



# *The Business Case for Launch Express for ISPF*

## Overview

- **Launch Express** is quick and easy to implement
- **Launch Express** delivers substantial and immediate performance enhancement
- **Launch Express** provides significant cost savings
- **Launch Express** significantly reduces LOGON EXCP and CPU overhead (on the order of 80%-90%)
- **Launch Express** has compelling ROI characteristics and offers immediate payback
- **Launch Express** is a return on investment no brainer!
- **Launch Express** – don't run ISPF without it!
- **Launch Express** – minimum investment, maximum ROI

---

**Bottom line:** process the same volume of work with fewer resources  
to achieve significant cost savings

## Questions about your TSO/ISPF Environment

- Have you ever wondered if there's a better way to install and maintain ISPF products and applications?
- Have you ever needed to compress a dataset only to discover 100 TSO users have it allocated but aren't using it?
- Do you maintain multiple TSO LOGON procedures for different groups of users?
- Must you change numerous LOGON PROCs, EXECs and CLISTs every time an ISPF application is added or modified? Do you sometimes overlook one or more of them?
- Must you pre-allocate your program libraries to avoid S806 abends?
- Do you have a large number of logon procs at your site?
- Do you ever get tired of adding ISPF application datasets to every proc?
- Is ISPF application management an issue?
- How many datasets are there in your LOGON procs?
- Do you merge ISPF dialog libraries for multiple products and applications into one huge dataset by type (panels, messages, skeletons, etc)?
- Do you use the ServerPac merge dialog, merge job (or other means) to merge ISPF dialog libraries?
- How many members are there in your largest ISPF dialog libraries?
- Do you allocate datasets with LOGON JCL, only to have a LOGON EXEC or CLIST free and then reallocate many of the same datasets (this is absurd but it occurs surprisingly often)
- Does your organization pre-allocate and pre-establish the environment to run any and all ISPF applications a user, group or department may utilize?
- Do you experience (high) contention on STEPLIB volumes in a shared DASD environment?

# The Business Case for Launch Express for ISPF

## Launch Express: a complete solution and methodology

- Launch Express offers a complete solution and methodology for managing ISPF applications and their dialog libraries
- Launch Express is *a better way* to install, maintain and use ISPF applications
- Launch Express is the missing piece of the puzzle to make dynamic ISPF a fully viable alternative
- Launch Express provides the missing facilities to dynamically establish the environment required to run any ISPF application, even troublesome ones that issue LINK, LOAD, ATTACH and XCTL macros
- Launch Express lets you invoke ISPF applications dynamically. This helps you shrink your logon procs to a manageable size, eliminate redundant procs and get better overall ISPF performance.
- Launch Express implements 'dynamic ISPF' as a commercial product
- Dynamic ISPF may have been attractive but was impractical until now because of load libraries and ill-behaved applications
- Launch Express is comprised of best of breed components
- Launch Express makes dynamic ISPF fully feasible
- Launch Express enables ill behaved applications to run without restriction and without being subject to S806 (program not found) abends
- Launch Express augments native TSO/ISPF with the facilities missing from a *complete* solution. Note that ad hoc solutions developed by systems programmers are often inefficient and may lack documentation and support
- In contrast, Launch Express facilities are backed by knowledgeable, responsive and professional support
- The overhead of dynamic allocation is offset by much faster searches for program modules and ISPF dialog objects. With Launch Express, there is usually only one library with relatively few members to search
- Problems with ISPF application installation and deployment are greatly reduced or completely eliminated with Launch Express
- Launch Express operates with a very simple and universal TSO LOGON procedure that can be used by all TSO users
- Bottom line – Launch Express provides the software and methodology required to adopt dynamic ISPF and enjoy its benefits

## Launch Express Benefits

- Dramatically reduce TSO logon times
- Reduce overhead and response times
- Add functionality while reducing overhead
- Improve Reliability, Availability and Serviceability
- Enable ill behaved ISPF applications to run successfully (without S806 abends)
- Easily convert ISPF applications incompatible with LIBDEF ISPLLIB into well-behaved dialogs
- Bring order and uniformity to ISPF application invocation
- Uncomplicate your ISPF applications
- Dataset allocations become more manageable and comprehensible
- Launch Express facilitates installation and subsequent upgrades of software products, site developed ISPF applications and z/OS itself!
- Launch Express delivers substantial performance improvement – benchmarked and verifiable
- Launch Express improves performance, increases flexibility, simplifies maintenance, reduces complexity and permits/promotes more granular security
- Launch Express lets you remove all those datasets from your LOGON JCL, execs and CLISTs

# The Business Case for Launch Express for ISPF

## Launch Express Benefits (continued)

- Optimize the search for ISPF dialog objects and achieve significant cost savings / performance improvement
- Shorten the search path for programs and ISPF dialog objects
- Launch Express streamlines the search so program modules and ISPF dialog objects are located immediately
- Launch Express allows the operating system to find program modules and dialog objects in the *first* library searched
- Eliminate major maintenance headaches
- Launch Express is consistent with IBM recommendations for optimizing the LOGON Process (from TSO/E Customization – Publication SA22-7783) that include allocating as few datasets as possible and avoiding the use of STEPLIBs in LOGON JCL

## Launch Express improves application dataset security

- Coarse security relies on LOGON procs to allow / disallow access to ISPF applications. Organizations often make use of multiple LOGON procedures that must be customized with specific ISPF dialog datasets to allow or disallow access to certain ISPF applications
- Launch Express promotes granular, genuine security at the dataset level (rather than false sense of security provided by LOGON JCL, CLISTs and EXECs)
- The granular security possible with Launch Express versus coarse LOGON proc security is analogous to the difference between row level locks and table space locks or record level locks versus dataset enqueues
- Launch Express lets you implement real security for ISPF applications with a security package like RACF to control access to ISPF application datasets
- Promotes more granular security because users are authorized to access only what they need and use
- Eliminate the need to maintain multiple TSO LOGON procedures for different groups of users

## Launch Express and ISPF Program Libraries

- Program libraries are the most confusing aspect of ISPF applications and perhaps the biggest obstacle (stumbling block) to invoking ISPF applications dynamically
- Program libraries and S806 abends are the biggest obstacle to invoking ISPF applications dynamically
- Every time you press ENTER, the system searches the STEPLIB, JOBLIB, LPA and LINKLIST concatenations for each TSO command
- TSO STEPLIBs can create poor response times and queuing - in a shared DASD environment with TSO users on different systems, the overhead of issuing RESERVEs on STEPLIB volumes is extremely high
- Launch Express reduces contention on STEPLIB volumes (i.e. delays for other users of the volume)
- Shorten the search path for load modules (PDS) and program objects (PDSE)
- DB2 subsystem libraries need not be in the LINK-LIST nor must they be pre-allocated in your TSO/ISPF session. Launch Express lets you dynamically allocate and use DB2 libraries within the ISPF logical screen in which DB2I is invoked
- Obviate the need to pre-allocate ISPF program libraries to avoid S806 abends

# ***The Business Case for Launch Express for ISPF***

## **Launch Express and Disaster Recovery**

- Launch Express enables much faster disaster recovery since you can LOGON once a minimum set of datasets are restored
- Ordinarily, nearly all program product datasets must be restored before anyone can logon to TSO
- In a disaster recovery situation, Launch Express ensures TSO is available as soon as the ISPF and site-specific datasets are recovered – you do not have to wait for every program product to be restored

## **Launch Express facilitates maintenance and deployment**

- Optimize the search for ISPF dialog objects and achieve significant cost savings / performance improvement
- Simplifies product installations and upgrades (including new releases of z/OS itself)
- Add, change or delete ISPF applications without impacting LOGON PROCs, EXECs and/or CLISTs
- Launch Express facilitates the install, upgrade and/or full replacement of z/OS releases and program products and site developed ISPF applications
- Launch Express is fully compatible with all z/OS software installation methods such as ServerPac, CBPDO, SystemPac, msys for Setup, etc.
- Shops can begin using Launch Express immediately
- Shops can cutover to Launch Express all at once or in phases
- Launch Express facilitates both software upgrades and full system replacements
- Launch Express lets you deploy new and revised ISPF applications immediately
- Facilitate upgrades to new versions of software products and applications
- Eliminate STEPLIBs and all those datasets in your LOGON procs
- Facilitate upgrades to new versions of ISPF products and applications
- Launch Express requires no additional knowledge or training in ISPF
- Launch Express can be deployed system wide or for selected users or groups
- Simplify installation and maintenance of new and existing ISPF applications
- Launch Express simplifies maintenance and increases flexibility

## **Launch Express Launchlets**

- Launchlets are small, driver REXX execs or TSO/E CLISTs used to invoke ISPF applications
- Launchlets are easy to create
- Launchlets can be generated by the Launch Definition dialog, coded with the LXLAUNCH composite command or coded using TASKLIB and native TSO/ISPF commands like ALLOCATE, LIBDEF and ALTLIB
- Product upgrades have minimal impact. Simply change the launchlet while the ISPF menu remains unaffected
- Launchlets permit quick prototyping and immediate availability of ISPF applications
- Your entire inventory of ISPF applications are invoked as launchlets from a site-wide library allocated in every TSO LOGON PROC

# The Business Case for Launch Express for ISPF

## Launch Express Features

- Launch Express offers unique, non overlapping functions and features
- LXCONCAT is a TSO/E command processor that lets you concatenate, de-concatenate and display allocated datasets. LXCONCAT lets you concatenate / deconcatenate datasets without the need to free and reallocate datasets
- The LXCONCAT command processor lets you add new datasets to an existing allocation without having to repeat the specification of datasets previously allocated
- In comparison with freeing and completely reallocating a set of datasets – LXCONCAT is *far* easier to code
- LXLAUNCH is a TSO command processor that can invoke any ISPF application (including programs that issue ATTACH, LINK, LOAD and XCTL macros) with a single command that specifies *all* application components. LXLAUNCH is a composite function which combines the ISPF SELECT and LIBDEF services as well as TSO ALTLIB and TASKLIB commands in a single command
- LXDB2S runs as a stand-alone ISPF dialog and can be invoked by DB2 sensitive launchlets. It's a facility to display defined DB2 subsystems and enable ISPF applications to switch subsystems

## Summary: Justifying Launch Express to Management

- On the basis of improved performance – Launch Express provides faster LOGON, lower BLDL overhead, and smaller TSO working set size
- On the basis of simplified ISPF maintenance and deployment - launchlets provide a single point of change and eliminate the need to LOGOFF and LOGON to access new or changed ISPF applications
- Reliability, Availability and Serviceability (RAS) – the single, simple LOGON PROC used with Launch Express is more reliable and much easier to service
- Reduced contention – dataset ENQ's are held only when an ISPF dialog is active. There are no superfluous enqueues to interfere with library maintenance
- Better security – Launch Express promotes granular and real security for ISPF applications – as opposed to the coarse and often false sense of security associated with LOGON JCL, execs and CLISTS
- Faster disaster recovery – Launch Express helps recover from disasters more quickly since TSO can run once a minimal set of system and site-specific ISPF datasets are restored