

Population GELLO MapReduce Example

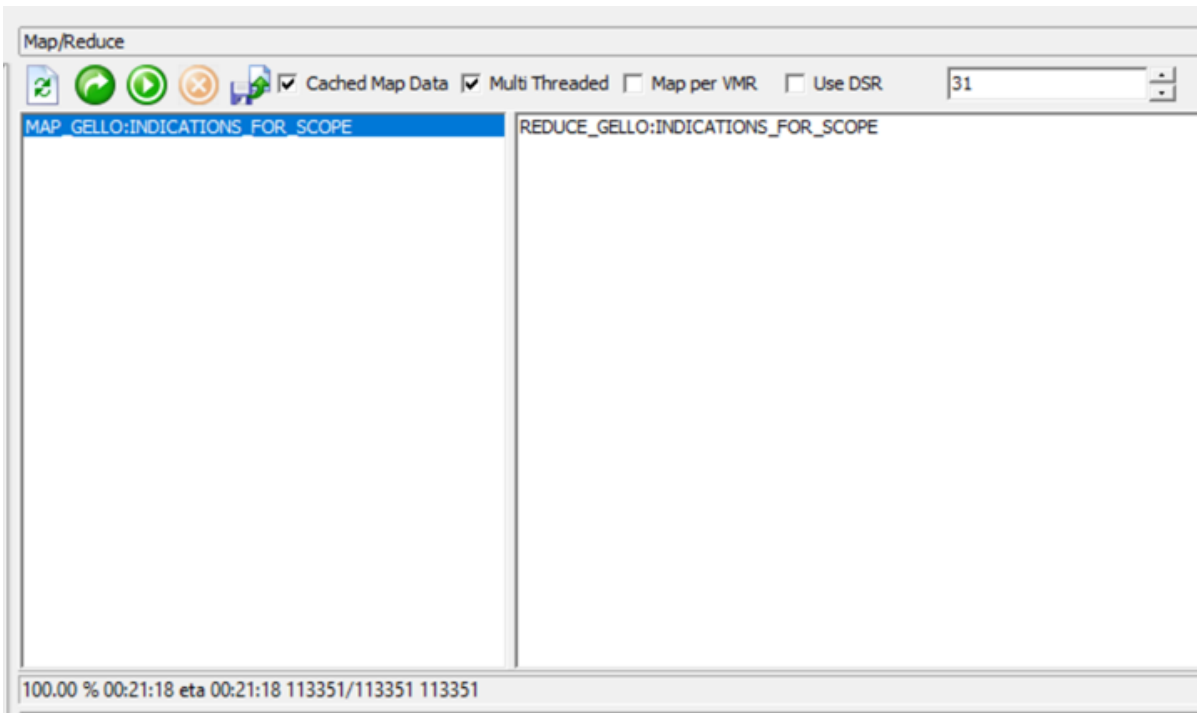
The Map gello looks similar to standard query with the result being Key Value pairs.

```
MAP_Demo_Indications_For_Scope.gello x
> 1 imports MapReduce, KeyType_Classes
  2 Context HL7_v2_VMR_V1::SinglePatient
  3
  4 Let metadata = Tuple{
  5     title = 'Map_Indications_For_Scope',
  6     keywords = Sequence{'MO', 'BGC', 'Upper endoscopy', 'colonoscopy'},
  7     version = 1,
  8     revision = 3,
  9     date = '20260619',
 10     author = 'PS Medical Objects',
 11     identifier = 'MAP_GELLO:INDICATIONS_FOR_SCOPE',
 12     purpose = ''
 13 }
 14
 15
 16 Let endoscopy_LN: CD = CD{code = '19780-6', codeSystem = '2.16840.1113883.6.1', codeSystemName = 'LOINC'}
 17 Let diagnosis_LN: CD = CD{code = '19811-9', codeSystem = '2.16840.1113883.6.1', codeSystemName = 'LOINC'}
 18 Let indication_LN: CD = CD{code = '19778-0', codeSystem = '2.16840.1113883.6.1', codeSystemName = 'LOINC'}
 19
 20 Let hasHadEndoscopy: Boolean =
 21     observations->exists(o| o.observationCode = endoscopy_LN and
 22         o.observationCode.displayName.value.toLower().matches('endoscopy'))
 23 Let hasHadColonoscopy: Boolean =
 24     observations->exists(o| o.observationCode = endoscopy_LN and
 25         o.observationCode.displayName.value.toLower().matches('colonoscopy'))
 26
 27 Let endoscopies =
 28     observations->select(o| o.observationCode = endoscopy_LN and
 29         o.observationCode.displayName.value.toLower().matches('endoscopy'))
 30
 31 Let indications = endoscopies.components->flatten()
 32     ->select(o|o.observationCode = indication_LN)
 33     ->collect(value.oclAsType(ST)).value
 34
 35 -----
 36 Let result: Sequence(KeyValue) =
 37     Sequence{
 38         KeyValue{
 39             key = KeyType_TestSandbox{
 40                 k_pid = patient.patientID.extension,
 41                 k_indications = indications
 42             },
 43             value = 1}
 44     }
 45
 46
 47 in
 48 result
```

The keys are customisable.

```
MAP_Demo_Indications_For_Scope.gello  REDUCE_Demo_Indications_For_Scope.gello  KeyType_Classes.gello_model  x
> 1 package KeyType_Classes
2
3 Imports iso_21090_datatypes , HL7_v2_VMR_V1
4
5 Let metadata = Tuple { version = 1,
6                       revision = 38,
7                       date = '20211214',
8                       author = 'Medical-Objects',
9                       title = 'GELLO KeyType Classes',
10                      keywords = Sequence{'Lib'},
11                      purpose = 'Shared Classes',
12                      identifier = 'GELLOLIB:KeyType_Classes'
13                    }
14 class KeyType_Base extends Any
15     k_pid: String
16     k_date: TS
17
18 class KeyType_MetaSyndrome extends KeyType_Base
19     k_trigs: PQ
20     k_hdl: PQ
21     k_tg_HDLRatio: Real
22     k_diag_list: Sequence(String)
23     k_has_marker: Boolean
24     k_has_bowel_cancer: Boolean
25
26 class KeyType_TestSandbox extends KeyType_Base
27     k_ProcDates: Sequence(TS)
28     k_hasHadColonoscopy: Boolean
29     k_hasHadEndoscopy: Boolean
30     k_indications: Sequence(String)
31     k_indicationsAsString: String
32     k_diagnoses: Sequence(String)
33     k_datesAndDiagnoses: String
34     k_hasPolyp: Boolean
35     k_hasCa: Boolean
36     k_hasRefluxChanges: Boolean
37     k_height_Real: Real
38     k_weight Real: Real
```

The Maps are run on the whole population to produce a map cache. This example of the indications for endoscopy for a gastroenterology practice shows the map stage took about 20 minutes for 113000 nested and at times complex patient records as VMRs.



Some Reduce gello is shown here; again its a fairly simple worked example. The loop is just to help the output strings be unique and somewhat cleaned.:

```

REDUCE_Demo_Indications_For_Scope.gello x
> 1 Imports MapReduce, HL7_v2_VMR_V1, KeyType_Classes
2 Package Local
3 --functions
4 UniqueStrings_Internal(i:Integer, accum: Sequence(String), strList:Sequence(String)): Sequence(String) =
5   If i <= strList.size()
6     then
7       Let result = UniqueStrings_Internal(i+1,
8         If accum.excludes(strList[i]) and strList[i].matchPos('') = 0
9         then accum->append(strList[i]) else accum endif ,
10        strList
11        )
12     in
13     result
14   else
15     accum
16   endif
17
18 UniqueStrings(strList: Sequence(String)): Sequence(String) =
19   UniqueStrings_Internal(1, Sequence(), strList)
20
21 EndPackage
22 Context Sequence(KeyValue)
23
24 Let metadata = Tuple{
25     title = 'indications For Scope',
26     keywords = Sequence{'MO', 'BGC', 'Upper endoscopy', 'colonoscopy'},
27     version = 1,
28     revision = 8,
29     date = '20260620',
30     author = 'PS Medical Objects',
31     identifier = 'REDUCE_GELLO:INDICATIONS_FOR_SCOPE',
32     purpose = '',
33     map_from = 'MAP_GELLO:INDICATIONS_FOR_SCOPE'
34 }
35
36 Let keys: Sequence(KeyType_TestSandbox) = Self->collect(key.oclAsType(KeyType_TestSandbox))
37 Let ids: Sequence(KeyType_TestSandbox) = keys->select(k_pid.oclIsDefined())
38 Let indications_raw: Sequence(KeyType_TestSandbox) = keys->select(k_indications.oclIsDefined() )
39 Let indications_cleaned = UniqueStrings(indications_raw.k_indications->flatten()->sort())
40 in
41
42 Tuple{
43   a=ids.k_pid,
44   b=indications_cleaned
45 }
46
47

```

The actual result for the combined MapReduce:

Results Explorer		
Name	Class	Data
[-] b[19073]	Sequence(String)	Sequence(\$ years mid-abdominal discomfort and bloating, BO normal. PH,'Dark' stools for 6
[1]	String	\$ years mid-abdominal discomfort and bloating, BO normal. PH
[2]	String	'Dark' stools for 6 weeks. P/H of GORD. Positive FOB
[3]	String	'Globus' like sensation. Post prandial and early morning nausea.
[4]	String	'Globus' like sensation. Sinusitis. Occasional dyspepsia.
[5]	String	'Histological' Barretts. GORD under control with Pariet
[6]	String	'Silent' reflux, regurgitation
[7]	String	'choking' sensation related to the upper oesophageal region.
[8]	String	+Coeliac serology.
[9]	String	+FOB
[10]	String	+FOB and weight loss.
[11]	String	+FOB for Investigation.
[12]	String	+FOB for further evaluation.
[13]	String	+FOB for investigation.
[14]	String	+FOB.
[15]	String	+FOB/Altered bowel habit.
[16]	String	+FOB/Anaemia.
[17]	String	+FOB/Erythropenia.
[18]	String	+FOB/Family history of stomach cancer.
[19]	String	+FOB/GORD.
[20]	String	+FOB/Iron deficiency.
[21]	String	+FOB/Pancreatitis.
[22]	String	+FOB/Prior gastric sleeve.
[23]	String	+FOB/Sub-optimal iron stores.
[24]	String	+FOB/Upper abdominal pain.
[25]	String	+ve FOB
[26]	String	+ve FOBT
[27]	String	+ve HP serology
[28]	String	+ve anti-gliadin Ab, -ve TTG Ab, +ve ANA, iron deficient.Mild

Known Packages | Results Explorer | Model Explorer | Stack | Data Explorer | Messages

The result can be saved into various formats with a right click on the result element, in this case 'b':

Results Explorer		
Name	Class	Data
[+] _Context_[113351]	Sequence(KeyValue)	Sequence{<KeyValue: TValue>, ...
[+] metadata	Tuple	<Tuple>
[+] keys[113351]	Sequence(KeyType_Test...)	Sequence{<KeyType_TestSandbox: ...
[+] ids[96564]	Sequence(KeyType_Test...)	Sequence{<KeyType_TestSandbox: ...
[+] indications_raw[112462]	Sequence(KeyType_Test...)	Sequence{<KeyType_TestSandbox: ...
[+] \$SCOPE1.result[19073]	Sequence(String)	Sequence{\$ years mid-abdominal dis
[+] indications_cleaned[19073]	Sequence(String)	Sequence{\$ years mid-abdominal dis
[-] _Result_	Tuple	<Tuple>
[+] a[96564]	Sequence(String)	Sequence{1000004^^^Medical-Obj
[-] b[19073]	Sequence(String)	Sequence{\$ years mid-abdominal dis
[1]	String	s mid-abdominal discomfort an
[2]	String	stools for 6 weeks. P/H of GOI
[3]	String	's' like sensation. Post prandial
[4]	String	's' like sensation. Sinusitis. Oc
[5]	String	'logical' Barretts. GORD under c
[6]	String	reflux, regurgitation
[7]	String	'choking' sensation related to the up
[8]	String	+Coeliac serology.

Context menu for 'b[19073]':

- Hide Null Results
- Write selected results element to file**
- Write all results to file
- Reload context data from file
- Reload data and execute

A clinical use case is for audit and research. The auditable and researchable Map cache could be updated on a regular basis to surface population clinical data of interest for bespoke Reduce gello queries.

In a gastroenterology practice doing colonoscopies, this includes the 'To caecum' rate (an assessment of the quality of the scope procedure) and an Adenoma rate (the number of adenomas on subsequent biopsy for biopsies performed with endoscopy). The example above is for upper endoscopies. Different clinical queries would be desired.

BACK