

BUSINESS CHANGE DOCUMENT

IT REQUEST NO: 68110270
DOCUMENT NAME: BCD_68110270_TUNING PERFORMANCE IS PROCESS
DOCUMENT VERSION: v1.0
CREATED DATE: 01/12/2025
LAST UPDATED: 19/12/2025

DOCUMENT INITIAL

PREPARED BY:	SUNTARA	01/12/2025
	SUNTARA, IT APPLICATION SERVICES	Date
VERIFIED BY:	CHUTIMA	01/12/2025
	CHUTIMA, IT APPLICATION SERVICES-SH	Date

DOCUMENT APPROVAL

The following signatures are required for approval of this document.

REQUEST BY:	SUNTARA	01/12/2025
	SUNTARA, IT APPLICATION SERVICES	Date
ACCEPTED BY:	SUNTARA	01/12/2025
	SUNTARA, IT APPLICATION SERVICES-SH	Date
APPROVED BY:	ORAPAN	19/12/2025
	ORAPAN, IT APPLICATION SERVICES-DH	Date

DOCUMENT CONTROL

Document History

History	Source	By	Version	Date	Comments
Created	-	ITAS.Suntara	v.0.1	1/12/2025	Create Document
Update	-	ITAS.Suntara	v.1.0	18/12/2025	Update acceptance

Reviewer/Inspector

Reviewer/Inspector Name	Date	Comments
DD.CHUTIMA	18/12/2025	

Distribution

Version	No. of Copy	To	Date	Telephone No.

References/Bibliography

No.	Title/ Document Name	Author	Version	Date	No. of Page

TABLE OF CONTENTS

1.0 BUSINESS REQUIREMENTS	4
<i>1.1 CHANGE REQUIREMENTS.....</i>	4
<i>1.1.1 OBJECTIVES.....</i>	4
<i>1.1.2 OVERVIEW CHANGE</i>	4
<i>1.1.3 CHANGE DETAIL FUNCTION.....</i>	4
<i>1.2 ACCEPTANCE CRITERIA.....</i>	6
<i>1.3 RISK</i>	10
<i>1.4 ABBREVIATION / FORMULA.....</i>	10
<i>1.4.1 Abbreviation</i>	10
<i>1.4.2 Formula</i>	10
2.0 REQUIREMENTS ANALYSIS.....	10
<i>2.1 FUNCTIONAL REQUIREMENTS.....</i>	10
<i>2.2 NON-FUNCTIONAL REQUIREMENTS.....</i>	23
<i>2.3 INTERFACE REQUIREMENT</i>	23
<i>2.4 TESTING SCENARIO.....</i>	24
3.0 SCHEDULE	28

File Name: BCD_68110270_TUNING PERFORMANCE IS PROCESS	Page: 3 OF 28
<i>Confidential and Restricted Circulation</i>	Template Version: V.3.0

1.0 BUSINESS REQUIREMENTS

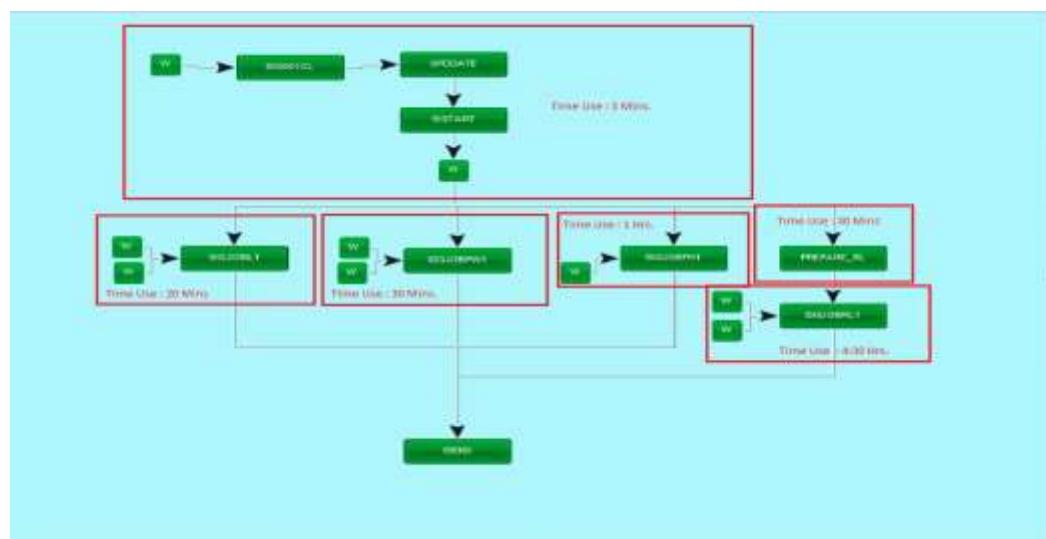
1.1 CHANGE REQUIREMENTS

1.1.1 OBJECTIVES

เพื่อแก้ไขข้อมูลในส่วนของ Tuning Performance program เกี่ยวกับขั้นตอนการทำงานของระบบให้สามารถทำงานได้อย่างมีประสิทธิภาพและทำงานได้เร็วขึ้นโดย Business Logic อ้างอิงด้วยเงื่อนไขเดิม

1.1.2 OVERVIEW CHANGE

Process run Batch IS ในปัจจุบันใช้เวลาการประมวลผลนานๆ ต้องการ query ข้อมูลแบบ row-by-row, ใช้ loop ซ้อน, ไม่มีการ cache ในกรณีที่มีการเรียกข้อมูลซ้ำๆ การใช้ Statement ที่ซับซ้อน, มีการ open/close connection บ่อยเกินไป



แสดง process run batch IS และเวลาที่ใช้แต่ละ process

จากข้อมูล process batch ได้สรุป tuning สำหรับแต่ละโปรแกรมโดยมีหัวข้อการ tuning ดังนี้

- ลด SQL calls:** ใช้ PreparedStatement และ Batch operations (addBatch/executeBatch) เพื่อลด overhead จาก connection
- Optimize Loops:** Parallel Streams สำหรับ processing ข้อมูลใหญ่
- Caching:** ใช้ HashMap และ Cache สำหรับข้อมูลที่ query ซ้ำ (เช่น ตาราง GNMSJB, RLMS73)
- DB Optimization:** เพิ่ม index บนคอลัมน์ที่ใช้ใน WHERE/JOIN, ใช้ Pagination (OFFSET/FETCH) สำหรับ large result sets

โดยการ tuning ครั้งนี้ จะทำการแก้ไขโปรแกรม Batch หลัก ๆ ดังนี้

1.1.3 CHANGE DETAIL FUNCTION

Functional	
Need No	Description
F-0001	1. IS2273 / IS2273S <ul style="list-style-type: none">- เปลี่ยนการอัปเดต จากการ Update ทีละ record เป็น Batch Update โดยใช้ `PreparedStatement.addBatch()` + `executeBatch()` เพื่อลด round-trip ไป database และทำใน transaction เดียว- ปรับ Statement ให้ใช้การ Fetch ด้วย min/max เอาตัว rollback static hisun ออกเนื่องจากไม่ได้ใช้

Functional	
Need No	Description
	<ul style="list-style-type: none"> - ลด PreloadData ไว้ก่อน เพื่อลดการ query ข้ามช้อน - ปรับ Array ไม่ใช้ new Array ไปใช้ clear แทน - ปรับ logic การแบ่ง SUBJOB จากเดิม loop row-by-row มาเป็นการคำนวณ min/max ของแต่ละ SUBJOB ก่อนเริ่มงาน ทำให้แบ่งงานได้เร็ว - เพิ่ม Prepare Data โดย query ตาราง CSMS25 และ RLMS73 ครั้งเดียวแล้วเก็บใน HashMap เพื่อ reuse ใน loop แทนการ query ซ้ำ <p>2. IS170RL / IS170RLS</p> <ul style="list-style-type: none"> - ปรับการ fetch ข้อมูลจากตาราง ISTARGET จากเดิม loop ที่ล็อค record (ทำให้แตก job ช้า) มาเป็นการ split งานตาม batch size โดยใช้ SQL หา min/max ของ Contract No. และ Branch เพื่อดึงข้อมูลเป็น chunk ในญี่ปุ่น ลดจำนวนครั้งที่เปิด cursor <p>3. ISD208S</p> <ul style="list-style-type: none"> - เพิ่มการใช้ MemCache กับตารางที่ข้อมูลไม่เปลี่ยนบ่อย เช่น PWTB60, ILTB60, RLTB60 เพื่อลดการ query database ช้า - สร้างตัวแปร substringT60fil เพื่อใช้ใน condition ต่าง ๆ แทนการ substring ช้าใน loop <p>4. IS175</p> <ul style="list-style-type: none"> - เพิ่ม Prepare Data โดย query ตารางข้อมูลขนาดเล็กแต่ถูกเรียกใช้บ่อย (เช่น GNTB16, GNTS17) ครั้งเดียวแล้วเก็บใน HashMap หลายตัว (ทุก queryMap) เพื่อ lookup O(1) - เพิ่มเงื่อนไข filter เฉพาะข้อมูลที่จำเป็นจริงเมื่อดึงตาราง RLMS01 ลดปริมาณข้อมูลที่ไม่จำเป็น <p>5. IS182S / IS182</p> <ul style="list-style-type: none"> - ปรับการเตรียม Statement Generation โดยใช้ import statement ร่วมกับโปรแกรม GNP025 และ GNP023 เพื่อลดการสร้าง object และ string concatenation ช้า ลด overhead ในส่วนการสร้างรายงาน <p>6. ISGPREJOBRL / ISCPYF03 / ISCPYF02 (Daily Backup Data)</p> <ul style="list-style-type: none"> - ใช้ Java 8 Native Batch Processing รองรับ Re-run และ Idempotency (initial ค่าตั้งต้น, execution context, mapping value) - แยกการสร้าง SQL เป็น Builder เพื่อความอ่านง่ายและ reuse ได้ - บันทึกสถานะงานลงตาราง Monitoring เพื่อรับรู้ Retry และป้องกัน duplicate - ใช้ Multi-Worker + Parallelism แบ่งงานตาม Schema+Table ลดเวลาโดยรวม - จัดการ Index โดย Disable Non-Clustered Index ก่อน Insert และ Rebuild หลังจบงาน - ใช้ Staging Table Workflow (RLMS01STG → RLMS01 → RLMS01H) พัฒนารองข้อมูลช้า ลดการ insert ช้าช้อน <p>7. IS2271 / IS2271S</p> <ul style="list-style-type: none"> - เปลี่ยน logic การแบ่ง SUBJOB จาก loop row-by-row เป็นการคำนวณ min/max ของแต่ละ SUBJOB เมื่อ IS2273 - เพิ่ม Prepare Data โดย query ตาราง CSMS25 และ RLMS19 ครั้งเดียวแล้วเก็บใน HashMap เพื่อ reuse ใน loop <p>8. IS302</p> <ul style="list-style-type: none"> - เปลี่ยนการ insert ตาราง ISMS32 และ ISMS32TM เป็น Bulk Insert เพื่อลดจำนวน transaction - ใช้ MemCache กับตาราง GNTB17 และ GNTB27 - ทำ Preprocessing Data ก่อนเข้าฟังก์ชัน business - ปรับการ query CSMS03 และ delete ISMS32 จากเดิมทำใน while loop มาเป็น query เก็บไว้ก่อนแล้ว process ทีเดียว ลด I/O และ memory heap <p>9. Submit Job Program</p> <ul style="list-style-type: none"> - ปรับการ fetch ข้อมูลจาก CSWKCSN จากเดิม loop ที่ล็อค record (ทำให้แตก job ช้า) มาเป็นการ split งานตาม batch size โดยใช้ SQL หา min/max ของ CSN เพื่อดึงข้อมูลเป็น chunk ในญี่ปุ่น ลดก่อน <p>10. ISG321RLS</p> <ul style="list-style-type: none"> - ปรับ ใช้ Memcache ใช้กับ ตาราง Rtb05, ischan, isappt, iscamps - ปรับใช้ Preload Data ใช้กับ ตาราง csms13, RLMS01, RLMS02

Functional	
Need No	Description
	<p>เพื่อลดการ query ข้ามช้อน</p> <ul style="list-style-type: none"> - ปรับคุณภาพของการทำงานในการทำงานในแต่ละรอบให้ดีขึ้น - ปรับการใช้ Batch Insert เพื่อลดจำนวน transaction <p>11. IS303RLS</p> <ul style="list-style-type: none"> - PrepareData เตรียมข้อมูลก่อนนำไปใช้เพื่อลด loop การทำงาน (RLTB05, RLTB01, RLMS40) - เปลี่ยน Statement Delete จาก Hisun เป็น Native Delete ลดการ Query ไม่จำเป็น <p>12. ISGM07</p> <ul style="list-style-type: none"> - ปรับการใช้ Batch Insert เพื่อลดจำนวน transaction <p>13. IS1701CS</p> <ul style="list-style-type: none"> - ปรับเรื่อง memcache (RLTB05) เพื่อลดการ connection db - ปรับการใช้ Batch Delete ISMS11TM เพื่อลด header ใน การส่ง query

1.2 ACCEPTANCE CRITERIA

Functional No.	Acceptance No.	Description
User ตรวจสอบ		
F-0001	AC-1001	<p>ตรวจสอบข้อมูลของ Table ISMS34, ISMD34, ISMS201, ISMS11RL, IS321, IS325MM, ISMS22, ISMS206, ISMD236, ISMS231, ISMS33, ISMD33, ISMS31, ISMS32, IS325 ตรงกับข้อมูล production ตามเงื่อนไขวันที่เลือก</p> <p>ตาม statement record count ดังนี้</p> <pre> SELECT (SELECT COUNT(*) FROM ISOD0001.ISMS34 WITH (NOLOCK) WHERE I34ASDT = 20251207)B198, (SELECT COUNT(*) FROM ISOD0001.ISMD34 WITH (NOLOCK) WHERE D34ASDT = 20251207)B202, (SELECT COUNT(*) FROM ISOD0001.ISMS201 WITH (NOLOCK) WHERE IPERIOD = 25681207)B208 SELECT (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS11RL WITH (NOLOCK) WHERE ICDTE = 25681207) A0001, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS321 WITH (NOLOCK) WHERE A21DTE = '2025-12-07') A0005, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS325MM WITH (NOLOCK) WHERE A25DTE = '2025-12-07') A0006, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS22 WITH (NOLOCK) WHERE ICDTE = 25681207) A0021, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS206 WITH (NOLOCK) WHERE IPERIOD = 25681207) A0044, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMD236 WITH (NOLOCK) WHERE DASOF = 25681207) A0057, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS231 WITH (NOLOCK) WHERE IPERIOD = 25681207) A0064, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS33 WITH (NOLOCK) WHERE I33ASDT = 20251207) A0109, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMD33 WITH (NOLOCK) WHERE D33ASDT = 20251207) A0111,ssss WITH (NOLOCK) WHERE M31ASDT = 20251207) A0132, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS32 WITH (NOLOCK) WHERE M32ASDT = 20251207) A0133, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS325_12 WITH (NOLOCK) WHERE A25DTE = '2025-12-07') A0134 </pre>

File Name:	BCD_68110270_TUNING PERFORMANCE IS PROCESS	Page:	6 OF 28
Confidential and Restricted Circulation		Template Version:	V.3.0

Functional No.	Acceptance No.	Description																																																																																																																																																																								
		<pre> SELECT (SELECT COUNT(*) FROM ISOD0001.ISMS34 WITH (NOLOCK))B198, (SELECT COUNT(*) FROM ISOD0001.ISMD34 WITH (NOLOCK))B202, (SELECT COUNT(*) FROM ISOD0001.ISMS201 WITH (NOLOCK))B208 SELECT (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS11RL WITH (NOLOCK))A0001, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS321 WITH (NOLOCK))A0005, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS325MM WITH (NOLOCK))A0006, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS22 WITH (NOLOCK))A0021, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS206 WITH (NOLOCK))A0044, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMD236 WITH (NOLOCK))A0057, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS231 WITH (NOLOCK))A0064, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS33 WITH (NOLOCK))A0109, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMD33 WITH (NOLOCK))A0111, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS31 WITH (NOLOCK))A0132, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS32 WITH (NOLOCK))A0133, </pre> <p>Result หลังจาก run statement, Before ต้องเท่ากันกับ After</p> <table border="1"> <thead> <tr> <th colspan="2">Before</th> <th>B198</th> <th>B202</th> <th>B208</th> <th>A0001</th> <th>A0005</th> <th>A0006</th> <th>A0021</th> <th>A0044</th> <th>A0057</th> <th>A0064</th> <th>A0109</th> <th>A0111</th> </tr> </thead> <tbody> <tr> <td>04 Where Date</td> <td></td> <td>9539</td> <td>0</td> <td>0</td> <td>323</td> <td>2391558</td> <td>0</td> <td>4273</td> <td>4676466</td> <td>1038</td> <td>426564</td> <td>2382019</td> <td>0</td> </tr> <tr> <td>04 Full</td> <td></td> <td>B198</td> <td>B202</td> <td>B208</td> <td>4646300</td> <td>688458</td> <td>439</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td>A0001</td> <td>A0005</td> <td>A0006</td> <td>5763443</td> <td>2391558</td> <td>913743375</td> <td>1749542</td> <td>4676466</td> <td>1246109</td> <td>180422860</td> <td>1047761392</td> <td>149933267</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="2">Before</th> <th>B198</th> <th>B202</th> <th>B208</th> <th>A0001</th> <th>A0005</th> <th>A0006</th> <th>A0021</th> <th>A0044</th> <th>A0057</th> <th>A0064</th> <th>A0109</th> <th>A0111</th> </tr> </thead> <tbody> <tr> <td>07 Where Date</td> <td></td> <td>0</td> </tr> <tr> <td>07 Full</td> <td></td> <td>B198</td> <td>B202</td> <td>B208</td> <td>4646310</td> <td>688458</td> <td>439</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td>A0001</td> <td>A0005</td> <td>A0006</td> <td>5763099</td> <td>2392362</td> <td>916135737</td> <td>1752777</td> <td>4677974</td> <td>1245796</td> <td>180390994</td> <td>1047762186</td> <td>149933787</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="2">After</th> <th>B198</th> <th>B202</th> <th>B208</th> <th>A0001</th> <th>A0005</th> <th>A0006</th> <th>A0021</th> <th>A0044</th> <th>A0057</th> <th>A0064</th> <th>A0109</th> <th>A0111</th> </tr> </thead> <tbody> <tr> <td>07 Where Date</td> <td></td> <td>9539</td> <td>0</td> <td>0</td> <td>323</td> <td>2391558</td> <td>2391558</td> <td>4273</td> <td>4676466</td> <td>1038</td> <td>426564</td> <td>2382019</td> <td>0</td> </tr> <tr> <td>07 Full</td> <td></td> <td>B198</td> <td>B202</td> <td>B208</td> <td>4655849</td> <td>688458</td> <td>439</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td>A0001</td> <td>A0005</td> <td>A0006</td> <td>5763428</td> <td>4783920</td> <td>918527295</td> <td>1757050</td> <td>9354440</td> <td>1246824</td> <td>1803817558</td> <td>1050144205</td> <td>149933787</td> </tr> </tbody> </table>	Before		B198	B202	B208	A0001	A0005	A0006	A0021	A0044	A0057	A0064	A0109	A0111	04 Where Date		9539	0	0	323	2391558	0	4273	4676466	1038	426564	2382019	0	04 Full		B198	B202	B208	4646300	688458	439									A0001	A0005	A0006	5763443	2391558	913743375	1749542	4676466	1246109	180422860	1047761392	149933267	Before		B198	B202	B208	A0001	A0005	A0006	A0021	A0044	A0057	A0064	A0109	A0111	07 Where Date		0	0	0	0	0	0	0	0	0	0	0	0	07 Full		B198	B202	B208	4646310	688458	439									A0001	A0005	A0006	5763099	2392362	916135737	1752777	4677974	1245796	180390994	1047762186	149933787	After		B198	B202	B208	A0001	A0005	A0006	A0021	A0044	A0057	A0064	A0109	A0111	07 Where Date		9539	0	0	323	2391558	2391558	4273	4676466	1038	426564	2382019	0	07 Full		B198	B202	B208	4655849	688458	439									A0001	A0005	A0006	5763428	4783920	918527295	1757050	9354440	1246824	1803817558	1050144205	149933787
Before		B198	B202	B208	A0001	A0005	A0006	A0021	A0044	A0057	A0064	A0109	A0111																																																																																																																																																													
04 Where Date		9539	0	0	323	2391558	0	4273	4676466	1038	426564	2382019	0																																																																																																																																																													
04 Full		B198	B202	B208	4646300	688458	439																																																																																																																																																																			
		A0001	A0005	A0006	5763443	2391558	913743375	1749542	4676466	1246109	180422860	1047761392	149933267																																																																																																																																																													
Before		B198	B202	B208	A0001	A0005	A0006	A0021	A0044	A0057	A0064	A0109	A0111																																																																																																																																																													
07 Where Date		0	0	0	0	0	0	0	0	0	0	0	0																																																																																																																																																													
07 Full		B198	B202	B208	4646310	688458	439																																																																																																																																																																			
		A0001	A0005	A0006	5763099	2392362	916135737	1752777	4677974	1245796	180390994	1047762186	149933787																																																																																																																																																													
After		B198	B202	B208	A0001	A0005	A0006	A0021	A0044	A0057	A0064	A0109	A0111																																																																																																																																																													
07 Where Date		9539	0	0	323	2391558	2391558	4273	4676466	1038	426564	2382019	0																																																																																																																																																													
07 Full		B198	B202	B208	4655849	688458	439																																																																																																																																																																			
		A0001	A0005	A0006	5763428	4783920	918527295	1757050	9354440	1246824	1803817558	1050144205	149933787																																																																																																																																																													
F-0001	AC-1002	<p>เข้าค่ารวมข้อมูล (Sum) เพื่อตรวจสอบผลรวมของยอดต่าง ๆ หรือ</p> <p>-- B198 --</p> <pre> SELECT SUM(I34PRIC) AS SUM_I34PRIC, SUM(I34TOAM) AS SUM_I34TOAM, SUM(I34PCAM) AS SUM_I34PCAM, SUM(I34UIDA) AS SUM_I34UIDA, SUM(I34INTR) AS SUM_I34INTR, SUM(I34CRUR) AS SUM_I34CRUR, SUM(I34DOWN) AS SUM_I34DOWN, SUM(I3412IN) AS SUM_I3412IN, SUM(I34NDUE) AS SUM_I34NDUE, SUM(I34OPR) AS SUM_I34OPR, </pre>																																																																																																																																																																								

Functional No.	Acceptance No.	Description
		<pre> SUM(I34OIN) AS SUM_I34OIN, SUM(I34OUS) AS SUM_I34OUS, SUM(I34OCOL) AS SUM_I34OCOL, SUM(I34OOTH) AS SUM_I34OOTH, SUM(I34P1PCP) AS SUM_I34P1PCP, SUM(I34P1INT) AS SUM_I34P1INT, SUM(I34P1CRU) AS SUM_I34P1CRU, SUM(I34P1PNT) AS SUM_I34P1PNT, SUM(I34P1OTH) AS SUM_I34P1OTH, SUM(I34P1T01) AS SUM_I34P1T01, SUM(I34P1T02) AS SUM_I34P1T02, SUM(I34P1TPP) AS SUM_I34P1TPP, SUM(I34P1T21) AS SUM_I34P1T21, SUM(I34P1T99) AS SUM_I34P1T99, SUM(I34ODAM) AS SUM_I34ODAM, SUM(I34SOSA) AS SUM_I34SOSA FROM ISOD0001.ISMS34 WITH (NOLOCK) WHERE I34ASDT = 20251207 -- B198 -- -- B202 -- SELECT SUM(D34TOAM) AS SUM_D34TOAM, SUM(D34OPR) AS SUM_D34OPR, SUM(D34P1PCP) AS SUM_D34P1PCP, SUM(D34P1INT) AS SUM_D34P1INT, SUM(D34P1CRU) AS SUM_D34P1CRU, SUM(D34P1T01) AS SUM_D34P1T01, SUM(D34P1T02) AS SUM_D34P1T02, SUM(D34P1TPP) AS SUM_D34P1TPP, SUM(D34P1T21) AS SUM_D34P1T21, SUM(D34P1T99) AS SUM_D34P1T99, SUM(D34P23) AS SUM_D34P23 FROM ISOD0001.ISMD34 WHERE D34ASDT = 20251207; -- B202 -- -- B208 -- SELECT SUM(ITCALLAC) AS SUM_ITCALLAC, SUM(ITCALLCU) AS SUM_ITCALLCU, SUM(ITCALL0AC) AS SUM_ITCALL0AC, SUM(ITCALL0CU) AS SUM_ITCALL0CU, SUM(ITCALL1AC) AS SUM_ITCALL1AC, SUM(ITCALL1CU) AS SUM_ITCALL1CU, SUM(ITCALL2AC) AS SUM_ITCALL2AC, SUM(ITCALL2CU) AS SUM_ITCALL2CU, SUM(ITCALL3AC) AS SUM_ITCALL3AC, SUM(ITCALL3CU) AS SUM_ITCALL3CU, SUM(ITCALL4AC) AS SUM_ITCALL4AC, SUM(ITCALL4CU) AS SUM_ITCALL4CU, SUM(IPPCALLAC) AS SUM_IPPCALLAC, SUM(IPPCALLCU) AS SUM_IPPCALLCU, SUM(IPPCAL0AC) AS SUM_IPPCAL0AC, SUM(IPPCAL0CU) AS SUM_IPPCAL0CU, SUM(IPPCAL1AC) AS SUM_IPPCAL1AC, SUM(IPPCAL1CU) AS SUM_IPPCAL1CU, </pre>

Functional No.	Acceptance No.	Description
		<pre> SUM(IPPCAL2AC) AS SUM_IPPCAL2AC, SUM(IPPCAL2CU) AS SUM_IPPCAL2CU, SUM(IPPCAL3AC) AS SUM_IPPCAL3AC, SUM(IPPCAL3CU) AS SUM_IPPCAL3CU, SUM(IPPCAL4AC) AS SUM_IPPCAL4AC, SUM(IPPCAL4CU) AS SUM_IPPCAL4CU, SUM(IPDCALLAC) AS SUM_IPDCALLAC, SUM(IPDCALLCU) AS SUM_IPDCALLCU, SUM(IPDCAL0AC) AS SUM_IPDCAL0AC, SUM(IPDCAL0CU) AS SUM_IPDCAL0CU, SUM(IPDCAL1AC) AS SUM_IPDCAL1AC, SUM(IPDCAL1CU) AS SUM_IPDCAL1CU, SUM(IPDCAL2AC) AS SUM_IPDCAL2AC, SUM(IPDCAL2CU) AS SUM_IPDCAL2CU, SUM(IPDCAL3AC) AS SUM_IPDCAL3AC, SUM(IPDCAL3CU) AS SUM_IPDCAL3CU, SUM(IPDCAL4AC) AS SUM_IPDCAL4AC, SUM(IPDCAL4CU) AS SUM_IPDCAL4CU, SUM(IRECCACCU) AS SUM_IRECCACCU, SUM(IRECCCUST) AS SUM_IRECCCUST, SUM(IRECAACCU) AS SUM_IRECAACCU, SUM(IRECACUST) AS SUM_IRECACUST, SUM(ICONC_DIR) AS SUM_ICONC_DIR, SUM(ICONO_PER) AS SUM_ICONO_PER, SUM(ICON_NA) AS SUM_ICON_NA, SUM(ICON_LM) AS SUM_ICON_LM, SUM(ICOMP_CA) AS SUM_ICOMP_CA FROM ISOD0001.ISMS201 WHERE IPERIOD = 25681207; -- B208 -- -- A0001 -- SELECT SUM(IRCL_11) AS SUM_IRCL_11, SUM(IRLAV) AS SUM_IRLAV FROM AS400DB01.ISOD0001.ISMS11RL WHERE ICDTE = 25681207; -- A0001 -- -- A0005 -- SELECT SUM(A21ARO) AS SUM_A21ARO, SUM(A21AME) AS SUM_A21AME, SUM(A21PRE) AS SUM_A21PRE, SUM(A21WOO) AS SUM_A21WOO, SUM(A21WOB) AS SUM_A21WOB, SUM(A21WCO) AS SUM_A21WCO, SUM(A21WPR) AS SUM_A21WPR FROM AS400DB01.ISOD0001.IS321 WHERE A21DTE = '2025-12-07' -- A0005 -- -- A0006 -- SELECT SUM(A25ARO) AS SUM_A25ARO, SUM(A25AME) AS SUM_A25AME, SUM(A25PRE) AS SUM_A25PRE, SUM(A25WOO) AS SUM_A25WOO, </pre>

Functional No.	Acceptance No.	Description
		<pre> SUM(A25WOB) AS SUM_A25WOB, SUM(A25WCO) AS SUM_A25WCO, SUM(A25WPR) AS SUM_A25WPR FROM AS400DB01.ISOD0001.IS325MM WHERE A25DTE = '2025-12-07' -- A0006 -- -- A0021 -- SELECT SUM(IDAYS) AS SUM_IDAYS, SUM(ISALARY) AS SUM_ISALARY, SUM(IETIME1) AS SUM_IETIME1 FROM AS400DB01.ISOD0001.ISMS22 WHERE ICDTE = 25681207; -- A0021 -- </pre>
F-0001	AC-1003	ลดเวลาของ process DayEnd IS (Refer production จาก 7 ชั่วโมง เหลือประมาณ 4 ชั่วโมง) ประมาณ 42.8%

1.3 RISKZ

Risk Information		Risk Analyze	Risk Mitigation
Risk No.	Risk Descriptions	Priority Rating	Plan
1		High	
2		Low	
3		Medium	

1.4 ABBREVIATION / FORMULA

1.4.1 Abbreviation

Abbreviation/Term	Definition

1.4.2 Formula

define	Formula

2.0 REQUIREMENTS ANALYSIS

2.1 FUNCTIONAL REQUIREMENTS

Feature code	Feature Description
F-0001	1. IS2273, IS2273S เป้าหมาย updateAsofIsms2273RL() เป็น Batch Update โดยใช้ PreparedStatement.addBatch() และ executeBatch() เพื่ออัปเดตหลาย record ใน transaction เดียว

File Name:	BCD_68110270_TUNING PERFORMANCE IS PROCESS	Page:	10 OF 28
Confidential and Restricted Circulation			Template Version: V.3.0

Feature code	Feature Description
	<pre> private void [REDACTED]() throws Exception { String business = phms.getValue(); writeTrc2("Update table IS000001.ISMS2273 ISSUE = "+business+"@"+(int)(Math.random()*10)); int count = 0; String sql = "UPDATE IS000001.ISMS2273 SET ISMS2273.CREATED = "+created.getYearValue() + " "+ created.getMonthValue() + " "+ created.getDateValue() + " "+ business+"@"+(int)(Math.random()*10); log.info("Update table by statement : "+sql); try{ Statement statement = connection.createStatement(); statement.executeUpdate(); connection.commit(); statement.close(); }catch (Exception e){ writeLog("Update table IS000001.ISMS2273 ISSUE = "+business+"@"+(int)(Math.random()*10)); writeTrc2("Update table IS000001.ISMS2273 ISSUE = "+business+"@"+(int)(Math.random()*10)); connection.rollback(); } } log.info("End Update table IS000001.ISMS2273 ISSUE = "+business+"@"+(int)(Math.random()*10)); </pre>

เปลี่ยน Logic การแบ่ง SUBJOB จากเดิม loop row by row เป็นการหา min/max ของ การ SUBJOB แต่ละตัว

```

private void [REDACTED] batchSize) throws Exception{
    String sqlTemplate = "WITH offset AS ("
        + "SELECT top(1) [P73CMT] FROM RL00001.RLMS73 AS T"
        + "WHERE IP73CMT > N' "
        + "AND ((IP73CBL = 0 OR IP73CBL) >= 'N') AND (IP73CBL <= 'X') "
        + "AND (IP73CBL) IN (Select R25C19 FROM CS000001.CSMS25)) "
        + "GROUP BY [P73CMT] ORDER BY [P73CMT] "
        + "Select ISNULL(CAST(max([P73CMT]) AS varchar),'''').ISNULL(CAST(max([P73CMT]) AS varchar),'''') from offset";
    String sql = String.format(sqlTemplate,batchSize,min.getValue(),max.getValue());
    Connection ds = getReadOnlyConnection(sql);
    ResultSet rs = null;
    try{
        writeLog(sql);
        writeTrc2(sql, "INFO");
        rs = ds.createStatement().executeQuery(sql);
        while(rs.next()){
            scontunning = rs.getString("scontunning");
            econtunning = rs.getString("econtunning");
        }
    }catch(SQLException e){
        writeLog(e.toString());
    }
}

```

```

private void AB1() throws Exception{
    writeTrc2("AB1()",getLineNumber());
    int batchSize = Integer.parseInt(trnrl.getValue());
    while(true{
        if(batchSize>0){
            writeLog("Batchsize");
            if(StringUtils.isNotBlank(scontunning) && StringUtils.isNotBlank(econtunning)){
                jsc.setValue(StringUtils.valueOf(jsc.getValue())+Long.parseLong((sc.getValue().trim().length()+(1+jsc.getValue().trim().length()))));
            }
            scnt.setValue(scontunning);
            scnt.setValue(econtunning);

            AB1();
        }else{
            break;
        }
    }
}

```

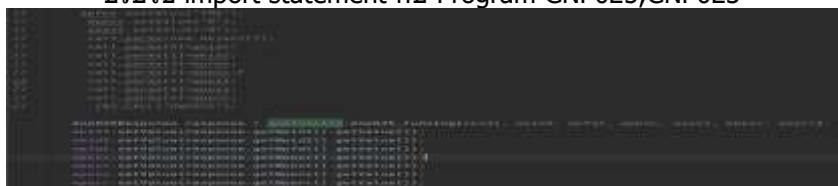
- เพิ่มการ Prepare Data สำหรับข้อมูลจาก CSMS25, RLMS73 ด้วย HashMap (query ครั้งเดียวแล้ว reuse ใน loops)

```

private void queryAllCms25() throws Exception{
    cms25CidList = new ArrayList<>();
    StringBuilder sql = new StringBuilder();
    sql.append("select M25CID from CS000001.CSMS25 group by M25CID");

    Connection ds = getReadOnlyConnection(sql.toString());
    ResultSet rs = null;
    try{
        writeLog(sql.toString());
        writeTrc2(sql.toString(), "INFO");
        rs = ds.createStatement().executeQuery(sql.toString());
        while(rs.next()){
            String cid = rs.getString("M25CID");
            cms25CidList.add(cid);
        }
    }catch(SQLException e){

```


Feature code	Feature Description
	<p>4. IS175 เตรียม Prepare Data สำหรับ Table ที่มีข้อมูลน้อยและถูก fetch บ่อย ๆ ด้วย HashMap สำหรับทุก queryMap (เช่น gntb16Map, gnts17Map)</p> <pre data-bbox="306 343 1144 714"> private void AMS01LG() throws Exception{ writeTrc(str: "AMS01LG()", getLineNumber()); //query prepare Map<String, String> gntb262Map = queryGntb262(); Map<String, String> rltb31Map = queryRltb31(); Map<String, String> rltb01Map = queryRltb01(); queryGntb61(); queryGntb16(); queryGnts16(); queryGnts17(); //##### Z-ADD *ZERO PXCONT 16 0### pxcont.setValue(CommonUtils.getSameCharStr(len: 16, cc: '0')); } </pre> <pre data-bbox="306 747 1144 1080"> String gntb262sto = gntb262Map.get(rlm01.get01Index()).get02Value(); //### IF NOTFOUND(GNTB262) ## if (!StringUtil.isEmpty(gntb262sto)){ gntb262.setValue(gntb262sto); } //### LEVEL(P) BASED IP02E00 lmss22.setValue(lmss22.isTrue(true, lmss22.get02e02.getValue()); //### } //### ELSEIF } //### ELSE } </pre> <p>เพิ่มเงื่อนไขการดึงข้อมูล Table RLMS01 เอาเฉพาะข้อมูลที่ใช้</p> <pre data-bbox="306 1136 1144 1522"> rlm02.setFilter("RLMS01 = "+rlm01.get01Index() + " AND RLAF01 = "+rlm01.get01Index()); rlm02.set2nd(true); rlm02.setcursor(); //### PRIM1 : PRIM2 : EPLSQL0000 if (rlm02.cursor.nextRow()) lms02.setSqlCode(1); else lms02.setSqlCode(2); if (rlm01.get01Index() > 0) if (Long.parseLong(lms02.getValue().trim(), 10) > Long.parseLong(rlm01.get01Index())) lms02.setSqlCode(3); //### OSB SQLCODE(8192)1000 while(rlm02.cursor.getValue() != null){ if (RLAF01 = 'S' AND RLAF02 = 'Y' AND RLAF03 = 'N' AND RLAF04 = 'N') ### if (CommonUtil.Equal("LMS01", lms02.getValue(), "S") && CommonUtil.Equal("RLMS01", lms02.getValue(), "S")) } </pre> <p>5. IS182S, IS182 – Statement Generation Preparation ปรับใช้ import statement กับ Program GNP025, GNP023</p> 

Feature code	Feature Description
	<p>6. ISGPREJOBRL, ISCPYF03, ISCPYF02 – Daily Backup Data</p> <p>1. Java 8 Native Batch Processing</p> <ul style="list-style-type: none"> - ออกแบบโครงสร้าง Job ให้รองรับการ Re-run และ Idempotency เช่น การ Initial ค่าตั้งต้น, การเตรียม Execution Context, และการ Mapping Value ที่จำเป็นก่อนเริ่มทำงาน - แยกขั้นตอนการสร้าง SQL Statement ออกเป็น Builder ชุดเดียว เพื่อเพิ่มความอ่านง่าย ลดการทำซ้ำช้อน และใช้ช้าได้ในการ Execute หลายรอบ - กำหนดมาตรฐานในการบันทึกสถานะงาน (Success/Fail) ลงตาราง Monitoring เพื่อรับ Retry และป้องกันการทำงานซ้ำโดยไม่ตั้งใจ (duplicate execution) <p>2. Multi-Worker Processing และ Parallelism</p> <p>ออกแบบระบบเป็น Worker หลายตัวเพื่อรองรับงานแบบ Parallel ลด Time-to-Completion โดยแบ่งข้อมูลเป็นชุด (Partition) ตามปริมาณงานของ Schema+Table มีการจัดการ Exception ที่เป็นมาตรฐานในทุก Worker เพื่อให้การ Re-run ทำได้ถูกต้องโดยไม่กระทบข้อมูลหลัก</p> <p>3. Index Management (DISABLE / REBUILD สำหรับ NONCLUSTERED Indexes)</p> <ul style="list-style-type: none"> - ปิด (DISABLE) NONCLUSTERED Indexes ก่อนเริ่ม Batch Insert เพื่อเพิ่ม Performance และลด Fragmentation - ทำการ REBUILD Index หลังงานจบ เพื่อคืนประสิทธิภาพการค้นหาและป้องกันปัญหาจาก Index เสียรูป (fragmented) - ใช้เฉพาะ Index ที่จำเป็นต่อ Query จริง ๆ เพื่อลด Overhead ของการ Maintain Index ในระหว่างโหลดข้อมูล <p>4. Staging Table Workflow</p> <ul style="list-style-type: none"> - ออกแบบ Data Flow ให้ชัดเจนระหว่าง Staging (RLMS01STG) → Master (RLMS01) → History (RLMS01H) เช่น Insert ข้อมูลเข้า Staging (RLMS01STG), Delete ข้อมูล Staging ตามเงื่อนไข, Insert ข้อมูลเข้า Table หลัก RLMS01, Insert ลง Table History (RLMS01H) <p><u>รูปแบบการทำงานของ Staging table:</u></p> <p>File03: Insert-BIOD0001.RLMS01 → BIOD0001.RLMS01H (ผ่านขั้นตอนกรองข้อมูลซ้ำมาแล้ว)</p> <p>File03: Insert-BIOD0001.RLMS01STG → BIOD0001.RLMS01H (ผ่านขั้นตอนกรองข้อมูลซ้ำมาแล้ว)</p> <p>File02: Insert-RLOD0001.RLMS01 → BIOD0001.RLMS01STG</p> <p>File02: Delete-BIOD0001.RLMS01STG เฉพาะข้อมูลที่ซ้ำออก</p> <p>File02: Insert Distinct-RLOD0001.RLMS01 → BIOD0001.RLMS01 (นำเข้าเฉพาะข้อมูลที่ซ้ำมากรอง)</p>

Feature code	Feature Description
	<p>7. IS2271, IS2271S</p> <p>เปลี่ยน Logic การแบ่ง SUBJOB จากเดิม loop row by row เป็นการหา min/max ของ การ SUBJOB แต่ละตัว</p> <pre data-bbox="314 399 1441 1403"> private void fetchData(int batchSize) throws Exception{ String sqlTemplate = "with offset as (" + "select top(%s) [P19CSN] from #L000001_RLMS19 ms " + "where [P19CSN] > ms " + "AND [P19PJB] >= '%s'" + "group by ([P19CSN]) order by ([P19CSN]) " + ")select ISNULL(CAST(min(P19CSN) AS varchar), '') ,ISNULL(CAST(max(P19CSN) AS varchar), '') from offset"; String sql = String.format(sqlTemplate,batchSize,ecsn.getDbValue(),pxdate8.getDbValue()); Connection ds = getReadOnlyConnection(sql); ResultSet rs = null; try{ writeLog(sql); writeTrc2(sql, lineNo+"S"); rs = ds.createStatement().executeQuery(sql); while(rs.next()){ sconTunning = rs.getString("columnIndex_1"); econTunning = rs.getString("columnIndex_2"); } }catch(SQLException e){ writeTrc2(sql, lineNo+"E"); } } private void ARI() throws Exception{ writeTrc2("ARI",getLineNumber()); //SQL SELECT RLMS19 FILTER([P19PJB] >= :PXDATE8).COLUMN(P19CSN P19PJB) int batchSize = Integer.parseInt(jtrnrl.getValue()); while(true){ fetchData(batchSize); if(StringUtil.isNotBlank(sconTunning) && StringUtil.isNotBlank(econTunning)){ job.setValue(String.valueOf(Long.parseLong(job.getValue().trim().length()==0?"0":job.getValue().trim())+1)); scon.setValue(sconTunning); econ.setValue(econTunning); ASJB(); }else{ break; } } } </pre> <p>- เพิ่มการ Prepare Data สำหรับข้อมูลจาก CSMS25, RLMS19 ด้วย HashMap (query ครั้งเดียวแล้ว reuse ใน loops)</p>

Feature code	Feature Description
<pre> private List<String> queryAllCsms25() throws Exception{ StringBuilder sql = new StringBuilder(); sql.append("select M25CID from CS00001.CSMS25 with(nolock) group by M25CID"); Connection ds = getReadOnlyConnection(sql.toString()); ResultSet rs = null; List<String> csms25CidList = new ArrayList<>(); try{ writeLog(sql.toString()); writeTrc2(sql.toString(), lineNumber: "0"); rs = ds.createStatement().executeQuery(sql.toString()); while(rs.next()){ String cid = rs.getString(columnIndex: 1); csms25CidList.add(cid); } }catch(SQLException e){ String nbr151=HboxCoreServer.getNbrFromOverDbf(transactionId, tableName: "R1MS19"); if(nbr151!=null){ nbr151+=""+nbr151+""; rms19rrl.addFilter(nbr151); } rms19rrl.setLimit(true); rms19rrlCursor=rms19rrl.openCursor(); } List<String> stoRms19List = new ArrayList<>(); while (rms19rrlCursor.readNext()) { stoRms19List.add(rms19rrl.getStoValue()); } return stoRms19List; } //AAA// 000 001/002/003/004 for(String stoRms19 : stoRms19List){ rms19rrl.setStoValue(stoRms19); //AAA// NOVEL(P) *BLANKS FNOVIC LARR fnocid.initializeValue(); //AAA// 1 00 N 4 3 0000 boolean isFoundCid = csms25CidList.stream().anyMatch(cid -> { try { return cid.trim().equalsIgnoreCase(rms19rrl.getP19pou().getDvValue().trim()); } catch (Exception e) { throw new RuntimeException(e); } }); //AAA// IF RLP19P00 = C1C1(J) *** if (isFoundCid){ //AAA// NOVEL(P) "Y" FNOVIC0000 fnocid.setValue(initialized: true, enabled: true, value: "Y"); } } </pre>	

Feature code	Feature Description
	<p>10. ISG321RLS</p> <ul style="list-style-type: none"> - ปรับใช้ Memcache ไข้กับ ตาราง ischan, isappt, iscamp <pre> private String query_ISCAMP(String wxbust, String wxcamp) throws Exception{ //NNNN// KXCOMP CHAIN ISCAMP NNN iscamp.setFilter(""); iscamp.addFilter("[CAMK01] = " + wxbust + "'"); iscamp.addFilter("[CAMK02] = " + wxcamp + " WITH (NOLOCK)"); //filter was added String[] orderFields43 = new String[2]; orderFields43[0] = "[CAMK01]"; orderFields43[1] = "[CAMK02]"; iscamp.setOrderFields(orderFields43); if (iscampCursor != null &amp; iscampCursor.isOpened()) iscampCursor.close(); iscamp.setIndexName("-1"); String tableName44 = HboxCoreServer.getTabNameFromOverDbf(transactionId, TableName.ISCAMP); if (tableName44 != null) { if (tableName44.startsWith("QTEMP")) tableName44 = tableName44.replaceFirst("QTEMP", "ISCAMP"); iscamp.setTableName(tableName44); } String mbr45 = HboxCoreServer.getMbrFromOverDbf(transactionId, TableName.ISCAMP); if (mbr45 != null) { mbr45 = "MBR=" + mbr45 + ""; iscamp.addFilter(mbr45); } iscamp.setLimit(false); iscamp.setStrictMode(true); iscampCursor = iscamp.openCursor(); if (iscampCursor.readNext()) iscamp.setSqlcode(0); else iscamp.setSqlcode(100); return iscamp.getSqlValue(); } </pre> <ul style="list-style-type: none"> - ปรับใช้ Preload Data ไข้กับ ตาราง csms13, Rlms01, Rlms02 <pre> private void query_Rlms02() { try{ //NNNN// RLMS02 SETL RLMS02 FILTER([P30EL]) = 0 AND [P2CNOT1] = 0 AND [P2CNOT2] = 0 AND [P2CRN] rlms02.setFilter(""); rlms02.addFilter("[P2CRN] = " + minnum.getDbValue() + "'"); rlms02.addFilter("[P2CNOT1] = " + minnum.getDbValue() + "'"); rlms02.addFilter("[P2CNOT2] = " + minnum.getDbValue() + "'"); rlms02.addFilter("[P2CNOT3] = " + minnum.getDbValue() + "'"); rlms02.addFilter("[P2CNOT4] = 0 AND [P2CNO1] = 0 AND [P2CNO2] = 0 AND [P2CNO3] = 0 WITH (INCLUDE0)"); //filter was added String[] orderFields25 = new String[]{}; orderFields25[0] = "[P2CRN]"; orderFields25[1] = "[P2CNOT1]"; rlms02.setOrderFields(orderFields25); if (rlms02Cursor != null &amp; rlms02Cursor.isOpened()) rlms02Cursor.close(); rlms02.setIndexName("-1"); String tableName26 = HboxCoreServer.getTabNameFromOverDbf(transactionId, TableName.RLMS02); if (tableName26 != null) { if (tableName26.startsWith("QTEMP")) tableName26 = tableName26.replaceFirst("QTEMP", "RLMS02"); rlms02.setTableName(tableName26); } String mbr27 = HboxCoreServer.getMbrFromOverDbf(transactionId, TableName.RLMS02); if (mbr27 != null) { mbr27 = "MBR=" + mbr27 + ""; rlms02.addFilter(mbr27); } //begin setSelectFields String[] fields = new String[13]; } } </pre>

Feature code	Feature Description
	<p>11. IS303RLS - ปรับใช้ PrepareData เตรียมไว้</p> <pre> private void queryRlms40() throws Exception { rlms40Map = new HashMap<>(); rlms40.setFilter(""); rlms40.addFilter("[P40CNT] >= '"+wiscnt.getDbValue()+"'"); rlms40.addFilter("[P40CNT] <= '"+wiecnt.getDbValue()+"'"); rlms40.addFilter("[P40FL6] = 'C'"); rlms40.addFilter("[P40DDT] = '"+wxasdtc.getDbValue()+"' WITH (NOLOCK)"); //filter was added String[] orderFields40 = new String[2]; orderFields40[0]=[P40CNT]; orderFields40[1]=[P40DDT]; rlms40.setOrderFields(orderFields40); if(rlms40Cursor!=null &&rlms40Cursor.isOpened()) rlms40Cursor.close(); rlms40.setIndexName("-1"); String tableName47=HboxCoreServer.getTabNameFromOverDbf(transactionId, fileName: "RLMS40L9"); if(tableName47!=null){...} String mbr48=HboxCoreServer.getMbrFromOverDbf(transactionId, fileName: "RLMS40L9"); if(mbr48!=null){...} rlms40.setLimit(true); rlms40Cursor=rlms40.openCursor(); //####// KXMS40 READE RLMS40L9##### while (rlms40Cursor.readNext()) { String contract = rlms40.getP40cnt().getDbValue(); rlms40Map.computeIfAbsent(contract, k -> new ArrayList<>()) .add(rlms40.getStoValue()); } } List<String> rlms40List = new ArrayList<>(); String p2Cont = rlms02.getP2cont().getDbValue(); if(rlms40Map.get(p2Cont) != null){ rlms40List.addAll(rlms40Map.get(p2Cont)); } //####// DOW NOT%EOF(RLMS40L9)##### for(String rlms40Sto : rlms40List){ rlms40.setStoValue(rlms40Sto); //####// ADD P40AMT I33WDAMT##### isms33.getI33wdamt().setValue(isms33.getI33wdamt().getValue().add(rlms40.getP40amt().getValue())); //####// ADD 1 I33WDTIM##### isms33.getI33wdtim().setValue(isms33.getI33wdtim().getValue().add(new BigDecimal(val: 1))); //####// ENDDO##### }</pre>

Feature code	Feature Description
	<p>- ปรับใช้ Native Delete แทนการใช้ Hisun query</p> <pre data-bbox="306 298 1441 1558"> private final static String DELETE_STATEMENT = "delete from ISOD0001.ISMS33 where " + "[I33ASDT] = '%s' " + "and [I33APP] = '%s' " + "and [I33BRN] = '%s' " + "and [I33CONT] = '%s' "; 1 usage 1 SIAMAC\o4507002 private void deleteIsms33() throws Exception { String asOf = kxasdte.getDbValue(); String app = wxapp.getDbValue(); String brn = rlms02.getP2brn().getDbValue(); String cont = rlms02.getP2cont().getDbValue(); String deleteStatement = String.format(DELETE_STATEMENT,asOf,app,brn,cont); writeTrc2(str "Delete table ISOD0001.ISMS33 : " + deleteStatement, lineNumber: "0"); int count = 0; log.info("Delete table by statement : {}",deleteStatement); try{ Statement statement = connection.createStatement(); count = statement.executeUpdate(deleteStatement); connection.commit(); statement.close(); wxdued.setValue(2); } String rltb05Sto = rltb05Map.get(rlms02.getP2cutd().getDbValue()); //##### IF %FOUND(RLTB05L1) ##### if (!StringUtils.isEmpty(rltb05Sto)){ rltb05.setStoValue(rltb05Sto); //##### Z-ADD T050DA WXUED### wxdued.setValue(rltb05.getT05dda().getValue()); //##### ENDIF### } //##### EYSR AWRT 01### } </pre>

Feature code	Feature Description
	<pre> 1 usage ▲ SIAMAC\o4507002 * private Map<String, String> queryRltb05Map() throws Exception { Map<String, String> rltb05Map = new HashMap<>(); rltb05.setFilter(""); rltb05.addFilter("[T05DEL] = ' '"); //filter was added String[] orderFields25 = new String[1]; orderFields25[0]=[T05CDA]; rltb05.setOrderFields(orderFields25); if(rltb05Cursor!=null &&rltb05Cursor.isOpened()) rltb05Cursor.close(); rltb05.setIndexName("-1"); String tableName26=HboxCoreServer.getTabNameFromOverDbf(transactionId, fileName: "RLTB05L1"); if(tableName26!=null){...} String mbr27=HboxCoreServer.getMbrFromOverDbf(transactionId, fileName: "RLTB05L1"); if(mbr27!=null){...} rltb05.setLimit(false); {...} rltb05.setStrictMode(true); rltb05Cursor=rltb05.openCursor(); while (rltb05Cursor.readNext()) { rltb05Map.putIfAbsent(rltb05.getT05cda().getDbValue(),rltb05.getStoValue()); } } String rltb01sto = rltb01Map.get(rms02.getP2brn().getDbValue()); //###/ IF \$FOUND(RLTB01L1) ### if (!StringUtils.isEmpty(rltb01sto)){ rltb01.setStoValue(rltb01sto); //###/ EVAL ISSAREA = SUBST(TIAREA:1:2) ### isms33.getIssarea().setValue(align: true, needPad: true,(rltb01.getTiarea().getStoValue()).substring(0,2)); //###/ ENDIF### } //###/ Z-ADD EXDUD ISSDUD ### isms33.getIssdue().setValue(exdud.getValue()); </pre>

Feature code	Feature Description
	<pre> private void queryRltb01() throws Exception { rltb01Map = new HashMap<>(); rltb01.setFilter(""); rltb01.addFilter("[T1CMCD] = 1 OR [T1SYST] = '02' OR [T1DEL] = '')"); //filter was added String[] orderFields34 = new String[1]; orderFields34[0]=[T1BRN]; rltb01.setOrderFields(orderFields34); if(rltb01Cursor!=null &&rltb01Cursor.isOpened()) rltb01Cursor.close(); rltb01.setIndexName("-1"); String tableName35=HboxCoreServer.getTabNameFromOverDbf(transactionId, fileName: "RLTB01L1"); if(tableName35!=null){...} String mbr36=HboxCoreServer.getMbrFromOverDbf(transactionId, fileName: "RLTB01L1"); if(mbr36!=null){...} rltb01.setLimit(false); rltb01.setStrictMode(true); rltb01Cursor=rltb01.openCursor(); while (rltb01Cursor.readNext()){ rltb01Map.putIfAbsent(rltb01.getT1brn().getDbValue(),rltb01.getStoValue()); } } </pre>

12. ISGM07

- ปรับการใช้ Batch Insert เพื่อลดจำนวน transaction

```

}
try (Statement st = connection.createStatement()) {
    int count = st.executeUpdate(" INSERT INTO IS000001.CSMS07TM ("+getColumnStr2(csmwkr)+") " +
        "SELECT "+getColumnStr2(csmwkr)+" FROM "+csmwkr.getTableNames());
    log.info("Insert : "+count+" , "+CSMS07TM.count);
}

```

13. IS1701CS

- ปรับเรื่อง memcache (RLTB05) เพื่อลดการ connection db



Feature code	Feature Description
	<p>- ปรับการใช้ Batch Delete ISMS11TM เพื่อลด header ในการส่ง query</p> <pre> private void delete_ISMS11TM(Long batchSize) throws Exception{ log.info("Delete ISMS11TM : {} ", batchSize); Connection conn = null; String jwt = thirdPartyUtility.getJwtToken(); conn = thirdPartyUtility.getConnection(jwt, "CENTER"); StringBuilder sql = new StringBuilder(); sql.append("with offset as (select top "+batchSize+" * from ISMS11TM WITH (NOLOCK))"); sql.append("where ISMS11ID IN (update.getDbValue() and (ISMS11ID between "+conn.getDbValue()+" and "+conn.getDbValue()+"))"); sql.append(")"); sql.append("\n"); sql.append("DELETE FROM offset"); int delete = 0; conn.setAutoCommit(false); try(Statement st = conn.createStatement()){ do{ delete = st.executeUpdate(sql.toString()); log.info("Delete ISMS11TM : {} ", delete); } while(delete > 0); conn.commit(); } catch(SQLException e){ writeLog(e.toString()); writeLog(e.toString(), "0"); writeException(e); conn.rollback(); } } </pre>

2.2 NON-FUNCTIONAL REQUIREMENTS

Non-Functional	
Need Code	Description

2.3 INTERFACE REQUIREMENT

No	Description	System	Link Ref.

2.4 TESTING SCENARIO

File Name:	BCD_68110270_TUNING PERFORMANCE IS PROCESS	Page:	24 OF 28
<i>Confidential and Restricted Circulation</i>		Template Version:	V.3.0

Acceptance No.	Test case No.	Test Scenarios
		<p>After Run Process Batch IS</p> <p>ทำการตรวจสอบจำนวน Data เปรียบเทียบกับจำนวน Data Production ที่มีวันที่ตรงกัน ของแต่ละ Table ดังนี้</p>
AC-1001	TS-1001	<p>ตรวจสอบข้อมูลของ Table ISMS34, ISMD34, ISMS201, ISMS11RL, IS321, IS325MM, ISMS22, ISMS206, ISMD236 , ISMS231, ISMS33, ISMD33, ISMS31, ISMS32, IS325</p> <p>ตาม statement record count ดังนี้</p> <pre> SELECT (SELECT COUNT(*) FROM ISOD0001.ISMS34 WITH (NOLOCK) WHERE I34ASDT = 20251207)B198, (SELECT COUNT(*) FROM ISOD0001.ISMD34 WITH (NOLOCK) WHERE D34ASDT = 20251207)B202, (SELECT COUNT(*) FROM ISOD0001.ISMS201 WITH (NOLOCK) WHERE IPERIOD = 25681207)B208 SELECT (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS11RL WITH (NOLOCK) WHERE ICDTE = 25681207) A0001, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS321 WITH (NOLOCK) WHERE A21DTE = '2025-12-07') A0005, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS325MM WITH (NOLOCK) WHERE A25DTE = '2025-12-07') A0006, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS22 WITH (NOLOCK) WHERE ICDTE = 25681207) A0021, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS206 WITH (NOLOCK) WHERE IPERIOD = 25681207) A0044, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMD236 WITH (NOLOCK) WHERE DASOF = 25681207) A0057, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS231 WITH (NOLOCK) WHERE IPERIOD = 25681207) A0064, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS33 WITH (NOLOCK) WHERE I33ASDT = 20251207) A0109, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMD33 WITH (NOLOCK) WHERE D33ASDT = 20251207) A0111,sssss WITH (NOLOCK) WHERE M31ASDT = 20251207) A0132, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS32 WITH (NOLOCK) WHERE M32ASDT = 20251207) A0133, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS325_12 WITH (NOLOCK) WHERE A25DTE = '2025-12-07') A0134 SELECT (SELECT COUNT(*) FROM ISOD0001.ISMS34 WITH (NOLOCK))B198, (SELECT COUNT(*) FROM ISOD0001.ISMD34 WITH (NOLOCK))B202, (SELECT COUNT(*) FROM ISOD0001.ISMS201 WITH (NOLOCK))B208 SELECT (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS11RL WITH (NOLOCK))A0001, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS321 WITH (NOLOCK))A0005, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.IS325MM WITH (NOLOCK))A0006, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS22 WITH (NOLOCK))A0021, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS206 WITH (NOLOCK))A0044, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMD236 WITH (NOLOCK))A0057, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS231 WITH (NOLOCK))A0064, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS33 WITH (NOLOCK))A0109,</pre>

		<pre>(SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMD33 WITH (NOLOCK))A0111, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS31 WITH (NOLOCK))A0132, (SELECT COUNT(*) FROM AS400DB01.ISOD0001.ISMS32 WITH (NOLOCK))A0133,</pre> <p style="color: red;">Remark : วันที่ ที่เป็นสีแดง จะเปลี่ยนแปลงตามวันที่ที่ต้องการตรวจสอบ</p>
AC-1002	TS-1002	<p>เช็คผลรวมข้อมูล (Sum) เพื่อตรวจสอบผลรวมของยอดต่าง ๆ หรือ -- B198 --</p> <pre>SELECT SUM(I34PRIC) AS SUM_I34PRIC, SUM(I34TOAM) AS SUM_I34TOAM, SUM(I34PCAM) AS SUM_I34PCAM, SUM(I34UIDA) AS SUM_I34UIDA, SUM(I34INTR) AS SUM_I34INTR, SUM(I34CRUR) AS SUM_I34CRUR, SUM(I34DOWN) AS SUM_I34DOWN, SUM(I3412IN) AS SUM_I3412IN, SUM(I34NDUE) AS SUM_I34NDUE, SUM(I34OPR) AS SUM_I34OPR, SUM(I34OIN) AS SUM_I34OIN, SUM(I34OUS) AS SUM_I34OUS, SUM(I34OCOL) AS SUM_I34OCOL, SUM(I34OOTH) AS SUM_I34OOTH, SUM(I34P1PCP) AS SUM_I34P1PCP, SUM(I34P1INT) AS SUM_I34P1INT, SUM(I34P1CRU) AS SUM_I34P1CRU, SUM(I34P1PNT) AS SUM_I34P1PNT, SUM(I34P1OTH) AS SUM_I34P1OTH, SUM(I34P1T01) AS SUM_I34P1T01, SUM(I34P1T02) AS SUM_I34P1T02, SUM(I34P1TPP) AS SUM_I34P1TPP, SUM(I34P1T21) AS SUM_I34P1T21, SUM(I34P1T99) AS SUM_I34P1T99, SUM(I34ODAM) AS SUM_I34ODAM, SUM(I34SOSA) AS SUM_I34SOSA FROM ISOD0001.ISMS34 WITH (NOLOCK) WHERE I34ASDT = 20251207 -- B198 -- -- B202 -- SELECT SUM(D34TOAM) AS SUM_D34TOAM, SUM(D34OPR) AS SUM_D34OPR, SUM(D34P1PCP) AS SUM_D34P1PCP, SUM(D34P1INT) AS SUM_D34P1INT, SUM(D34P1CRU) AS SUM_D34P1CRU, SUM(D34P1T01) AS SUM_D34P1T01, SUM(D34P1T02) AS SUM_D34P1T02, SUM(D34P1TPP) AS SUM_D34P1TPP, SUM(D34P1T21) AS SUM_D34P1T21, SUM(D34P1T99) AS SUM_D34P1T99, SUM(D34P23) AS SUM_D34P23 FROM ISOD0001.ISMD34 WHERE D34ASDT = 20251207; -- B202 -- -- B208 -- SELECT</pre>

File Name:	BCD_68110270_TUNING PERFORMANCE IS PROCESS	Page:	26 OF 28
<i>Confidential and Restricted Circulation</i>		Template Version:	V.3.0

		<pre> SUM(ITCALLAC) AS SUM_ITCALLAC, SUM(ITCALLCU) AS SUM_ITCALLCU, SUM(ITCALL0AC) AS SUM_ITCALL0AC, SUM(ITCALL0CU) AS SUM_ITCALL0CU, SUM(ITCALL1AC) AS SUM_ITCALL1AC, SUM(ITCALL1CU) AS SUM_ITCALL1CU, SUM(ITCALL2AC) AS SUM_ITCALL2AC, SUM(ITCALL2CU) AS SUM_ITCALL2CU, SUM(ITCALL3AC) AS SUM_ITCALL3AC, SUM(ITCALL3CU) AS SUM_ITCALL3CU, SUM(ITCALL4AC) AS SUM_ITCALL4AC, SUM(ITCALL4CU) AS SUM_ITCALL4CU, SUM(IPPCALLAC) AS SUM_IPPCALLAC, SUM(IPPCALLCU) AS SUM_IPPCALLCU, SUM(IPPCAL0AC) AS SUM_IPPCAL0AC, SUM(IPPCAL0CU) AS SUM_IPPCAL0CU, SUM(IPPCAL1AC) AS SUM_IPPCAL1AC, SUM(IPPCAL1CU) AS SUM_IPPCAL1CU, SUM(IPPCAL2AC) AS SUM_IPPCAL2AC, SUM(IPPCAL2CU) AS SUM_IPPCAL2CU, SUM(IPPCAL3AC) AS SUM_IPPCAL3AC, SUM(IPPCAL3CU) AS SUM_IPPCAL3CU, SUM(IPPCAL4AC) AS SUM_IPPCAL4AC, SUM(IPPCAL4CU) AS SUM_IPPCAL4CU, SUM(IPDCALLAC) AS SUM_IPDCALLAC, SUM(IPDCALLCU) AS SUM_IPDCALLCU, SUM(IPDCAL0AC) AS SUM_IPDCAL0AC, SUM(IPDCAL0CU) AS SUM_IPDCAL0CU, SUM(IPDCAL1AC) AS SUM_IPDCAL1AC, SUM(IPDCAL1CU) AS SUM_IPDCAL1CU, SUM(IPDCAL2AC) AS SUM_IPDCAL2AC, SUM(IPDCAL2CU) AS SUM_IPDCAL2CU, SUM(IPDCAL3AC) AS SUM_IPDCAL3AC, SUM(IPDCAL3CU) AS SUM_IPDCAL3CU, SUM(IPDCAL4AC) AS SUM_IPDCAL4AC, SUM(IPDCAL4CU) AS SUM_IPDCAL4CU, SUM(IRECCACCU) AS SUM_IRECCACCU, SUM(IRECCCUST) AS SUM_IRECCCUST, SUM(IRECAACCU) AS SUM_IRECAACCU, SUM(IRECACUST) AS SUM_IRECACUST, SUM(ICONC_DIR) AS SUM_ICONC_DIR, SUM(ICONO_PER) AS SUM_ICONO_PER, SUM(ICON_NA) AS SUM_ICON_NA, SUM(ICON_LM) AS SUM_ICON_LM, SUM(ICOMP_CA) AS SUM_ICOMP_CA FROM ISOD0001.ISMS201 WHERE IPERIOD = 25681207; -- B208 -- -- A0001 -- SELECT SUM(IRCL_11) AS SUM_IRCL_11, SUM(IRLAV) AS SUM_IRLAV FROM AS400DB01.ISOD0001.ISMS11RL WHERE ICDTE = 25681207; -- A0001 -- -- A0005 -- SELECT </pre>
--	--	---

		<pre> SUM(A21ARO) AS SUM_A21ARO, SUM(A21AME) AS SUM_A21AME, SUM(A21PRE) AS SUM_A21PRE, SUM(A21WOO) AS SUM_A21WOO, SUM(A21WOB) AS SUM_A21WOB, SUM(A21WCO) AS SUM_A21WCO, SUM(A21WPR) AS SUM_A21WPR FROM AS400DB01.ISOD0001.IS321 WHERE A21DTE = '2025-12-07' -- A0005 -- -- A0006 -- SELECT SUM(A25ARO) AS SUM_A25ARO, SUM(A25AME) AS SUM_A25AME, SUM(A25PRE) AS SUM_A25PRE, SUM(A25WOO) AS SUM_A25WOO, SUM(A25WOB) AS SUM_A25WOB, SUM(A25WCO) AS SUM_A25WCO, SUM(A25WPR) AS SUM_A25WPR FROM AS400DB01.ISOD0001.IS325MM WHERE A25DTE = '2025-12-07' -- A0006 -- -- A0021 -- SELECT SUM(IDAYS) AS SUM_IDAYS, SUM(ISALARY) AS SUM_ISALARY, SUM(IETIME1) AS SUM_IETIME1 FROM AS400DB01.ISOD0001.ISMS22 WHERE ICDTE = 25681207; -- A0021 -- Remark : วันที่ ที่เป็นสีแดง จะเปลี่ยนแปลงตามวันที่ที่ต้องการตรวจสอบ </pre>
AC-1003	TS-1003	<p>ตรวจสอบเวลาในการ Run process Day End IS จะต้องลดลงประมาณ 42.8%</p> <p>Refer : production จาก 7 ชั่วโมง เหลือประมาณ 4 ชั่วโมง</p>

3.0 SCHEDULE

No	Events	Start	Finish	Duration (Days)
1	Requirement	10/11/2025	14/11/2025	5
2	Design & Develop	17/11/2025	12/12/2025	18
3	SIT			
4	UAT	23/12/2025	25/12/2025	5
5	Go-live	26/12/2025		1