Module 19 Suspect Duplicate Processing

IBM InfoSphere Master Data Management Fundamentals

Module Objectives

After completing this topic, you should be able to explain:

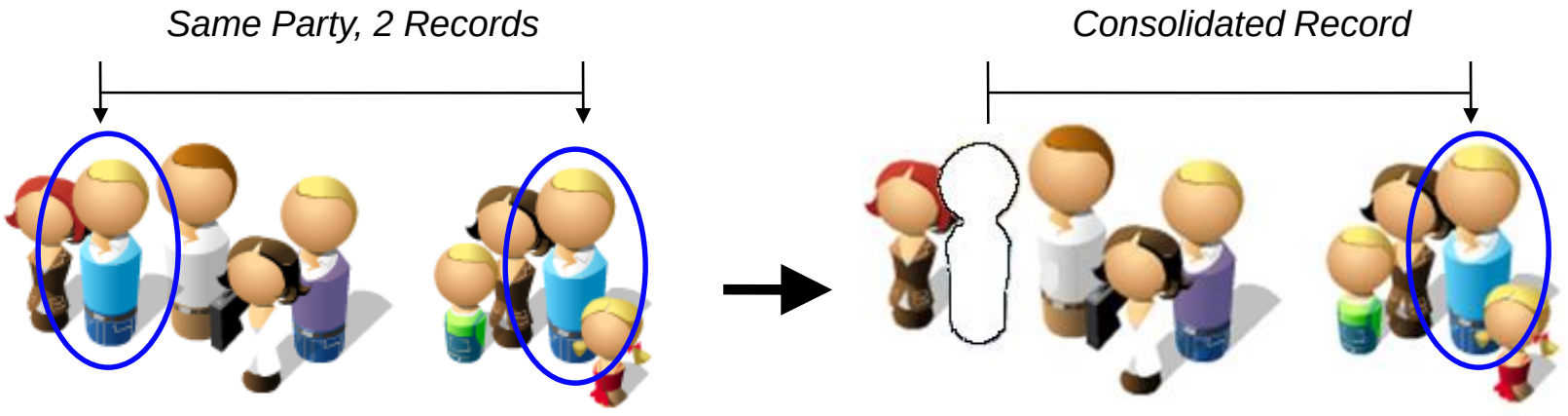
- What & Why & When Suspect Duplicate Processing
- Different MDM Matching Styles & Engines
- Other Matching Engine Options
- Different Search Styles & Engines
- Undo Collapse, Party Lineage and History

Learning Objectives

- Investigating Suspect Duplicates
  - What & Why & When Suspect Duplicate Processing
  - Different MDM Matching Styles & Engines
    - Deterministic matching
    - Probabilistic matching
    - Other Matching Engine Options
  - Undo Collapse, Party Lineage and History
  - Different Search Styles & Engines (ILO Optional)

What is Suspect Duplicate Processing

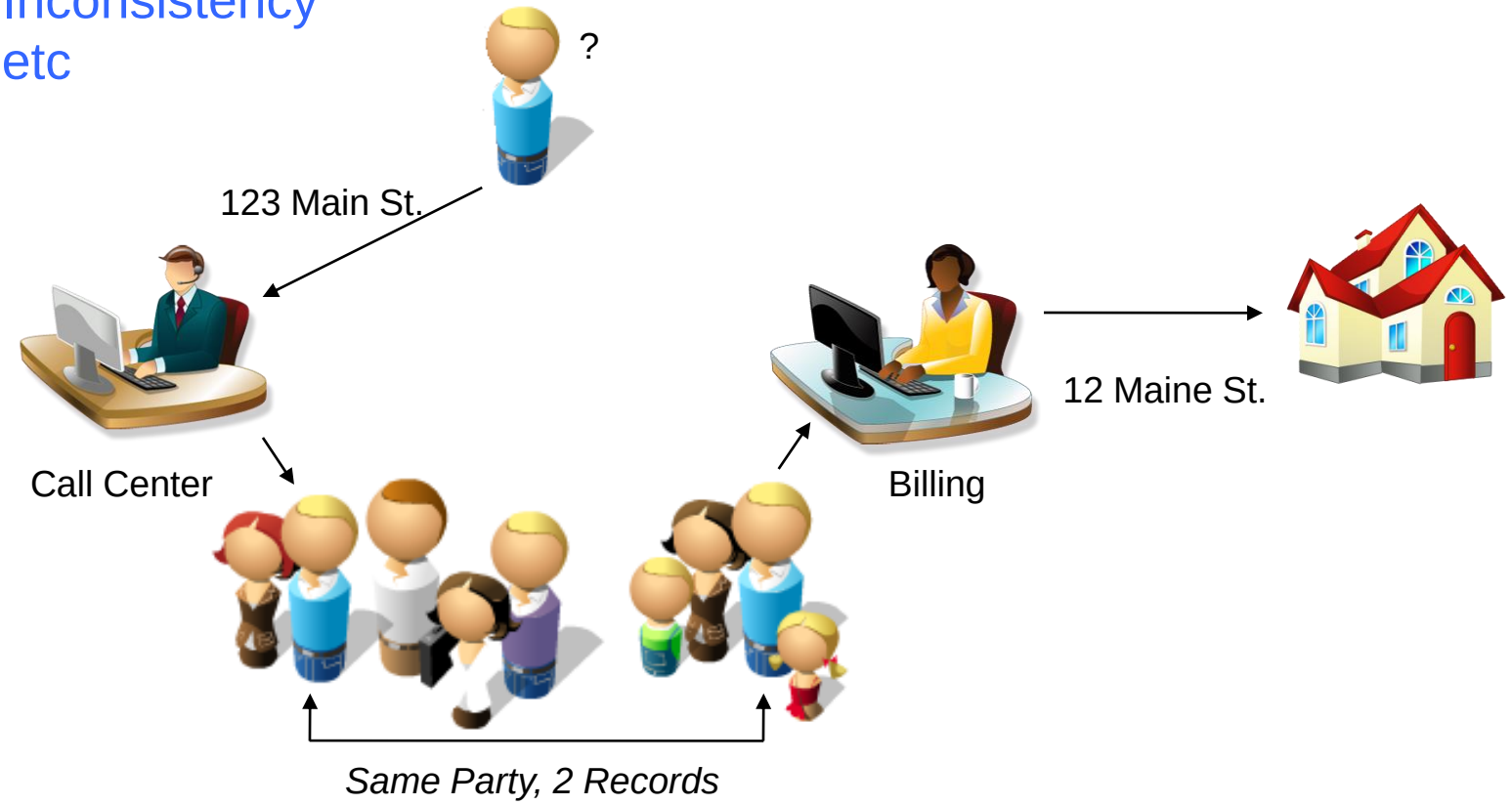
- The process of determining and handling multiple records that may represent the same entity (suspects) to maintain the quality of data



Rick Smith	Richard Smyth	Richard Smyth and Rick Smith
123 Main Street	123 Maine Street	123 Maine Street
DOB: Jan 4, 1967	DOB: April 1, 1967	DOB: April 1, 1967 or Jan 4, 1967 (Requires Investigation)
Gender: M	Gender: M	Gender: M

Why Suspect Duplicate Processing is needed

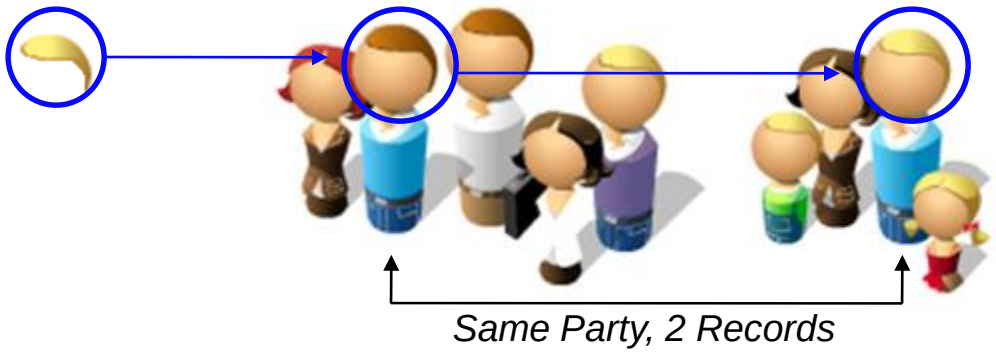
- Duplicates create data issues:
  - Data redundancy
  - Synchronization of records
  - Inconsistency
  - etc



When Suspect Duplicate Processing occurs

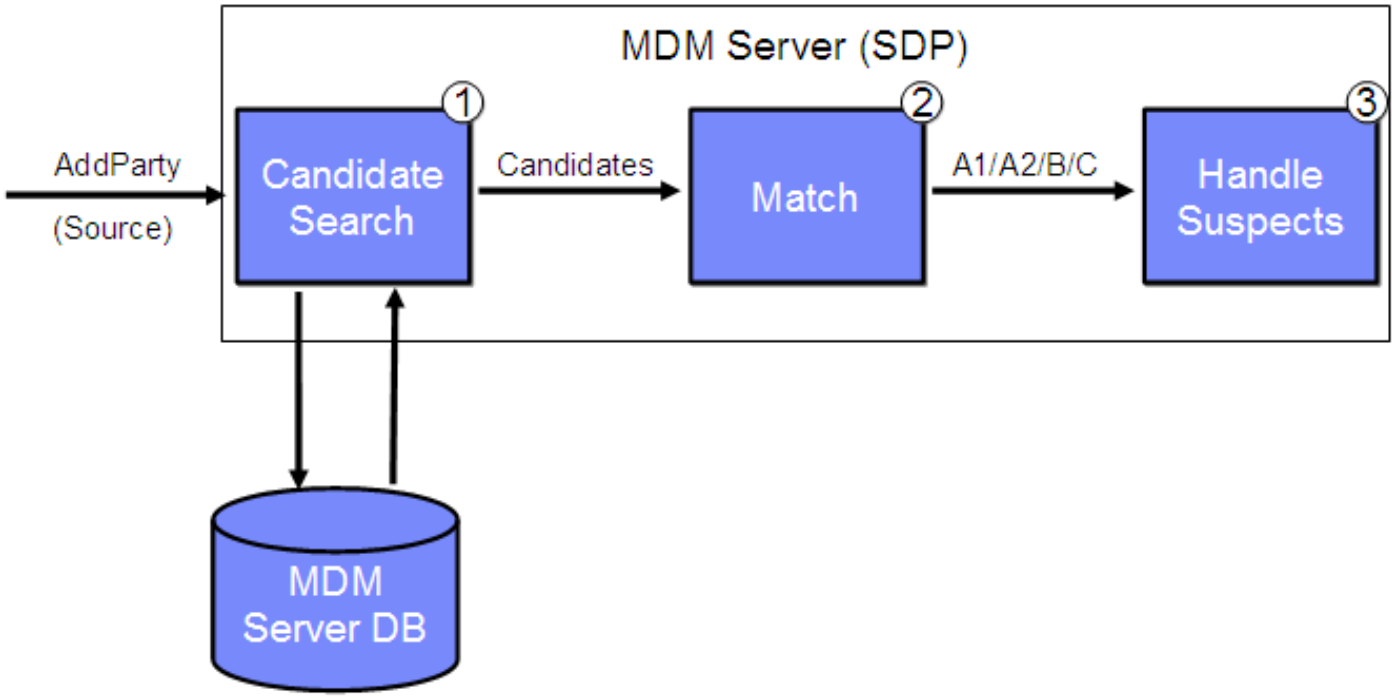
- Adds (record could already exist)

- Updates (new updated information could help identify the duplicate)



Suspect Duplicate Processing

- Searching and matching – identifies and determines which suspect categorization each candidate match belongs to
- Survivorship – determines how to handle suspect records



Learning Objectives

- Investigating Suspect Duplicates
  - What & Why & When Suspect Duplicate Processing
  - Different MDM Matching Styles & Engines
    - Deterministic matching

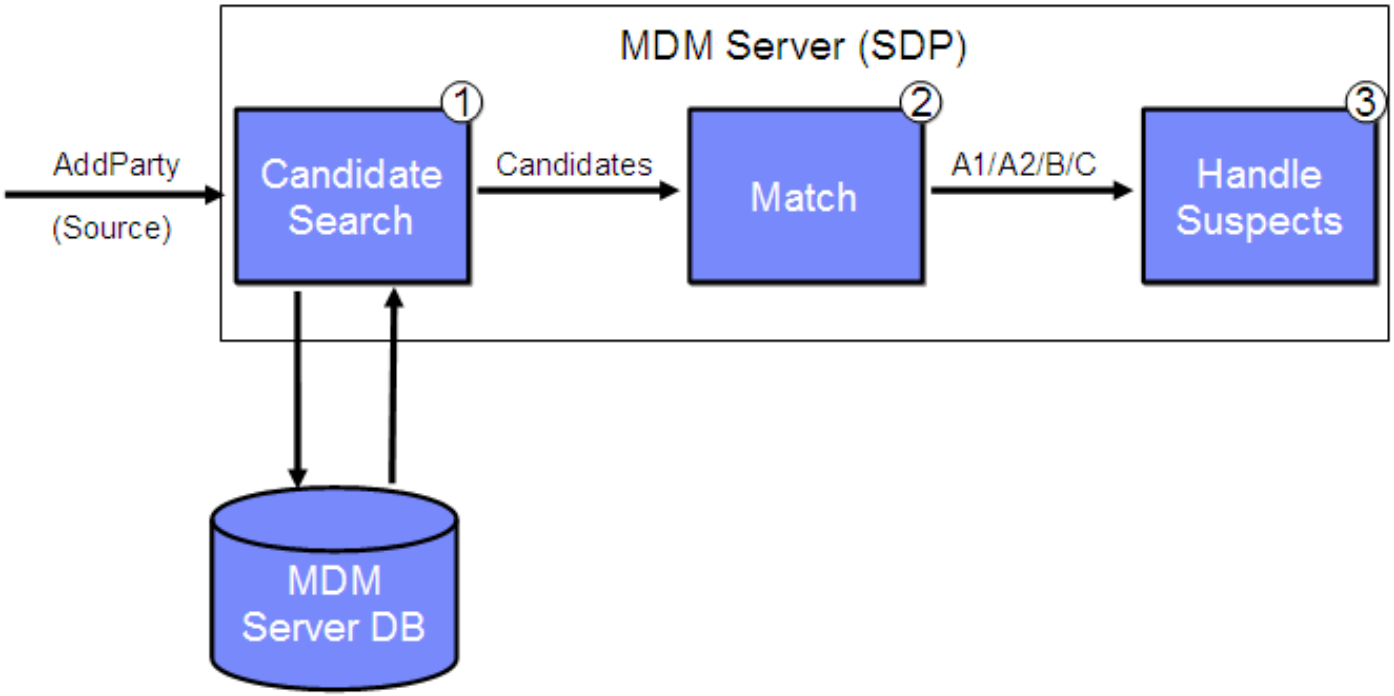
      MDM Classic (Deterministic) Matching Engine
    - Probabilistic matching
    - Other Matching Engine Options
  - Undo Collapse, Party Lineage and History
  - Different Search Styles & Engines (ILO Optional)

Deterministic Matching Style

- Compares the set of values for all of a given party's critical data elements with those of another
- Compares the presence, absence, and content of the values
- Results in a matching score and a non-matching relevancy score
- Provided by MDM Classic Matching Engine

Using MDM Classic (Deterministic) Matching Engine

- Identifies suspected duplicate parties searching and comparing the set of values for all of a party's ***critical data elements*** with those of another party
- Indicate the likelihood of a matching using match and non-match relevancy scores

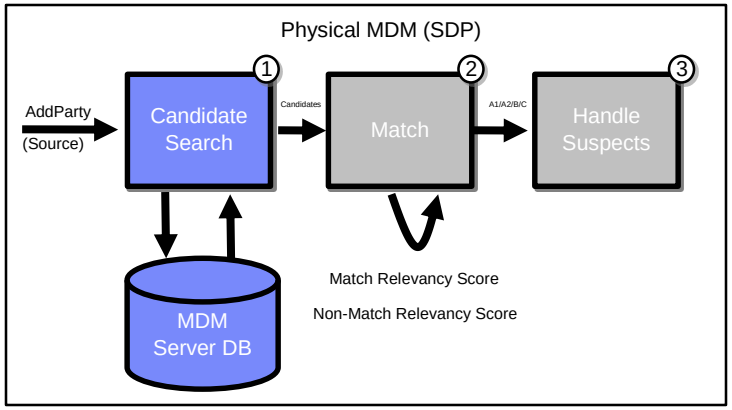


Critical Data Elements

- Key elements of party information
- SDP is driven by changes to critical data
- Can be customized to match business requirements
- Person critical data elements
  - Last name
  - Given name one
  - Date of birth
  - Address
  - Social Security Number
  - Gender
- Organization critical data elements
  - Name
  - Address
  - Tax Identification Number

1. Candidate Search

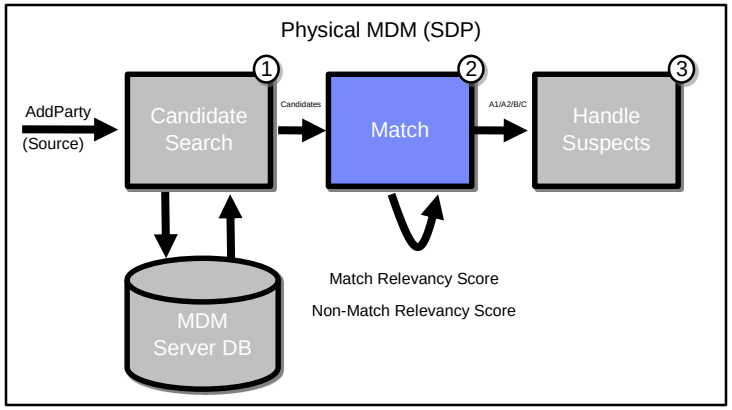
- Purpose: To create a subset of parties that may be matches to the Source Party (Party being added or updated)
- Suspect searches are done using a subset of the defined critical data elements or combinations of that subset
- External Rule (Rule Id = 3)
  - PartySuspectSearchRule



Default Behaviour for Person	Default Behaviour for Org
•Search Person by Address •Search Person by ID	•Search Org by Name •Search Org by Address •SearchOrg by ID

2. Matching

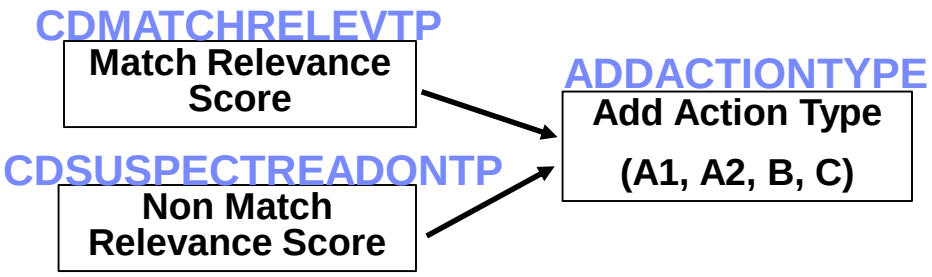
- Purpose: To compare candidates to the Source Party and determine how close they are to a match (A1/A2/B/C)
- Configuration and Management element
  - /IBM/Party/SuspectProcessing/PartyMatcher/className
  - com.dwl.tcrm.coreParty.component.TCRMPartyMatcher
- Matcher find the following:
  - Match Relevancy Score: attributes that match
  - Non-Match Relevancy Score: attributes that do not match



Match Category	Description
A1	The parties are definitely the same.
A2	It is highly probable that the parties are the same. These are essentially guaranteed matches, but require a data steward to review a piece of critical data to confirm.
B	The parties are potentially the same.
C	The parties are definitely not the same.

2. Matching

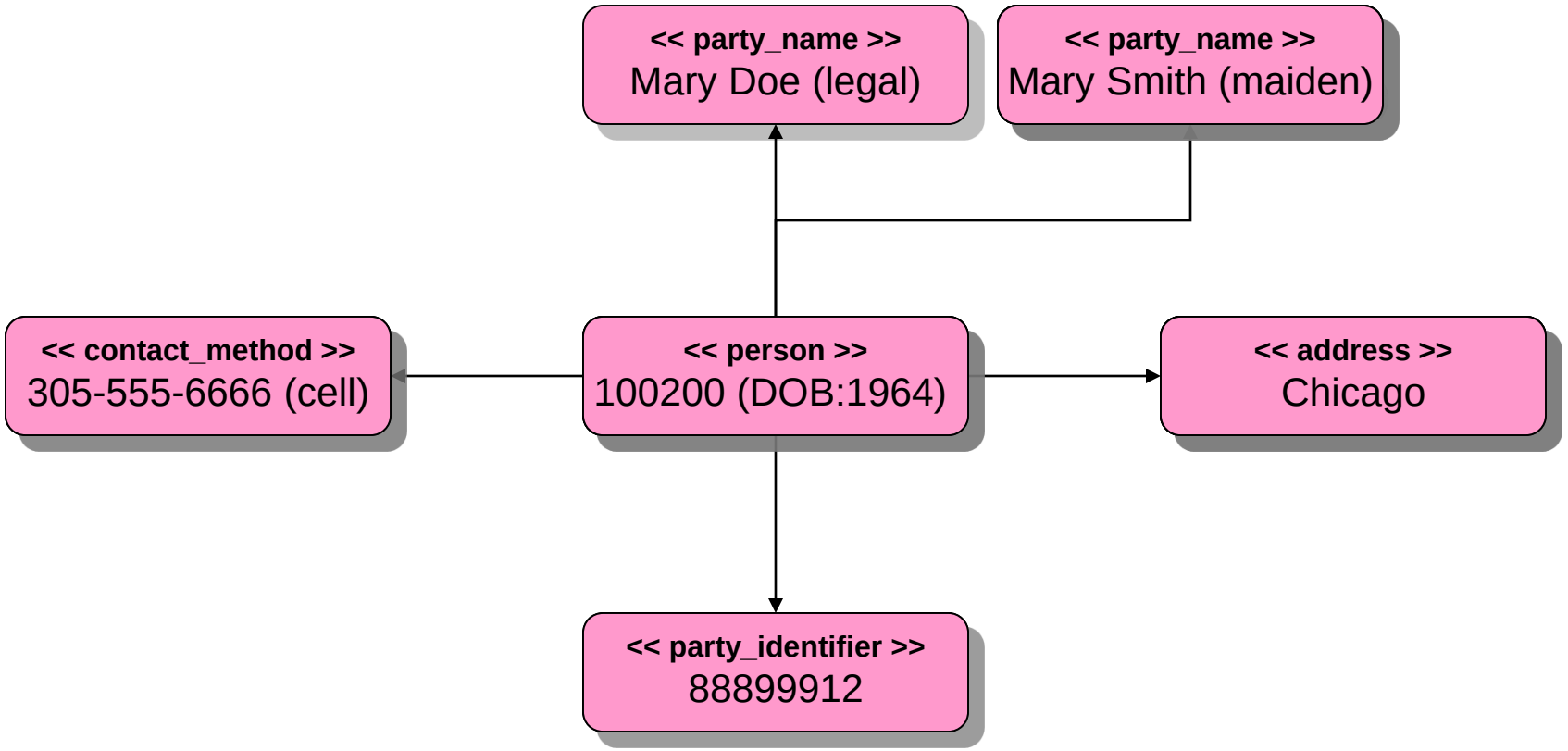
- Match Relevancy Score and Non Match Relevancy Score product Add Action Type (A1, A2, B, C) from ADDACTIONTYPE table



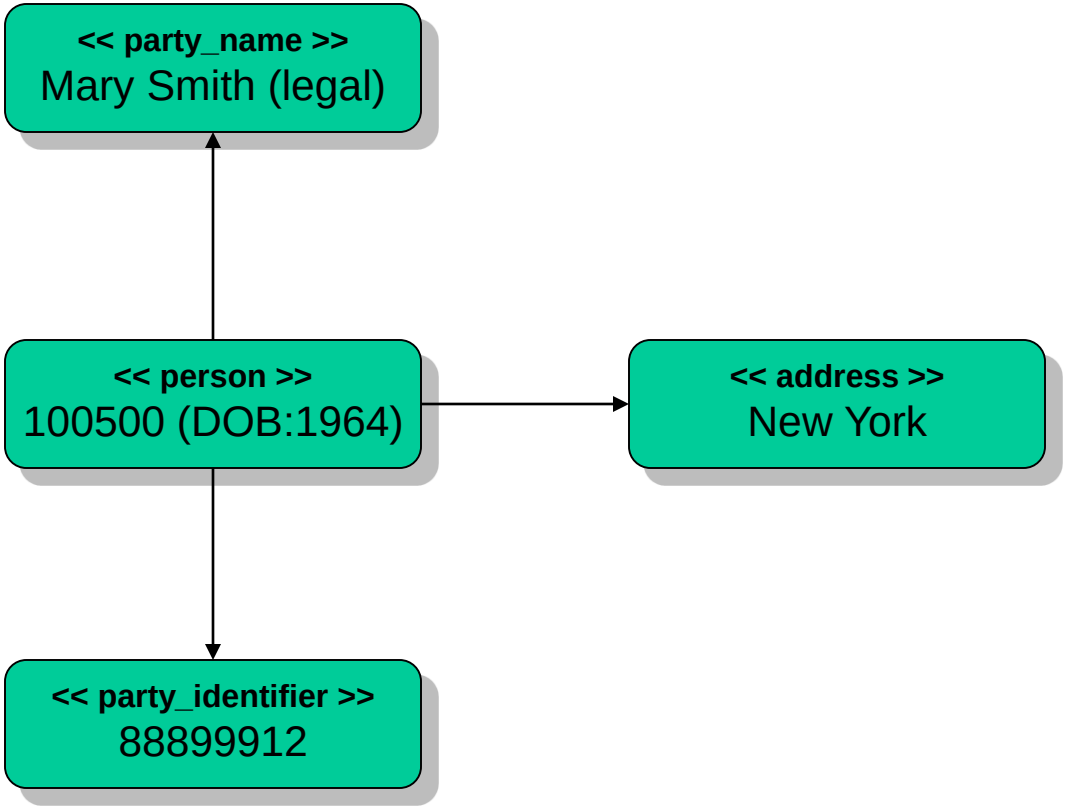
ADDACTIONTYPE

MATCH_RELEV_TP_CD	SUSP_REASON_TP_CD	ADD_ACTION_CODE	…
10	6	B	…
16	46	A1	…
…	…	…	…

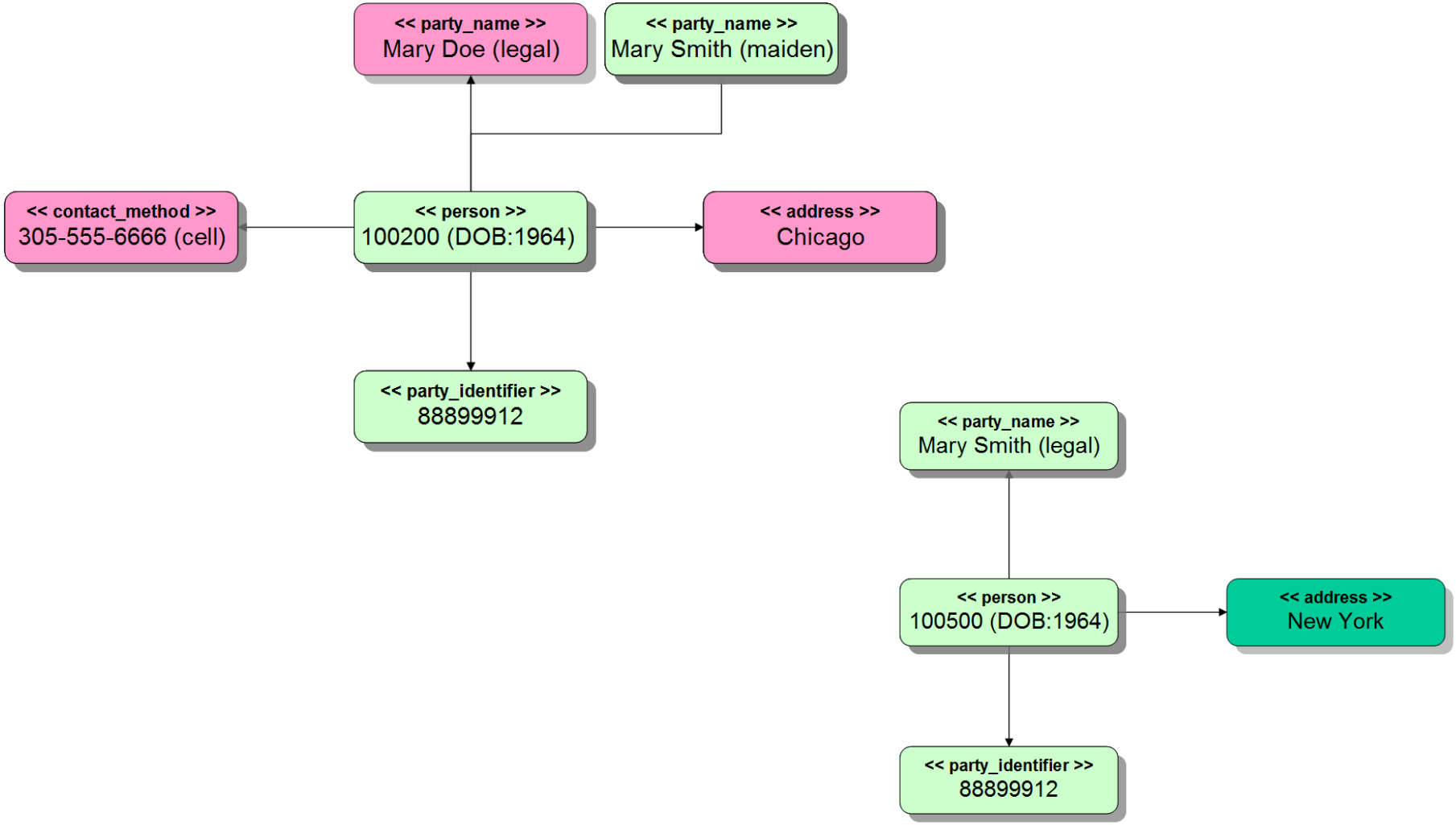
2. Matching – Example: Mary Doe



2. Matching – Example: Mary Smith



2. Matching – Example: Mary Doe & Mary Smith



2. Matching - Example

- Mary Doe & Mary Smith
- Attributes that match: Last name (SName), Given name one (G1Name), Date of birth (DOB), Social Security Number (SSN), Gender (G)
- Match Relevancy Score = 55
- Attributes that do not match: Address
- Non-Match Relevancy Score = 4
- Match Category = A2

Party Id	Family Name / Organization Name	Given Name	Party Type	Created Date	Best Match Detected	Task Status	Due Date	Priority	Owner
100200	Doe	Mary	Person	April 27, 2011					

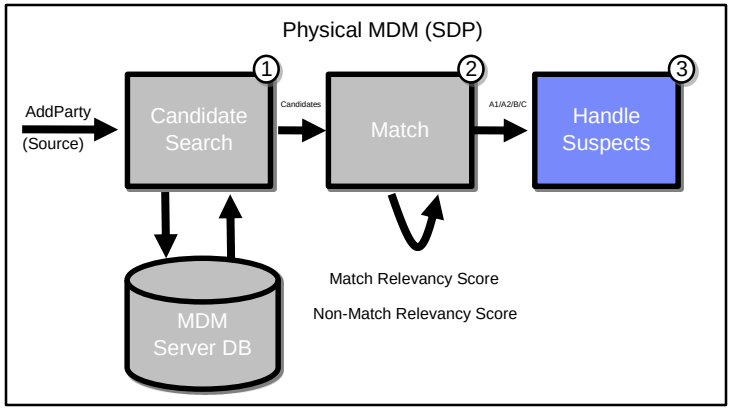
Suspects

Party Id	Family Name / Organization Name	Given Name	Match Score	Non-Match Score	Best Match	Match Reason	Modified Date	Suspect Status	Source	Match Category	Probal Score
100100	Doe	John	10	41		SName, Addr	April 27, 2011	Parties are Suspect Duplicates	System marked	C	
100500	Smith	Mary	55	4		G, G1Name, SName, DOB, SSN	May 6, 2011	Party Pending Critical Change	System marked	A2	

Page 1

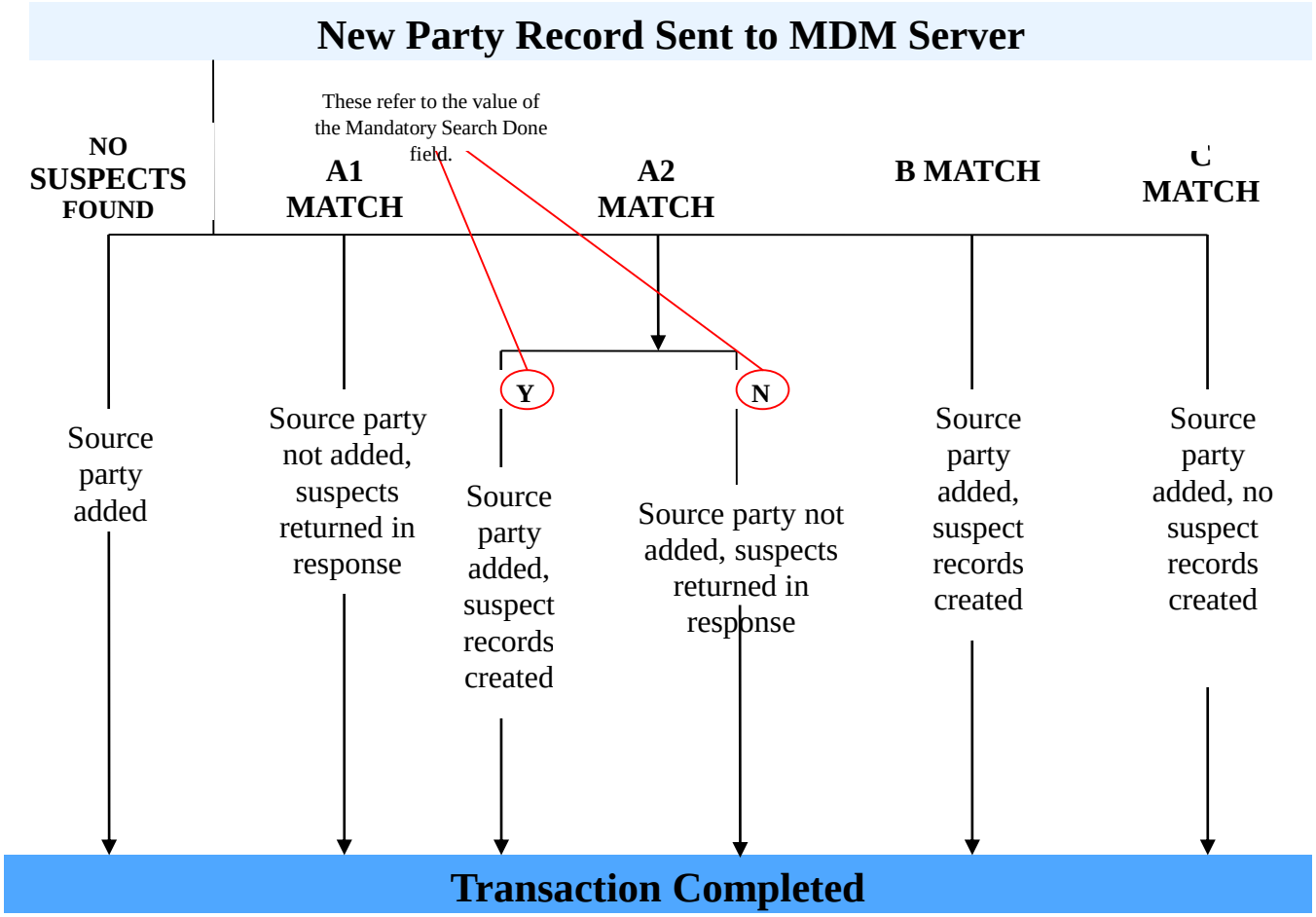
3. Handling Suspects

- Purpose
  - To collapse Parties if a match is found
  - To add a suspect record to the database if needs more investigation



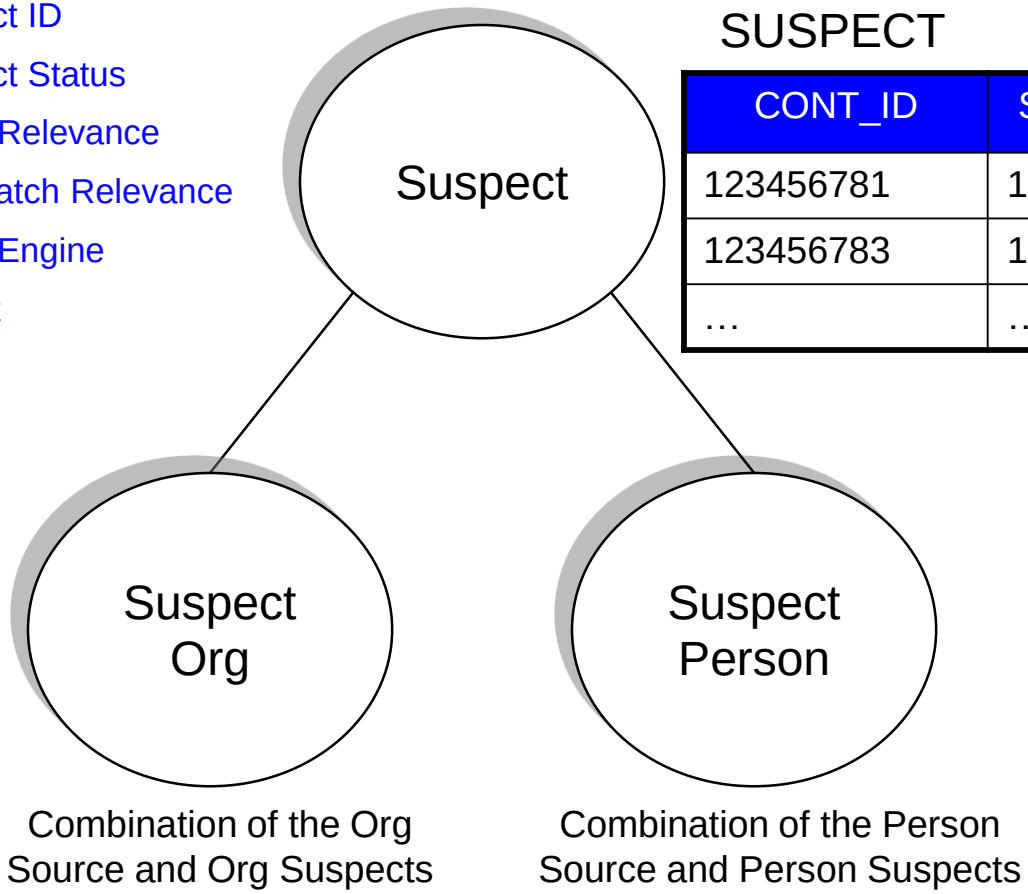
- External Rule:
  - A1SuspectsActionRule
  - A2SuspectsActionRule
  - BSuspectsActionRule
  - CSuspectsActionRule

3. Handling Suspects



Suspect Entity

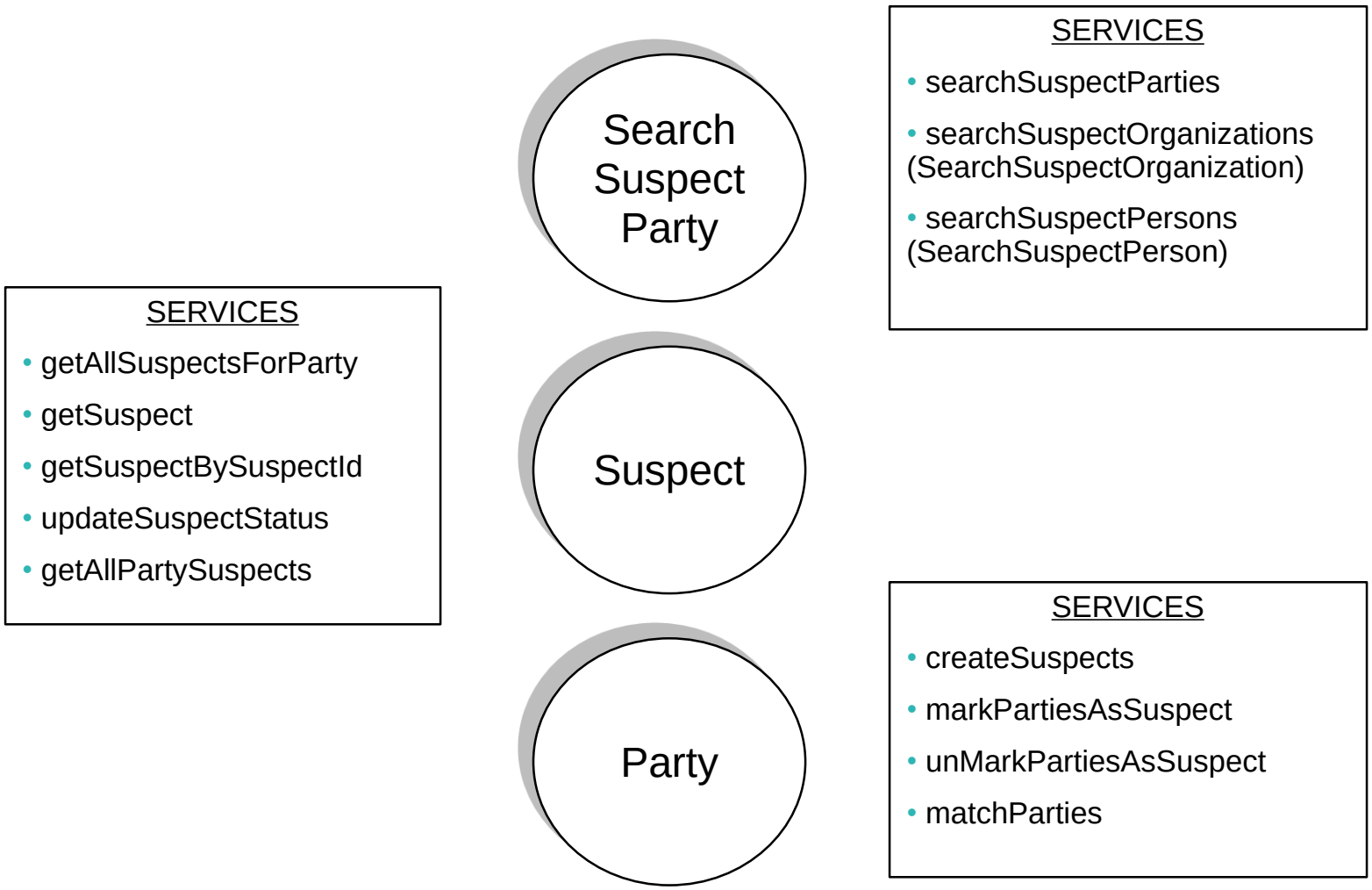
- Source Party ID
- Suspect ID
- Suspect Status
- Match Relevance
- Non Match Relevance
- Match Engine
- Weight
- …



SUSPECT

CONT_ID	SUSP_CONT_ID	Weight	…
123456781	124124	243	…
123456783	12412512	245	…
…	…	…	…

Meet the Services (Suspect Search)



Learning Objectives

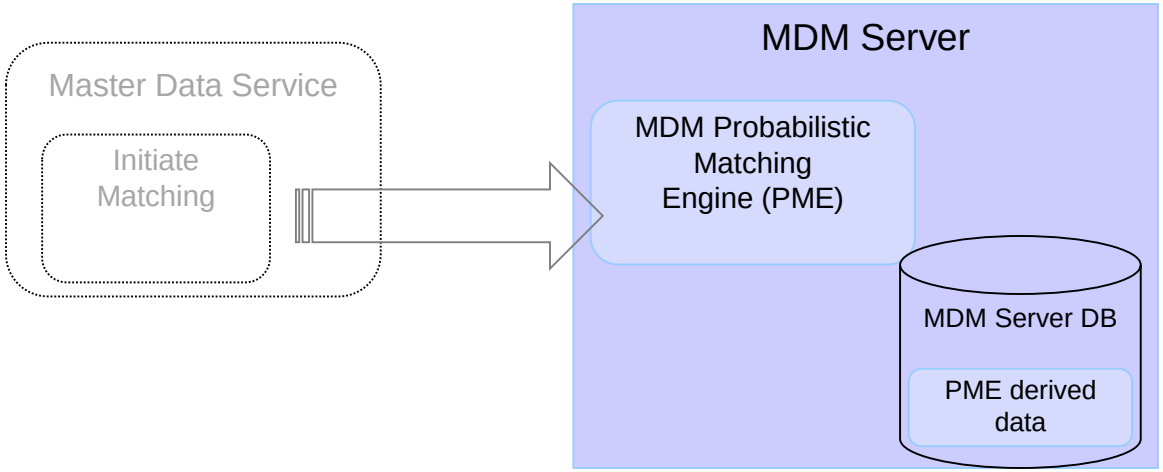
- Investigating Suspect Duplicates
  - What & Why & When Suspect Duplicate Processing
  - Different MDM Matching Styles & Engines
    - Deterministic matching
    - Probabilistic matching
      - InfoSphere MDM Probabilistic Matching Engine (default)
      - IBM InfoSphere QualityStage Matching Engine
    - Other Matching Engine Options
  - Undo Collapse, Party Lineage and History
  - Different Search Styles & Engines (ILO Optional)

Probabilistic Matching Style

- Compares a preconfigured set of attributes between two or more parties and derives a final score
- Generates scores that take into consideration the frequency of the occurrence of a data value
- Calculates only one composite weight score
- Provided by MDM Probabilistic Matching Engine and InfoSphere QualityStage matching engine

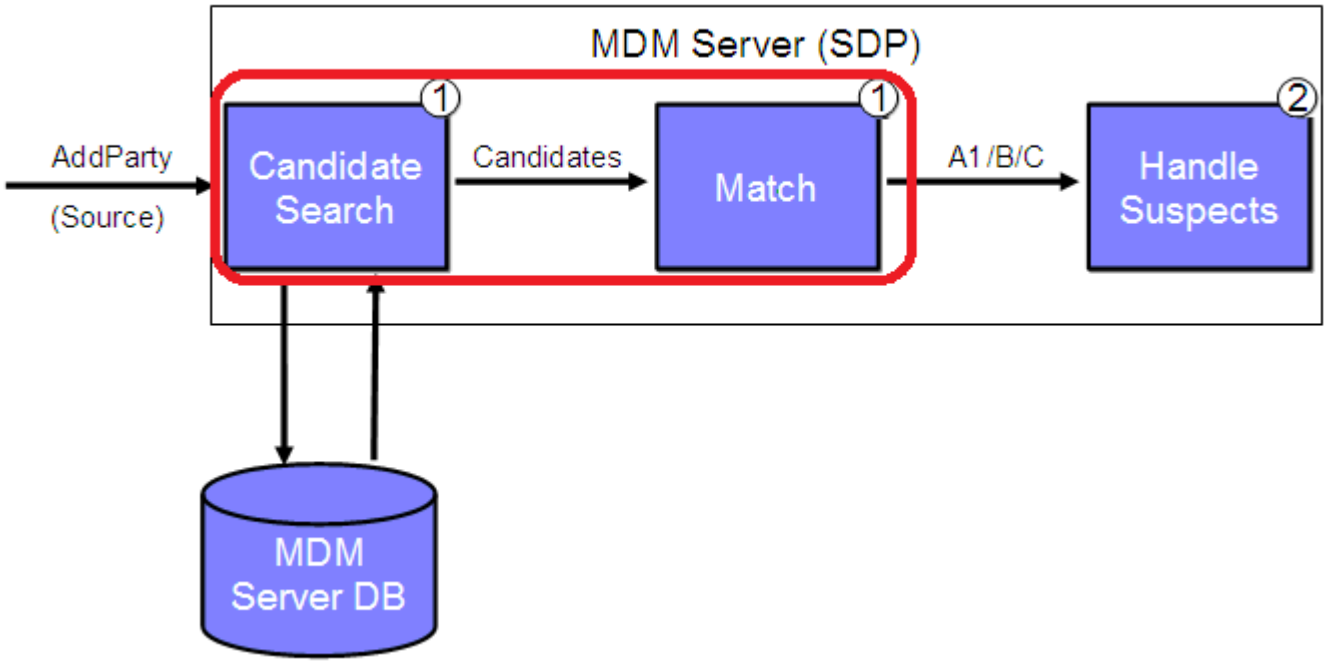
Using Probabilistic Matching Engine (PME)

- What is the Probabilistic Matching Engine (PME)?
  - an embedded, preconfigured and integrated version of the matching engine used by IBM Initiate Master Data Service (MDS)
  - the preconfigured default matching engine for new InfoSphere MDM installations.

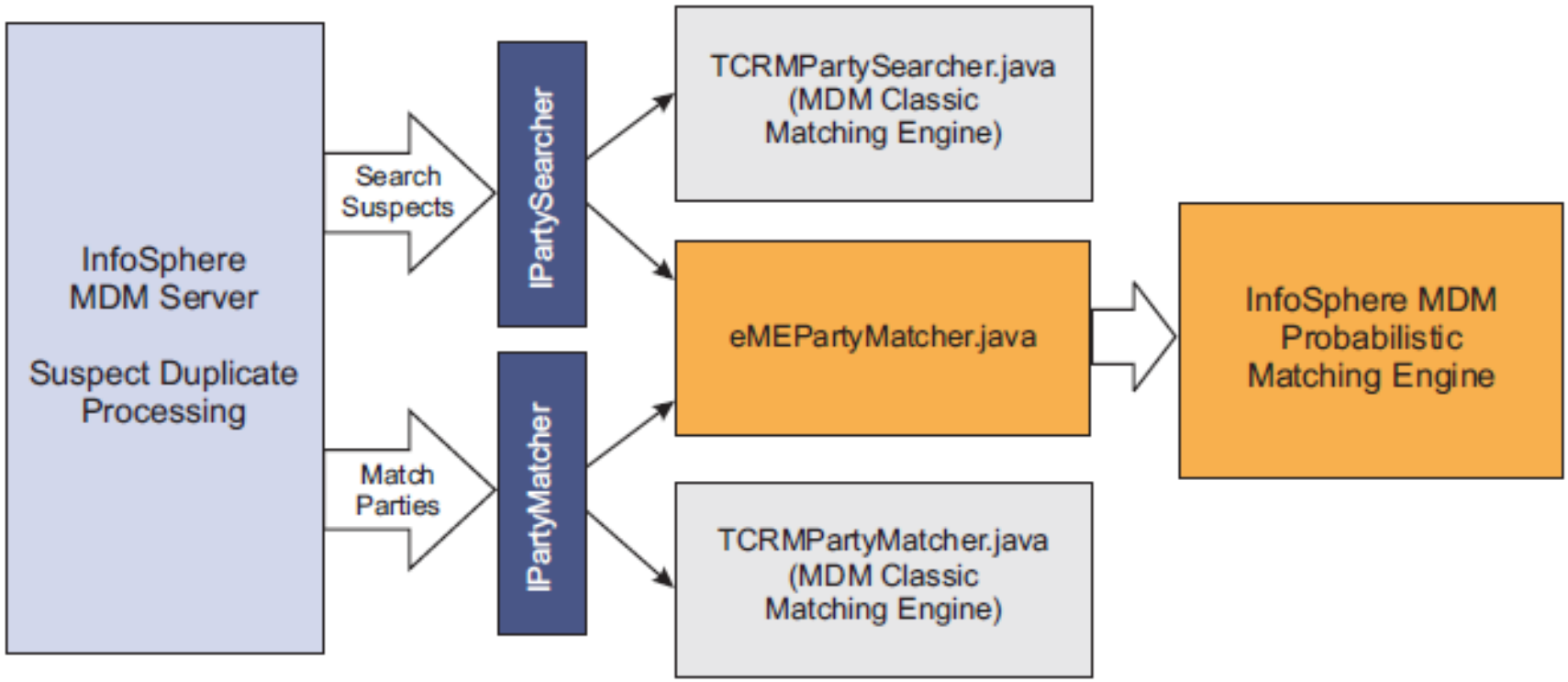


Using Probabilistic Matching Engine (PME)

- What is the value to the customer?
  - better matching outcomes
  - improved efficiency → suspect search and matching combined into one step
  - configurable state of the art algorithm



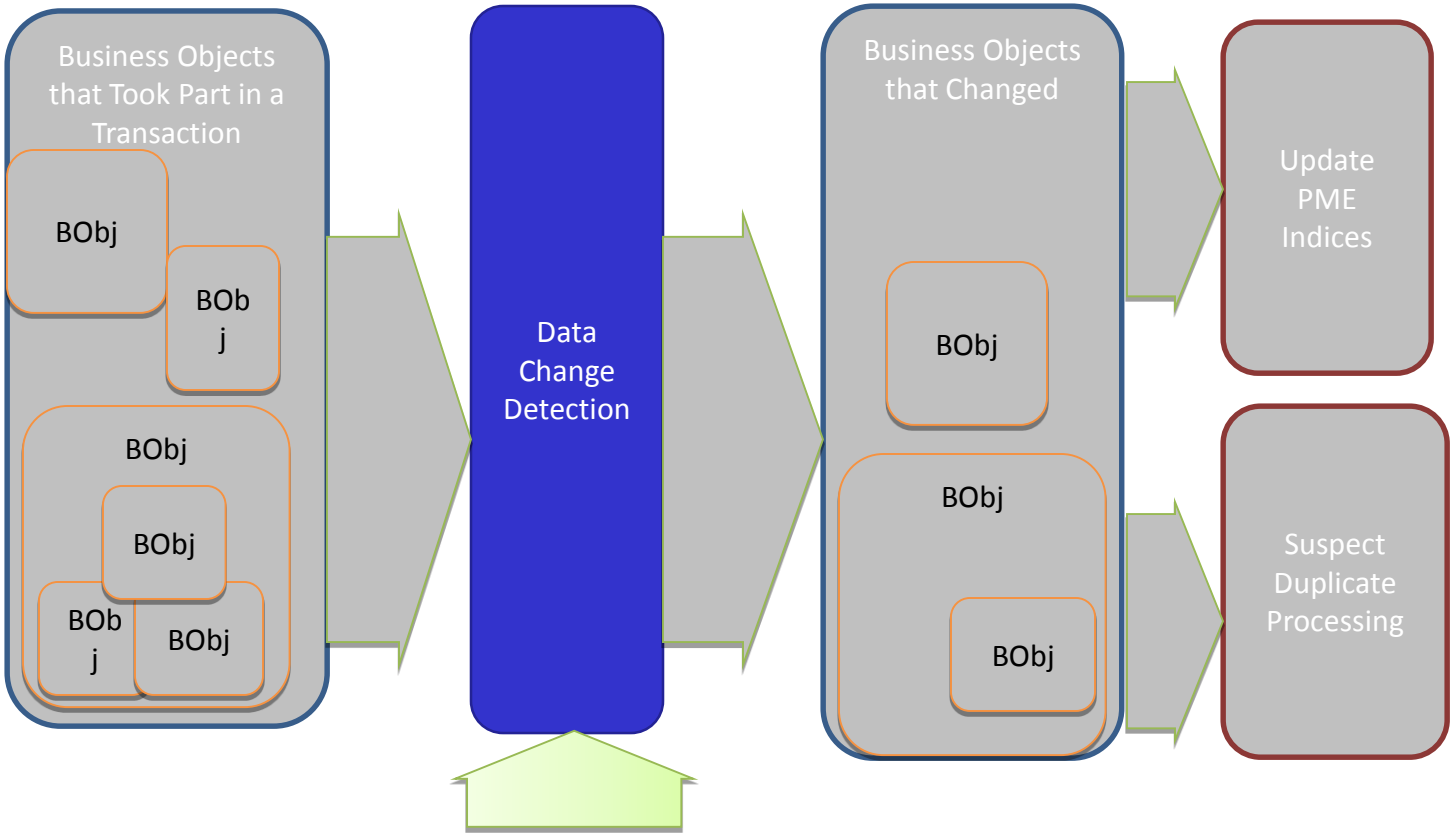
MDM Classic Matching Engine vs MDM PME



Using Probabilistic Matching Engine (PME)

- PME uses *a set of indices which are derived data* generated by PME algorithm to optimize matching Persons or Organizations based on a preconfigured set of attributes
- These indices must be *kept synchronized* with MDM Server data that indices must be updated if party data changes
- This synchronization relies on data change detection to determine if any changes that affect the indices have occurred
- The same mechanism that is used to detect data changes for SDP is used to detect changes for synchronization
- If the out-of-the-box configuration of the PME algorithm and data model is not appropriate for the business requirements, *MDM Workbench can be to use for the algorithm adjustment*

Data Change Detection



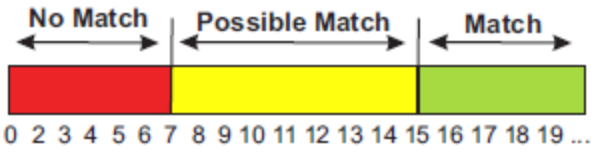
CRITICAL_ELEMENT_ID	APPLICATION	GROUP_NAME	ELEMENT_NAME	SYNCPURPOSE_TP_CD	ENTITY_TYPE	INSTANCE_PK	ACTIVE_IND	ULTIMATE_PARENT_GROUP_NAME
1	TCRM	Person	GenderType	1			Y	
2	TCRM	Person	BirthDate	1			Y	
3	TCRM	PersonName	GivenNameOne	1	NameUsageType	1	Y	Person
4	TCRM	PersonName	GivenNameTwo	1	NameUsageType	1	Y	Person
5	TCRM	PersonName	LastName	1	NameUsageType	1	Y	Person
6	TCRM	PersonName	GivenNameOne	1	NameUsageType	2	Y	Person
7	TCRM	PersonName	GivenNameTwo	1	NameUsageType	2	Y	Person
8	TCRM	PersonName	LastName	1	NameUsageType	2	Y	Person

- Critical data elements (CDE) are defined in the Critical data element table.
- CDE entries can be made Active/Inactive
- CDE entries can be supported for specific code types

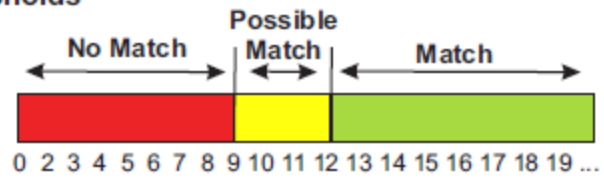
PME Matching Scores

- PME scoring algorithm compares a preconfigured set of attributes between two or more parties and derives a final score
- PME matching score is a numerical value which indicates the degree of the match and signifies the likelihood that any two parties are the same

Person thresholds

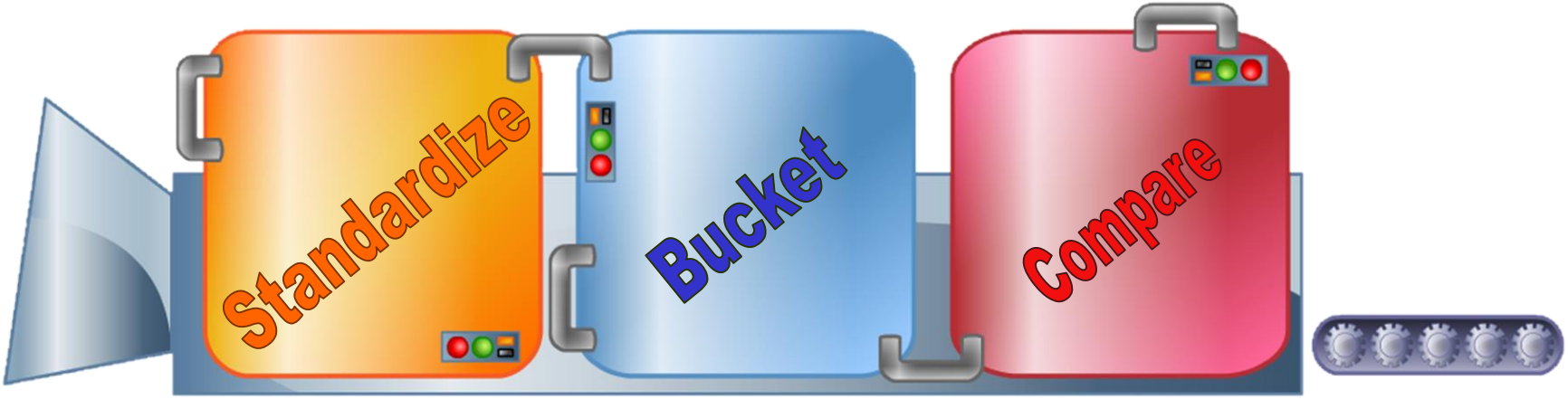


Organization thresholds



InfoSphere MDM Server Suspect Types	InfoSphere MDM Probabilistic Matching Engine Match Type Codes
A1	Match
A2	[not used]
B	Possible Match
C	Non-Match

PME Algorithm



Converts data to simplest form for easier use during matching process.

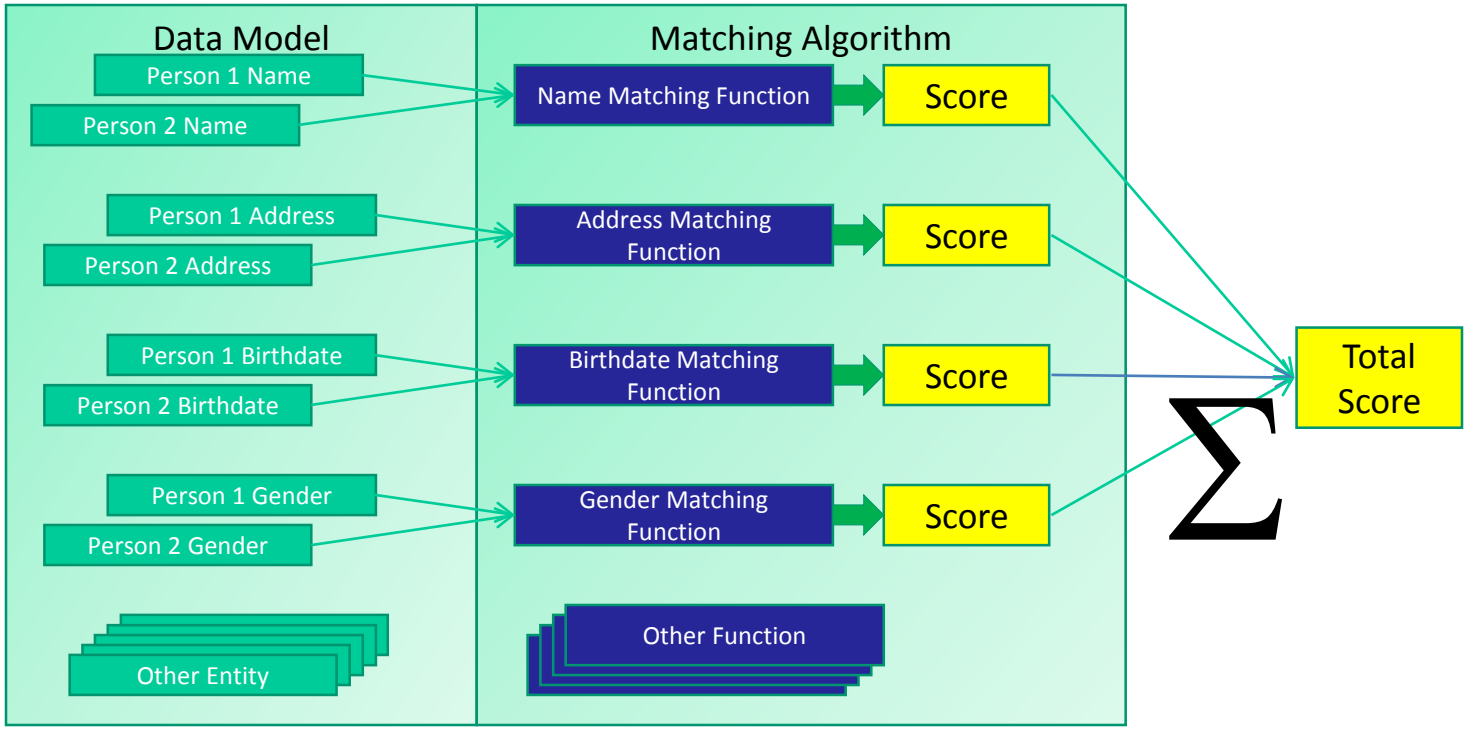
Organizes records that share common values for faster search retrieval.

Compares pairs of records using the probabilistic method to calculate a score.

Jim Smith ➔ SMITH:JIM ➔ JN + SNT ➔ Jim Smith vs. Jim Smith
➔ Jim Smith vs. James Smith
➔ Jim Smith vs. John Smith

PME Matching Algorithm

- The algorithm is the brain behind searching and matching
- The algorithm sends data through a series of three steps, which break the data down, organize the records, and compare data for overall match
- PME matching algorithm consists of a set of predefined functions that are applied to Physical MDM party attributes



PME Algorithm - Standardization

Original Data	Standardized Data
Maria R. Fontana	FONTANA:MARIA:R::.
391-20-1923	391201923
(832) 812-1193	8121193
208 W. Chicago Ave. Apartment 3 Oak Park, IL 60302	N-208:S-W:S-CHICAGO:S-AVE:S-APT:N-3:S-OAK:S-PARK:S-IL:N-60302
mfont91@us.ibm.com	MFONT91USIBM
1973-09-21	19730921

Standardized data is stored in the **EME_RECCMPD** table:

```
FONTANA:MARIA:R::.^391201923^8121193~8112915^N-208:S-W:S-CHICAGO:
S-AVE: S-APT:N-3:S-OAK:S-PARK:S-IL:N-60302.^MFONT91USIBM^19730921
```

PME Algorithm - Comparison

- Zodiac signs may hint at compatibility, but it's the score that counts

Compare Records | Total Score: 23.6				
Comparison Code	BROSE OLIVER PERRY, HOS:91881028	BROSE O PERRY, CL:435202	Weight	Match Code
XNM	BROSE|OLIVER|PERRY|	BROSE|O|PERRY|	6.9	Partial
PERADDR	5520|BUCHANAN|ST|APACHE|JUNCTION|AZ|85217|	5520|BUCHANAN|ST|APACHE|JUNCTION|AZ|85217|	5.95	Equal
DOB	19430922	19430922	4.34	Equal
SSN	217824877	217824877	6.14	Equal
IDN			0.0	Missing
BUSADDR			0.0	Missing
SEX	M	M	0.35	Equal

PME Algorithm - Comparison

- Zodiac signs may hint at compatibility, but it's the score that counts
  - Comparisons issue a weight for each attribute

Compare Records | Total Score: 23.6				
Comparison Code	BROSE OLIVER PERRY, HOS:91881028	BROSE O PERRY, CL:435202	Weight	Match Code
XNM	BROSE|OLIVER|PERRY|	BROSE|O|PERRY|	6.9	Partial
PERADDR	5520|BUCHANAN|ST|APACHE|JUNCTION|AZ|85217|	5520|BUCHANAN|ST|APACHE|JUNCTION|AZ|85217|	5.95	Equal
DOB	19430922	19430922	4.34	Equal
SSN	217824877	217824877	6.14	Equal
IDN			0.0	Missing
BUSADDR			0.0	Missing
SEX	M	M	0.35	Equal

PME Algorithm - Comparison

- Zodiac signs may hint at compatibility, but it's the score that counts
  - Comparisons issue a weight for each attribute
  - Individual weight scores come from lookup tables

Compare Records | Total Score: 23.6

Comparison Code	BROSE OLIVER PERRY, HOS:91881028	BROSE O PERRY, CL:435202	Weight	Match Code
XNM	BROSE|OLIVER|PERRY|			Partial
PERADDR	5520|BUCHANAN|ST|AP		5	Equ
DOB	19430922		.34	Equal
SSN	217824877		.14	Equal
IDN			0.0	Missing
BUSADDR			0.0	Missing
SEX	M	M	0.35	Equal

XNM Detail

CL:435202	HOS:91881028	Weight	Match Code
BROSE	BROSE	4.4	Equal Word
O	OLIVER	0.5	Initials matches first character of word
PERRY	PERRY	3.21	Equal Word

PME Algorithm - Comparison

- Zodiac signs may hint at compatibility, but it's the score that counts
  - Comparisons issue a weight for each attribute
  - Individual weight scores come from lookup tables
  - Weights are aggregated into the total score

23.6 is above the Threshold of 15.0

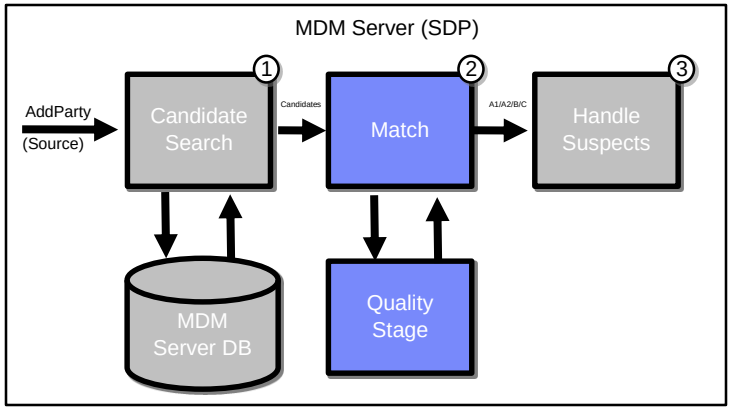
Compare Records | Total Score: 23.6				
Comparison Code	BROSE OLIVER PERRY, HOS:91881028	BROSE O PERRY, CL:435202	Weight	Match Code
XNM	BROSE|OLIVER|PERRY|	BROSE|O|PERRY|	6.9	Partial
PERADDR	5520|BUCHANAN|ST|APACHE|JUNCTION|AZ|85217|	5520|BUCHANAN|ST|APACHE|JUNCTION|AZ|85217|	5.95	Equal
DOB	19430922	19430922	4.34	Equal
SSN	217824877	217824877	6.14	Equal
IDN			0.0	Missing
BUSADDR			0.0	Missing
SEX	M	M	0.35	Equal

Learning Objectives

- Investigating Suspect Duplicates
  - What & Why & When Suspect Duplicate Processing
  - Different MDM Matching Styles & Engines
    - Deterministic matching
    - Probabilistic matching
      - InfoSphere MDM Probabilistic Matching Engine (default)
      - IBM InfoSphere QualityStage Matching Engine
    - Other Matching Engine Options
  - Undo Collapse, Party Lineage and History
  - Different Search Styles & Engines (ILO Optional)

IBM InfoSphere QualityStage Matching Engine

- Probabilistic Matching engine QualityStage
  can be configured to be the Matcher
  - Weighted scoring system
  - Determines A1, A2, B, or C
  - Various Algorithms
    - Character comparison
    - Uncertainty character comparison
    - Data comparison
    - etc



- Configuration and Management element
  - /IBM/Party/SuspectProcessing/PartyMatcher/className
  - com.ibm.mdm.thirdparty.integration.iis8Adapter.InfoServerPartyMatcherAdapter

Learning Objectives

- Investigating Suspect Duplicates
  - What & Why & When Suspect Duplicate Processing
  - Different MDM Matching Styles & Engines
    - Deterministic matching
    - Probabilistic matching
    - Other Matching Engine Options
  - Undo Collapse, Party Lineage and History
  - Different Search Styles & Engines (ILO Optional)

Other Matching Engine Options

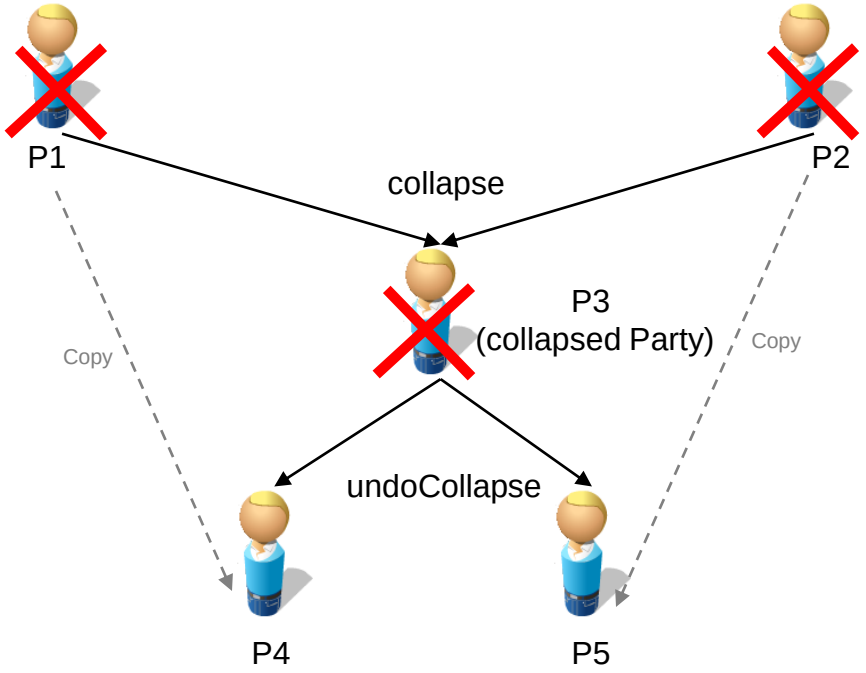
- Trillium
- Custom

Learning Objectives

- Investigating Suspect Duplicates
  - What & Why & When Suspect Duplicate Processing
  - Different MDM Matching Styles & Engines
    - Deterministic matching
    - Probabilistic matching
    - Other Matching Engine Options
  - Different Search Styles & Engines
  - Undo Collapse, Party Lineage and History

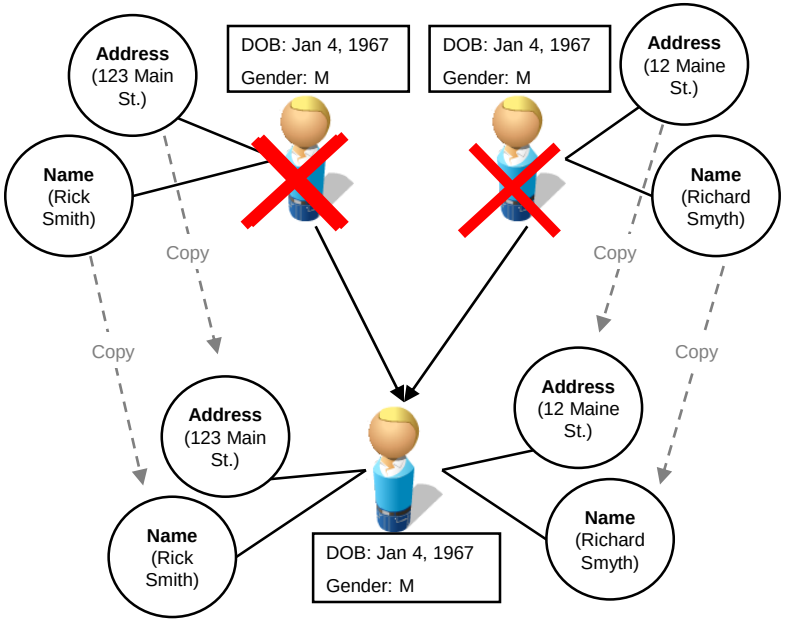
Undo Collapse

- ***undoCollapseMultipleParties*** transaction is introduced in v10. This transaction will undo the previous collapse of multiple parties. The original source parties remain inactivated and clones of all of those original source parties (and their child business objects) are created. Optionally, new party definitions may be provided for some or all of the new parties. The consolidated party created through the collapse is inactivated.



- The Party Ids are not being reused
- Each Party record has its own party Id
- The Party whose being crossed is not deleted but inactivated

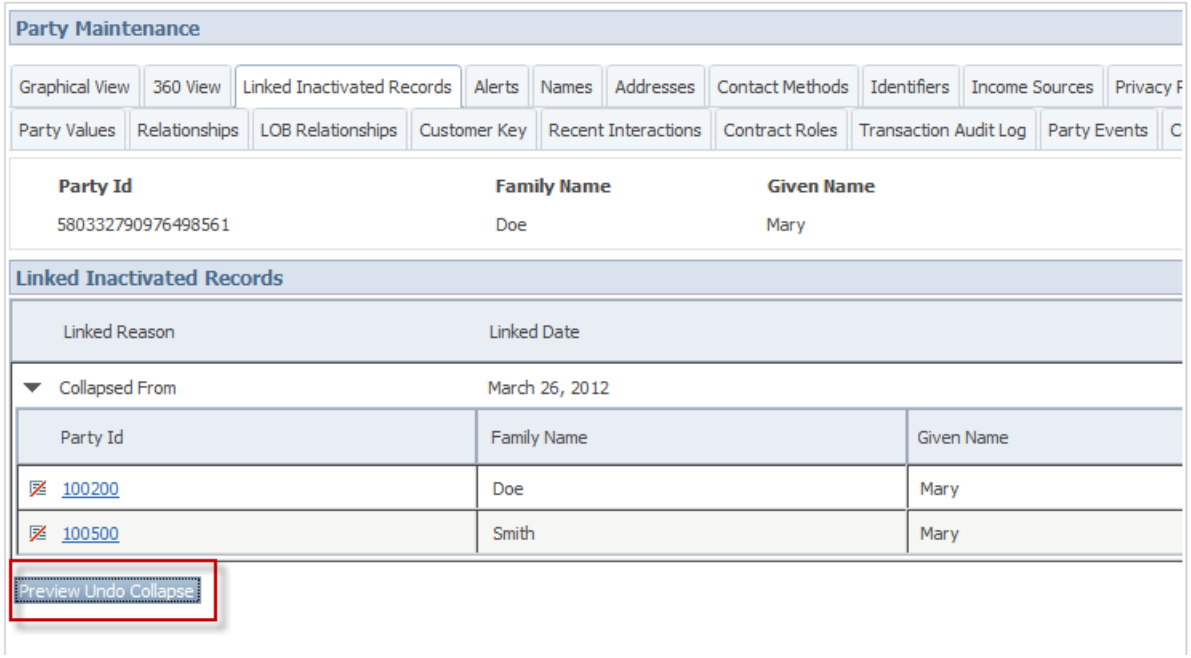
Review Collapse Duplicates



- The two original two parties will be inactivated and the other related data will copy over to the new collapsed party
- The new collapsed party will have a new party id

Party Lineage

- A tab called Link Inactivated Records in the Party Maintenance section of the Data Stewardship UI enables users to view the lineage of a given party record within Physical MDM. The screen shows the fully history of related collapse, split, and undo collapse operations.
  - For Master Data Lineage, ***getLinkedParties*** transaction is invoked behind scene to find the lineage.

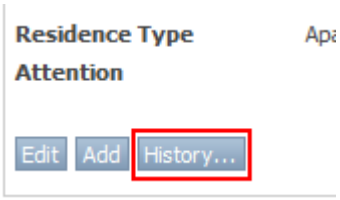


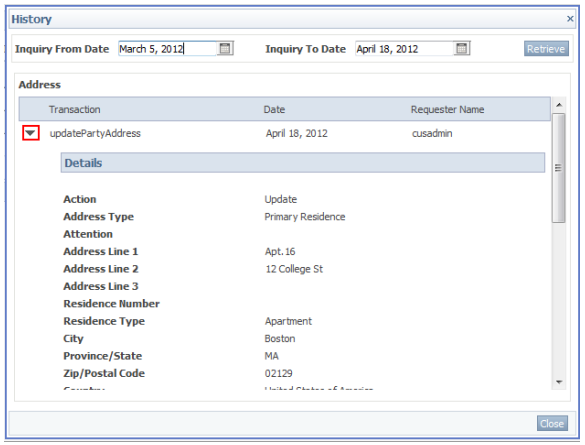
- Will see similar screen during the lab exercise
- The list will shows us what were the related party records and through what activity

Party History

- Many of the Party Maintenance tabs in the DSUI have been enhanced to provide users with the ability to view historical information about the current record within a History pop-up. The content for the History pop-up is a set of historical records based on information found in the transaction log for a specified date range. Each record is represented by a twistie and shows summary information such as transaction type, date, and requester name. Users can view detailed information for each data item by clicking on the associated twistie to expand the display.

  - For Party History, ***getTransactionLog*** transaction is invoked to get transaction history and then use the timestamp from transaction to get history data through inquiry transaction with time. In order to use this feature, Transaction Audit Information Log has to be turn on in Physical MDM.





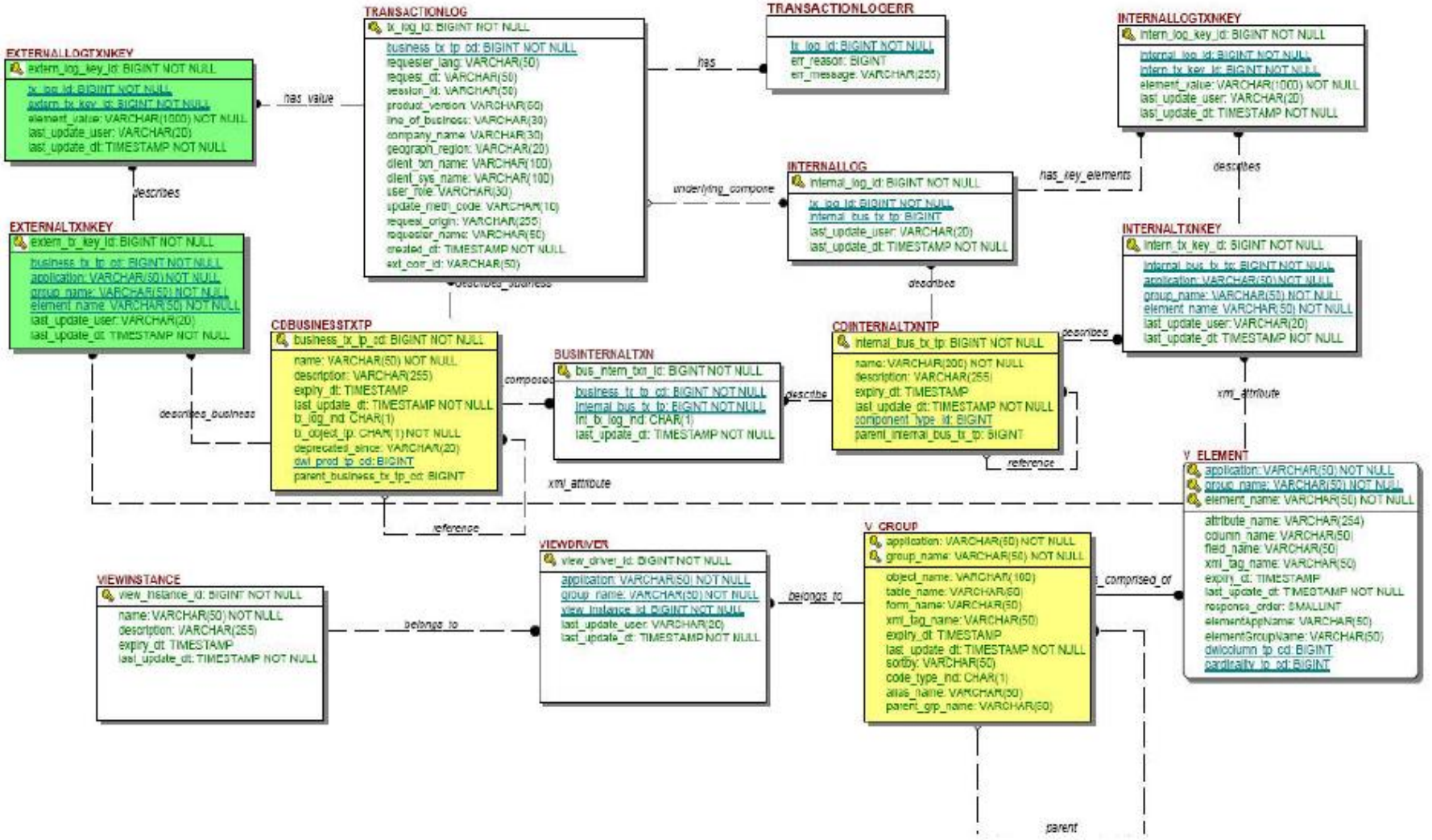
Transaction Audit Information Log (TAIL)

- The Transaction Audit Information Log (TAIL) module provides services for the storage and retrieval of transaction log information for the MDM Physical Implementation.
- You can log the following:
  - External/business transactions
  - Associated internal transactions
  - Key elements associated with the external transactions, such as the party ID
  - Key elements associated with the internal transactions, such as the party ID
  - Successful transactions, failed transactions, or both
- The Transaction Audit Information Log feature has mainly database-driven configuration options. TAIL may be configured to log any persistent or inquiry-based business transactions, as well as some or all of their associated internal transactions. Both successful and failed transactions can be logged. The execution of search services may not be logged.

TAIL Configuration

- How to turn on or off TAIL globally?
  - To turn TAIL logging on, in the Configuration and Management component, set /IBM/DWLCommonServices/TAIL/enabled to true.
  - To turn TAIL logging off, in the Configuration and Management component, set /IBM/DWLCommonServices/TAIL/enabled to false. TAIL logging is turned off by default.

TAIL Related Meta Tables



Instructor Led Online Course (ILO) Optional Material

Learning Objectives

- Investigating Suspect Duplicates
  - What & Why & When Suspect Duplicate Processing
  - Different MDM Matching Styles & Engines
    - Deterministic matching
    - Probabilistic matching
    - Other Matching Engine Options
  - Undo Collapse, Party Lineage and History
  - Different Search Styles & Engines (ILO Optional)

Probabilistic Party Search

- **Probabilistic search option available by invoking the two services:**
  - SearchPersonProbabilistic
  - SearchOrganizationProbabilistic
- Identifies party matches using statistical methods
  - Search criteria must be an attribute takes part in the algorithm
- Leverages the same PME search API used for PME matching
  - Requires PME configured as the engine
- Search criteria are based on matching data defined for PME matching
  - Derived data are synchronized with the physical master data
- Returns a search score for each party which allows ranking of search results
- Can override default threshold or minimum score to adjust search results
- Supports pagination
- Not intended as a replacement for MDM classic search such as
  - searchPerson
  - searchOrganization

Overview of Probabilistic Search Service

Service Name	**searchPersonProbabilistic**
Description	**Searches for persons given a set of criteria using the probabilistic search capabilities of the PME**
Request Object	**ProbabilisticPersonSearchBObj**
Response Object	**ProbabilisticPersonSearchResultBObj** **Party Data returned by search: name, address, contact method, identification number, gender and birth date**

- Existing Physical MDM Data Model remains no change
- Existing searchPerson transaction continue provides classic search
- searchOrganizationProbabilistic service is very similar

Example from the lab exercise

Party record in the system

Name	**Mary Smith**
Address	**1 Main Street, New York**
Date of Birth	**July 31, 1964**

Search criteria

Name	**Mary Smith**

Only first and last name are provided as the search criteria.

Using **probabilistic** search:

1. The candidate party is identified through the PME Name 'bucket'.
2. A search score is determined by comparing attributes between the candidate party and the search criteria.
3. Although the name is an exact match, the score for name alone is not high enough to meet the default Possible Match threshold (7).
4. No party is returned by the probabilistic search transaction.
5. If the search request override the default minimum level to No Match, the party can be returned for the search request.

Using **classic** search:

1. According to the priority of search strategy and the criteria provided, the search transaction uses the Person Name search strategy to identify candidates.
2. Because there is an exact match on name, the party is found.
3. Party is returned by the classic search transaction.

Possible usage on Probabilistic search and Classic search

Probabilistic search	Classic search
▪ **Search using many search criteria**	▪ **Broad result set is desired to return using minimal search criteria (e.g. 1 attribute)**
▪ **Find a single specific record, determine that the specific record doesn't exist in the hub, search for high potential matches before adding a new party**	▪ **Take advantage with wildcard or look-alike searches**
▪ **Reduced overhead if PME matching is configured and derived data is synchronized**	▪ **Configured with Classic Engine and populating the PME tables for search will be extra overhead**

For some scenarios, either search approach will work fine.