example, if we are using identity columns for primary key generation we may need to retrieve the value to provide the foreign key of a child table row that is to be inserted after the primary key is generated. DB2 provides the IDENTITY_VAL_LOCAL() function that can be used to retrieve the value of an identity column after insertion. For example, we can run the following statement immediately after the INSERT statement that sets the identity value:

```
VALUES IDENTITY_VAL_LOCAL() INTO :IVAR;
```

The host variable IVAR will contain the value of the identity column.

Summary

In the data-warehouse identity columns are used in the following tables:

```
ADDRESS_KEY_XREF
PARTY_KEY_XREF
```

Identity columns are useful when:

- Only one column in a table requires automatically generated values
- Each row requires a separate value
- An automatic generator is desired for a primary key of a table

- The LOAD utility is not used to load data into the table
- The process of generating a new value is tied closely to inserting into a table, regardless of how the insert happens

Problems with Identity Columns

Identity columns can be useful, depending on your specific needs, but the problems that accompany identity columns are numerous. Some of these problems include:

- Handling the loading of data into a table with an identity column defined as GENERATED BY DEFAULT. The next identity value stored by DB2 to be assigned may not be the correct value that should be generated. This can be especially troublesome in a testing environment.
- LOAD INTO PART x is not allowed if an identity column is part of the partitioning index.
- What about environments that require regular loading and reloading (REPLACE) for testing? The identity column will not necessarily hold the same values for the same rows from test to test.
- Prior to V8, it was not possible to change the GENERATED parameter (such as from GENERATED BY DEFAULT to GENERATED ALWAYS).
- The IDENTITY_VAL_LOCAL() function returns the value used for the last insert to the identity column. But it only works after a singleton INSERT. This means you cannot use INSERT INTO SELECT FROM or LOAD, if you need to rely on this function.
- When the maximum value is reached for the identity column, DB2 will cycle back to the beginning to begin reassigning values - which might not be the desired approach.
- The counter for the identity column is stored in SYSIBM.SYSSEQUENCES and **cannot be** altered by the DBA. The only time the value in SYSIBM.SYSSEQUENCES is alterable is by dropping and recreating the table.
- It is more than likely that when moving data from one environment to another via a unload/load utility the max value in SYSIBM.SYSSEQUENCES will not match between test and production. Any attempt to do an insert into the table will result in a -803 – primary index violation because of duplicate insert.
- The GENERATED BY DEFAULT option is usually selected on the identity column because otherwise the value will change when the table is loaded and that would be undesirable for other tables that might care what the value is in that column. The GENERATED BY DEFAULT option allows us to specify a value when you do an insert into a table. However, if a value is specified, the value in SYSIBM.SYSSEQUENCES is overridden. Thus when SYSIBM.SYSSEQUENCES reaches the value that was inserted it will result in a -803 again.
- Anytime the table is dropped/recreated the max value for the identity column needs to be noted before dropping the table. When the table is re-created the max value of the identity column will have to be reset to this value again.

If we can live with these caveats, then identity columns might be useful to our applications. However, in general, these "problems" make identity columns a very niche solution. IBM has intentions to rectify some of these problems over time in upcoming versions of DB2.

Enterprise Job Scheduling For Informatica

Overview

The enterprise scheduler solution for ODJFS is Control M. Instead of using Power Mart scheduler, the standard is to run on production scheduled jobs off the EM Scheduler. There may still be a case where one time jobs need to run off your desktop, but all production scheduled jobs need to be in EM scheduler. The EM scheduler can run any type of Power Mart script, and the In and Out condition can communicate to the mainframe scheduler and other jobs that are in the EM scheduler.

This document deals with the write-up dealing of scheduling and Informatica scripting so that Informatica jobs can execute on the EM scheduler. Please follow the standard laid in this document so that upgrades and maintenance can be easier. I would advice similar standards to be in test as well as production whenever possible.

Justification

The following are reasons for using enterprise Control M.

- 1) Production Control can monitor the jobs, and call you or send you e0-mail when the job fails. Also production control can rerun or force complete an EM scheduled job. .
- 2) Remove the number of trigger files.
- 3) Move towards using the standard product for scheduling the product. This may include the Cognos cubes being triggered by the completion of a Control M out condition on the mainframe.
- 4) EM Schedule has the ability to FTP files efficiently. Also, EM scheduler can provide a graphic representation of job monitoring.
- 5) Control M is the standard package for scheduling and monitoring production jobs. .

Standards

ODJFS has certain stands for name the Job Name associated with a UNIX script or a UNIX commands All EM jobs has 11 characters, and sometimes 12 Character to make the name unique but the script is the same as a previous session. This is done by the DBA, but the application programmer rarely needs to do this. Below is the 11 character layout that should be used for naming all Informatica jobs that goes to EM scheduling. The examples are based on the Data Warehouse, but can be changed using your specific Major and Minor Code.

- 1 – 1 Major Code – F for the Data Warehouse and G for CRISE Other non Data Warehouse group will use their own Major and Minor system codes associated with their group and project.
- 2 – 3 Minor Code – DM for DataMart and DW for Data Warehouse.
- 4 – 7 A 4 digit numeric numbers (see below for data warehouse numbering).
- 8 – 8 Frequency of the run like W = Weekly, M = Monthly, and R = On Request
- 9 – 11 DEV = Development or PRD = Production

A Data Warehouse standard should have the 4 digit should be based on some type of project. Below is the initial break down, and when new projects are added this break down can change.

- 0001 – 0499 = Child Care
- 0500 – 0999 = Sets
- 1000 – 1499 = Elig. Pre mart
- 1500 – 1999 = Pharmacy Mart
- 2000 – 2499 = Case Management
- 2500 – 2999 = Application Timeliness
- 3000 – 3499 = Address
- 3500 – 3999 = ORAA
- 4000 – 4500 = Elig Mart and Post Mart
- 9500 – 9999 = Special Request

Some thought might be given to use some of the last two digits to have meaning. For example, number that ends with a 00 should be reserve for the submitting a Workflow for creating the Parm file. Also, a session that ends with a 99 can stand for a trigger or dummy session specifying the beginning or ending of a workflow within Informatica.

Procedures

All request adding new jobs to Enterprise Control M (EM) need to be sent as a Groupwise task to SCHEDPRO and CC your team lead. All emergency request needs to be sent to SCHEDEPR and CC your team lead or manager. All new scheduled EM needs an EM SCHEDULE MODIFICATION REQUEST FORM, and the following information might be helpful in filling out this form.

- 1) Informatica runs on a UNIX machine (AX-DEV-A08) while Cognos runs on a NT server.
- 2) UNIX is case sensitive. Keep this in mind when doing any forms associated with the EM scheduler. Also try to avoid placing blanks in name fields.
- 3) For Informatica processing the **Node Id/Group** (Server Name) is AX-DEV-A08.
- 4) For monthly job that might have ODATE problems, request that the condition is setup with a STAT.
- 5) If you want to be paged if the job gets a return code greater than 0, remember to place the 11 character job name in OCES. Also, specify this information with the instruction for job abend.
- 6) Notify the Informatica Administrator if you want to setup a new script
- 7) The **OWNER** is mart7adm and keeps the **GROUP** field blank.
- 8) All condition between the server and the mainframe needs to be defined as a global condition.
- 9) As a general rule, the EM and Control M work in the same way with in condition and out condition.
- 10) The **TASK TYPE** is Command where the **PATH** name always start with /mart_install/info/scripts/XXXX where XXXX = ELIG, CSLD, PMRT, ADDR, APPL, CRISE, ERIC, MMIS, SETS or ORAA The Path can be considered the directory where the file name exists.
- 11) The **FILENAME** is the same where the actual pmcmd exists. The only internal standard is the 4 digit for the JOBNAME is part of the FILENAME. For the Data Warehouse ETL team the FILENAME will be the same as the JOBNAME but in lower case and a .sh extension
- 12) The **Report Information** should be kept blank, but in the future this might be used when a reporting tool gets attached to this project.
- 13) The **Cycle** should only be used when a job is executed more than once per day.
- 14) The **Database, Read, and Update** are not required even if the job updates a DB2 table. Currently, this information is for documentation purposes only, and might be dropped from the request form in the future.
- 15) The **Description** is information about the process that the Job name is executing. This information can be seen by production support under the property tab. The standard is to place at least the session or workflow name in the Description field.

EMAILS and Other Notification

I strongly advised to continue to use Informatica e-mails for both success and failure, because the Informatica can attach the log or the row counts to the e-mail. Since session log history is not kept on the UNIX box, the session logs can be useful in doing research.

For non critical jobs, E-Mail Address needs to be specified on the EM Scheduling Request form. You can specify POWERMART_RESOURCE as an e-mail location or some other group box. The Data Warehouse ETL team is using POWERMART_RESOURCE as the group e-mail box. However, for critical jobs, the 11 character job name must be in OCES, and under **Instructions for job abends** specify the contact the individual(s) who is responsible for the job (on call programmer(s)). In general, EM notification and Control M notification is similar except that the Production Control person has less insight and experience in identifying and preparing a job for restart.

Informatica and Other Changes

The following action needs to occur before EM is used for scheduling a subject area instead of Informatica.

- 1) The Workflow in Informatica must be unscheduled and only run **on demand**.
- 2) Eliminate the wait at the start of the workflow
- 3) If you run with the Starttask command, all session must have "fail parent if the task fails", and NO task should have the "fail parent if the task does not run".
- 4) Request the CMEM trigger to the mainframe to be stopped or dropped.
- 5) Eliminate the FTP trigger step or job to the UNIX box. Instead this will be an out condition that controls the triggering of the Informatica sessions.
- 6) Send in the EM Schedule Modification Forms to SCHEDPRO.

- 7) Create the scripts on the AX-DEV-A08 box that are needed to support the schedule. The script must be executable by permission.
- 8) Have a quick review meeting between the programmers, production support and scheduling.
- 9) A job flow in VISO that provides a picture view of the job flow.

Example Scripts

Below are some example scripts examples for production

A starttask that has the ability to run on the failover server

```
#####
# Job Name
# HCAP2 INF30 workflow
# Description Run HCAP2 INF30 workflow session
#####
./mart_install/info/scripts/ABC/pm_setup_env_vars.ksh
cd $PM_PCHOME
$PM_PCHOME/pmcmd starttask -s ofis:4003 -uv PM_REPUSER -pv PM_REPPWD -wait -f 'BOSS - OAKS
Interface Project' -w wfl_INF30_HCAP2 s_INF30_HCAP2_CREATE_BATCH_ID
```

A startworkflow command that has the ability to run on the fallover server

```
#####
# Job Name OYDP006D.sh
# Business Unit SACWIS
# Description Delete data for valid batches from OAKS staging tables
#####
./mart_install/info/scripts/ABC/pm_setup_env_vars.ksh
cd $PM_PCHOME
$PM_PCHOME/pmcmd startworkflow -s ofis:4003 -uv PM_REPUSER -pv PM_REPPWD -wait -f 'BOSS - OAKS
Interface Project' wfl_SACWIS_OAKS_TABLES_DELETE
```

The Data warehouse example of a warehouse startworkflow

```
./mart_install/info/scripts/ABC/pm_setup_env_vars.ksh

cd $PM_PCHOME
$PM_PCHOME/pmcmd startworkflow -uv PM_REPUSER -pv PM_REPPWD
-s ax-dev-a08:4003 -wait -f 'DW – Food Stamp Accuracy Project' Wf_FDSTMP_PYMNT_ACCRCY_MART_LOAD
```

The difference between BOSS and Warehouse is the same except for server name. The warehouse uses the actual server and name. For testing you need to specify ax-dev-a16 instead of ax-dev-a08. The difference between starttask and startworkflow is that the session name is specified as the last field instead of the workflow name.

If you use starttask, you can only run one starttask within any given workflow. Therefore in many cases its best to setup the worklows to run with a startworkflow. Also startworkflw can better handle the link command.

Setting up E-mail notifications for Informatica sessions – Successful and Failed

SUCCESS

\$PMSuccessEmailUser

SUCCESSFUL : %s

ETL Subscribers,

Repository Name : %n

Folder Name : %d

Session %s completed successfully.

Session %b

Session %c

Session %i

%l

%r

Refer to the attached session log for LOAD SUMMARY to get individual target load statistics. %g

PLEASE DO NOT REPLY TO THIS E-MAIL.

Regards.

FAILURE

\$PMFailureEmailUser

FAILED : %s

ETL Subscribers,

Repository Name : %n

Folder Name : %d

Processing for session/job %s failed and more information on the cause of the problem is shown in the attached log file. %g

Session %b

Session %c

Session %i

PLEASE DO NOT REPLY TO THIS E-MAIL.

Regards.

Parameter Management – Quick Reference

This section summarizes the process of using parameters for any project folder in Informatica PowerCenter:

Function	Detailed Description	Frequency	Role
Encrypt PMADMIN password.	<ul style="list-style-type: none"> Logon as PMSERVER (mart7adm). Use “pmpasswd mypwd” command to determine encrypted password for PMADMIN. Use the returned value to assign to \$PM_REPPWD when setting up the profile. 	Once-off	PowerCenter Administrator
Update .profile for PMSERVER	<ul style="list-style-type: none"> Logon to PMSERVER and edit the profile in /home/PMSERVER where PMSERVER → Informatica Server Manager (mart7adm). Ensure that the profile calls the script Logoff PMSERVER and log on again using “su –” to ensure the profile is applied. Stop and restart all the Informatica daemons. 	Once-off	PowerCenter Administrator
Create project folder	<ul style="list-style-type: none"> Define necessary user roles and groups using Repository Manager Create folder and apply security standards Logon as PMSERVER and . Update parameter control table (ABC_SUBJECT_AREA) with the Development Group and Folder Abbreviation. Notify the project team of the Development Group and Folder Abbreviation to be used for the sessions in the folder. 	On request of a new project folder.	PowerCenter Administrator
Setup necessary folder level parameters in control tables.	<ul style="list-style-type: none"> Define DB Schema/Table Creator parameters for the project folder by updating parameter control table (ABC_PARM_LIST). 	On request (<i>when creating project folder</i>)	PowerCenter Administrator and Developer
Develop mappings and define necessary mapping parameters and variables.	<ul style="list-style-type: none"> Define necessary parameters and defaults in the mapping, including DB Schema/Table Creator parameters and refer to them in the mapping. Verify that all SQL or reference to relational tables are prefixed by DB Owner parameter (e.g. \$\$DB2TableOwner01.DM_MONTH) 	On-going	Developer
Verify necessary mapping level parameters in control tables.	<ul style="list-style-type: none"> Run script pm_setup_std_parms.ksh in UNIX directory \$PM_SCRIPTS. This will automatically extract all mapping parameters from repository into control tables. 	On request	PowerCenter Administrator and Developer
Develop sessions and workflows	<ul style="list-style-type: none"> Verify that all directories are suffixed by Development Group and Subject Area Code (folder abbreviation)(e.g. \$PMSessionLogDir\DSS5\OHP_CS) Verify that all Database Connections are referenced by a parameter. (e.g. \$DBConnectionSource01 or \$DBConnectionTarget01) Verify that all sessions use the parameter file named \$PMRootDir/folder_abbr.par Verify that all PRE- and POST-session commands use UNIX environment and/or session parameters. 	On-going	Developer

Create parameter file for a folder.	<ul style="list-style-type: none"> • Create a special workflow that will execute the following processes: <ul style="list-style-type: none"> ○ Run \$PM_SCRIPTS/ABC/pm_setup_std_params.ksh ○ Run \$PM_SCRIPTS/ABC/pm_upd_folder_params.ksh <i>folder_abbr</i> (only when there's a mapping defined to define and update the folder specific parameters. ○ Run \$PM_SCRIPTS/ABC/pm_build_parm_file.ksh <i>folder_abbr</i> • Run workflow to create parameter file for the folder. 	On-going (<i>before starting workflows and sessions</i>)	Developer
Unit test sessions or workflows	<ul style="list-style-type: none"> • No special consideration is required. 	On-going	Developer
Setup EM Scheduler configuration files	<ul style="list-style-type: none"> • Define the process dependencies and label the processes as per defined EM Scheduler Standards. • Prepare for EM Scheduler Setup: <ul style="list-style-type: none"> ○ Follow EM Schedule approval procedure. ○ Provide picture view of the job flow. ○ Setup review session with Informatica Administrator. • For each labeled process, define a configuration file that will identify the particular task to be performed as discussed in Section xxxxx. The configuration file has the following format: <ul style="list-style-type: none"> ○ JOB_TYPE=SCRIPT or WORKFLOW or SESSION or LOADUTIL (<i>mandatory</i>) ○ DEV_GROUP=<<i>development group assigned (mandatory)</i>> ○ PROJECT_ABBR=<<i>project/folder abbreviation assigned (mandatory)</i>> ○ WF_NAME=<<i>name of the Informatica Workflow (mandatory, if INFA session or workflow)</i>> ○ SESS_NAME=<<i>name of the Informatica Session (optional)</i>> ○ INFA_PARM_FILE=<<i>name of parameter file to be used (optional, if different from default)</i>> ○ SCRIPT_NAME=<<i>name of UNIX script to be executed(optional)</i>> ○ UTIL_NAME=<<i>name of Database utility to be executed (optional)</i>> 	On-going	Developer
Setup EM Scheduler Application	<ul style="list-style-type: none"> • The owner of the schedule is PMSERVER (mart7adm). • The path name is ALWAYS set to \$PM_SCRIPTS/ABC (or equivalent hard-coded – not recommended) • The task should ALWAYS be set as pm_start_process. • The parameter passed to the task should ALWAYS be the configuration filename defined above. • Define dependencies as per process flow diagram. 	On request schedule change	Development Lead and Informatica Administrator
Publish EM Scheduler Configuration files	<ul style="list-style-type: none"> • Place all the defined configuration files on the UNIX machine. <ul style="list-style-type: none"> ○ Location: \$PM_SCRIPTS/<project/folder abbreviation assigned> 	On request schedule change	Informatica Administrator

UNIX Environment List – Quick Reference

This section summarizes the list of UNIX environment variables that may be used by any project folder in Informatica PowerCenter.

Variable Name	Sample Value	Usage
\$PM_REPUSER	pmadmin	Repository Administrator Login (currently pmadmin)
\$PM_REPPWD	xmDe23Jj (<i>illustration only!</i>)	Repository Administrator Password (encrypted - see next paragraph on securing password)
\$PM_INFOBASE	/mart_install/info7	The main entry point to all the sub-directories required by Informatica toolset that will execute under PMSERVER login.
\$PM_SCRIPTS	\$PM_INFOBASE/sc ripts	Sub-directory with all the scripts required during processing.
\$PM_FILESYS	\$PM_INFOBASE/m art_dev	Sub-directory leading to the file system that hosts all the working directories required by the sessions and workflows. The 'mart_dev' represents the environment and should be changed to 'mart_test' or 'mart_prod' accordingly. This directory path is synonymous to the <u>\$PMRootDir</u> set for the particular server instance.
\$PM_PARMS	\$PM_FILESYS/Par ms	Sub-directory where the parameter files and the process control files are stored.
\$PM_FOLDER_LIST	\$PM_PARMS/INFA. Assigned.Folder.List	A list of folders which have been assigned a Development Group and a folder Abbreviation.
\$PM_REPHOME	\$PM_INFOBASE/re posit	Sub-directory where the Repository Server software was installed.
\$PM_PCHOME	\$PM_INFOBASE/pc	Sub-directory where the PowerCenter Server software was installed. Similar to PM_HOME in Powercenter terminology.
\$PM_PCHOST	`hostname`	The machine name hosting the PowerCenter daemons. Note the type of quote used when defining this variable. Example, AX-DEV-A08.
\$PM_PCPORT	4003	The machine port from which the PowerCenter Server daemon is listening for instructions. Set at installation time.
\$ORACLE_HOME	Oracle Clint Home Directory	Sub-directory where the Oracle Client Software is installed on the machine.
\$DB2_HOME	DB2 Client Home Directory	Sub-directory where the DB2 Client Software is installed on the machine.

Standard Server Parameter List – Quick Reference

This section summarizes the list of standard server parameters that may be used by any project folder in Informatica PowerCenter.

Server Variable	Description
\$DBConnectionName	A connection parameter used to facilitate flexibility in the definition of the connections used by the sessions. The actual connection name will be set at runtime in the parameter file and will not require changing the session during migration to production. The Name may be replaced by any meaningful and descriptive suffix, for example, \$DBConnectionDW for Data Warehouse Connection.
\$PMRootDir	A root directory to be used by any or all other server variables. For ODJFS, this is the same as the \$PM_FILESYS defined in the environment variable.
\$PMSessionLogDir	Default directory for session logs. Defaults to \$PMRootDir/SessLogs.
\$PMBadFileDir	Default directory for reject files. Defaults to \$PMRootDir/BadFiles.
\$PMCacheDir	Default directory for the lookup cache, index and data caches, and index and data files. Defaults to \$PMRootDir/Cache.
\$PMTargetFileDir	Default directory for target files. Defaults to \$PMRootDir/TgtFiles.
\$PMSourceFileDir	Default directory for source files. Defaults to \$PMRootDir/SrcFiles.
\$PMExtProcDir	Default directory for external procedures. Defaults to \$PMRootDir/ExtProc.
\$PMTempDir	Default directory for temporary files. Defaults to \$PMRootDir/Temp.
\$PMWorkflowLogDir	Default directory for workflow logs. Defaults to \$PMRootDir/WorkflowLogs.