z/OSMF Value Proposition

Updated on : May 31, 2024

** Based on IBM laboratory results, your results may vary

Incident Log ⚠️

Use Case	**Without** Incident Log	**With** Incident Log
Recognizing a system-detected (dumped) problem occurred	• Requires 5 to 7 manual steps, • Skill in effective use of IPCS to extract data from each of the dumps. **Up to 5-6 minutes**	• Display in 1 click. Greatly reduced skill required **As little as 5 seconds**
Allow new dump to be taken for the same symptom	• Requires 7 to 12 manual steps • Skill in effective use of IPCS to locate the dump data set, obtain the symptom string, get into the IPCS DAE display, locate the matching symptom string (could be non-trivial) and indicate TakeNext on the IPCS display **Up to 15 minutes**	• Make the update happen in 3 mouse clicks **As little as 10 seconds**
Collecting and sending diagnostic data	• Requires 7 to 15 manual steps • Skill to locate the right log files • Build and run jobs • Rename the output datasets • Use an FTP job to send the different data sets and additional documents to the target destination **Up to 20 minutes** **Up to 30 minutes for sysplex components**	Send the material in 8 clicks: • Select the incident materials and additional documents • Specify the FTP destination information • Send the material • Check if FTP was successful **As little as 30 seconds**

Incident Log ⚠ (cont.)

Use Case	Without Incident Log	With Incident Log
Sending diagnostic data via PDUU or SFTP	• Requires 2 to 3 manual steps • Build and run jobs to send the different data sets to the target destination via PDUU or SFTP **Up to 10 minutes** **Up to 15 minutes for sysplex components**	Send the material in 8 clicks: • Select the incident materials • Specify the PDUU or SFTP destination information • Send the material • Check if file transfer was successful **As little as 30 seconds**
Viewing diagnostic datasets within context	• Context switch to ISPF green screen interface, • Log-in if necessary, manual input of dataset name. **Up to 1 minute**	• 2 clicks to open diagnostic dataset **As little as 7 seconds**
Searching APAR for matching service when a dump is captured.	• Requires 5 to 7 manual steps • Skill in effective use of IPCS to locate the dump data set • Obtain the symptom string and other diagnostic information • Visit IBM Support website to search for the matching APARs. **Up to 10 minutes**	Search for matching service in 3 clicks: • Select the incident • Click on "Search for Matching Service" • Click on the generated search for service URLs **As little as 5 seconds**

Sysplex Management

Use Case	Without Sysplex Management	With Sysplex Management
Display detailed Sysplex information including CFs, Structures, Couple data sets and CF connections.	• Users need to submit multiple JCL and run several commands to retrieve all Sysplex data • Users need to summarize and restructure above data to build the complete Sysplex information • Sysplex data cannot be exported easily **Hours even days**	• A few clicks to display Sysplex information on graphical views. • 1 click to export Sysplex data **3 - 5 minutes**
Perform complex Sysplex change operations.	• Users need to search help documentation, to figure out the detailed list of commands including the specific syntax for the desired change they want to make • Users may need to run multiple commands to implement the desired change operation. If one command is missed, the whole operation will fail and may not be recoverable • Users need to check the entire syslog to find out the command history **10 - 20 minutes**	• Normally just 1 click on graphic topology view. Sysplex Management task hides the complexity and runs all needed commands sequentially for end user • The history of all performed commands are saved automatically. With 1 click, a user is able to review a detail history of all operations performed **1 or 2 minutes**

Sysplex Management 🔄 (cont.)

Use Case	Without Sysplex Management	With Sysplex Management
Check Sysplex Couple Data Set health status and fix the potential issues	• Need to run several commands to retrieve each CDS's information • Personal judgment is needed to analyze whether the current CDS is in a healthy or not. If there are potential issues, the administrator needs to submit multiple JCL and commands to resolve the issues **30 - 60 minutes**	• System Management task discovers the potential health issues for users and warns them of any issue found via the graphical UI • A few clicks via the UI can resolve potential issues **1 or 2 minutes**
Update Sysplex CFRM policy data	• Users need to run JCL utility to retrieve current policy information including CF and structure data. • Cut and paste the new CF and structure data into the CFRM policy batch utility manually (can be tedious for a large installation with hundreds of structures) • Manual process is very error prone **Hours even days**	• 1 click to retrieve all the policy data and display it in the graphical view. • A few clicks to update policy data. Structure bulk editing with cross data validation is supported so that users do not need to update structures individually. **A few minutes**

Desktop UI

(Personal web-based desktop for managing z/OS)

Use Case	Without Desktop UI (using ISPF)	With Desktop UI
Multi-tasking	• Multiple window support in ISPF is less flexible and limited by the ISPF display	• Works similar to x86 OS desktops with a modern windowing system and taskbar that enables multi tasking support including running multiple z/OSMF tasks at the same time • This is particularly efficient in scenarios like data set editing
Look for target data set or USS file	• Needs to know Z specific way for searching data set or USS file • Always have to start from the beginning dialog regardless of how frequently the target data set or USS file is used **Repeated 4 steps to open a frequently-used data set member.**	• Common search window is provided with type-ahead support which avoids the need to remember exact data set name pattern. • Users can create their own folders and collect hot links of z/OS resources, start with data sets and USS files. • Users can open a data set or USS file for editing directly from desktop or specific folder with 1 click on the hot link. **1 double-click to open a frequently-used data set member.**

Desktop UI (cont.)

Use Case	Without Desktop UI (using ISPF)	With Desktop UI
Edit text format of z/OS data	• Very different editing experience with personal computer. (E.g., usage of Function keys for page up/page down/left/right) **Require Z unique editing skill.** **Repeated 4 steps to open a data set or file referenced in current data set or file.**	• VS-Code like Editor is built into z/OSMF Desktop to provide modern and consistent editing experience with a personal computer. (E.g., side preview, scroll bar, Ctrl+Z for Undo, GUI based Find and Replace, hot links to referenced data set or USS file) **Very little Z skill is needed.** **1 click to open a referenced data set or file.**
Create data set	• Requires deep understanding of lots parameters and their relationships • Can only access admin defined SMS constructs but not necessarily have knowledge about what is inside of each SMS construct **Deep knowledge of data sets and SMS constructs.** **Takes minutes to change every parameter and verify.**	• Embedded cross checking capabilities reduces the required knowledge and complexity. • Minimum set of parameters required, with size recommendations • IBM provided templates • User-defined data set templates • Admins can define global templates and share with all users **Much less knowledge required.** **3 clicks with a data set template in seconds.**

z/OS Operator Consoles 🎧

Use Case	Without z/OS Operator Consoles	With z/OS Operator Consoles
Monitor console messages in real time	• Only a small number of latest messages can be displayed **Can not view messages in real time**	• A scrollable view that refreshes console messages in real time • Bar chart graphic shows message trends and is associated with the individual messages in the console view **Refresh messages in real time**
Filter logs with keywords, and severity or color	• Only basic support on 3270 with find, find next. • No support to filter with severity or color. • Need to call native function programmatically or offload the data from the platform **Hard to accomplish**	• Find and highlight all support in only a few seconds • Easy action to filter message by severity, color, time, system, or job in a matter of a few seconds **Only a few clicks that takes seconds.**

z/OS Operator Consoles (cont.)

Use Case	Without z/OS Operator Consoles	With z/OS Operator Consoles
Find corresponding help of a message	• Copy message ID and Google it. • Scan through results and (hopefully) will find a relevant link **Minutes**	• Instantly show help messages on mouse hover actions directly in the message view. **Only a few seconds**
REST API for issue command, retrieve operations log	• Redevelopment using native REXX script. **Require knowledge of z/OS native function or programming language.**	• RESTful services support, can be driven from any platform using any programming language. **No z/OS specific programming is required.**

Workflows

Use Case	Without Workflows	With Workflows
Performing complex step-by step tasks(eg: 50 short running JCL steps)	• Requires manual steps to select JCL, submit JCL and monitor the job status • Requires manual selection of the next JCL based on different job return code **Up to 50 minutes**	• Perform the steps as a workflow instance automatically with 1 click. • Greatly reduced human interactions in monitoring and manual actions **As little as 5 minutes**
Junior System Programmer needs to understand how to perform complex step-by step daily tasks.	• A skilled system programmer will need to manually gather all the required information to build the multiple sequential steps required for a task including JCLs, sequence and relationship between the steps among other details **Hours**	• Senior system programmer is able to capture and persist their experience into a workflow, with or without automation • Workflows also supports dependency checking to ensure the right order of steps • Junior system programmers can now follow a guided set of workflow steps to complete tasks manually or automatically **No extra time to be spent**

Workflows 📊 (cont.)

Use Case	Without Workflows	With Workflows
System Programmer needs to perform similar step-by-step tasks in multiple z/OS systems.	• JCLs and scripts need to be copied and pasted to each desired z/OS system • They need to be manually adjusted and performed on each system. **Hours**	• Shared workflow definition file is used in combination with a system specific variable input file • User only needs to choose target sysplex and system when creating the workflow instance to execute. **No extra time to be spent**
Collect historic records for z/OS System Operation task	• Records are dispersed across each system • Need to collect records for all steps from each system and manually consolidate them into a singular report **Minutes to Hours**	• The workflow execution history and job output are saved centrally and can be reviewed together at any time with a few clicks. **Seconds**

Security Configuration Assistant

Use Case	Without Security Configuration Assistant	With Security Configuration Assistant
Discover security requirement for specific function/product/solution	• Look up the information from the product's setup or configuration book or online help • Cross-book references can be a challenge if a solution references several products **Minutes to Hours**	• A flexible, well-organized and browser-based Web UI that displays the security requirements at once from top to bottom by function/product/solution • The JSON format is used to document the security requirements. JSON is an industry standard with education and skill widely available **Seconds**
Investigate if security requirement of specific function/product/solution is satisfied.	• Enter commands one by one to verify the security requirement • It takes some time when there are many security requirements for a product • Manual steps may have to be repeated across different products or the user may create a custom script for each product. **Minutes to Hours**	• 1 click from browser-based Web UI to automatically check if security requirements are satisfied or not. • Avoid repeated execution/failure/fix loops **Seconds**
Take care differences of all Enterprise Security Manager (ESM).	• Normally user has to translate RACF commands into sample jobs for other non-RACF ESM products • Then they need to issue commands of specific ESM accordingly to validate if security requirements are satisfied or not **Minutes to Hours**	• 1 click to validate if security requirements are satisfied or not which is consistent among all ESM products. **Seconds**

z/OSMF ISPF

Use Case	Without z/OSMF ISPF	With z/OSMF ISPF
Operate multi windows in ISPF	• Users can use F2 key to split the whole ISPF window into two small windows, but this cannot increase the whole window size so the split windows are small and hard to use • Users can use F9 key to do switch between different windows, but native ISPF cannot display two whole windows in one screen simultaneously	• WebISPF supports horizontal and vertical split window • The sub-windows size can be adjusted easily by clicking and dragging • The entire web browser can be used to display ISPF windows so that users can operate multiple windows together while making use of the maximum possible screen real estate available for each window
Require ISPF operations which can not be done in other z/OSMF plugins during daily job in z/OSMF UI	• Users need to open both z/OSMF UI and traditional ISPF • May need to constantly switch between z/OSMF UI and traditional ISPF **Frequent switch between Browser and PCOM, also require different operation behavior.**	• Users can open z/OSMF ISPF and stay in z/OSMF UI to efficiently complete all tasks. **1 click to switch to ISPF and operation behavior keeps the same with web-UI operations.**

z/OSMF RESTful Services

Use Case	Without z/OSMF REST Services	With z/OSMF REST Services
Skill/Language to drive z/OS operations (E.g., job operations, console operations, data set/USS file operations, OPERLOG/SYSLOG download, etc.)	• Require z/OS native programming skills like REXX, PLX, JCL **Up to several months in order to pick up z/OS native programming skills**	• Industry standard skills including HTTPS, JSON parsing • User can choose their preferred programming language because most languages support HTTPS communication **As little as several days typically**
Drive z/OS operations from distributed platforms	• Hard to implement • Sometimes user may need use FTP, which is not secure	• z/OS operations can be driven in a consistent way from almost any platform that supports HTTPS
Browse API document and try APIs	• User needs to download books and refer to the book in advance • In order to try APIs, user needs to write code which may require z/OS native programming skills **Up to several days initially**	• z/OSMF REST API supports Open API standard • A live API document is integrated into z/OSMF • Any user with a connection to z/OSMF can access the live API document from the z/OSMF desktop • User can try APIs directly from the API document without need to write any code **As little as several minutes**

Network Configuration Assistant

Use Case	Without Network Configuration Assistant (using Policy Agent only)	With Network Configuration Assistant

Filter unwanted network traffic from your z/OS system	• Learn how to set up IP filters
	• Review the IP Configuration Guide
	    • Configure the Policy Agent application
	    • Create configuration policy for IP Filter rules
	    • Configure default filter rules in the TCP/IP profile
	    • Configure the TRMD application
	    • Configure the Syslogd application
	Hours (or even days for initial setup)
		• Configuration Assistant provides guidance
		• Go to IP Security Perspective
		• Add a connectivity rule for an IP Filter
		• Use Application Setup Tasks to assist with the configuration and setup of the required applications
		• The Configuration Assistant will generate and help you deploy the configuration files to your z/OS system
		
		As little as 30 minutes

Secure your TN3270 server connections with TLS	• Manual process
	• Review the IP Configuration Guide
	    • Configure the Policy Agent application
	    • Configure TTLS in the TCP/IP profile
	    • Configure the Syslogd application
	    • Create configuration policy for AT-TLS for your TN3270 Server
	Hours (or even days for initial setup)
		• Configuration Assistant provides guidance
		• Go to AT-TLS Perspective
		• Select the AT-TLS rule for the TN3270 server and enable
		• Use Application Setup Tasks to assist with the configuration and setup of the required applications
		• The Configuration Assistant will generate and help you deploy the configuration files to your z/OS system
		
		As little as 30 minutes

Network Configuration Assistant (cont.)

Use Case	Without Network Configuration Assistant (using Policy Agent only)	With Network Configuration Assistant
Set up TCP/IP configuration for images at alternate locations (for planned or unplanned failover)	• Create system symbolics in TCP/IP configuration to manage differences between the main and alternate locations • Use an out-of-band method (such as a spreadsheet or other notes) to keep track of symbol values in the alternate locations **(error-prone)** • Manually update symbol values by location, making sure they stay consistent **(error-prone)** • Manage multiple copies of the configuration file at each location (e.g., when one is updated, they all have to be) **Hours for every change, and error-prone**	• Configuration Assistant provides guidance • Use the Alternate Locations function of NCA TCP/IP profile perspective to manage a single configuration with different symbol values at different locations (symbol values managed in the NCA) • Use View All Symbol Details reports to accurately track and manage symbol values in multiple locations • Use NCA multiple installation function to push updates of a configuration to multiple locations with a single action • See attached slides 7-8 for illustration **As little as 30 minutes, and not error-prone**
Create zERT Policy-Based Enforcement Rules for TLS (V2R5 new function)	• Manual process • Review the IP Configuration Guide • Manually configure a rule for general enterprise requirements • Manually configure exception and catch-all rules, making sure to order them carefully **Hours (or even days for initial setup)**	• Configuration Assistant provides guidance • Go to zERT Perspective • Create required TLS rules in any order • Create a TLS rule set • Use checkboxes to select the TLS rules in the rule set, and their order. • The Configuration Assistant will generate and help you deploy the configuration files to your z/OS system **As little as 30 minutes**

IBM zERT Network Analyzer

Use Case	**Without** IBM zERT Network Analyzer	**With** IBM zERT Network Analyzer
Identify z/OS TCP/IP connections using weak SSL/TLS versions (SSLv2-TLSv1.1)	Either: • Write your own program that parses zERT SMF records, format the data into a usable form, and provide users a method to quickly locate the specific data of interest OR • Purchase an IBM or ISV product that can process zERT SMF records and perform user-directed analysis **Write your own: Weeks to months** **Purchase: Cost of software product** **Once solution is in place, time for the stated task varies based on solution's query/analysis capabilities. At best, it might be comparable to the zERT Network Analyzer**	• Create your own customized queries for the exact protection attributes you're interested in • Building a query for SSLv2-TLSv1.1 takes about 2 minutes • Once you save a query, you can run it whenever you want with a single click • Query results are displayed in a summary format that allows you to easily drill down and identify the client/server pairs that are using weak protocols **Less than 1 minute**
Similar scenarios (weak algorithms, short key lengths, unprotected connections, etc.)	**Same as above**	**Same as above**

Resource Monitoring

Use Case	Without Resource Monitoring (using RMF ISPF Monitor III Reporter)	With Resource Monitoring
Checking the performance status for several sysplexes	• You need a Monitor III Reporter session on each sysplex, • Need to manually consolidate data from different reports. • Monitoring of Linux resources has to be done with other tools **Up to 15 minutes to look up each sysplex and high degree of skill needed to interpret reports**	• Cross-sysplex performance monitoring from a single point of control • Quick easy to consume red-yellow-green health indicator for all your system in a single panel • Linux monitoring features are fully integrated **Just seconds to see the health of all your sysplexes (and Linux images)**
Explore & compare the processor usage of specifics jobs	• Tabular reports are a fixed layout • Can be viewed only one at a time with limited ability to customize and filter the presentation of the data • You have to manually consolidate data from different reports **A significant amount of time, depending on data required and correlations needed. In some cases, generating reports is not possible.**	• The monitoring desktops are fully customizable • Specific metrics of selected resources can be added to a desktop and are visually presented as charts • Multiple desktops can be started in parallel via different tabs • Advanced filtering features allow you to conduct sophisticated performance analysis. **About 5 minutes to set up a custom monitoring desktop, 3 mouse clicks to view real time statistics**

Software Management 🖥️

Use Case	Without Software Management	With Software Management
Software deployment according to IBM recommendations	• Only choice is to develop and maintain in-house z/OS software deployment tools and processes • Limited capability to add best practices for deploying software • Possibly incomplete documentation	• Official strategic solution from IBM • Reduce or reallocate resources normally used to develop and maintain in-house deployment tools • IBM best practices for deployment built in • Complete solution with documentation and official IBM support.

Error checking	• Manual processes that can be error prone including:
	  • Missed steps during deployment process due to limited internal documentation of in-house tools
	  • Incorrect or incomplete requisite checking prior to deploying software resulting in potential software compatibility issues
		• Reduced errors and missed steps during the deployment process
		• A deployment checklist provides sequential end-to-end guidance throughout the entire deployment process
		• Requisite and regression checks identify dependencies and potential errors prior to deploying software

Skills and risks	• High SMP/E skill level needed	• Can help reduce SMP/E skill level required by providing a guided task flow to complete a deployment • IBM estimates a reduction of 30% in the required SMP/E skill level needed when deploying software (following IBM-provided task flow and instructions)

Software Management 🖥️ (cont.)

Use Case	Without Software Management	With Software Management
Identify missing requisite PTFs on instances that will share resources with the deployed software.	• Manually create and run SMP/E jobs to identify required missing service on other software instances • Analyze SMP/E report output manually • Missing coexistence PTFs can cause sysplex wide outages that can require fallback to prior levels **Deep SMP/E skills required**	• Prior to the actual deployment of software, a full missing requisite PTF analysis and report is generated identifying any missing PTFs • Prevent potential outages caused by missing coexistence PTFs • Supports cross-system checking **Fewer skills, simpler**
Identify regressed software on the prior level instance. Identify required actions from PTF HOLDs.	• Manually create and run SMP/E job to compare source with prior instance • Can not be done if source and prior instance are on different systems • Manually identify the delta and any required actions • No SMP/E report capability to compare source with the prior instance • Ignoring actions or regressing service on the target system can cause critical problems	• Prior to the actual deployment of software, a full regression analysis and report is generated identifying any actions from PTF HOLDs that need to be taken • Prevent critical errors or corruption of data as a result of a deployment on the target system • Supports cross-system checking • Built into the deployment process and can be executed in a few minutes (combined with requisite checking action)

Software Management 🖥 (cont.)

Use Case	Without Software Management	With Software Management

Identify complete content of software to be deployed.	• Manually analyze SMP/E inventory to identify the correct content to deploy
	• More typical is to deploy entire volumes or data sets by prefix. This requires strict volume and data set name conventions, which contribute to user errors including:
	   • Copy PDSE without UNIX file system (common with WAS V6)
	   • Copy one data set without another causing partial APAR fixes.
	   • Renaming a data set causes it not to be copied
		• Data correctness and integrity of a software instance being deployed is guaranteed because z/OSMF Software Management will automatically identify all the data sets that are part of an instance being deployed.
		• Prevent all errors related to missing content of a software instance being deployed

Modify the target software layout (data set names, location, and catalog).	Manual:
	• Define target system datasets, volumes and catalogs
	• Ensure the desired catalogs will be updated and new HLQs are identified
	• Make best-guess estimates for volume free space and required space for target data sets
	• Mistakes typically require cleanup and deployment jobs to be re-run
	
	Hours
		• Automated guided deployment checklist and wizard will:
		   • Analyze target system catalogs to identify which will be updated for the target data sets - accounting for new, deleted, and replaced data sets
		   • Calculate volume free space and ensure target data sets will fit, accounting for data sets added, deleted, and replaced
		   • Ensure no existing data sets are accidentally overwritten or lost
		
		Few minutes

Software Management 🖥️ (cont.)

Use Case	Without Software Management	With Software Management
Create deployment jobs	• Create jobs from scratch or copy IBM supplied samples. Manual and error prone process • Easily allows users to mistakenly deploy incomplete subsets of software • Inhibits exploiting new technologies, like zFS	• Complete and accurate jobs are automatically created that always deploy complete software instances • Supports current technologies
View the planned target system updates before running the jobs.	• If done at all, must manually compare source with the target system	• Automatically generate reports to preview and summarize the changes that will be made to the target system prior to making the actual updates • Save deployment reports for later audits or problem determination
Identify software products that have reached end of service dates, or will reach the end of service dates shortly.	• Identify all products and features in use on all systems in the enterprise • Research each of those products, by looking at announcements for end of service dates, across software vendors • Gather information together to understand an enterprise picture of what software is end of service **Days or even months, which is usually too costly to do comprehensively.**	• Select all software instances in the enterprise, in as little as one click • Ensure that the latest end of service inventories have been retrieved, in a couple of clicks • Run the End of Service report, in two clicks **Seconds**

z/OS Change Tracker

Use Case	Without z/OS Change Tracker	With z/OS Change Tracker
Protection of critical system libraries and monitoring of change activities	• Limited to no ability to prevent changes from being made beyond External Security Manager. • No real-time way to monitor who's making changes to what libraries, when, and how • Requires multiple manual steps to discern if and what changes were made, and why • Manual tracking methods can cause unnecessary overhead and lack real-time insights **Up to hours/ days**	• Add and display monitored resources in a few clicks • Establish base versions for protected members by default • Real-time, system-wide change tracking and control for system libraries • Ability to 'lock' data sets and members until a checkout is performed, regardless of user permissions, to permit only changes when you want them. **Automatic monitoring**
Member-level backups and recoveries	• Requires many manual steps to search through potentially thousands to identify what has changed, locate previous saved versions if they exist, and recover an old version • Can be very time consuming and expensive if changes made are harmful to an environment **Up to hours/ days**	• Ability to locate what changes were made quickly in just a few clicks • Customizable rollback capabilities to quickly undo the effect of unplanned/ undesired changes • Can compare current version with previous versions side by side to identify differences • Ability to recover into another data set, or recover in place, in just a few clicks **As little as seconds**

z/OS Change Tracker 🗒️ (cont.)

Use Case	Without z/OS Change Tracker	With z/OS Change Tracker
Need for auditable tracking mechanisms	• Potential that changes are not tracked at all, making evidence collection very difficult • Lack of auditable protection and change evidence can fail to meet compliance requirements **Can lead to failed audit requirements**	• Provides auditable evidence for all protected data sets and their members • Notes action taken (add, update, rename, delete, etc.), users who took the action, time/date, and the program used • Audit log reports are accessible with just a few clicks in the z/OSMF plugin • Meets DORA requirements and other upcoming compliance regulations **Automatic auditing logs**
Compare data sets and volumes across environments	• Manual identification of changes/additions/ deletions between data sets or volumes can be very time consuming • Can be very hard/ impossible challenge to spot differences between large sets of data **Up to hours- days**	• Data sets and volumes can be compared against one another to find member-level changes, additions, and deletions. • Comparisons are simple to create and run and can be saved and re-run at any point • Long-running comparisons are run in the background to ensure productivity • Results are displayed in an easily consumable summary page with filter, sort, and export capabilities **Automatic or manually in seconds**

Workload Management

Use Case	**Without** Workload Management (using ISPF WLM Application)	**With** Workload Management
Optimization of a service definition based on best practices	• Read through WLM-related manuals and identify best-practices. • Print out the service definition and investigate it with respect to proposed best-practices • If required, modify the corresponding policy elements **Hours (or days when done initially)**	• Check the best-practice hints the GUI displays for policy elements • If required, modify the corresponding policy elements **Minutes (or hours when done initially)**

Review of service definitions for daily changes, migration, consolidation	• To get an overview of a service definition you:
	    • Print it to a data set
	    • Download the data set
	    • Print or feed it into the Service Definition Formatter tool to filter and sort policy elements
	5-10 minutes until review can start
		• Open a service definition from the service definition repository
		• Navigate through it using links
		• Filter and sort policy elements in the tables
		
		Seconds until review can start

Transfer policy elements from a test service definition to a production service definition	• Print out the test service definition • Update the production service definition by typing in the changes **Up to several minutes per policy element**	• Simultaneously open the test and production service definitions • Copy over the changed policy elements via copy and paste operations **Seconds per policy element**

Capacity Provisioning 📊

Use Case	Without Capacity Provisioning	With Capacity Provisioning
View active CP policy and compare with data provided by RMF and WLM	• Start stand-alone Windows-based client (CPCC) for Capacity Provisioning • Connect to CPM and display active configuration report • Open z/OSMF in a browser and inspect RMF and WLM data **5 – 10 minutes until all tasks are completed**	• Use integrated z/OSMF GUI solution to work with CP, RMF and WLM • Analyze and compare data provided by each exploiter **2 – 3 minutes until all tasks are completed**
Operating person needs to reuse existing connection	• Connection information like hostname, protocol and port needs to be manually gathered • Available domains must be known **Up to 5 minutes**	• Usage of shared connection repository in z/OSMF • List of available domains is retrieved from server and shown to user **No extra time to be spent**
Installation of the capacity provisioning UI application	• Install Windows client (CPCC) on workstation. **Hard to install on managed clients, 20 minutes otherwise**	• Centrally managed z/OSMF application available to all authorized users **No extra time to be spent**

Cloud Provisioning & Management for z/OS

Use Case	Without CP&M	With CP&M
Create new z/OS systems from scratch, repeatedly and consistently, in a monoplex configuration	Deploying new z/OS systems in an IBM Z LPAR is a complex and arduous process requiring the skill of experienced system programmers. **Days or weeks**	Automate z/OS provisioning with reusable templates and a guided process for less experienced system programmers **As little as one hour**
Transfer key skills required for building a new z/OS configuration from senior system programmers to early tenure system programmers	Because configuration for z/OS is complex and difficult, educating early tenure system programmers can be difficult as well. There is no centralized documentation that describes all the steps required to setup a z/OS instance. **Weeks or months**	All the steps associated with z/OS provisioning are described in an IBM supplied best practices workflow. System programmers can customize any step and add documentation specific to their environment in the workflow. Early tenure system programmers can quickly learn about the z/OS provisioning process from one central place. **Couple of days**

Cloud Provisioning & Management for z/OS 🖼️ (cont.)

Use Case	Without CP&M	With CP&M
Create new IBM middleware and other software instances repeatedly and consistently	Deploying new middleware and other software instances on existing z/OS systems is a manual and complex process requiring the skill of experienced system programmers **Days**	Rapidly provision z/OS subsystems and software with reusable and easy-to-configure, IBM sample templates and a guided process for less experienced system programmers **As little as one hour**
Manage and orchestrate provisioning resources	Keeping track of and allocating multiple resources across environments involves several manual steps and can become complicated quickly **Up to 15 minutes**	• Support a robust, multi-tenant environment with defined resource pools that isolate services and resources for different lines of business • Dynamically orchestrate resource requests, leaving behind the days of manually tracking and orchestrating z/OS resources for each instance **As little as 10 seconds**
Track and manage provisioned instances	Tracking multiple provisioned instances and performing lifecycle actions on them requires manual steps across different tools and interfaces **Hours**	Automatically track and see details on provisioned instances and their resources in a single registry, and directly perform lifecycle actions (e.g. deprovision, start/stop instances) in a few clicks **As little as 10 seconds**