IMS 13

Migration to IMS V13: help reduce the costs and leverage the opportunities

- IMS Configuration Manager V2R1
- IMS Performance Analyzer V230
- IBM Transaction Analysis Workbench V110

Information Management software

© 2014 IBM Corporation

Please note

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

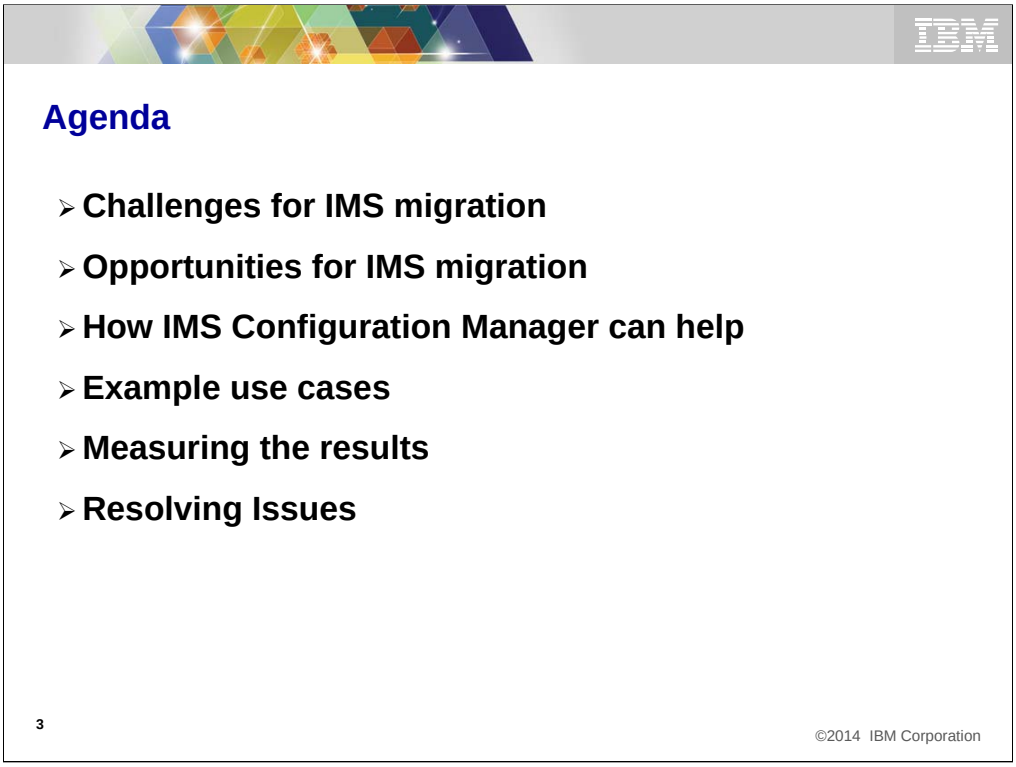
The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

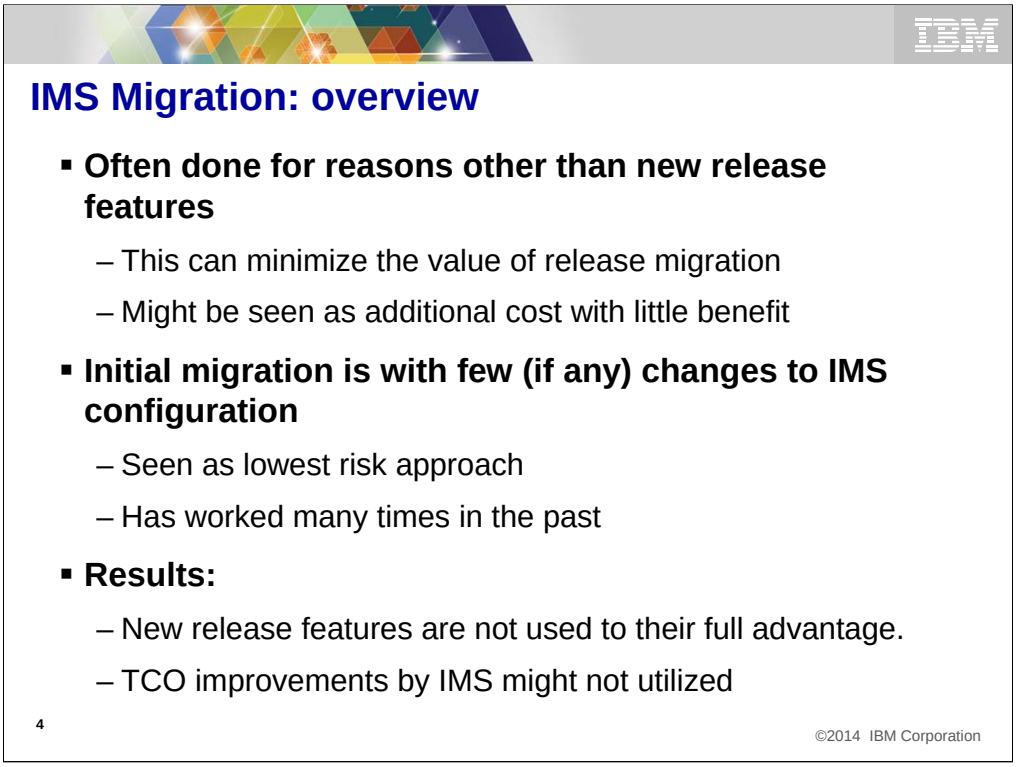
2

©2014 IBM Corporation

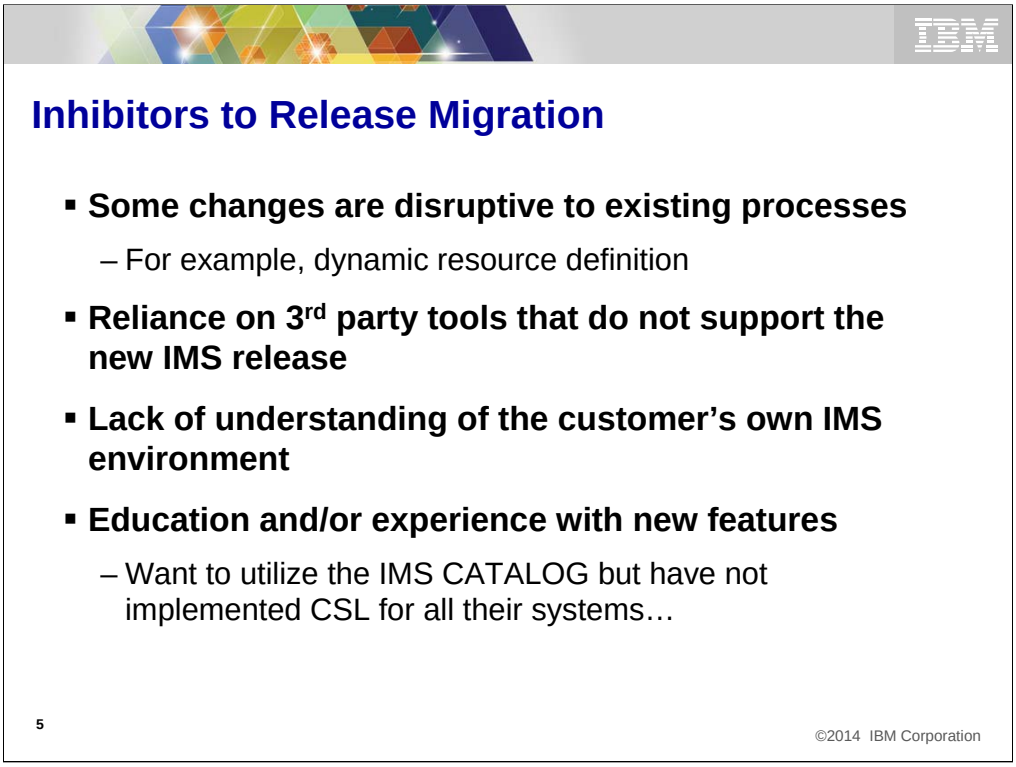
What would we do without these?



These are the things we are going to talk about today but I will focus mainly on how to build and test an evaluation system using IMS and IBM tools so that the best migration plan for your company can be achieved.



It is very common that the primary reason for migration is for IMS currency rather than the new function provided in the target release. Since there is no perceived advantage to the migration other than continued support, there is often little if any urgency to complete the migration. Couple that with the tendency to migrate without changing the current IMS system there is often a long time between the start of the migration process and the yielding of any benefits from the new release. If during the migration planning phase, the target IMS environment could be tested and evaluated easily, the potential for improved performance and TCO could be determined. An early look at the release quality and the potential value of new features could be assessed. The results of this early evaluation process might justify a more aggressive migration plan or it might indicate that the benefits of the new system do not justify any increased emphasis on the duration of the migration process. Most of you have done several migrations and have old processes to fall back on that you have confidence in and that work. Just as with most things, time and innovation usually creates better ways to accomplish things. I think that todays tools can help the migration process be done much quicker and easier. Once you have seen the success these tools can provide, you may find that these same tools are useful in maintaining your environments as you go forwards. Using them for test system creation and maintenance can simplify and reduce the time it takes to manage your test environments. With the reductions in staff sizes and experience, that is something we all should be looking for.



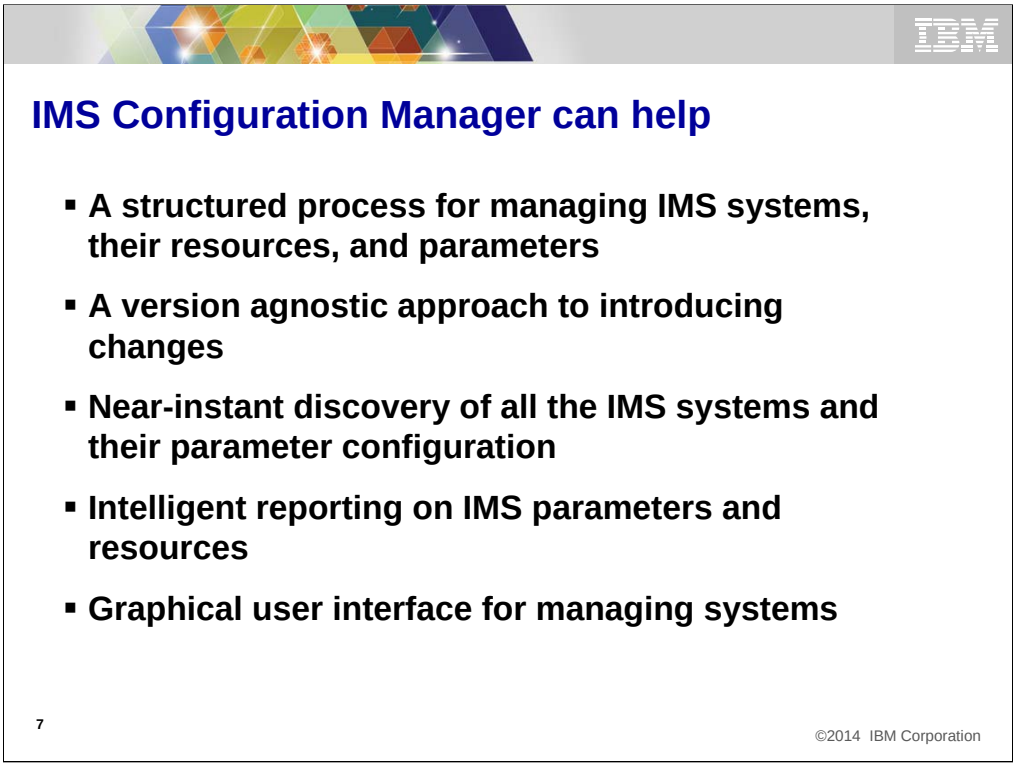
I do not think that I have to explain much of this chart to most of you. Most of you have gone through this and many of you more than once. The ability to easily and quickly create an evaluation system can provide significant benefits. If you have 3rd party tools, maybe a system without the tools can still be useful for the initial evaluation. It might allow evaluation of newer tools that ***do*** support the new release and or at least enable early performance and stability testing of the new release level. If it has been some time since the last migration, their may have been significant changes to your environments that are not well known. Using tools that automatically understand the source environments can save time during migration due to items not being missed during the migration process.

Release Migration Planning

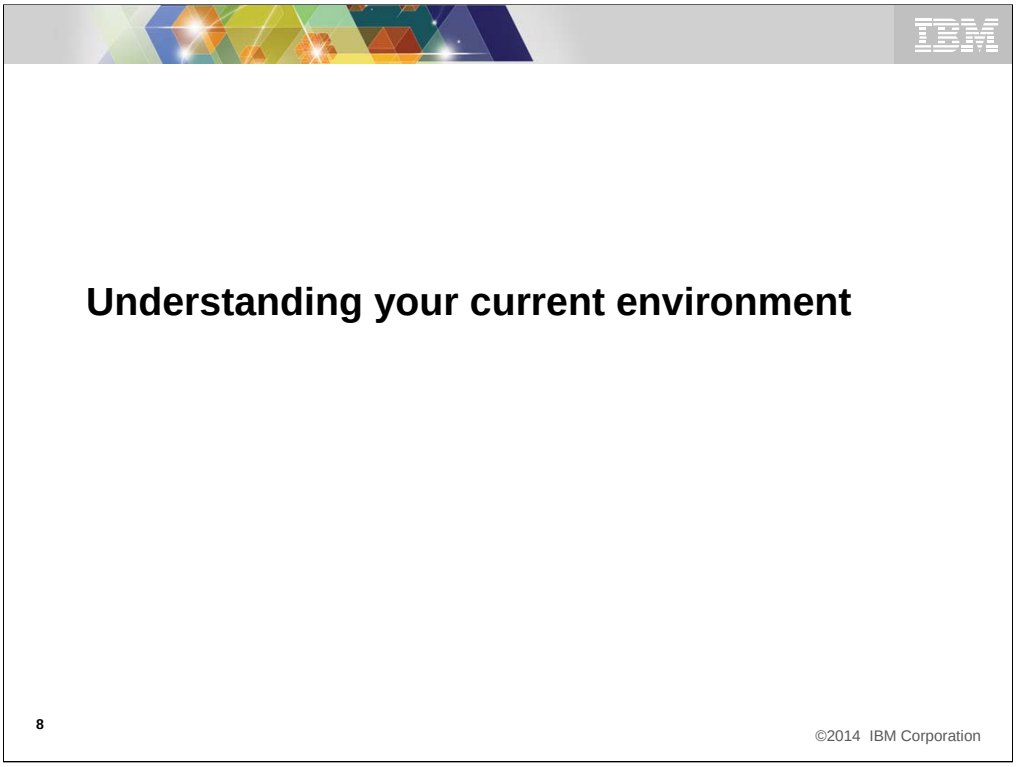
- **Early evaluation of new release features**
  - – May identify TCO opportunities in new release
- **Use of tools to quickly create evaluation system**
  - – IMS Cloning tool can quickly create a ‘cloned’ system
    - Includes data sets and databases
  - – IMS Configuration Manager eases ‘cloned system configuration’
    - copy parameter members to ‘cloned’ IMS system and add new release keywords and/or members
    - Copy resources and create updated modblks for ‘cloned’ system.

©2014 IBM Corporation

This slide shows how you can actually use the target IMS level to provide input into the migration planning process. Using the tools I will talk about, you can be running the new evaluation system in a matter of hours or days. Lets begin to see how this process can work.



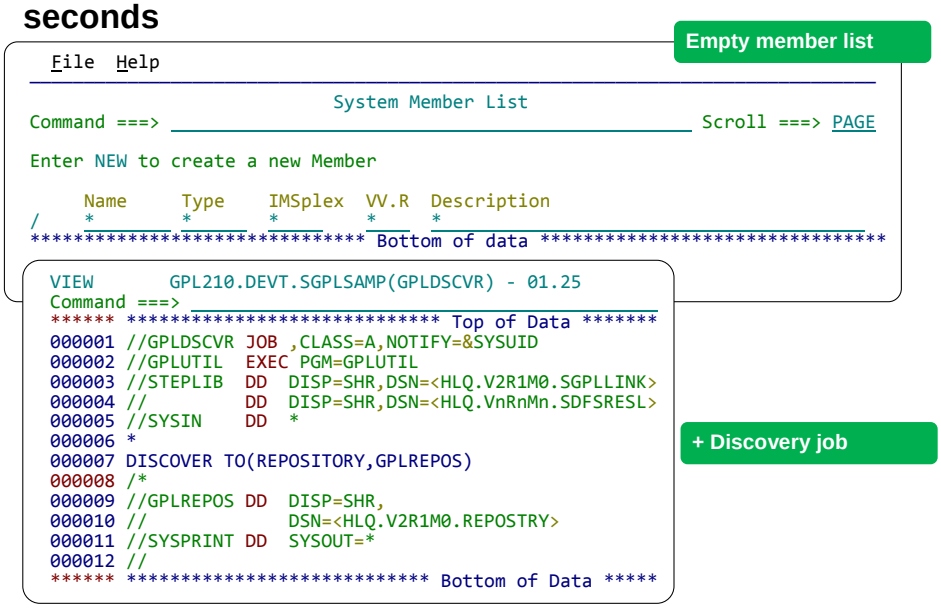
I will discuss the IMS Cloning tool later in the presentation but here are some ideas in terms of how the IMS Configuration manager can help you in both the planning and creation of the evaluation system(s). An important part of the support provided by ICM is its agnostic approach to resource management. The tool is aware of the features and capability of each IMS release and so you can manage resources and proclib parameter members for several IMS systems at different IMS levels. That means that implementing resource definition at the evaluation systems level is easy, even if you need to add and/or change some of the resource definitions to take advantage of new IMS features. I know that we all think we know how many IMS systems and associated address spaces but often we are just close. Using ICM to discover all the IMS and related jobs such as IMS Connect can make the planning process much more accurate. I know that many of you are thinking, all I have to do is change my STEPLIBS and I am migrated. Maybe so, but to take advantage of new function that can yield benefits, you will actually need to migrate the systems sometime. In the old method, this is done after you have the release running on the current IMS systems. Especially for test systems, maybe it makes sense to do both the migration and release change at the same time.



The first place to start might be to create a map of your current environment or at least the environment you need to migrate. That should give you an idea of the scope of the effort based upon how many IMS environments have to be converted.

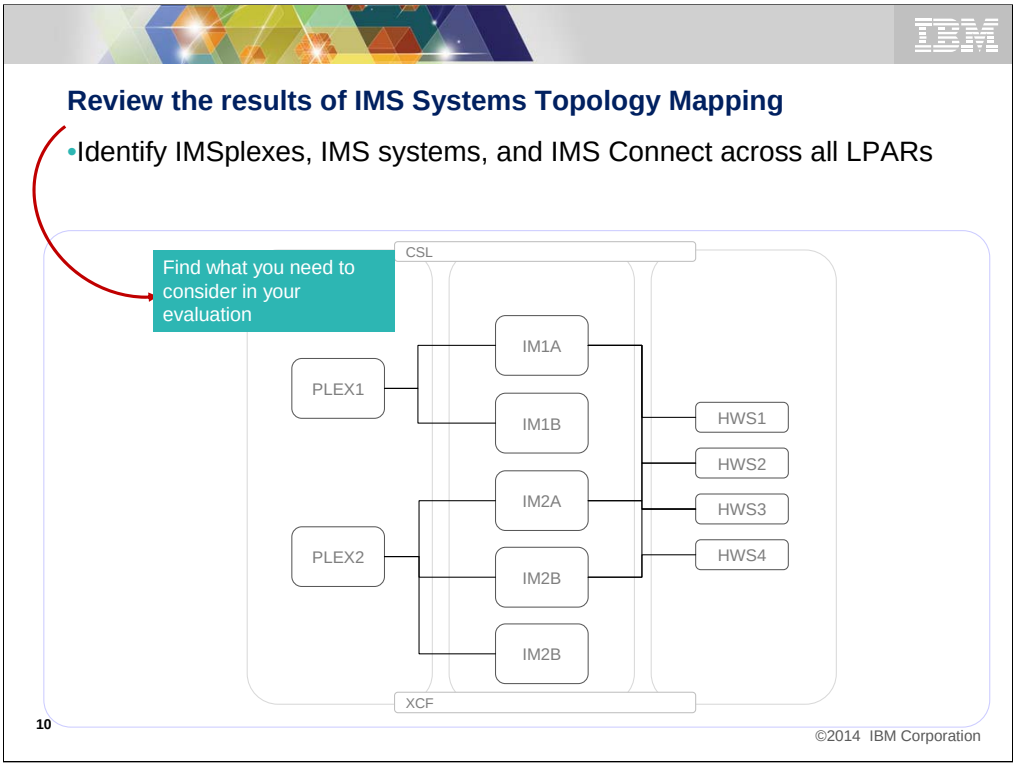
Auto discovery of IMS Systems

- **IMS Configuration maps an entire IMS topology in seconds**



9

A well planned migration requires identifying the IMS systems that are to be upgraded: their version, and their IMS configuration. Once these systems are identified, it is then possible to plan and validate the migration itself. IMS Configuration Manager allows you to start with an empty repository and use a DISCOVER job or an AUTODISCOVER server parameter to populate all the IMS systems. Here we can see that we start with a blank repository and then run a DISCOVER job. The job requires no additional parameters to perform an extensive discovery of your environment, but you can optionally set parameters to limit discover to certain plexes, system types, or to perform discovery specifically for IMS systems that are **not** part of a plex.

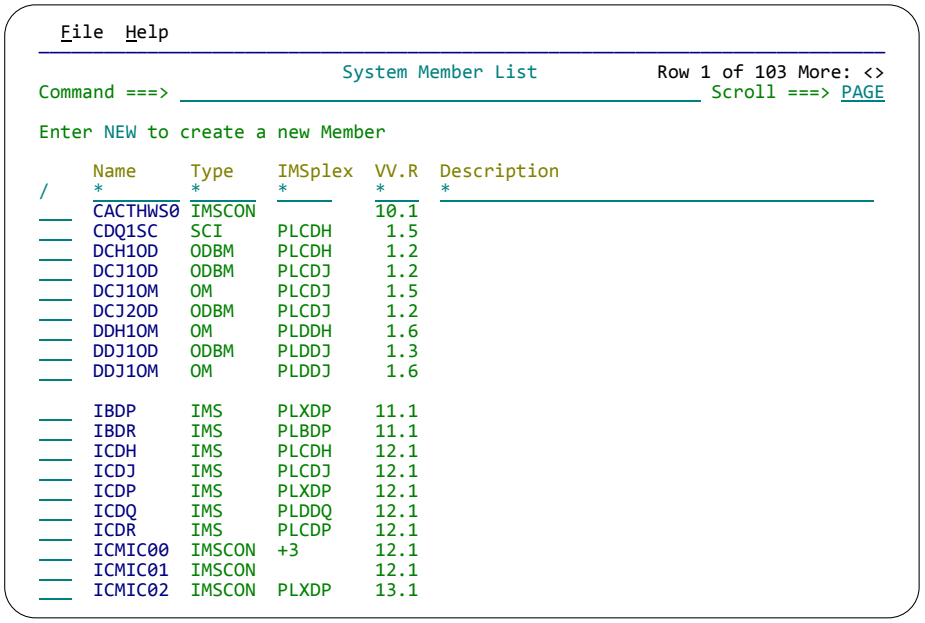


The discovery process then finds all the IMSplexes registered to XCF and the associated IMS systems.
•The IMSplexes must be registered to XCF
•Discovery of the IMS systems relies on an OM and SCI for each plex being available on the LPAR on which the DISCOVER job (or server is run)

The discovery process will also identify all associated IMS Connect systems. These IMS Connect systems *may* be member of the plex. But they do not have to be: all that is required is a DATASTORE association between any discovered IMS and the IMS Connect.

Today the auto discovery feature facilitates IMS parameter management, but the ability to include mod blocks data during discovery is being considered.

Complete IMS topology



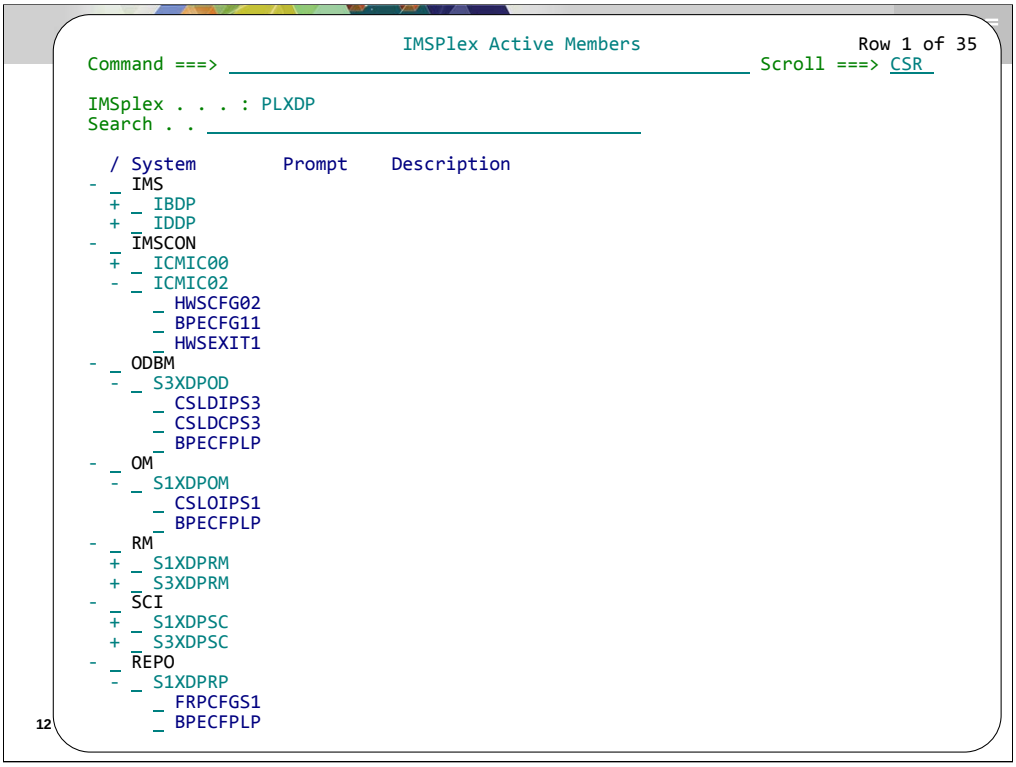
11

The job itself typically takes no more than a couple of minutes to complete. At the end of the process IMS-related address spaces are mapped. Here, we can see the repository with the output of a DISCOVER job. It has found a number of IMSplexes across multiple MVS images and has mapped their IMS and IMS Connect address spaces, CSL address spaces, and the related PROCLIB settings for each of these systems.

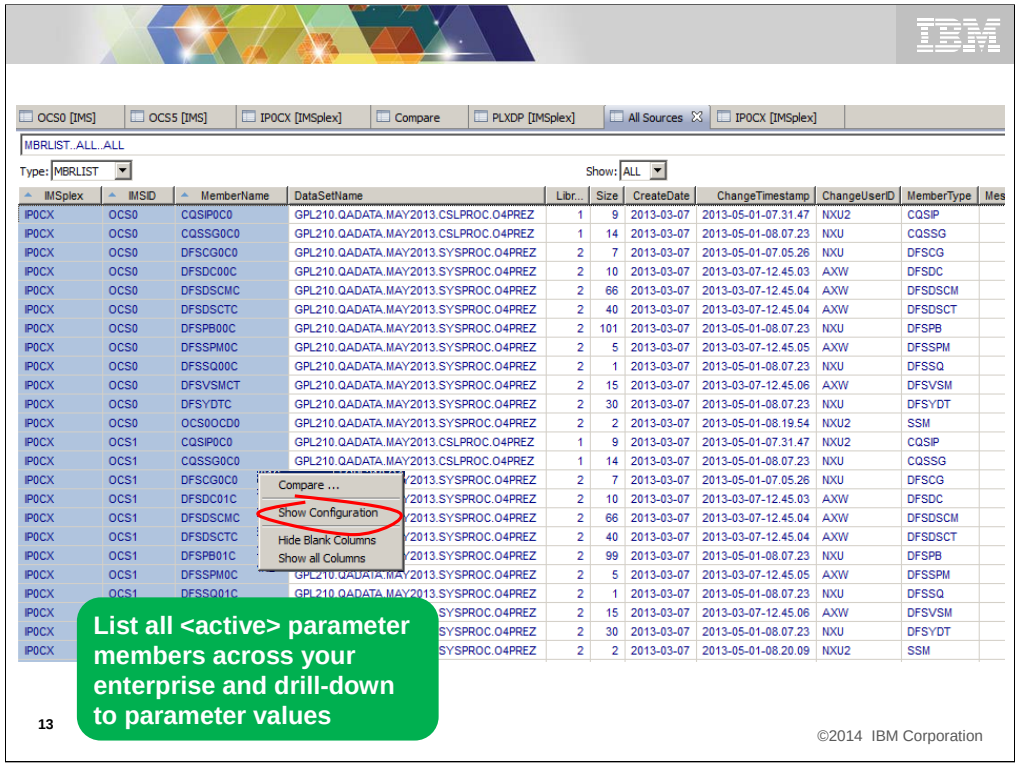
Notice that in the case of IMS Connect, the discover job discovers systems that are not part of any plex (even if they relate to systems within a particular plex) and that it can also identify IMS Connect systems that serve multiple IMSplexes.

For IMS systems without a PLEX, you can run the DISCOVER job with the NOPLEX option. However, this type of discovery is restricted to the MVS image on which the job is executed.

Nevertheless, the result is likely to be a complete representation of even complex IMS environments and gives unique and instantaneous access to the active PROCLIB members for each of these systems.

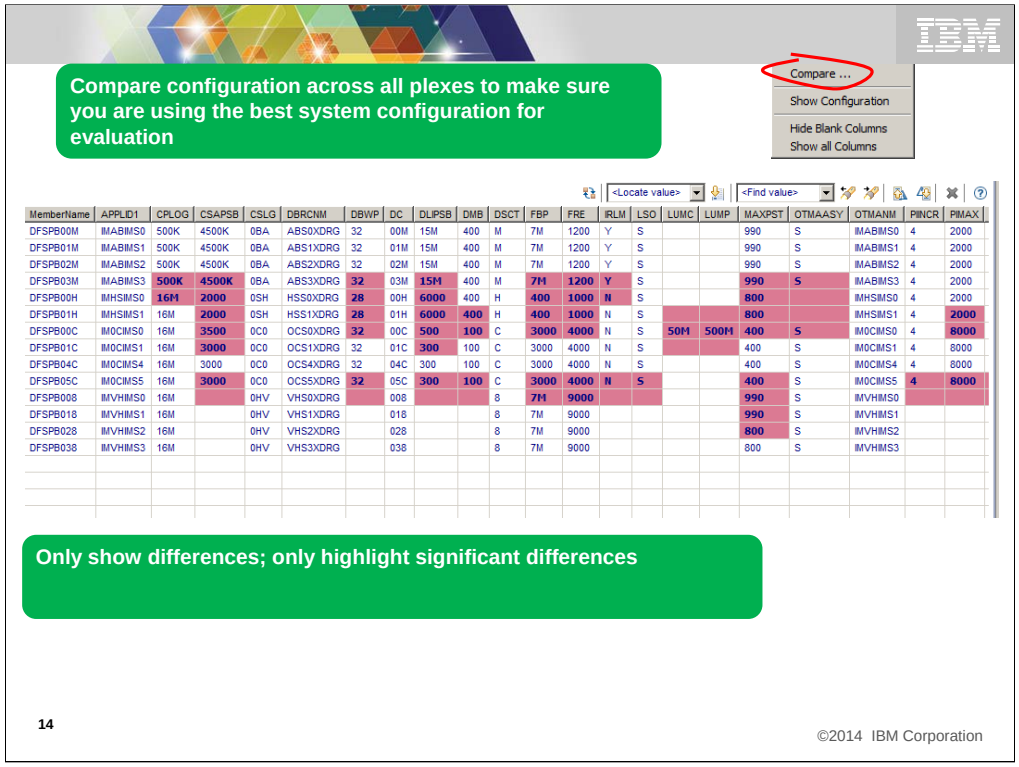


You can now use a “p” line action against any of the systems, or any of the plexes and view the PROCLIB members for any of those member types. But where is gets really interesting is when we start using the GUI to interrogate these objects. Remember that, through the Connection Server, both the GUI and ISPF can browse the same repositories. The main advantages of the ISPF – is the additional editing capabilities that it provides, including smart search, which lets you implement the features you want; the main advantages of the GUI is that it makes it easier to view and analyze configurations across plexes and that it provides better analytic capabilities including smart-compare (which we will look at in a minute), export configuration to a spreadsheet, as well as filtering, sorting, etc…. Let’s have a look at it now.



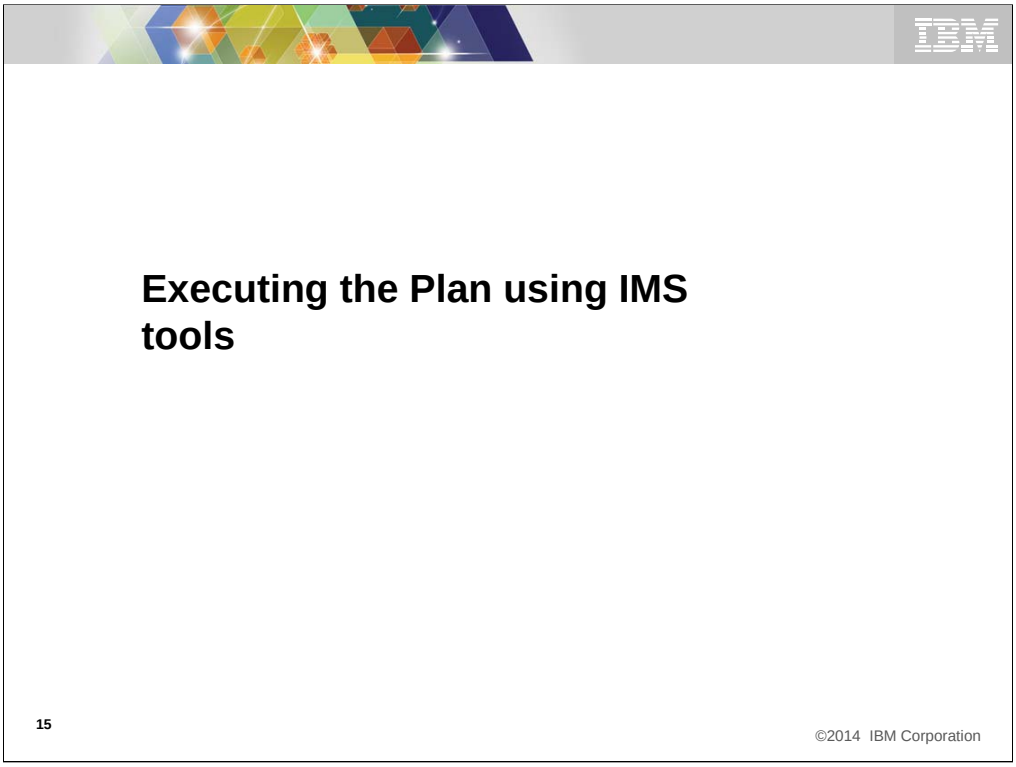
Just like we just saw in the ISPF, the GUI lets you view the *active* PROCLIB members for the system. The difference is that the GUI makes it easy to look at configurations across a number of IMSplexes. A consolidated result set is available for all the repositories in the scope of all defined servers within a given client install instance.

The GUI is also able to provide context sensitive actions. For example, you can right-click a member to tabulate that particular PROCLIB's configuration. The tabulated form makes it easier to filter, export, and report on members, but perhaps the most powerful feature is compare…

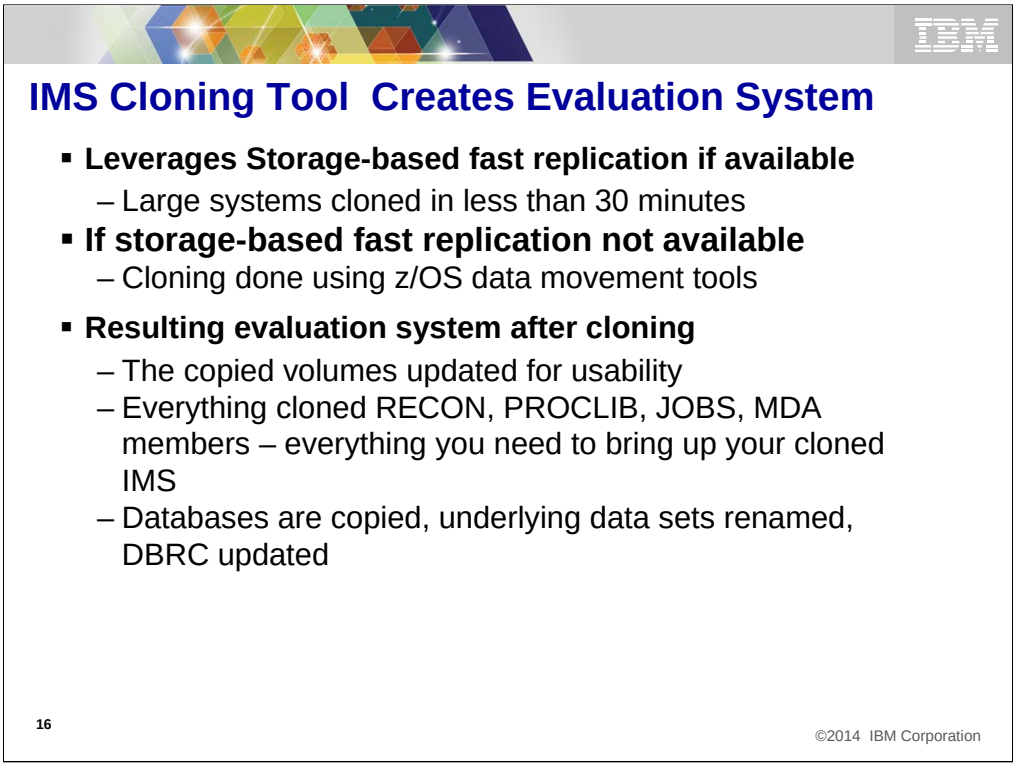


MemberName	APPLID1	CPLOG	CSAPSB	CSLG	DBRCNM	DBWP	DC	DLIPSB	DMB	DSCT	FBP	FRE	IRLM	LSO	LUMC	LUMP	MAXPST	OTMAASY	OTMANM	PIINCR	PIMAX
DFSPB00M	IMABIMS0	500K	4500K	0BA	ABS0XDRG	32	00M	15M	400	M	7M	1200	Y	S			990	S	IMABIMS0	4	2000
DFSPB01M	IMABIMS1	500K	4500K	0BA	ABS1XDRG	32	01M	15M	400	M	7M	1200	Y	S			990	S	IMABIMS1	4	2000
DFSPB02M	IMABIMS2	500K	4500K	0BA	ABS2XDRG	32	02M	15M	400	M	7M	1200	Y	S			990	S	IMABIMS2	4	2000
DFSPB03M	IMABIMS3	500K	4500K	0BA	ABS3XDRG	32	03M	15M	400	M	7M	1200	Y	S			990	S	IMABIMS3	4	2000
DFSPB00H	IMHSIMS0	16M	2000	0SH	HSS0XDRG	28	00H	6000	400	H	400	1000	N	S			800		IMHSIMS0	4	2000
DFSPB01H	IMHSIMS1	16M	2000	0SH	HSS1XDRG	28	01H	6000	400	H	400	1000	N	S			800		IMHSIMS1	4	2000
DFSPB00C	IM0CIMS0	16M	3500	0C0	OCS0XDRG	32	00C	500	100	C	3000	4000	N	S	50M	500M	400	S	IM0CIMS0	4	8000
DFSPB01C	IM0CIMS1	16M	3000	0C0	OCS1XDRG	32	01C	300	100	C	3000	4000	N	S			400	S	IM0CIMS1	4	8000
DFSPB04C	IM0CIMS4	16M	3000	0C0	OCS4XDRG	32	04C	300	100	C	3000	4000	N	S			400	S	IM0CIMS4	4	8000
DFSPB05C	IM0CIMS5	16M	3000	0C0	OCS5XDRG	32	05C	300	100	C	3000	4000	N	S			400	S	IM0CIMS5	4	8000
DFSPB008	IMVHIMS0	16M		0HV	VHS0XDRG		008			8	7M	9000					990	S	IMVHIMS0		
DFSPB018	IMVHIMS1	16M		0HV	VHS1XDRG		018			8	7M	9000					990	S	IMVHIMS1		
DFSPB028	IMVHIMS2	16M		0HV	VHS2XDRG		028			8	7M	9000					800	S	IMVHIMS2		
DFSPB038	IMVHIMS3	16M		0HV	VHS3XDRG		038			8	7M	9000					800	S	IMVHIMS3		

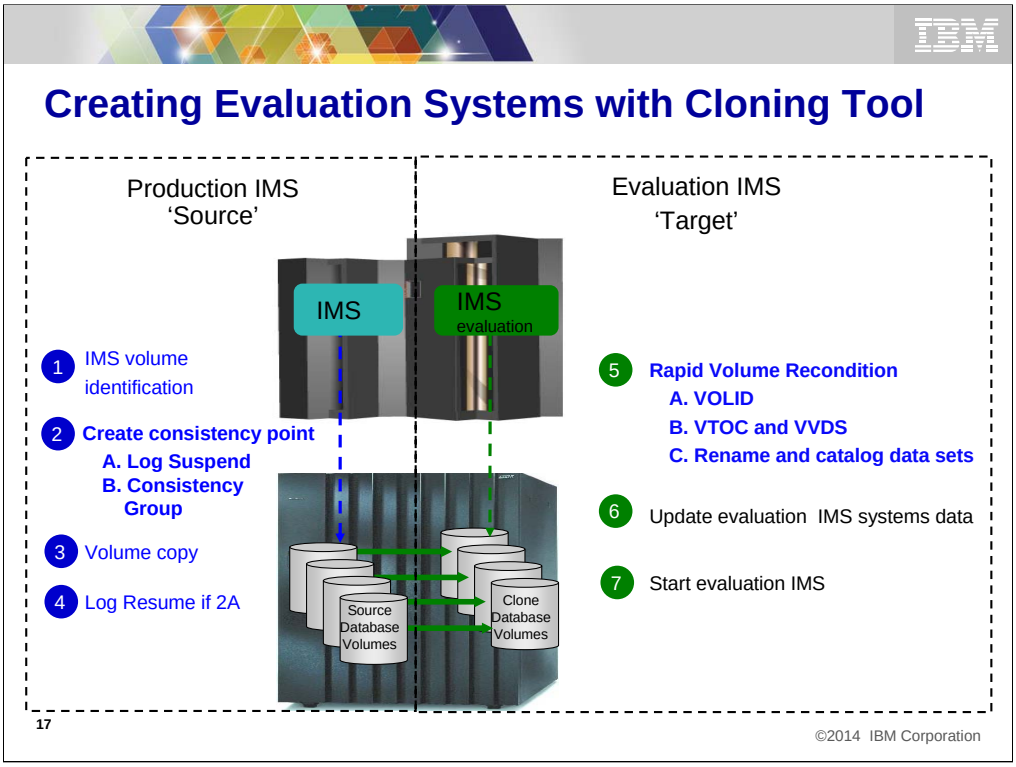
Compare provides an intelligent function that compares the values for the member and allows you to highlight significant differences. Crucially, the compare function actually understands the members it is comparing and is able to differentiate between spurious differences and significant differences. The above display compares rows within the same list and highlights cells with functional differences between two consecutive rows.



Once we have created a complete migration plan or just an plan to create and evaluate the new target IMS release, we can use these tools to quickly execute the plan.



You start the process by cloning the IMS system that you wish to emulate as your evaluation system. If you have and can use storage-based fast replication the process is faster and maybe less disruptive than what I call the ***standard*** cloning process. Regardless of which method you use, the results are the same it just takes longer with the ***standard*** process. The final results is that you have an evaluation system that is ready for the final tailoring process.



This example shows cloning an existing production system using fast-replication capable hardware. It is not necessary to have or use fast-replication hardware to accomplish the cloning process but it makes it faster and less intrusive. If not using the fast-replication software, the source system must be inactive during the cloning process.

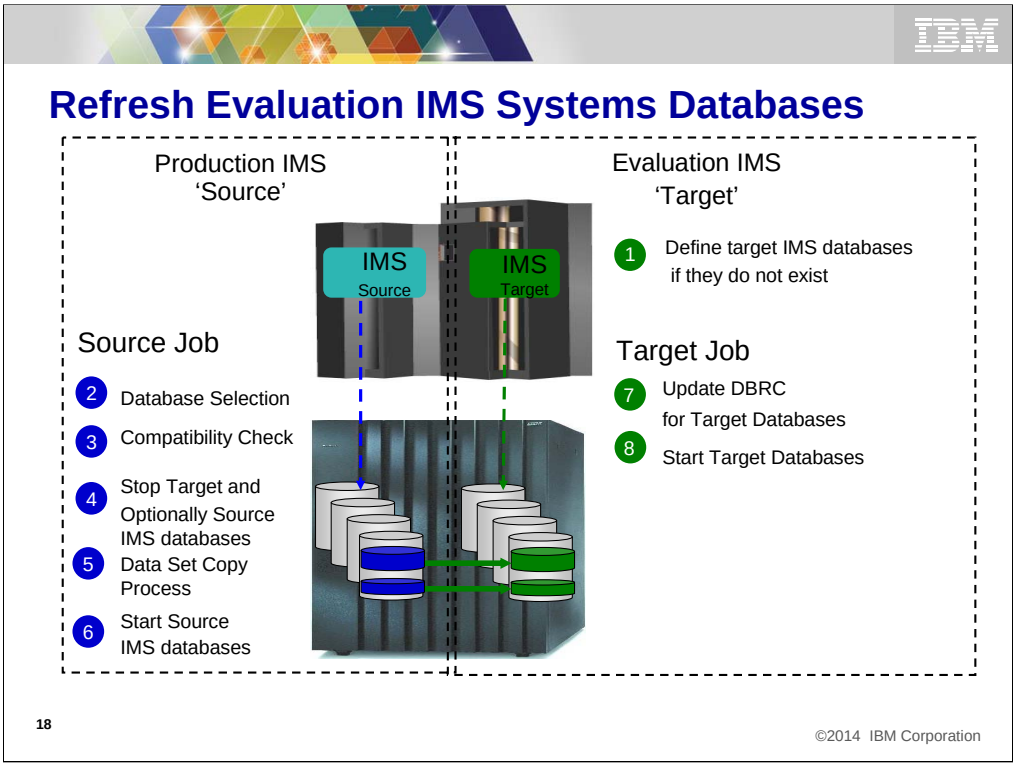
The left side of the slide represents the source IMS, in this case, a Production IMS system. The right side of the slide represents the target or cloned IMS system.
On the source,
Step 1 --- The disk volumes that make up the production IMS system are identified by using specific VOLSER IDs, VOLSER mask, or SMS storage group.
Step 2-4 ---To gain a static copy of your data and the source ICF catalog entries, there are several options:

a. IMS is up and the IMS log is suspended (this is proprietary code in IMS Cloning Tool and it works similar to DB2's log suspend.)

b. IMS is up and can use either IBM FlashCopy, FlashCopy Manager, or EMC consistency group support

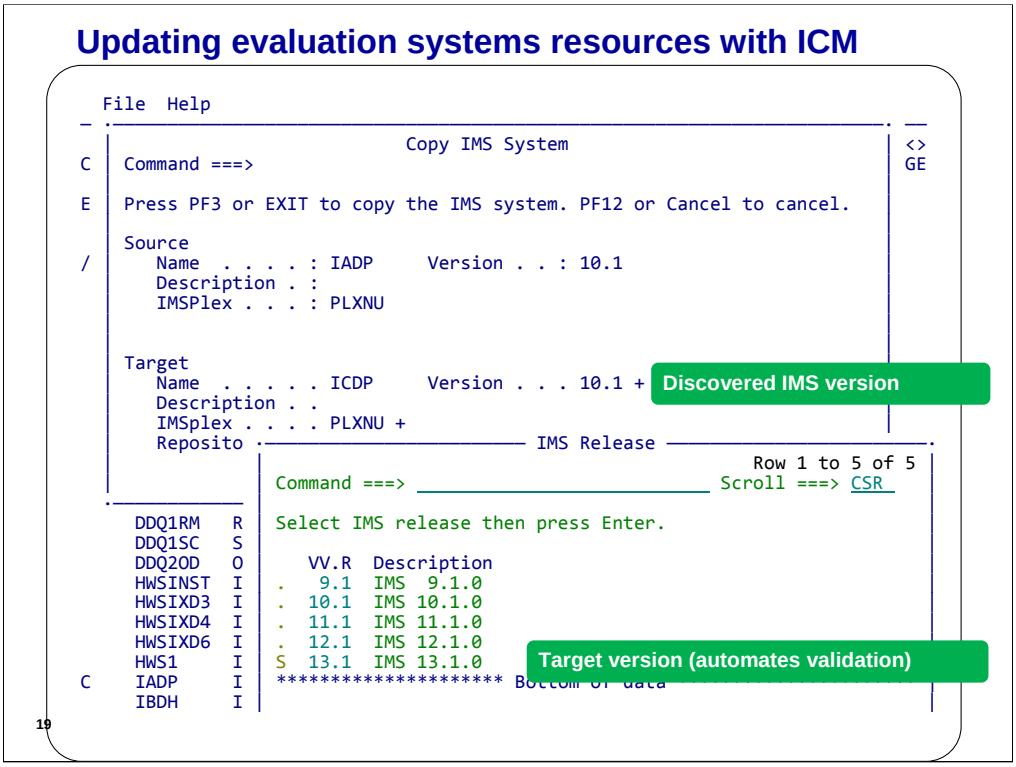
----With either of these techniques, IMS Cloning Tool immediately invokes a DFSMSdss Copy to issue IBM FlashCopy or SnapShot commands to copy the data instantly and backs up the ICF catalogs that point to the data sets being copied.
----When cloning IMS systems that reside on EMC or Hitachi Storage Systems, an appropriate storage-based fast-replication process is performed before the IMS Cloning Tool cloning automation is invoked and a list of copied storage volumes is passed to IMS Cloning Tool for use in later processing steps.
----Once the copy is complete which is in seconds, IMS Cloning Tool can resume the source log. At this point, we are done using the production IMS system.

The following steps will be performed on the target or cloned IMS system shown on the right side of



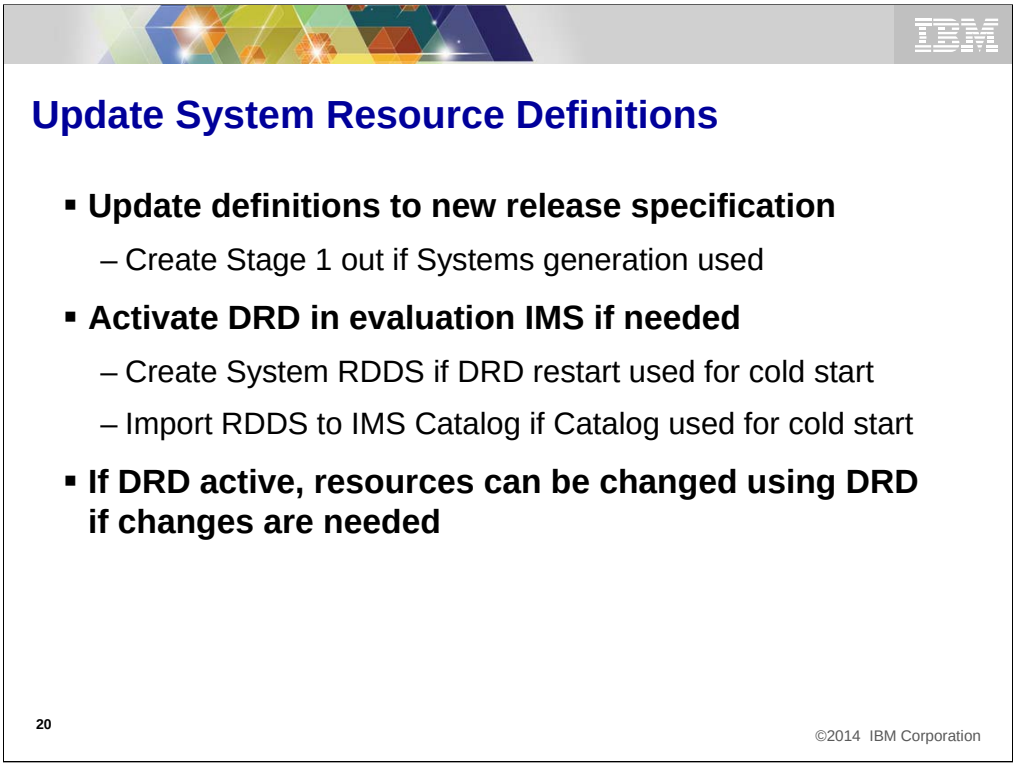
This is just an example of the process that the Cloning tool uses to refresh the IMS databases that are needed for the new evaluation system that is created. If storage-based fast replication is available it is used, if not, traditional techniques are used to build the databases. The difference in the results is time not results. Either technique results in all the required databases being make available to the evaluation system.

The left side of the slide represents the source IMS, in this case, a Production IMS system. The right side of the slide represents the target IMS system – where you want to refresh the IMS data to.
Step 1 – is on the target side. The definitions need to exist in advance
Step 2 - The source databases to be refreshed are selected by database name. IMS Cloning Tool finds the IMS subsystem name and determines if it is active, then it finds the source and target databases and indexes (if the targets already exist), determines the data set names for each database and index and then verifies their existence
Step 3 - Checks are performed to ensure the characteristics of the source and target IMS databases are compatible. IMS Cloning Tool gets the attributes of the IMS databases and indexes from the source and target IMS RECON, ACBLIB, and MDALIB data sets. Some characteristics that are checked include: Type, Access Method, Number of segments, data set groups, Blocksize, Randomizer Parameters, etc.
Step 4 - The source and target databases are stopped (DBR commands automated). Optionally, the source databases can be copied while they are running to create a fuzzy copy. However, the fuzzy copy option does not guarantee transactional integrity on the refreshed copy.
Step 5 – Data sets are copied. For sites using FlashCopy or SnapShot, IMS Cloning Tool will invoke these copy facilities. For sites using EMC TimeFinder or Hitachi ShadowImage, IMS Cloning Tool will produce the output files describing source and target data set information so users can create their own data set fast-replication job streams.
Step 6 - The source IMS databases are started unless a fuzzy copy was specified in step four.
The following steps will be performed on the target IMS system.



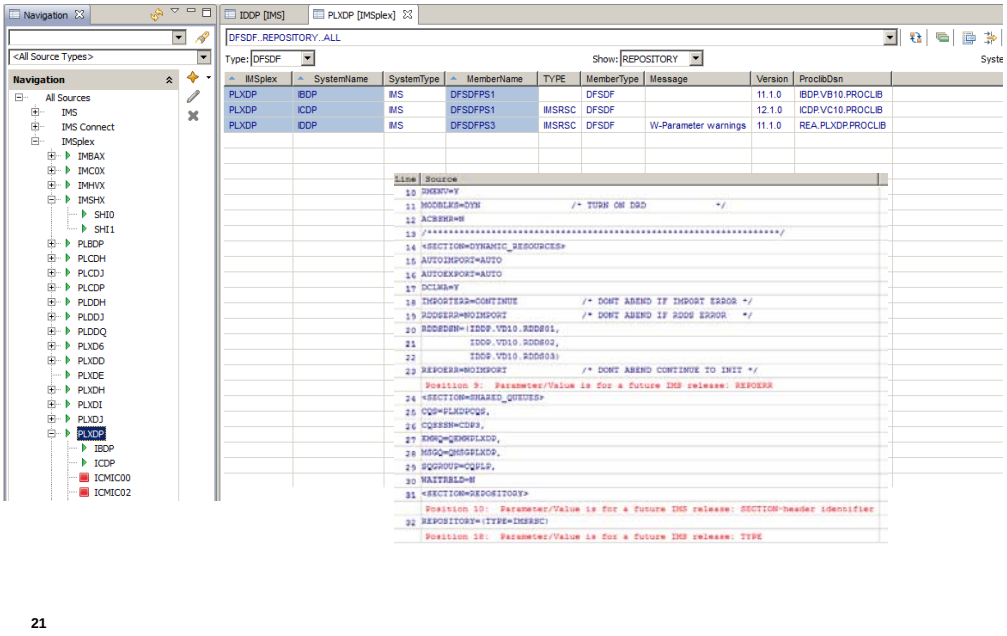
Having discovered your IMS environment, you can begin to plan and validate the migration to a newer version of IMS. A good starting point is to copy the discovered system into a new repository setting the IMS version to the target IMS version. The newly copied IMS system will contain the same PROCLIB configuration but these PROCLIBs will now be validated using IMS V13 rules, instead of the rules for the original system. This means you can quickly identify obsolete parameters as well as validate and introduce new values.

The advantage of this approach is that, as we have seen before, the GUI can consolidate information from multiple repositories (as well as multiple servers) so that as you build your proposed map you can view and interrogate the differences between the proposed configuration and the current, automatically discovered, configuration.

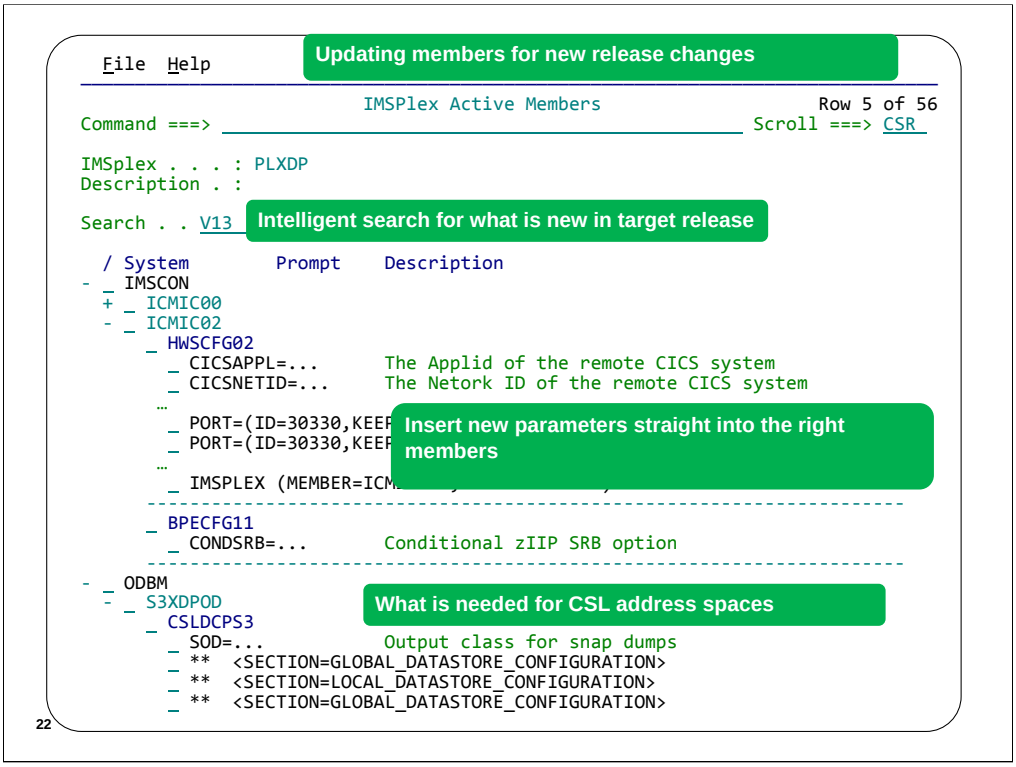


Almost always you have to update your system resource definitions (i.e. MODBLOCKS) to the level of the new evaluation system. Using ICM you have several options on how you accomplish this. If you have the new system enabled for DRD, the quickest way to do this might be to just created system level RDDS datasets and cold start the evaluation system from these. That might remove the need to do a systems generation during initial evaluation system testing. If you need to activate DRD because your source system does not have it active, ICM will walk you through setting up the DRD environment for the parameter members that need to be changed to support. You have lots of options on how you make the new evaluation system capable of supporting the DRD environment.

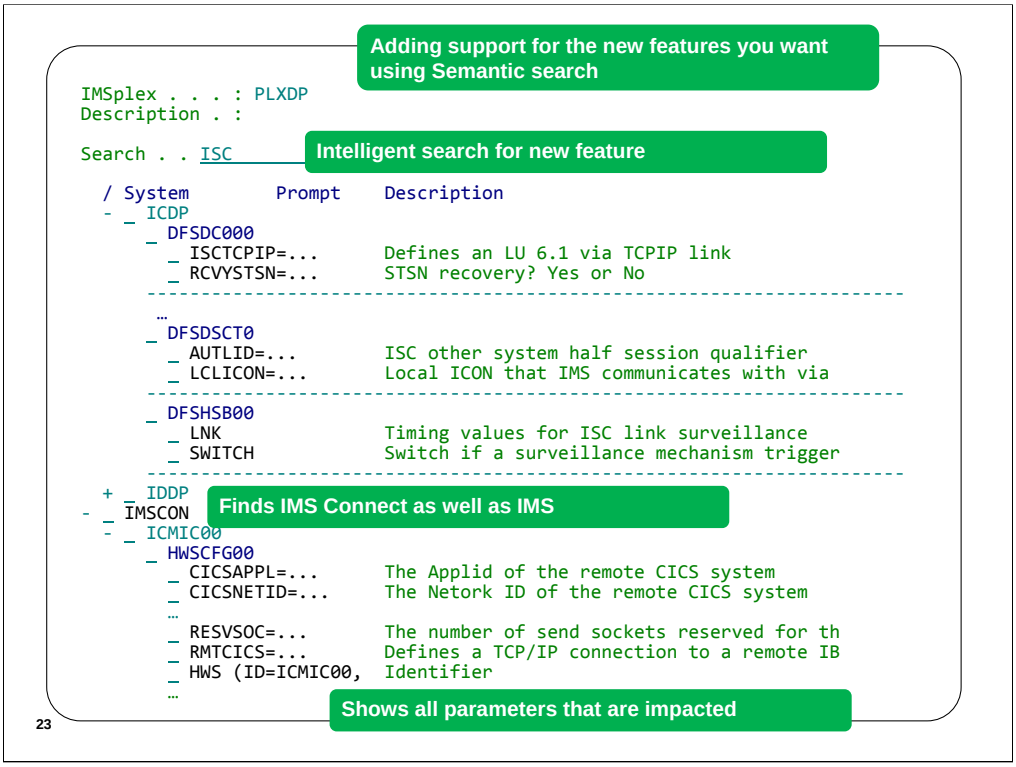
Find Parameter changes needed for copied parameter members in evaluation system



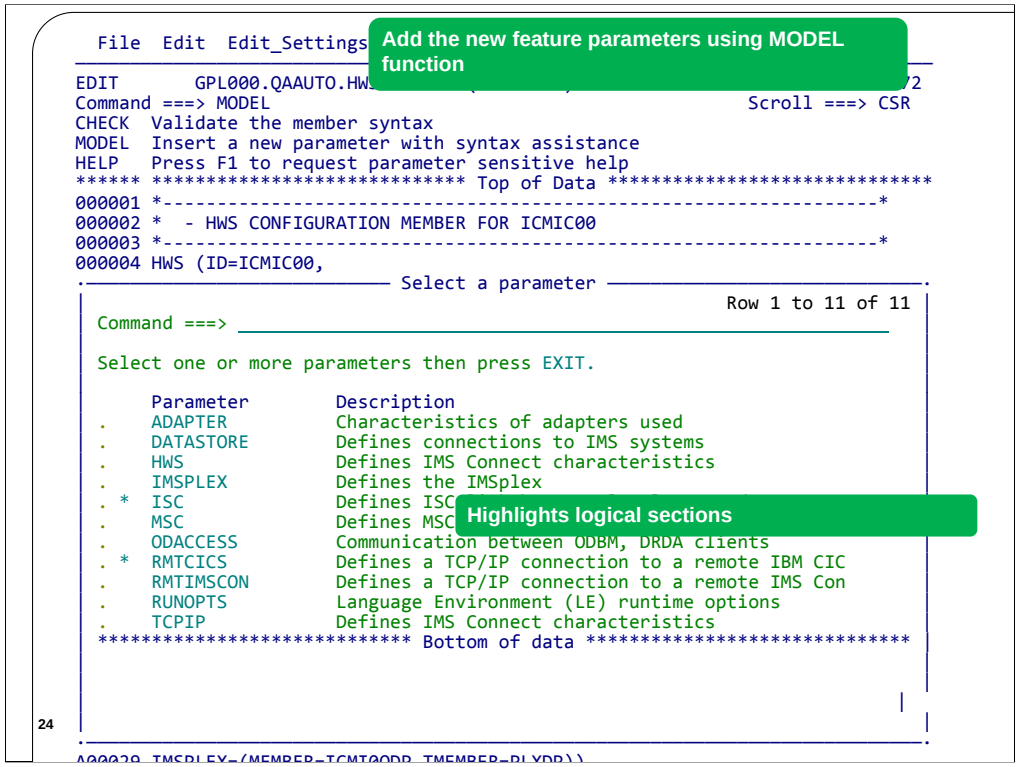
Here we show an example of the GUI validation. If introduce a parameter member from the source system at an earlier version, the server will provide a Parameter Warning to the GUI for that particular member, and you can then retrieve the member to see exactly where and why the warnings have been generated. This allows you to update the parameter member as needed in the evaluation system.



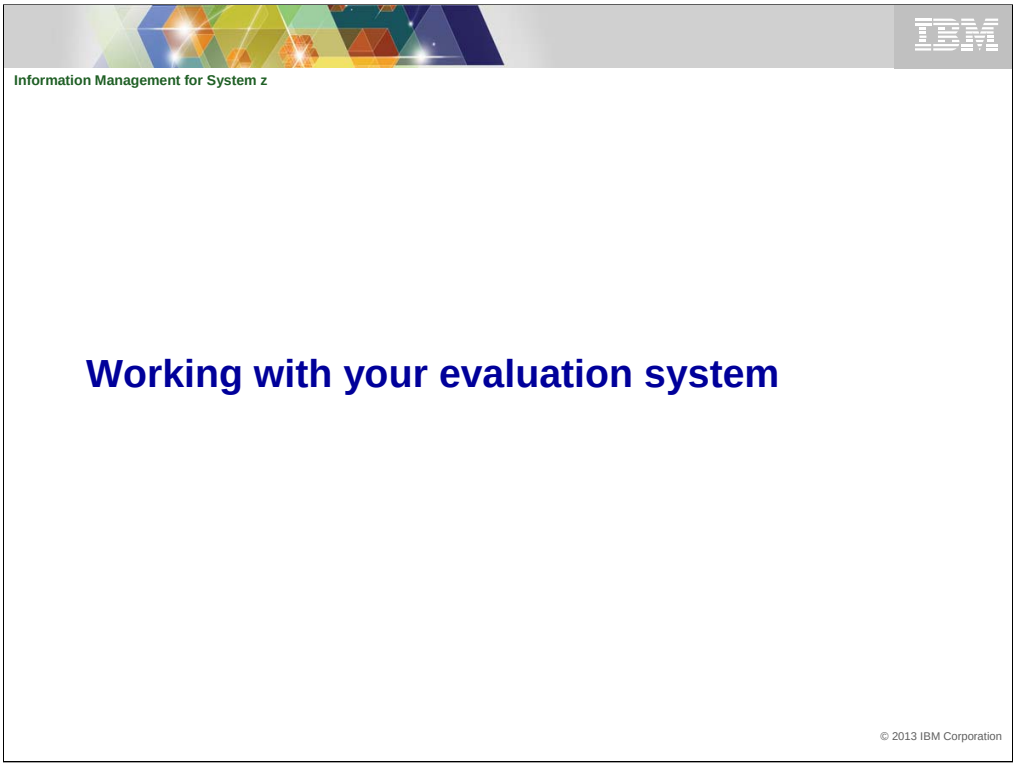
From the ISPF side, the capability of IMS Configuration Manager to support CSL member types and IMS Connect provides significantly more power to the existing intelligent search capabilities. For example, when you search for new V13 parameters for a particular PLEX, it will show you each PROCLIB with new parameters, that are applicable for each member identified in that PLEX. You can then select the PROCLIB for the appropriate member straight from the display and insert the new parameters.



The value of this is particularly apparent when you try to introduce new features that require changes to multiple members. You can see each member that is impacted by the given change in a single search and then proceed to implement your changes directly on each of those members.



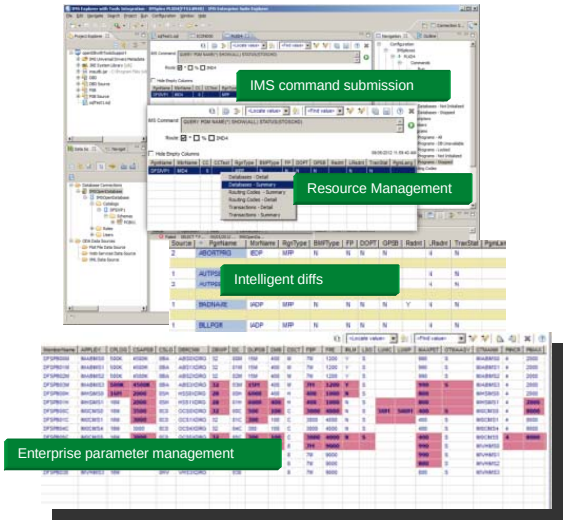
When you select a member you can edit it, just like in ISPF edit but with a few significant enhancements that aid version-to-version migration. The first is the MODEL function, that allows you to insert a basic template for the new parameters, and the second is the CHECK function which will validate the member based on the rules that are appropriate for the given version of the member. Hence you can validate the member both as a “Version Previous” member and as a “Version Next” member, simply by altering the IMS version.



Once you get the evaluation system built, you want to do some initial testing of it as quickly as possible. You probably have ways of driving workload in your shop but if not, there are IVP's and other ways to do some initial testing. Once the evaluation system has proven to be stable, they you might want to do some actual performance testing. Usually your application teams can do this as a part of their release testing.

Centralized management of IMS systems

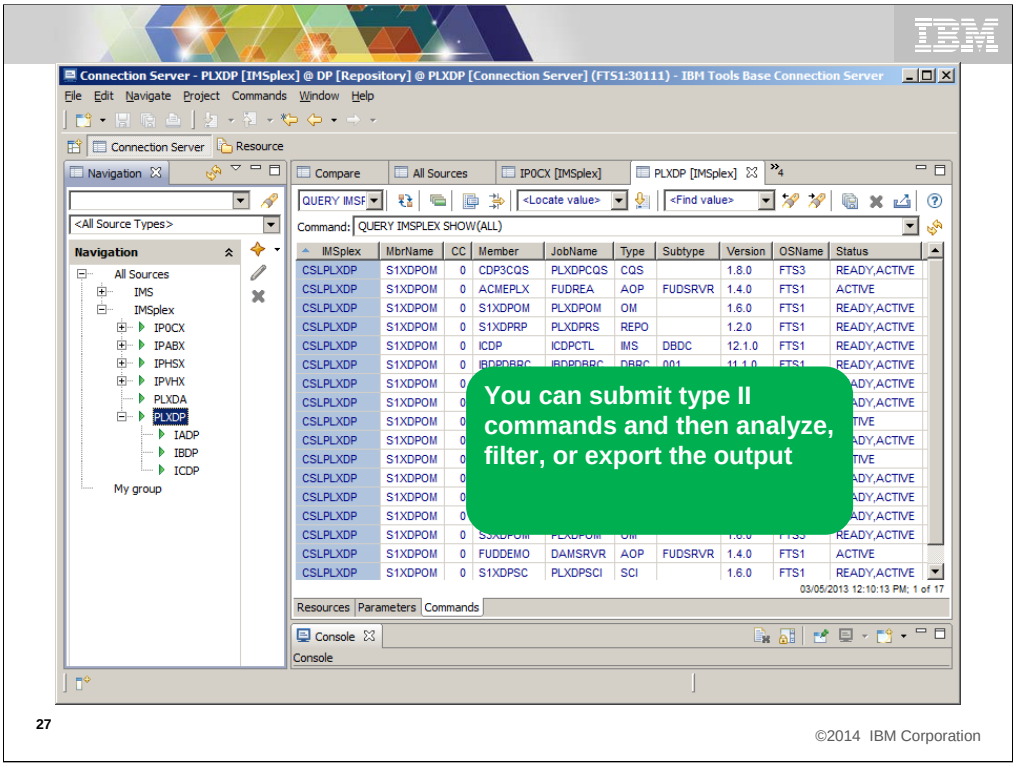
- **Map IMS topology**
- **Analyze PROCLIB parameters across global sites**
- **Run CSL commands**
- **Manage MODBLK resources**
- **Search, filter, compare, and export results to spreadsheet applications**
- **Provides tight integration with IMS Connect Extensions GUI**
- **Works with z/OS Explorer, IMS Explorer, CICS Explorer, and Rational offerings**



26

©2014 IBM Corporation

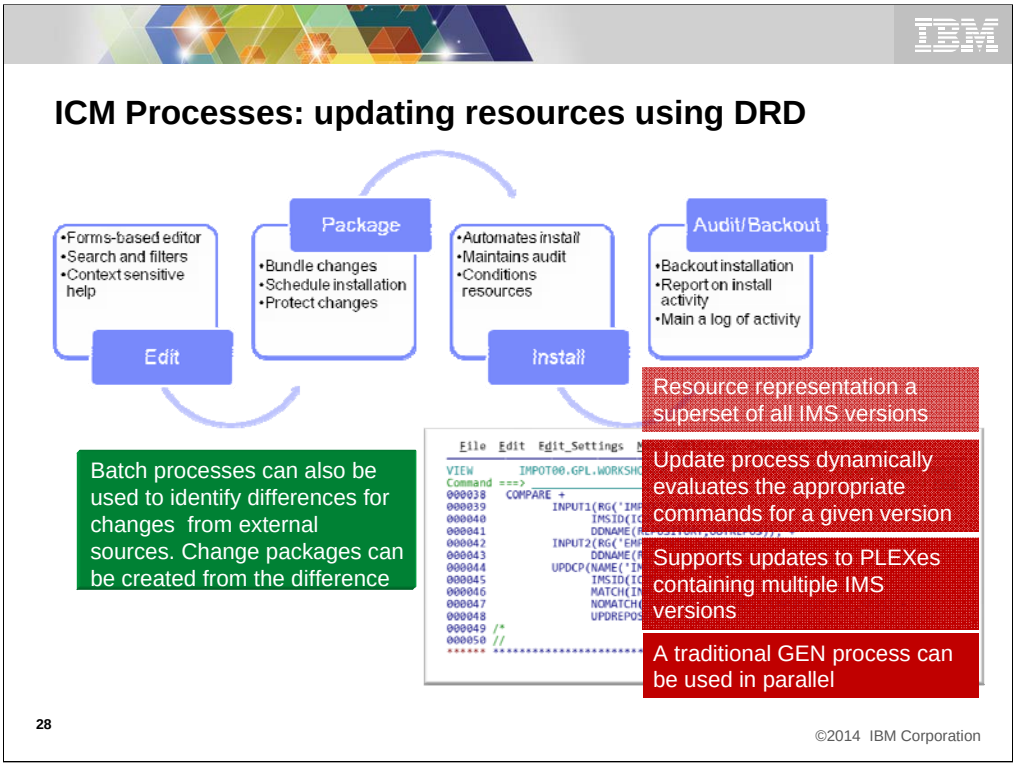
One of the things many customers are asking for is the ability to manage all or a group their systems from a single view. Using the ICM GUI enables a wide range of new function such as the items listed on the slide. You should take special notice of the fact that the GUI can shell share with several other Eclipse GUI offerings so it can also be a part of everything you need to do with IMS and or IMS Connect if you also have IMS Connect extensions.



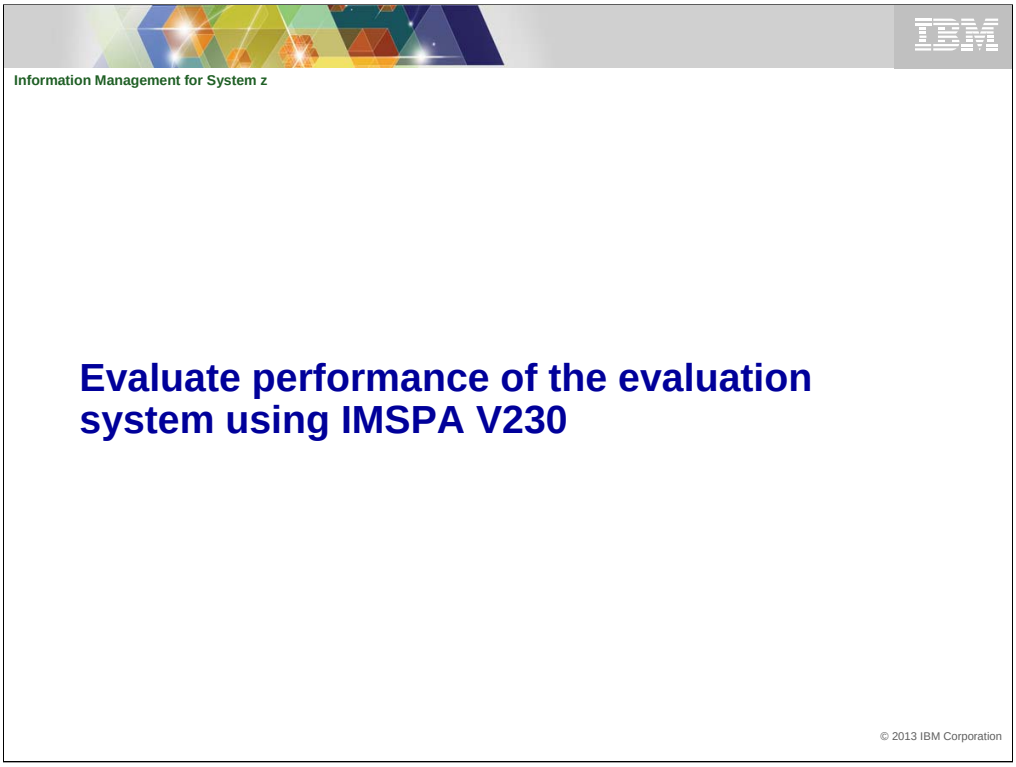
27

©2014 IBM Corporation

The next few slide show different parts of the GUI. This is a single point of control (SPOC) part of the GUI. It allows you to enter any IMS commands that can be processed via the CSL/OM interface. It provides many productivity components that remember commands and allow you to do special things with the command output such as compare the results from one system against other systems.



The IMS DRD feature introduced in IMS V10 continues to be enhanced. It provides a lot of flexibility in dealing with IMS resource definitions and changes. It does however require the same level of management that we have with the system generation process today. It is easy to forget that when you make changes with DRD, the change process still has to be managed just like we do our systems generation process. Often our current processes have been in place for so long that we have forgotten all the things it provides. If we are making changes with DRD, we still need to honor traditional change control standards, maintain a history of what has changed, be able to automate installation of new changes, back out changes in error and create an environment where these dynamic changes are most likely to be successful. The ICM change process satisfies all of these requirements. You use ICM to make the changes to the required resource definitions. You then create a 'Change Package' that encompasses the changes you wish to make. Once the change package is complete and closed, it can no longer be updated. This satisfied one of the basic change control standards that what is submitted to the change control committee is what will be installed. Once approved, since the actual update of resources is done via a batch job, it can be scheduled using your job scheduling system. A key concept of changes made via DRD is that some changes require pre-conditioning of resources before they can be changed. The ICM batch install process does this using IMS supplied best practices to ensure that the changes are made successfully. The process gives you lots of flexibility to control the process when not all of the changes are successful. You can have the successful ones removed and/or rerun the job to accomplish installation of any changes that were not successful the first time. Of course, an audit log is produced of the installation process.



Once you have run some application testing runs, you need to be able to quickly understand in high level terms if there is any significant change either way.

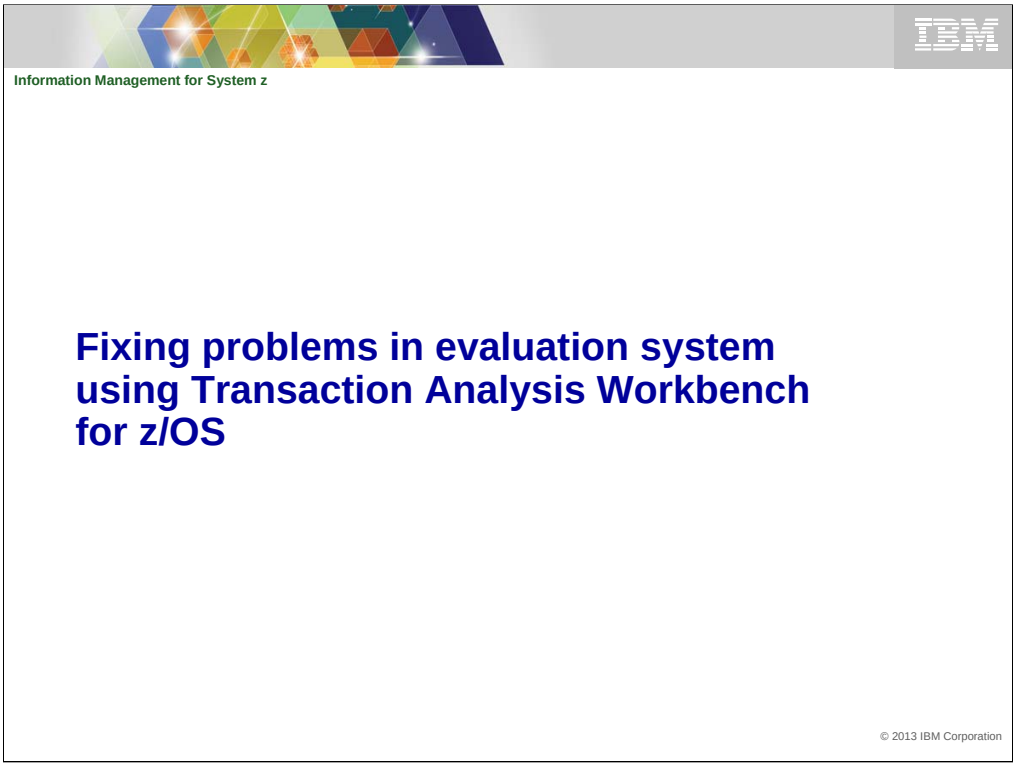
Performance Before and After Migration

Trancode	Proc Vers	Tran Count	Avg InputQ Time	Avg Process Time	Avg CPU Time	Avg OutputQ Time	Avg Total IMS Time	Avg IMS Resp Time	Avg DB Get Count	Avg DB Updat Count	Avg DB Wait Count	Avg DC Call Count
ACCOUNT	1210	167	0	345	30	0	345	341	14	135	0	3
	1310	178	0	356	31	0	356	347	14	135	0	3
BALANCE	1210	273	0	93	8	0	93	99	11	42	0	4
	1310	298	0	97	9	0	97	101	11	42	0	4
INVOICE	1210	546	0	174	43	0	174	182	710	67	0	4
	1310	563	0	177	45	0	177	185	710	67	0	4
LOGON	1210	444	0	274	9	0	274	282	14	40	0	3
	1310	423	0	281	11	0	281	287	14	40	0	3
MENU	1210	165035	6	134	16	0	140	84	50	0	0	1
	1310	167381	7	145	19	0	152	85	50	0	0	1
ORDER	1210	342	0	258	10	0	259	266	38	42	0	3
	1310	376	0	263	11	0	263	269	38	42	0	3
STOCK	1210	272	0	155	33	0	156	171	498	38	0	3
	1310	298	0	134	37	0	134	162				
WITHDRAW	1210	134	0	175	9	0	175	184				
	1310	156	0	182	11	0	182	191				

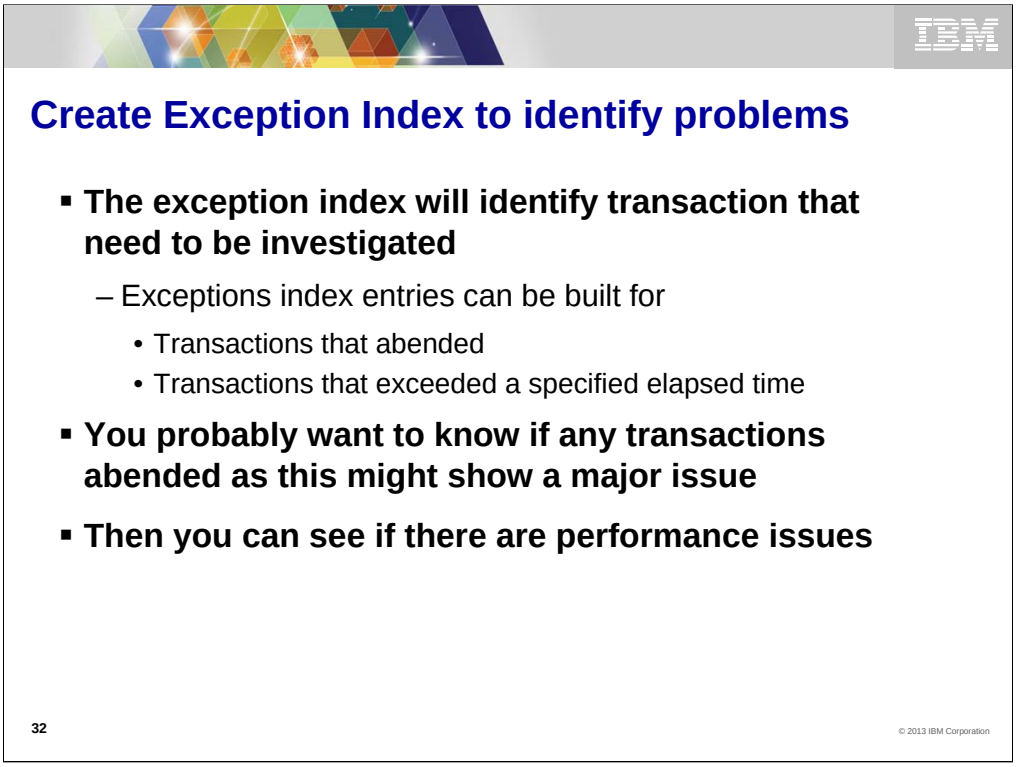
STOCK Transaction 9ms faster response time

© 2013 IBM Corporation

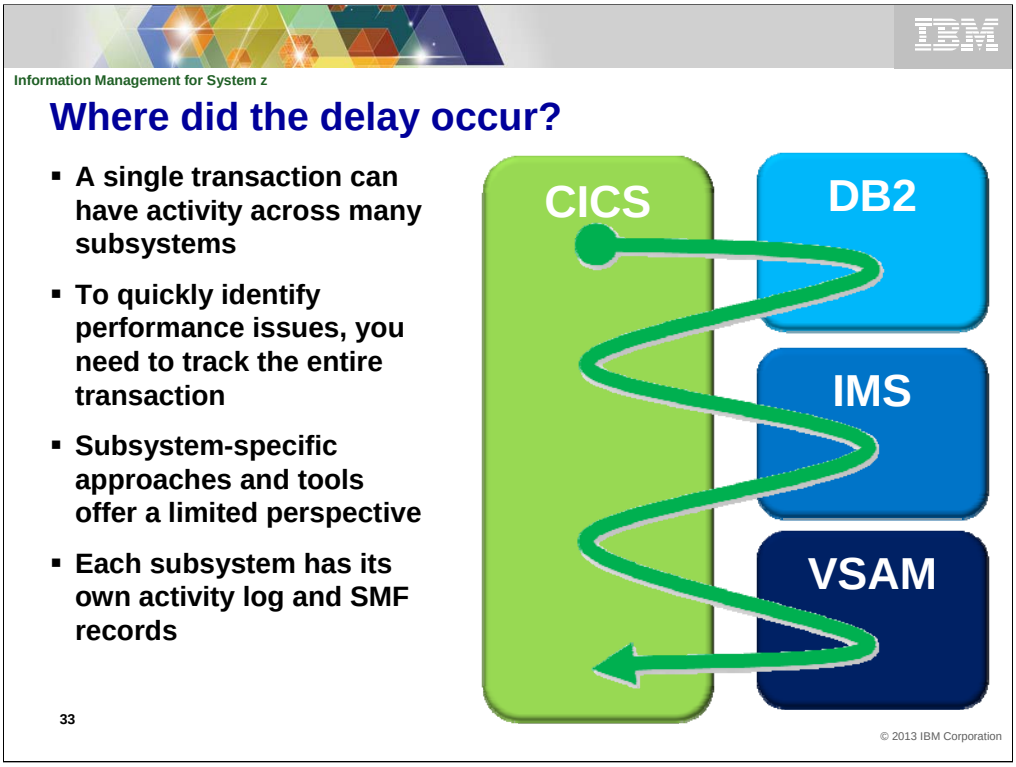
IMS PA allows you to quickly see on a transaction basis the results of the new run compared to the old one. You create this report by including logs from both level systems. For this example, you would want IMS logs from the same application testing runs. One set of logs from each application test.



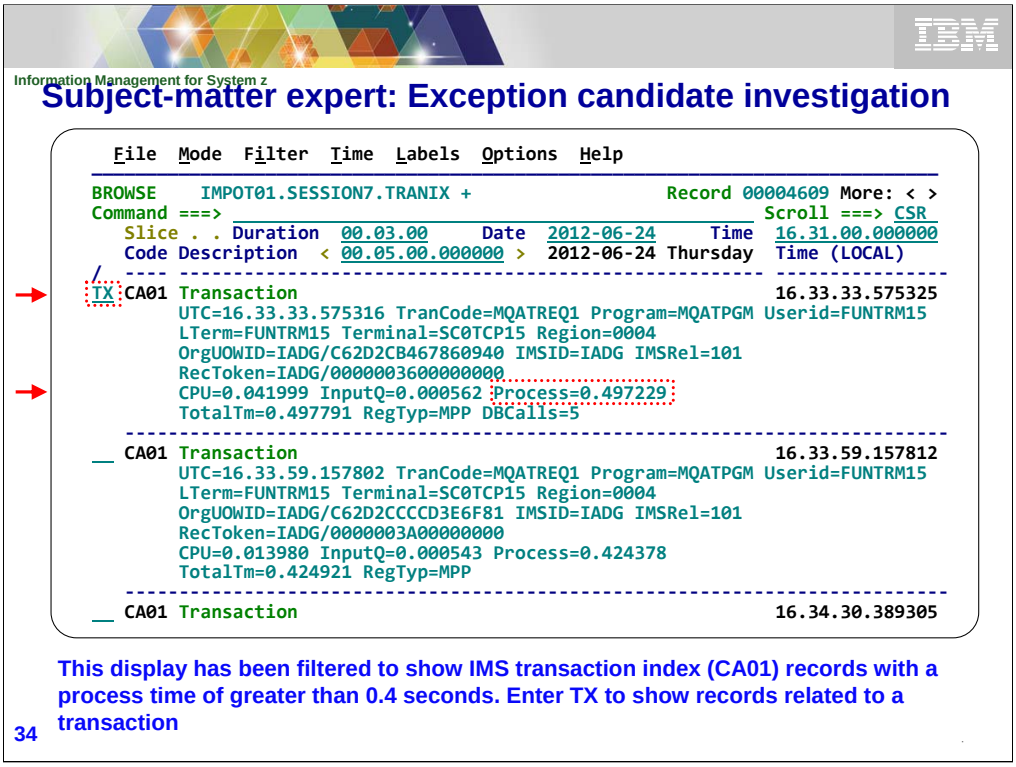
If there are issues with the evaluation system, you need to be able to find out what they are and fix them as easily and quickly as possible. The next slide show how you can use the IBM Transaction Analysis Workbench for this purpose.



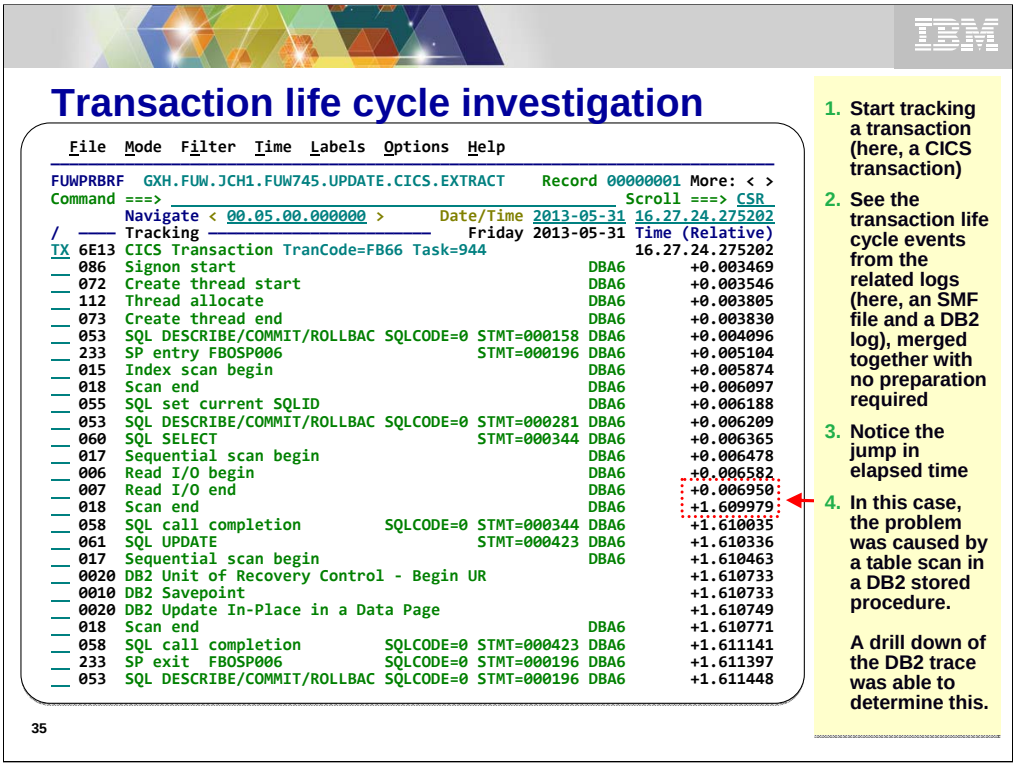
One of the first things you might do is to use the exception process of TAW to process the log data from the application tests to see immediately where you stand compared to the same application runs using the current level IMS system. The exception process can identify from the IMS logs any transactions that either abended and/or exceeded your specified response time criteria. This process will break down all the transactions that executed into only those that might need additional investigation. Once the exception index is created, you can filter its contents many ways to find the transactions you want to investigate. For example, if you find a transaction that abended U3303 (i.e. deadlock), you might create a filter to see how many transactions abended with a U3303 abend. Maybe the changed performance of the new release has exposed an application issue that allows for more lock contention and deadlocks.



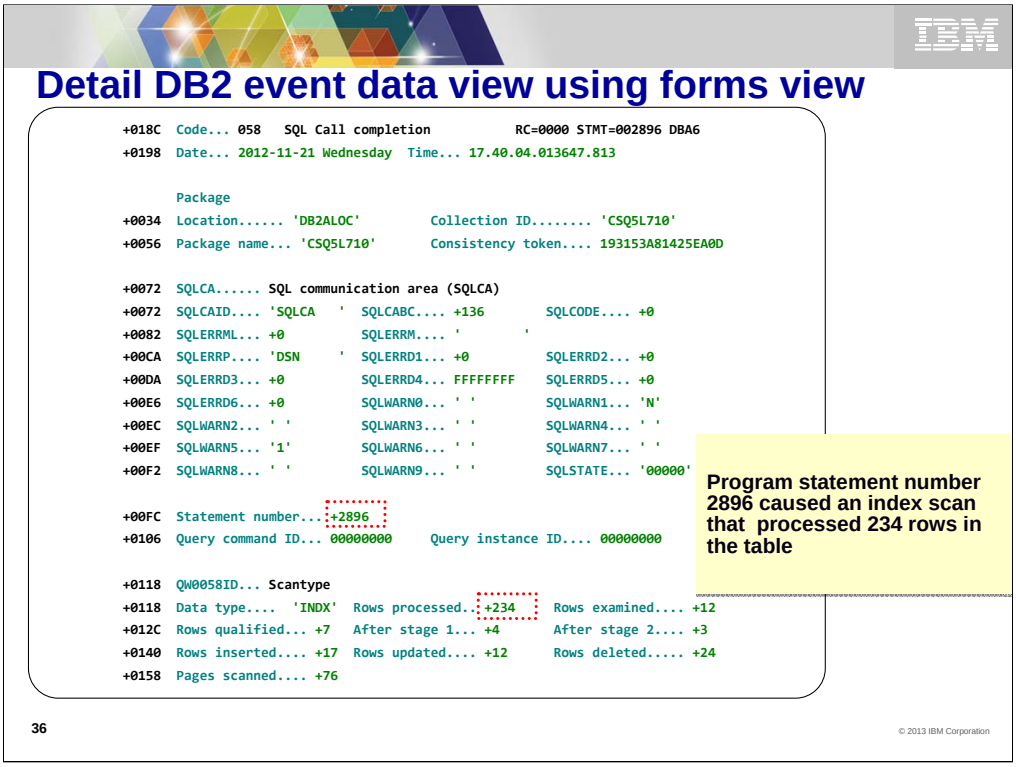
With the complexity of modern transactions, one of the first things we need to do when evaluating a transaction is find out what the transactions lifecycle looks like and where within the lifecycle the delay is occurring. That is one of the things that the IBM Transaction Analysis Workbench provides. It uses instrumentation data from all involved transaction managers and database subsystems to show the life cycle in terms of events with either the time between events or the event time relative to a point within the transactions life cycle.



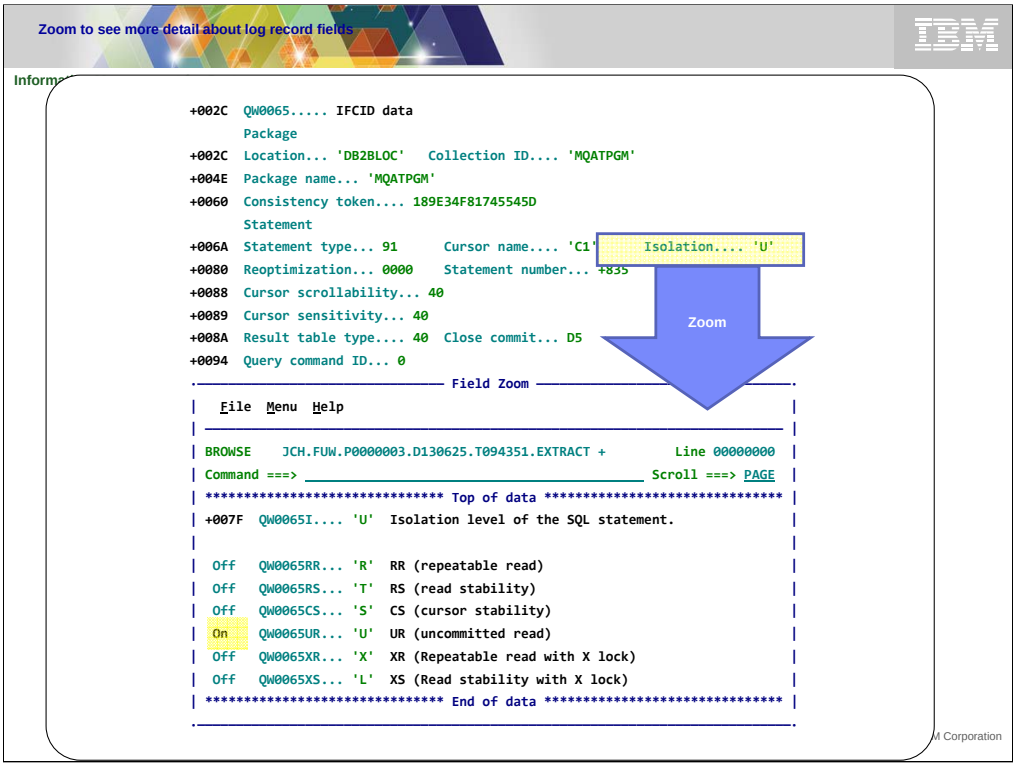
The next few slides are examples of how TAW displays the transaction life cycle. The foils are not related so you can see that TAW supports CICS, IMS, DB2, MQ, z/OS, and other instrumentation data. This is an example of looking at a set of IMS exception records. In this example a filter has been created that results in only those transaction exceptions that have exceeded .4 seconds being shown. You can adjust the filter to show transactions with greater and/or lower process times. Once you have determined a transaction you wish to evaluate by viewing the transactions lifecycle, you use the TX line command as shown against the exception entry. This results in the transactions lifecycle being displayed.



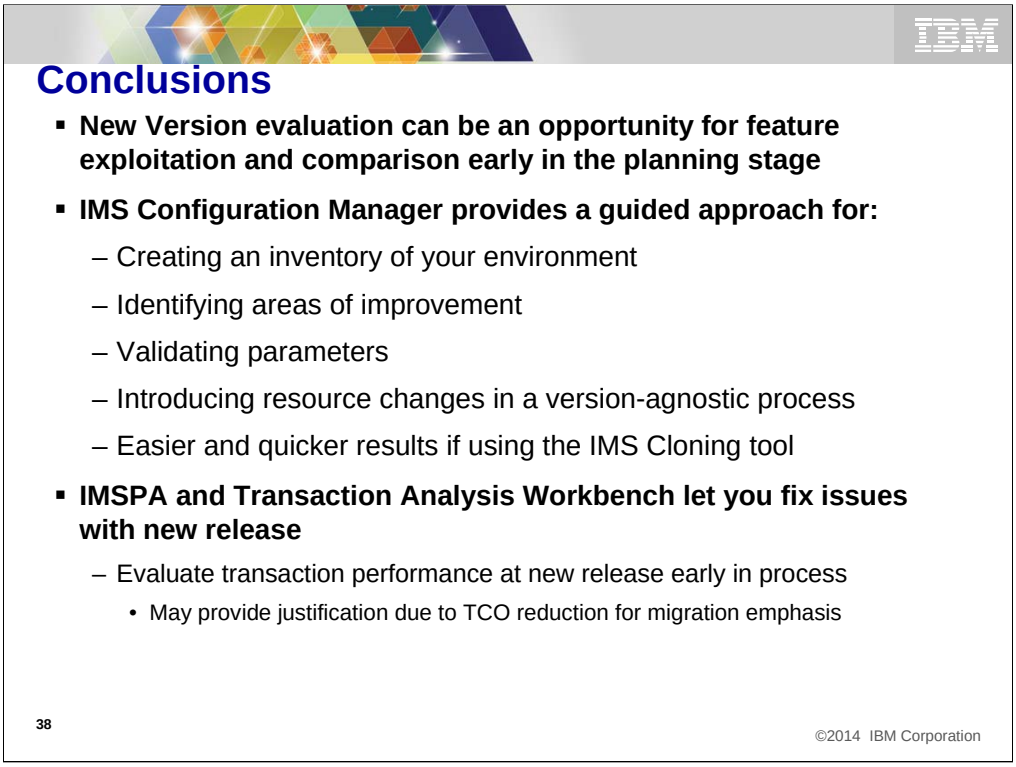
This is an example of a transaction but in this instance it is a CICS DB2 transaction. Note that the times shown in this example are relative from the beginning of the transaction. Using the relative time display, you can visually spot significant jumps in elapsed time that might indicate a delay in processing. This example is of a DB2 stored procedure spawned by a CICS transaction. Some of the trace events shown require some DB2 IFCID trace records that might not normally be collected by your installation. There is a tremendous amount of value to the information provided by the DB2 IFCID records but they can be so many that it is difficult to manually associate the ones that apply to a give transaction. TAW does this for you. When you track on a transaction using DB2 only the IFCID's for the transaction are shown in the life cycle view. This is what I meant in the earlier for that it is often easy to see where the delay occurred. It is not seen on this screen but TAW can also format the IMS internal traces for DLI Calls, Lock, Dispatcher, and other trace entries. If included in the instrumentation input, the lock entries are shown for the transaction in their relative position within the transactions life cycle.



With literally 100's of individual instrumentation records supports, you will be able to see the content of the instrumentation record you are interested in. This is example of IFCID trace record type 58 contents. The record contents is displayed showing the DSECT name and the data contained in the record. Some of these records can be very large so in this case, a Form was created that displays only the fields within the record that you are interested in. You can create any number of forms for each individual record. You can change or turn forms on and off at will if you need to see a different set of fields or the entire record contents. In this example I have highlighted two fields that often contain useful information about the SQL call. One is the statement number and the other is the number of rows processed by the SQL statement. Both of these might be necessary to resolve and/or understand how the individual SQL call is performing.



When you are viewing the contents of a instrumentation record you might find a field that appears to contain interesting values but you might not know what the field represents. You an place the cursor in any DSECT name field of the record and by hitting enter key, you will be shown a ‘Zoom’ box display that contain additional information about the field and its contents. In this example, the field named ‘Isolation’ contains a value of ‘U’. The zoom box display tells you that the field represents the Isolation level of the SQL statement. In this case, the Isolation level is uncommitted read. This can be a great teaching tool for the newer members of your team as well.



We have covered a lot of things today with the perspective of release to release migration and how IBM tools can help. This is by no means a complete list of ways other tools might help. Also keep in mind that these tools can be used in your day to day workings with IMS and the job you do. One of the key things I hope to have accomplished is to show you how easy and quickly you can create an evaluation IMS system at the new release and provide a quick understanding of the benefits of migration to the new release.

©2014 IBM Corporation

Break for questions about every 20-30 minutes.

Contact Info:

james_martin@fundi.com.au

Jim_martin@fundi.com.au

©2014 IBM Corporation

If you need more information or have questions later.

©2014 IBM Corporation