TSI

MOVE FORWARD · TOGETHER

Product Updates

TPFUG 2019, Denver

Agenda

- zTPFGI Releases
- JavaNow
- Multiple System Support
- DFDL with Rest API
- zQDC Automation
- Dump Viewer – Source, TPFDF, DETAC
- Color Customization
- Other Enhancements

zTPFGI Releases

zTPFGI continues to deliver 2 releases per year, focused on priority customer requests, fixes, and the growing feature set of z/TPF®

- This presentation covers features from three different zTPFGI releases:
  - 1.3.8.0
    - Being rolled out at customers now
  - 1.3.9.0
    - Testing for rollout soon
  - 1.4.0.0
    - Scheduled for release this summer
- Unless otherwise marked, features belong to 1.3.8.0
- Other features will be marked with a * for 1.3.9.0 and a ** for 1.4.0.0

JavaNow for zTPFGI**

JavaNow

What is JavaNow?

- z/TPF shops' workflow will include a Java-specific IDE for Java resources (e.g. NetBeans, IntelliJ, Eclipse), and…
- **JavaNow** in zTPFGI is an additional option that allows Java for z/TPF to be coded, compiled, loaded and debugged directly in zTPFGI

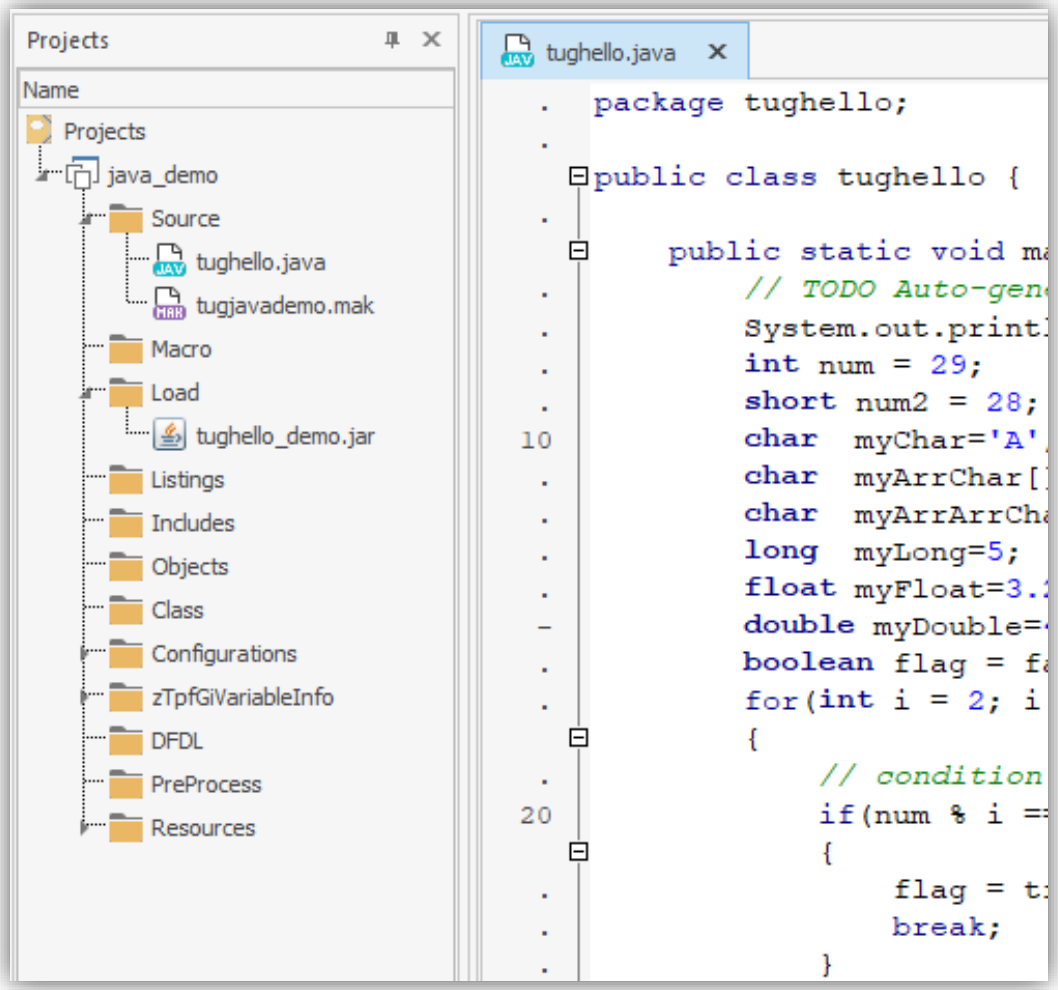
JavaNow Benefits

- Saves **time** for developers who are primarily interested in traditional z/TPF development but who need to interact with Java for development, testing, debugging, and maintenance
- Saves **training costs** for developers who know zTPFGI and also need to work with Java

JavaNow Edit/Build/Load

JavaNow

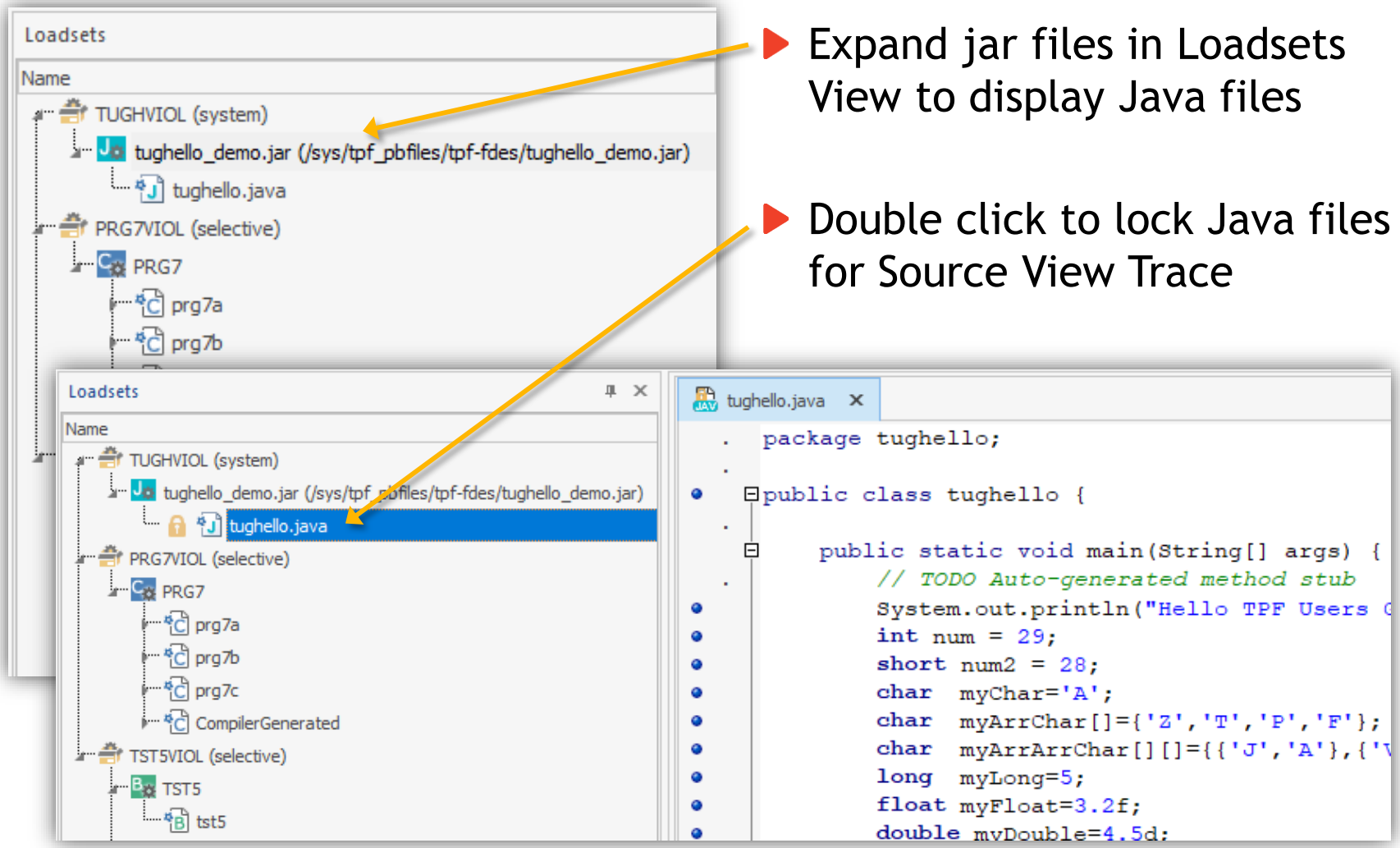
- Support for editing Java files (syntax highlighting)
- Support for Java files in projects
- Support for compile/build of Java
- Support for Loading Jar files



TSI

JavaNow Loadsets View

JavaNow

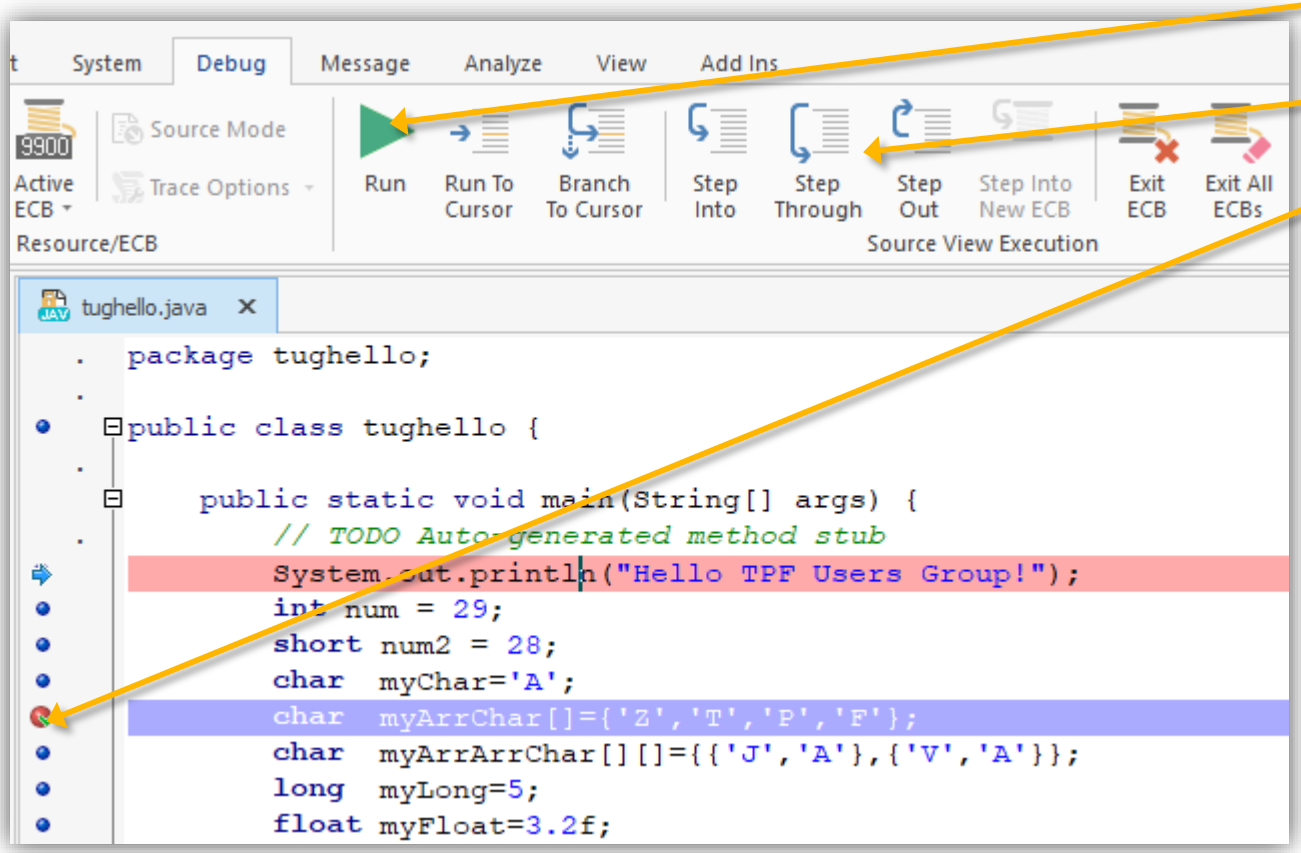


- Expand jar files in Loadsets View to display Java files
- Double click to lock Java files for Source View Trace

JavaNow Source View Trace Execution

JavaNow

Source View Trace Java in same execution and control buttons as Assembler/C/C++/SabreTalk

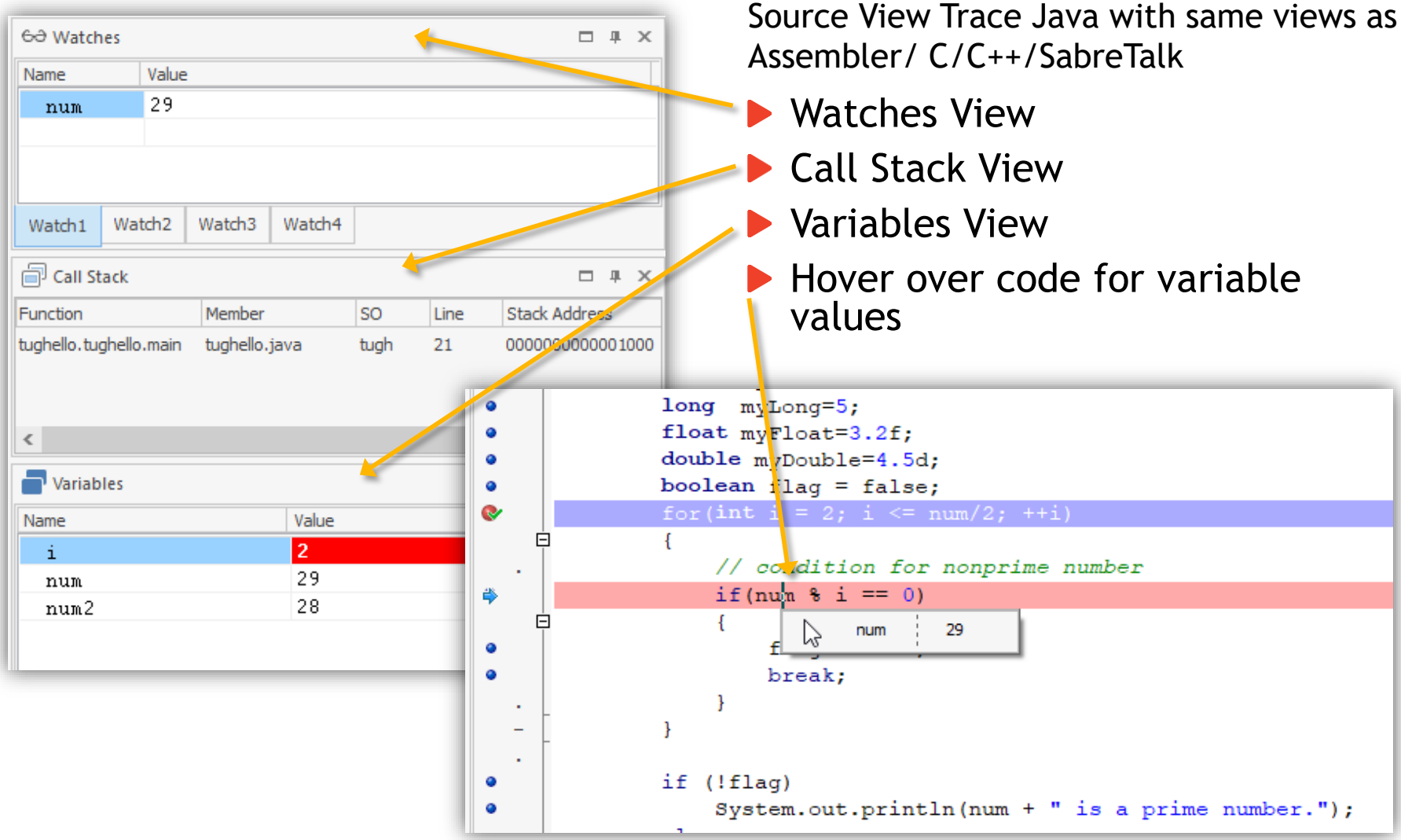


- Run
- Step
- Set Breakpoints
- Etc.

TSI

MOVE FORWARD · TOGETHER

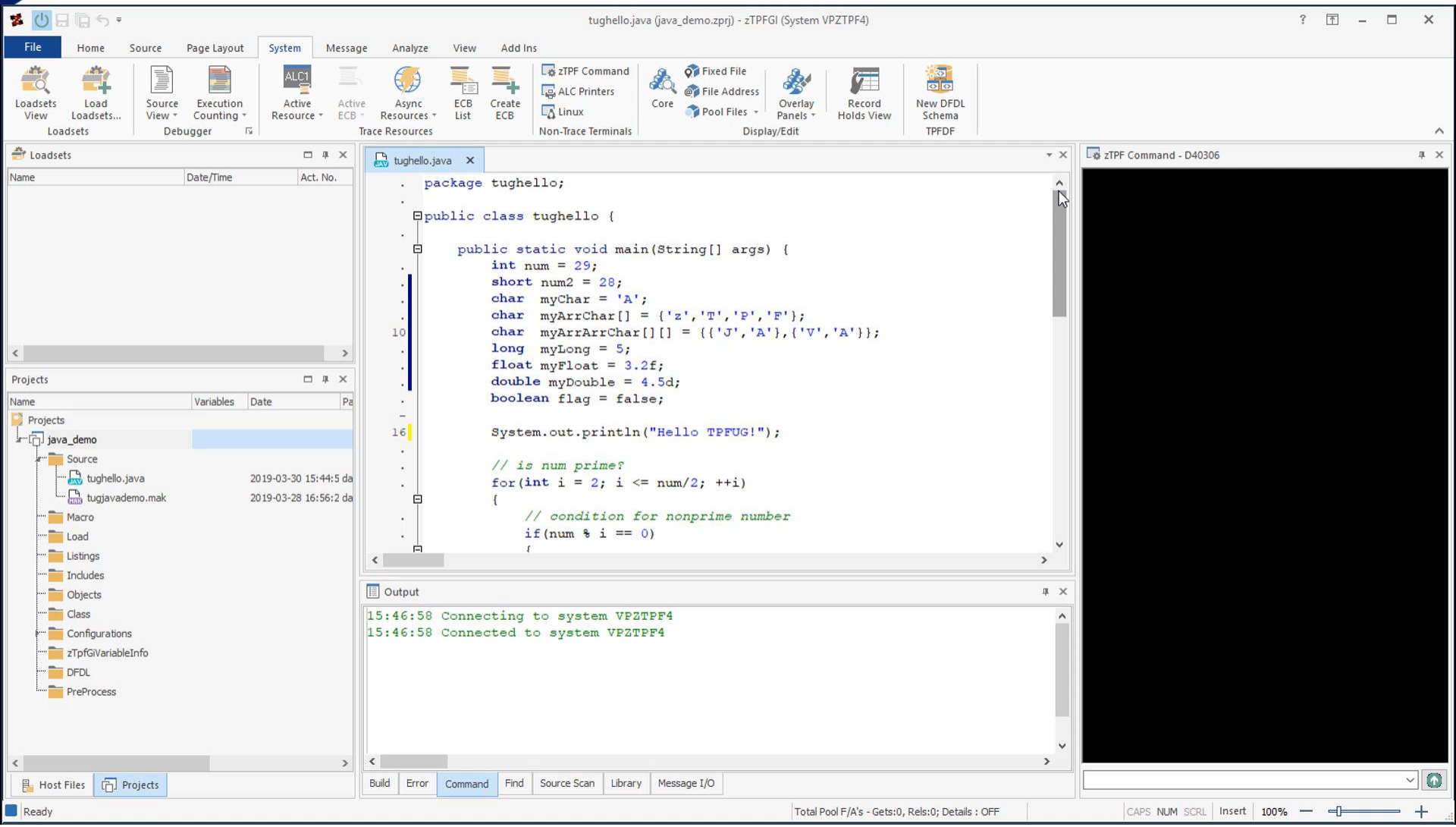
JavaNow Source View Trace Variables



Source View Trace Java with same views as Assembler/ C/C++/SabreTalk

- Watches View
- Call Stack View
- Variables View
- Hover over code for variable values

JavaNow Demo



Multiple System Support*

What is it?

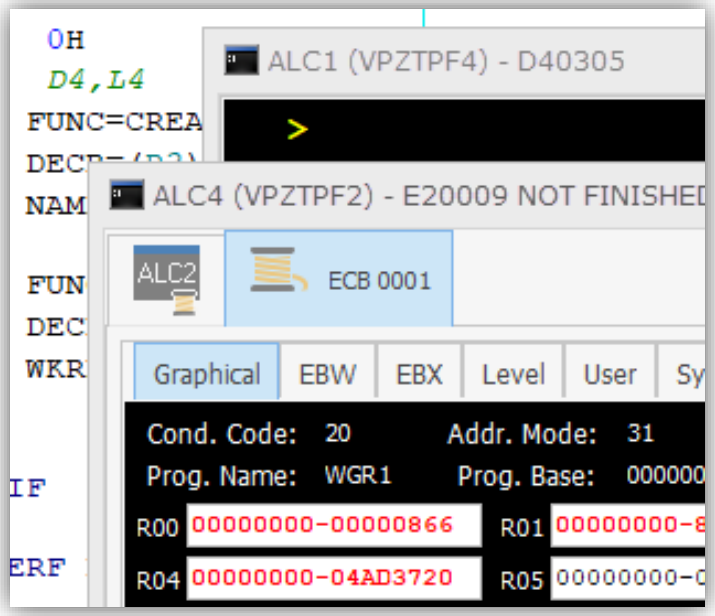
- zTPFGI now supports connections to multiple system/CPU configurations from a single instance of zTPFGI. (Eliminates the need for multiple instances of zTPFGI.)

Benefits

- Much quicker to set up for debugging multiple systems and loosely coupled systems

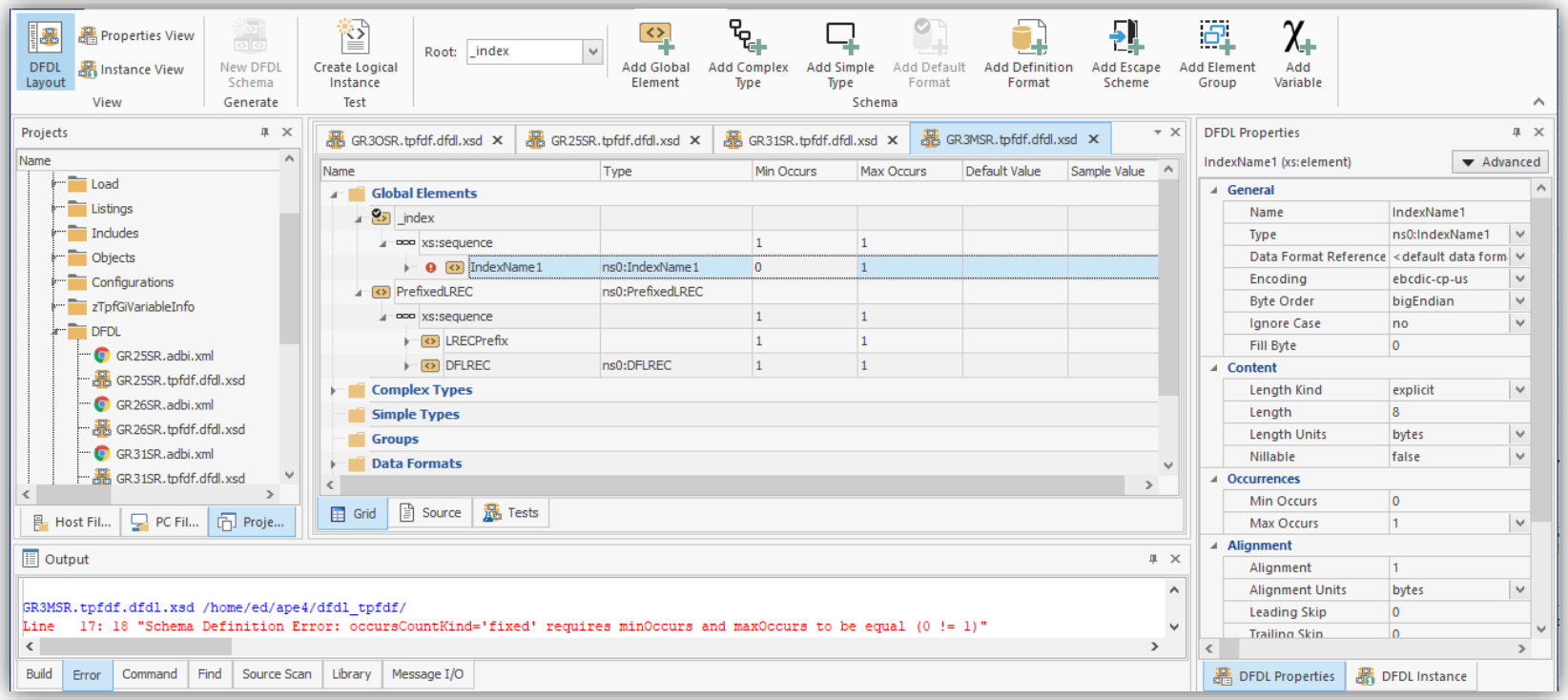
How it works

- Connecting to multiple systems is one step, the same as connecting to a single system
- Multiple system configurations are set up by zTPFGI administrator
- The system being targeted is identified on each zTPFGI window/view



DFDL Support

zTPFGI's existing DFDL support includes editing, testing and loading of DFDL schemas



TPF Rest Services Tool**

Need

- z/TPF systems need to provide services to be consumed through Rest API

What is TPF Rest Services Tool?

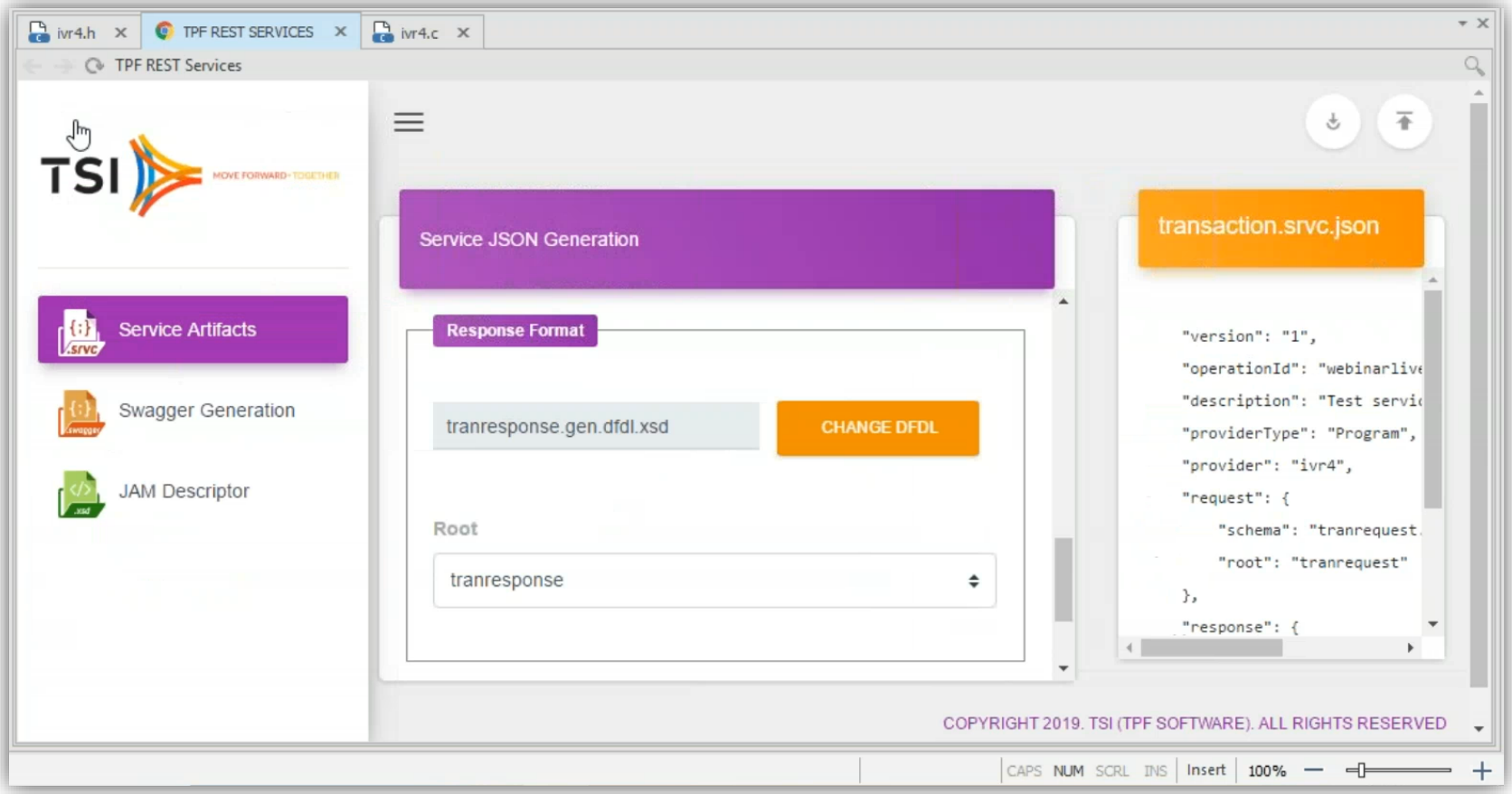
- The TPF Rest Services Tool is integrated with zTPFGI
- It allows users to build the files (service JSON files, swagger schemas and JAM descriptors) needed to make API communication within or outside the TPF system

Benefits

- All that is needed to create a service is available in a tool in zTPFGI
- Uses standard files. Users can edit and reuse product files created by other tools.
- Simple to use with immediate feedback, saving time

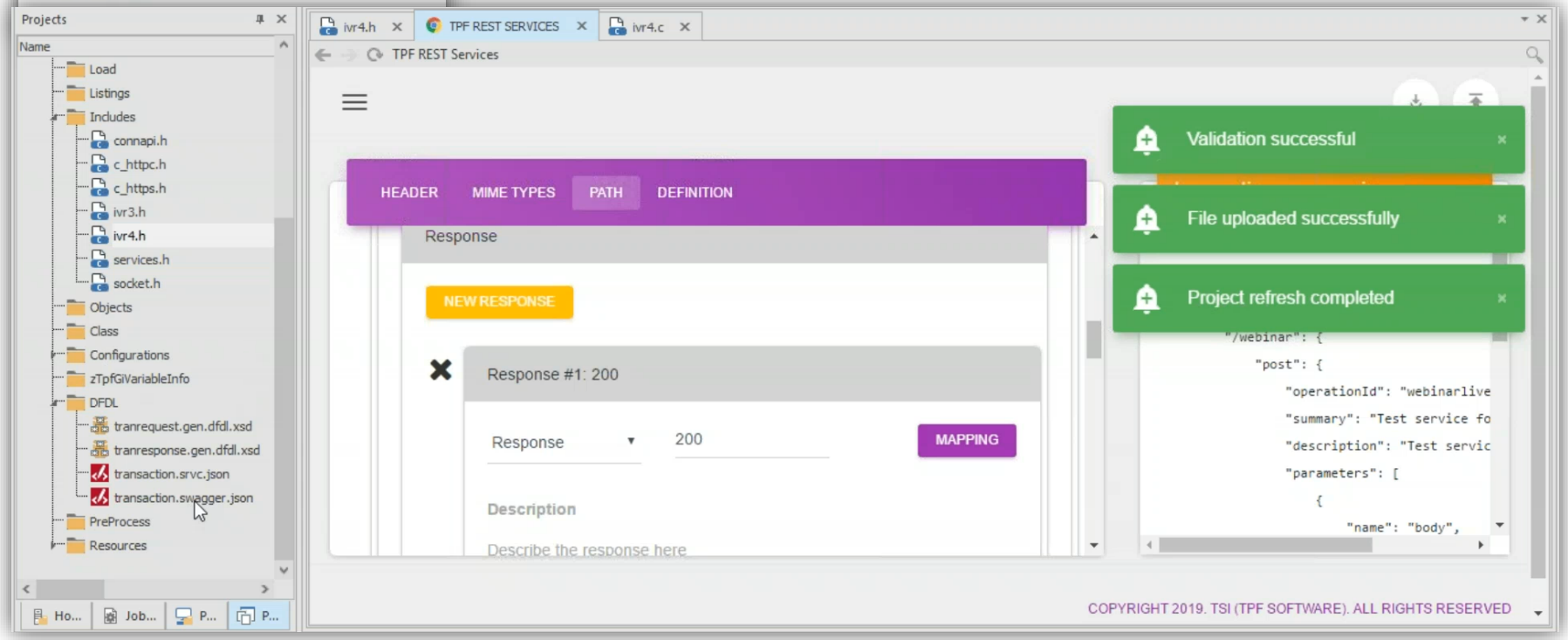
TPF Rest Services – Services Artifact

- Allows the user to generate the service descriptor (Service JSON) file that describes the implementation for z/TPF services
- Generates based on DFDL schemas that are easy to select in the zTPFGI environment



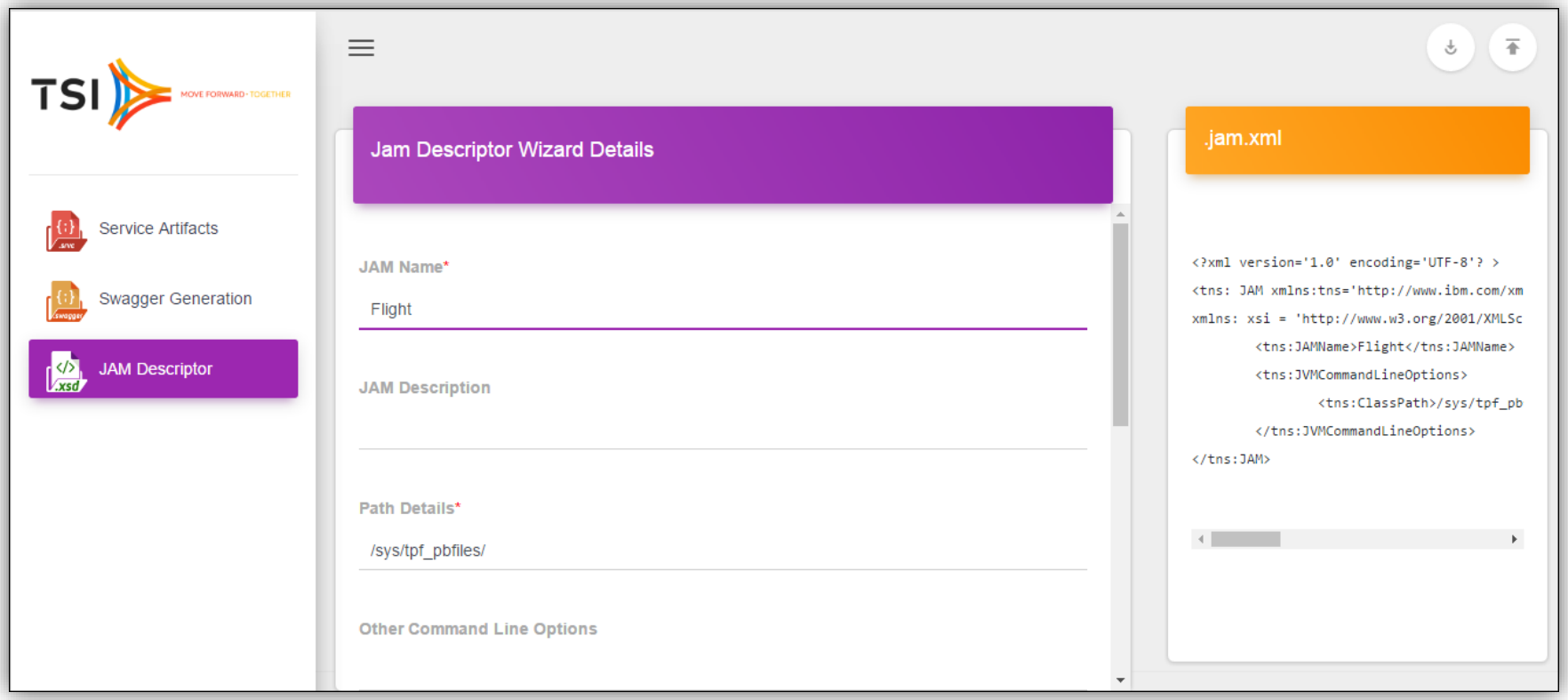
TPF Rest Services – Swagger

- Allows the user to generate OpenAPI descriptor (Swagger) that describes a set of Rest APIs
- The swagger JSON is generated based on a selected service JSON file



TPF Rest Services – JAM Descriptor

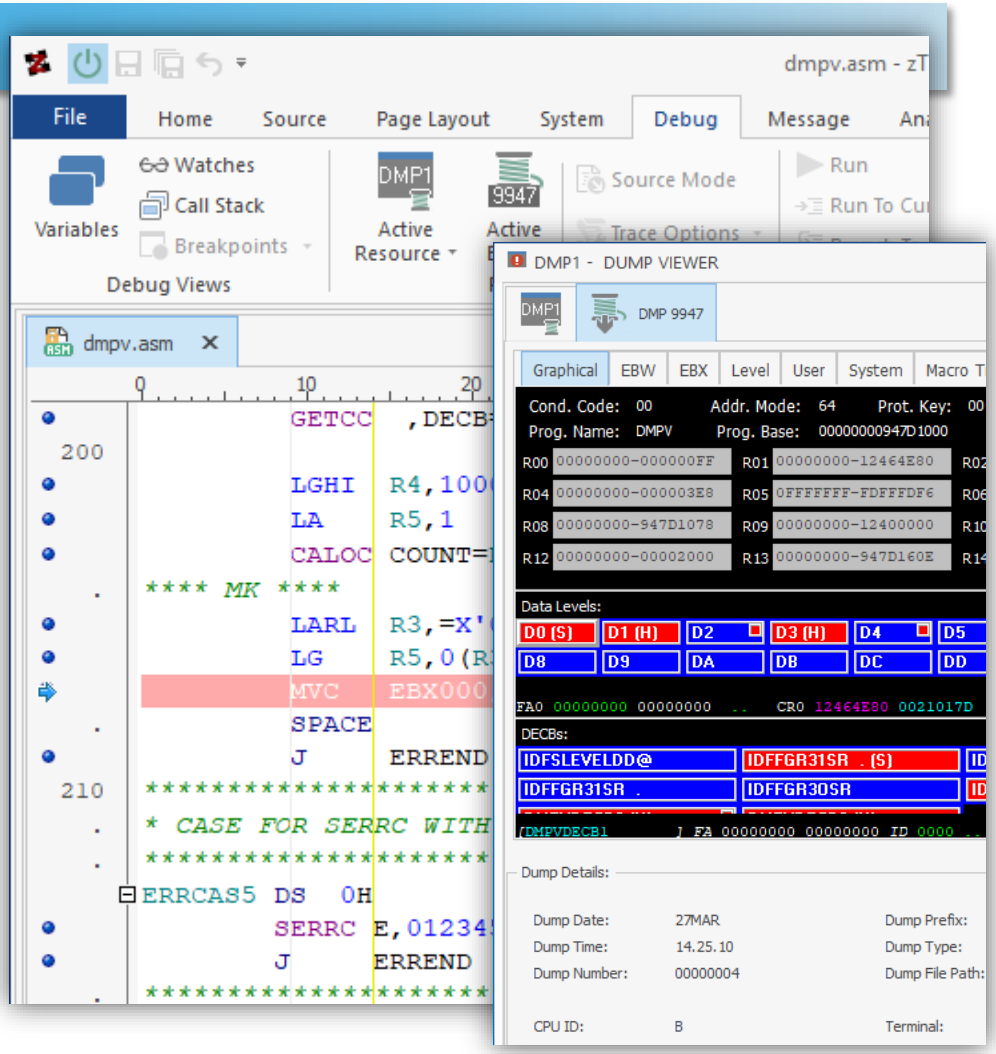
- Allows the user to generate JAM descriptor that contains the necessary information to define JVMs for Java applications on z/TPF
- The JAM.XML file is generated based on the service JSON and can be loaded to the development server
- Then the JAM can be uploaded to the z/TPF system to make API communication



Dump Viewer*

View Dump Source

- Option to open read-only source file, if available, showing execution line where dump occurred



TSI

MOVE FORWARD · TOGETHER

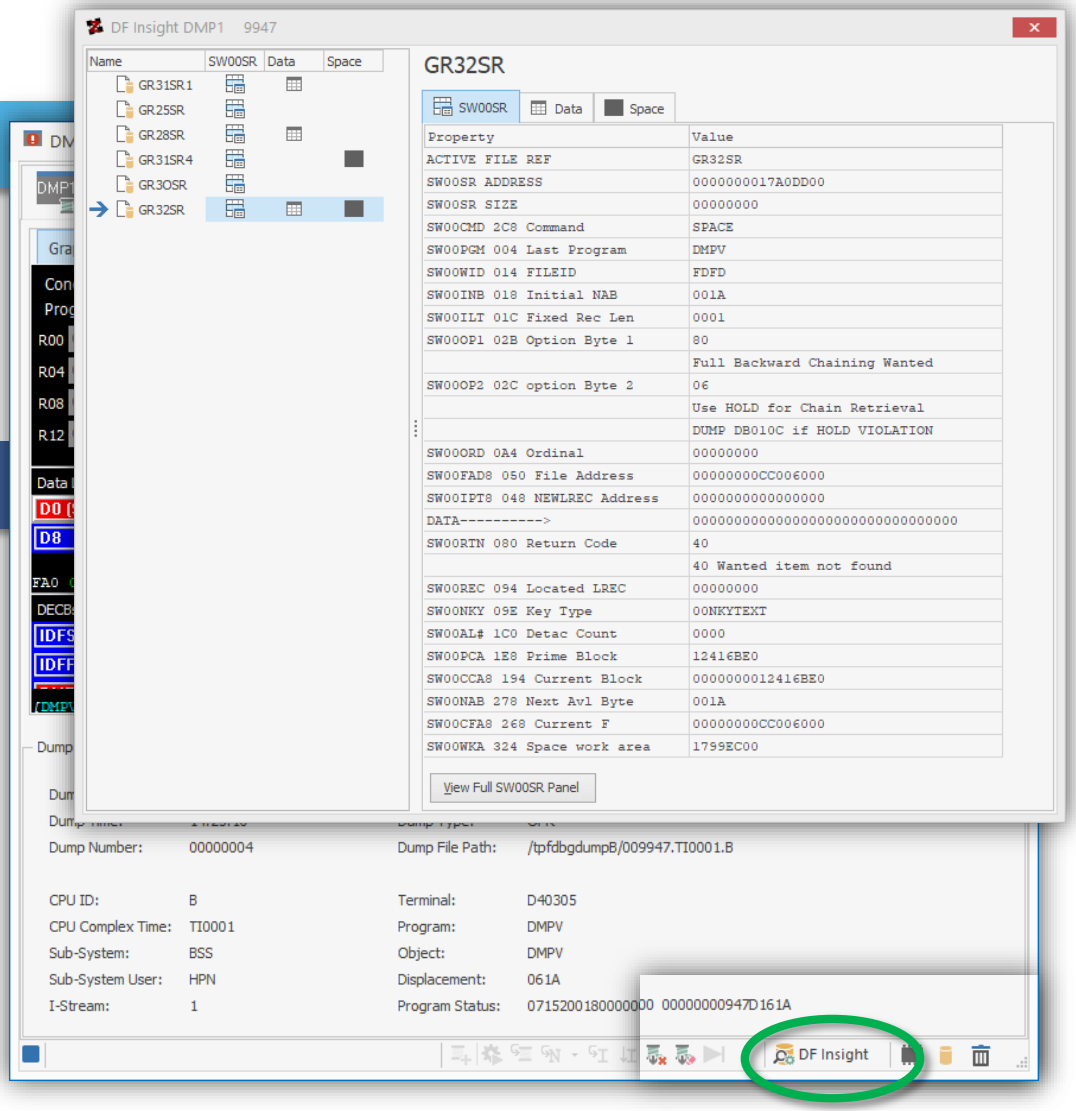
Dump Viewer

View Dump Source

- Option to open read-only source file, if available, showing execution line where dump occurred.

View TPFDF in DF Insight

- Click on DF Insight in the Dump Viewer to view TPFDF just as you would when tracing an ECB.



Dump Viewer

View Dump Source

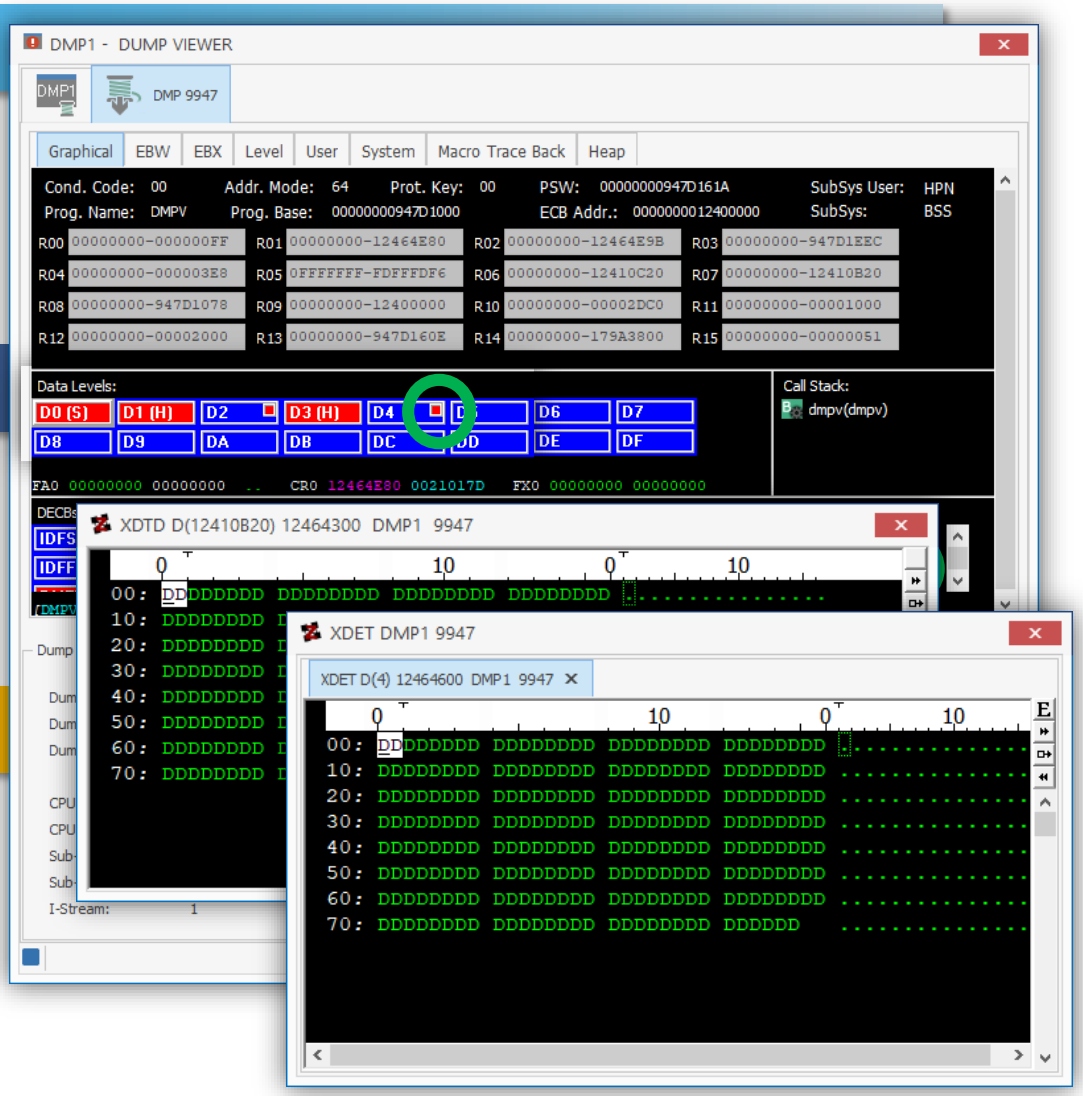
- Option to open read-only source file, if available, showing execution line where dump occurred.

View TPFDF in DF Insight

- Click on DF Insight in the Dump Viewer to view TPFDF just as you would when tracing an ECB.

View Detached Core Blocks

- Double-click a detached core block in a data level or DECB to open and view it in a Core Block window.



zQDC Automation

What is it?

- Updates the collected data from the z/TPF machine in the customer DB in the customer defined structure through Jenkins (previously a manual step was required via a Web UI to select options & upload data to DB)

Benefits

- No manual intervention required – faster, with no manual errors

How it works

- Database table is customizable based on the customer need
- Data update event triggering is customizable. Can be based on…
  - The new data from z/QDC
  - Time initiated
  - Manual trigger

Color Enhancements**

What is it?

- zTPFGI now supports user customization of editor colors, along with an option for an overall dark color scheme

Benefits

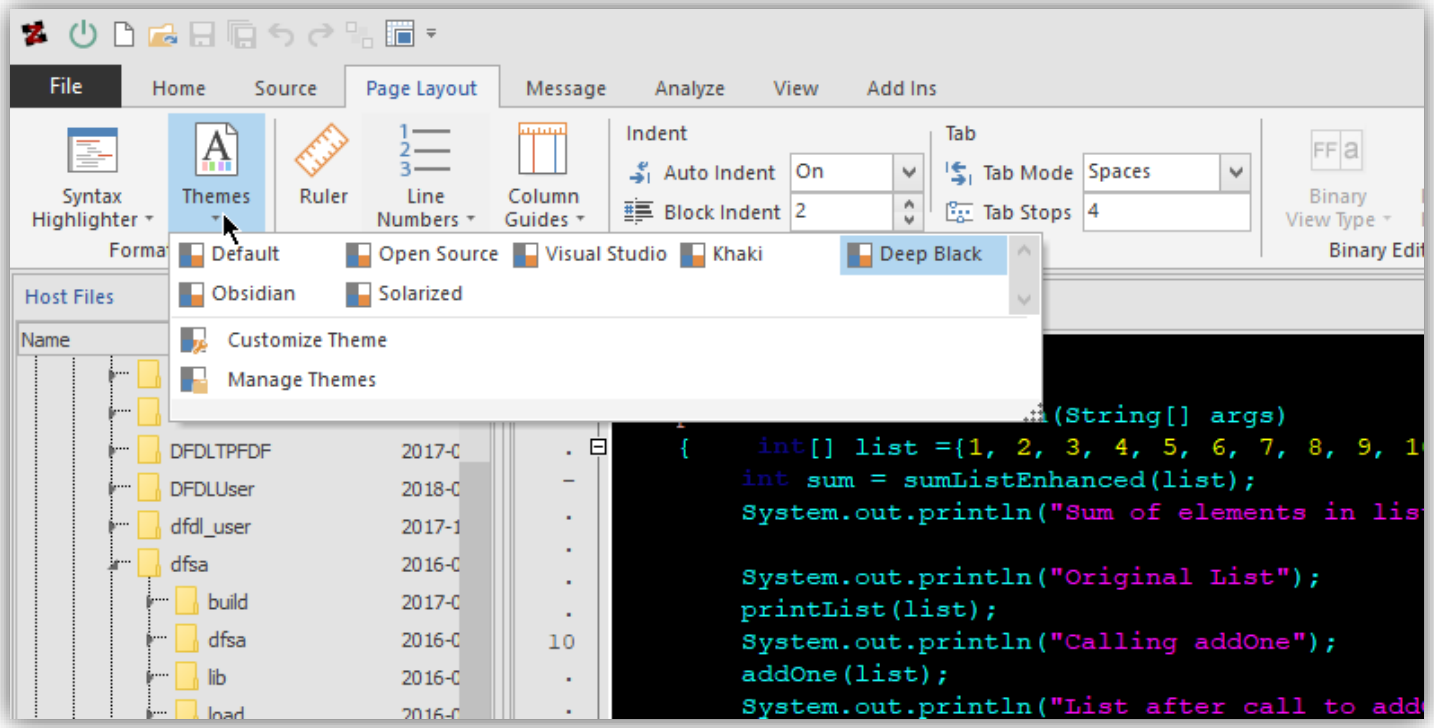
- Greater user productivity
- Greater accommodation of different visual abilities
- Greater user satisfaction

How it works

- A greater range of standard editor themes now offers an array of non-white editor backgrounds
- In addition, editor themes may be duplicated, customized, and even shared
- An additional overall color scheme offers a darker choice for the ribbon and other menus and windows

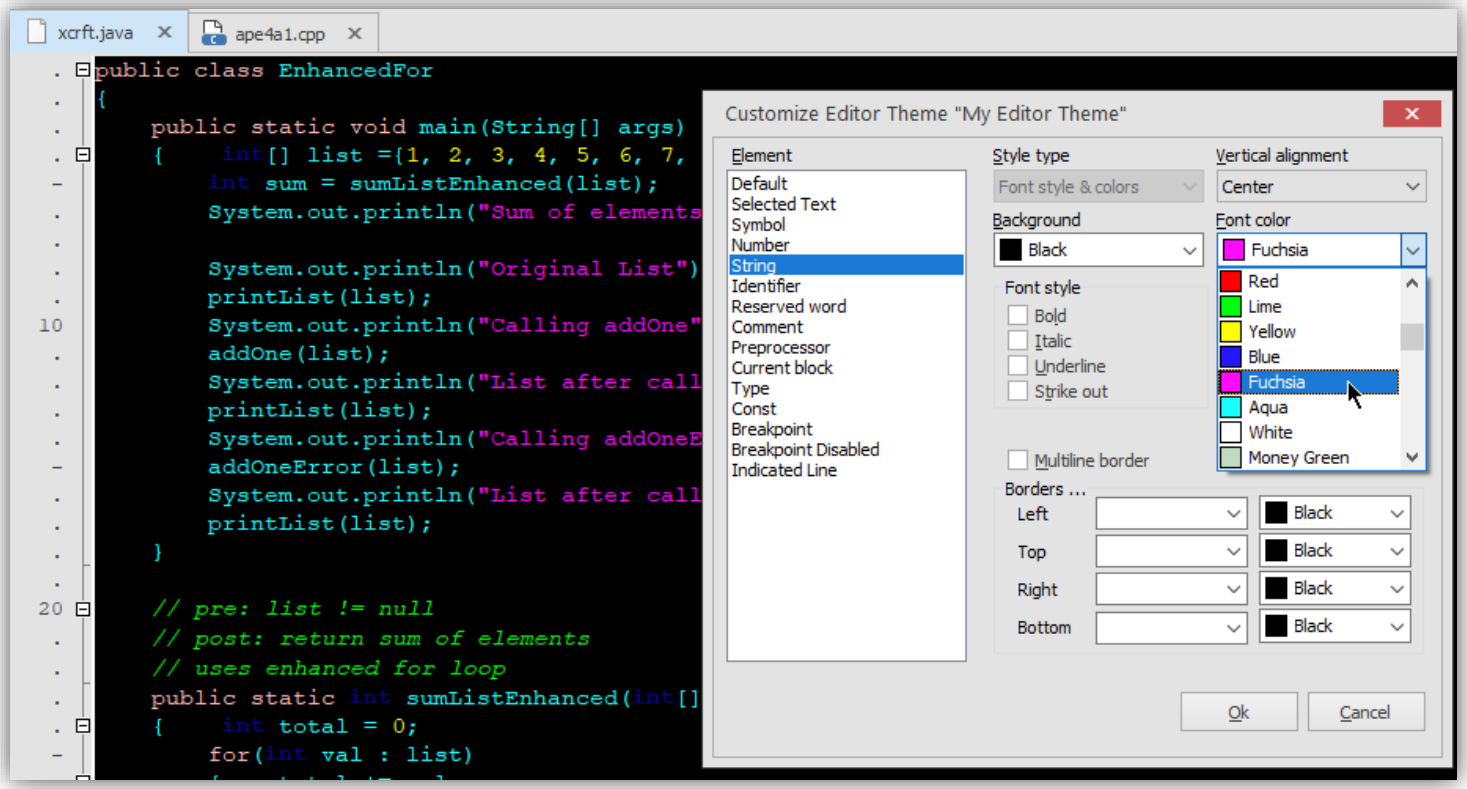
Color Enhancements – Standard Editor Themes

- Users can hover over an expanded gallery of standard editor themes to instantly preview different syntax highlighting choices
- A single click makes the choice for that file type
- A preference is also available to change the default editor theme, affecting every file type without a different choice



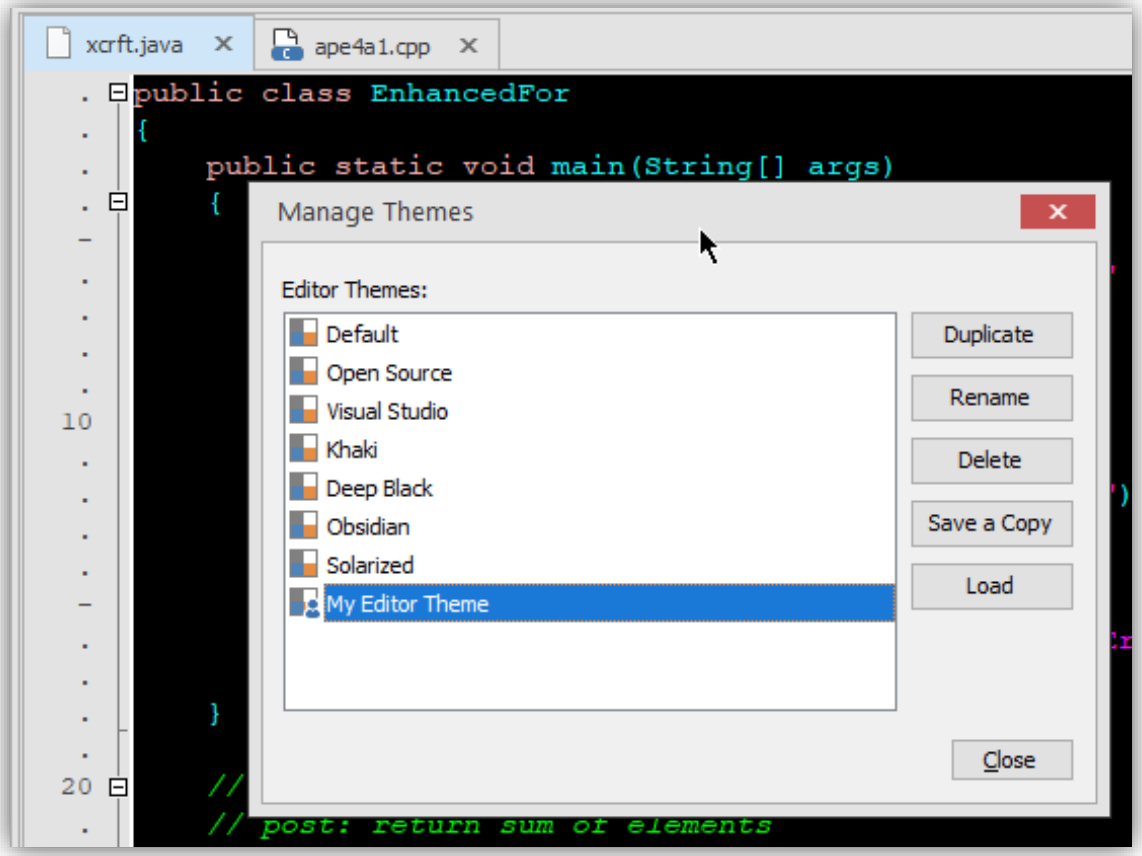
Color Enhancements – Custom Editor Themes

- Users can customize duplicate standard editor themes and customize every syntax element
- Even breakpoint and other colors can be customized
- Color and style choices are previewed instantly in the editor



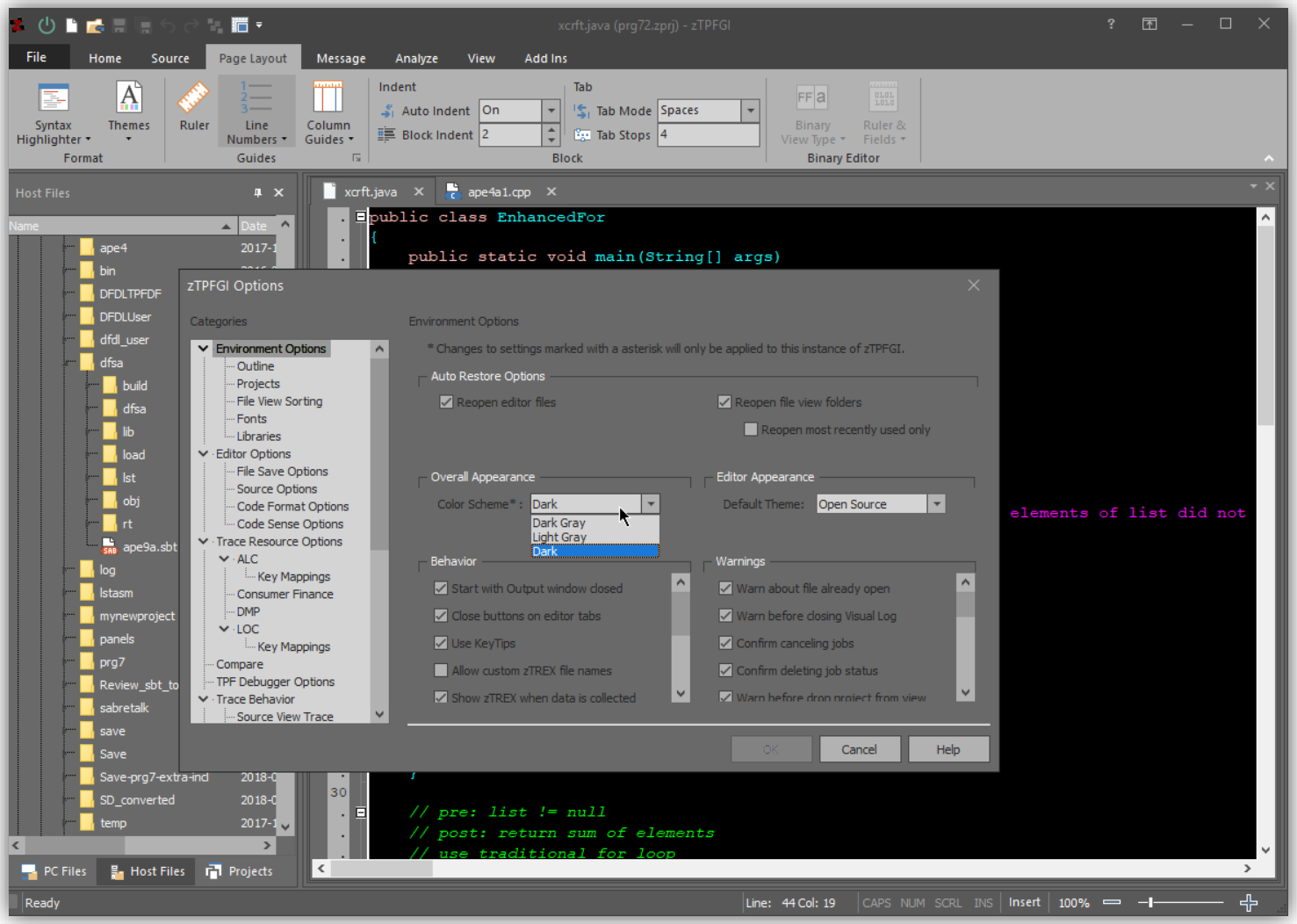
Color Enhancements – Editor Theme Management

- Users can manage their custom editor themes
- They can also save a copy for co-workers, who can load it into their zTPFGI UI



Color Enhancements – Dark Color Scheme

- The new Dark color scheme affects the non-editor areas of zTPFGI



Other Enhancements

- ALC
  - Message Capture: facility to Capture and replay ALC inputs
  - Create ECB - Support to attach input message block to D0
  - Execute large entries
- DF Insight
  - Highlight current LREC in overlay view
  - Display complete DF file including all forward chains
- Dump Viewer
  - Support multiple sort options - Sort by date/time, program name
- Trace
  - Async Trace support for C functions
  - Performance improvements in Async
  - Modify debugger to catch overwrites of system area of ECB
  - Shortcut key for ECB
- Usability
  - Option to open a file in desired mode (text, binary)*
  - Font size support for Host Files, PC Files, Projects, & Other Views*
  - Tooltips for view tabs*
  - Personalize system names for private labs*
  - Ability to copy paste variable visible in variable/watch list for C program*

Questions?

TSI

MOVE FORWARD · TOGETHER

Thank you