**Large Data Extract Query Strategies**

Salesforce Enterprise Scale Team

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| # | Date | Description | Author |
| 1 | 2/20/13 | Initial draft created | SF Enterprise Scale Team |
| 2 | 2/28/13 | Amended phase2 strategy. Add “query 5” in phase 2 | qiang.hun.tn1x@statefarm.com |
|  |  |  |  |

In response to State Farm concerns around large scale extraction of data (issue 310 among others), the Enterprise Scale team and Salesforce R&D have indicated that extracts of the size required must be “chunked” into multiple smaller requests to provide adequate performance. The team has also laid out a set of strategies and product enhancements for effective chunking of extract queries. The following is a description of the delivery phases for this work, and what State Farm can expect out of each phase. At this point, we have not determined release dates for the second or third phases of this work.

**Phase 1** – Extract Based on Autonumber and Formula Fields

**What is it?**

A strategy using existing product features that requires an Autonumber field on each object to be extracted (and a corresponding Formula field to convert the Autonumber text into an INT).

**How is it used?**

State Farm would first add an Autonumber field to each object to be extracted, and a Formula field (for these purposes call it ‘SequenceNum’) that converts the Autonumber text into an integer. They could then extract the data in chunks defined by ranges within this sequence.

For example, suppose that there are 100M Policy records to be extracted in this manner. This could be done by executing parallel bulk queries of the form:

select \* from Policy\_\_c where SequenceNum > 0 and SequenceNum <= 10,000,000

select \* from Policy\_\_c where SequenceNum > 10,000,000 and SequenceNum <= 20,000,000

.

.

.

select \* from Policy\_\_c where SequenceNum > 90,000,000 and SequenceNum <= 100,000,000

**Availability?**

NOW - In the Spring ’13 release

**Phase 2** – Primary Key Chunking for a Complete Table Extract

**What is it?**

A strategy developed by Salesforce R&D that would allow chunking of extract queries using the existing Primary Key (record ID) of a table.

**How is it used?**

The process has two phases:

1. State Farm would perform a pre-query to discover chunk boundaries in an object table based on the Primary Key.

For example, State Farm could loop the following query to find the Primary Key boundaries of every 250,000 records until it returned a NULL

select Id from Policy\_\_c where Id > [range] order by Id asc limit 1 offset 249999

An example of extracting 750K rows in this manner would look like the following:

*Query 1*

select Id from Policy\_\_c where Id > ‘000000000000000’ order by Id asc limit 1 offset 249999

Returns: ‘a01000000250000’

*Query 2*

select Id from Policy\_\_c where Id > ‘a01000000250000’ order by Id asc limit 1 offset 249999

Returns: ‘a01000000500000’

*Query 3*

select Id from Policy\_\_c where Id > ‘a01000000500000’ order by Id asc limit 1 offset 249999

Returns: ‘a01000000750000’

*Query 4*

select Id from Policy\_\_c where Id > ‘a01000000750000’ order by Id asc limit 1 offset 249999

Returns: NULL

Indicating the upper boundary is within the range of 24999, if there

are no more records, then ‘101000000750000’ is the upper bound. Since there is uncertainty, another query is needed.

*Query 5*

select Id from Policy\_\_c where Id > ‘a01000000750000’ order by Id asc limit 1 offset 0

Scenario 1:

Returns any id that is not null

Indicating there are more records, so ‘a01000000750000’ will be the last lower bound. (last upper bound does not matter here)

Scenario 2:

Returns null

Indicating ‘a010000007500000’ is the upper bound. This only happens when we can perfectly divide the record by 250000.

This would result in the following Primary Key chunk boundaries:

‘000000000000000’

‘a01000000250000’

‘a01000000500000’

‘a01000000750000’

State Farm would use the chunk boundaries identified in Step 1 to construct the following bulk queries to extract the data:

select \* from Policy\_c where Id > ‘00000000000000’ and Id <= ‘a0100000025000’

select \* from Policy\_c where Id > ‘a0100000025000’ and Id <= ‘a0100000050000’

select \* from Policy\_c where Id > ‘a0100000050000’ and Id <= ‘a0100000750000’

select \* from Policy\_c where Id > ‘a0100000750000’

\* notice the last query is necessary only if there are more record.

Each query would return 250K of the 750K rows.

**Availability?**

NOW - In the Spring ’13 release

**Phase 3** – Primary Key Chunking for a Subset of Table Data

**What is it?**

A strategy under developed by Salesforce R&D that would allow chunking of extract queries using the existing Primary Key (record ID) of a table, including the ability to define a filter to extract only part of the data.

**How is it used?**

The process has two phases:

1. As in the Phase 2 strategy, State Farm would perform a pre-query to discover chunk boundaries in an object table based on the Primary Key.
2. State Farm would then use the chunk boundaries identified to construct a chunked query to extract data in the table, with the option to use a filter to extract only a portion of the data.

For example, the sequence of queries:

select \* from Policy\_c where Id > ‘000000000000000’ and Id <= ‘a01000000250000’ and CreatedDate\_\_c = ‘2103-01-01’

select \* from Policy\_c where Id > ‘a01000000250000’ and Id <= ‘a01000000500000’ and CreatedDate\_\_c = ‘2103-01-01’

select \* from Policy\_c where Id > ‘a01000000500000’ and Id <= ‘a01000007500000’ and CreatedDate\_\_c = ‘2103-01-01’

would extract all records in the complete set created on January 1st, 2013.

**Availability?**

TBD, Summer ’13 or later