

Component-Based Simulation Development

Brad Spearing
Ternion Corporation



Ternion History

Ternion was founded to satisfy the growing requirement for simulation

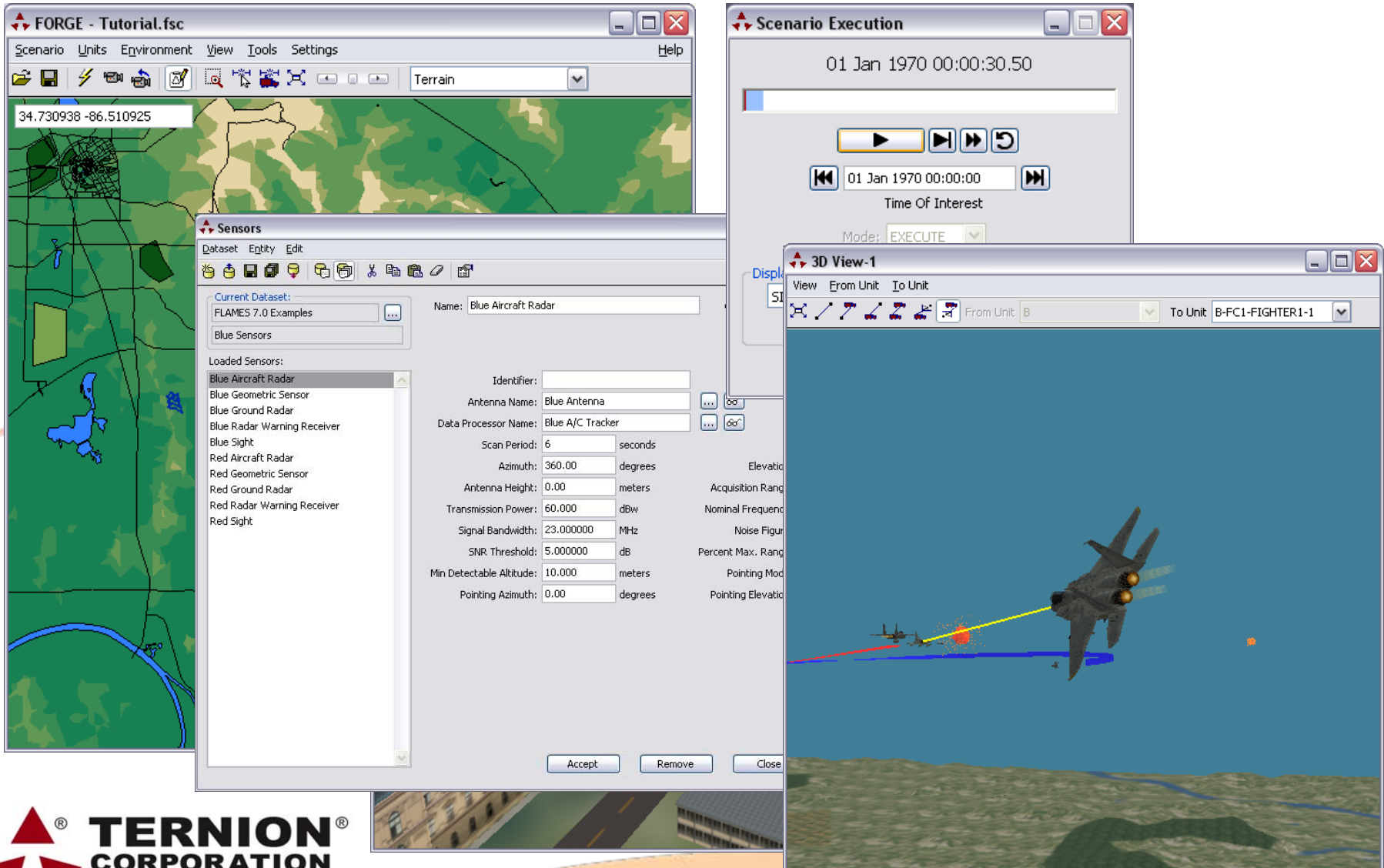
- Founded in April of 1989 in Huntsville, Alabama
- Single product line is the FLexible Analysis Modeling and Exercise System (FLAMES®)
- Ternion also helps customers develop FLAMES-based simulations

Some of Ternion's Customers

Ternion's products and services support simulations that are used around the world

- United States Air Force, Army, Navy, and Marine Corps and NATO
- Militaries of Canada, Finland, France, Greece, Israel, Italy, Japan, Germany, Republic of China (Taiwan), Republic of Korea, Spain, Sweden, Turkey, the United Arab Emirate, and the United Kingdom
- Numerous U.S. and foreign commercial organizations

Simple FLAMES-based Simulation



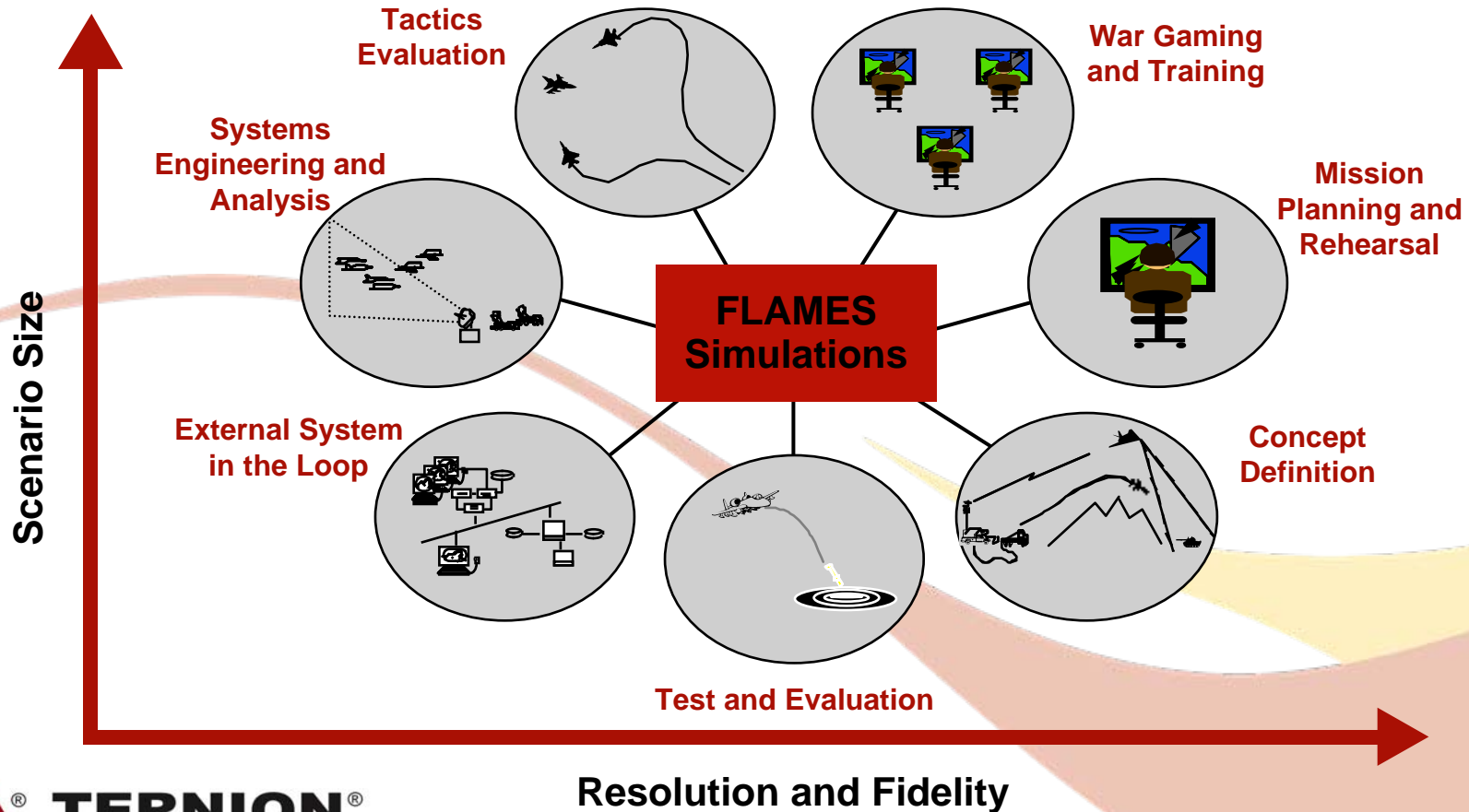
The screenshot displays the FLAMES simulation environment with several key components:

- FORGE - Tutorial.fsc:** The main simulation window showing a 2D terrain map with a radar footprint. The coordinates 34.730938 -86.510925 are visible in the top left.
- Sensors:** A configuration window for the "Blue Aircraft Radar".

Parameter	Value	Unit
Name	Blue Aircraft Radar	
Identifier		
Antenna Name	Blue Antenna	
Data Processor Name	Blue A/C Tracker	
Scan Period	6	seconds
Azimuth	360.00	degrees
Antenna Height	0.00	meters
Transmission Power	60.000	dBw
Signal Bandwidth	23.000000	MHz
SNR Threshold	5.000000	dB
Min Detectable Altitude	10.000	meters
Pointing Azimuth	0.00	degrees
- Scenario Execution:** A control panel showing the simulation time as 01 Jan 1970 00:00:30.50. It includes play, stop, and fast-forward buttons, and a "Time Of Interest" field set to 01 Jan 1970 00:00:00. The mode is set to EXECUTE.
- 3D View-1:** A 3D perspective view of a B-FC1-FIGHTER1-1 jet flying over a terrain. A yellow radar beam is shown originating from the jet and illuminating a target on the ground.

Types of FLAMES-based simulations

Simulations have been developed using FLAMES to support nearly every requirement

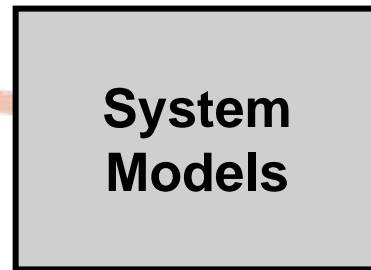


Component-Based Development

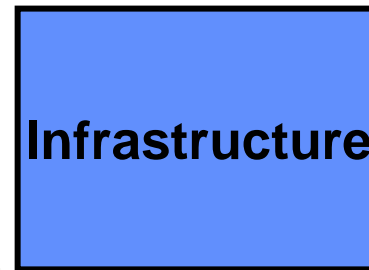


Typical Simulation Architecture

Simulation software can be divided into two categories



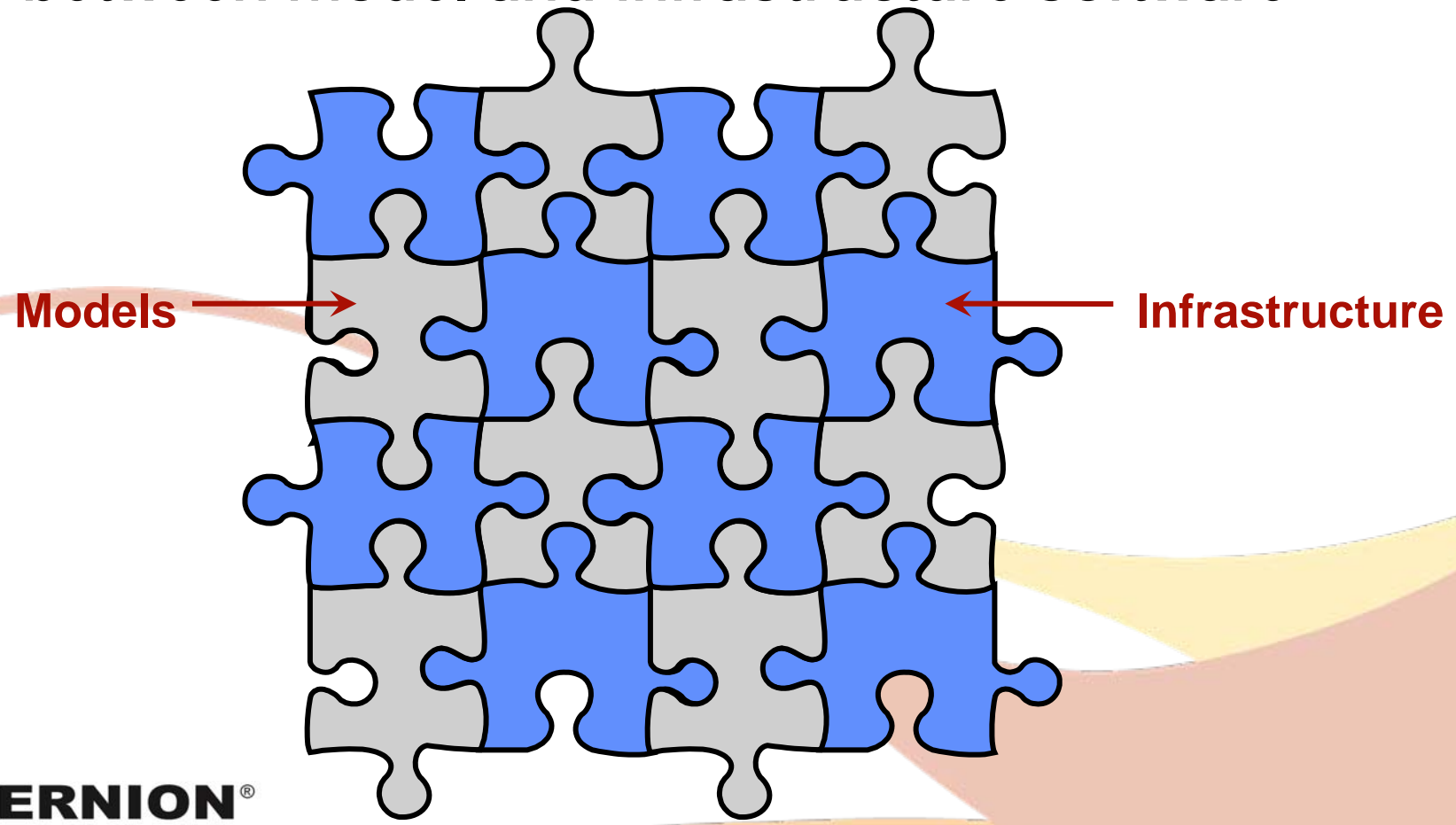
**Humans, equipment, and
the environment
(simulation specific)**



**Simulation management
and support software
(necessary for all
simulations)**

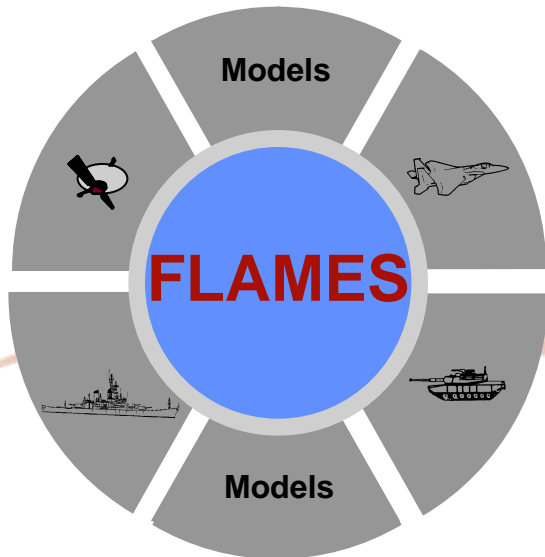
Typical Simulation Architecture

Most simulations make no clear distinction between model and infrastructure software



The FLAMES Approach

FLAMES is a framework for composable, object-oriented, component-based simulations

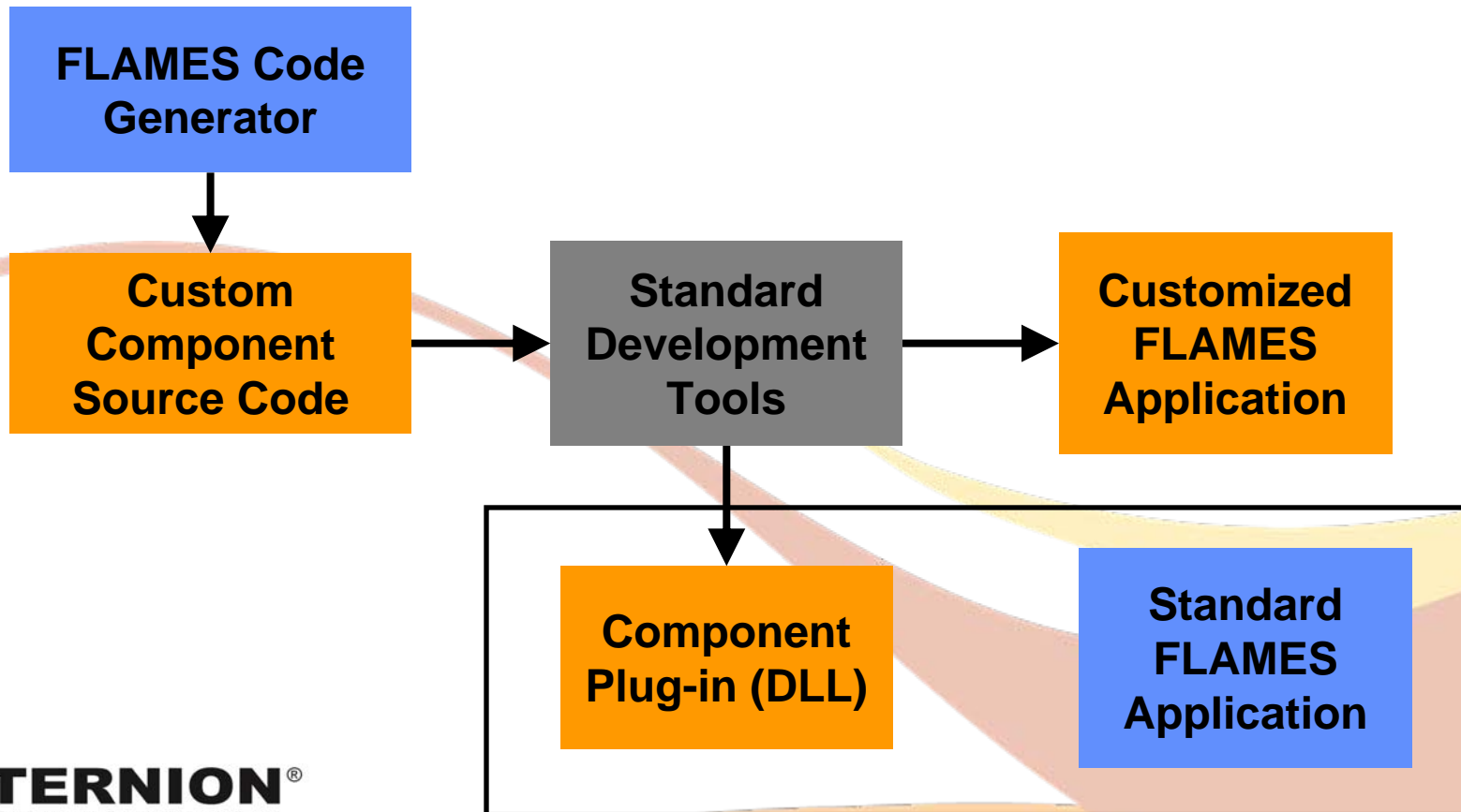


**FLAMES-Based
Simulation**

- FLAMES provides modern, feature-rich infrastructure and model interface software
- Real-world system models (components) are completely external to and independent of FLAMES
- Components “plug” into FLAMES and are automatically recognized by FLAMES
- New components can be developed quickly and inexpensively

Component Development Process

New FLAMES components can be developed and integrated easily



FLAMES Components

Many types of components can be developed for FLAMES-based simulations

- Physical Device (Equipment) Models
- Human Cognition Models
- Communications/Message Models
- Weapon Effects Models
- Atmosphere and Weather Models
- Management Services
- 2D and 3D Custom Visualization
- Interfaces to external live and virtual systems

Engineering and Analysis Simulation



Engineering and Analysis Simulation

FLAMES readily supports non-interactive, engineering design and analysis simulations

1

Configure the Scenario

- System Concepts
- Design Parameters
- Rules of Engagement
- Tactics and Plans

2

Execute the Scenario



3

Analyze the Results

- Automated Parametric Trade Studies
- Monte Carlo Analysis
- Proof of Concept
- System Demonstrations

Using Scenario Variables

Almost any input parameter can be specified using a Scenario Variable (Enhanced Analysis option)

Sensor Parameters as Constants

RANGE = 300

AZIMUTH = 90

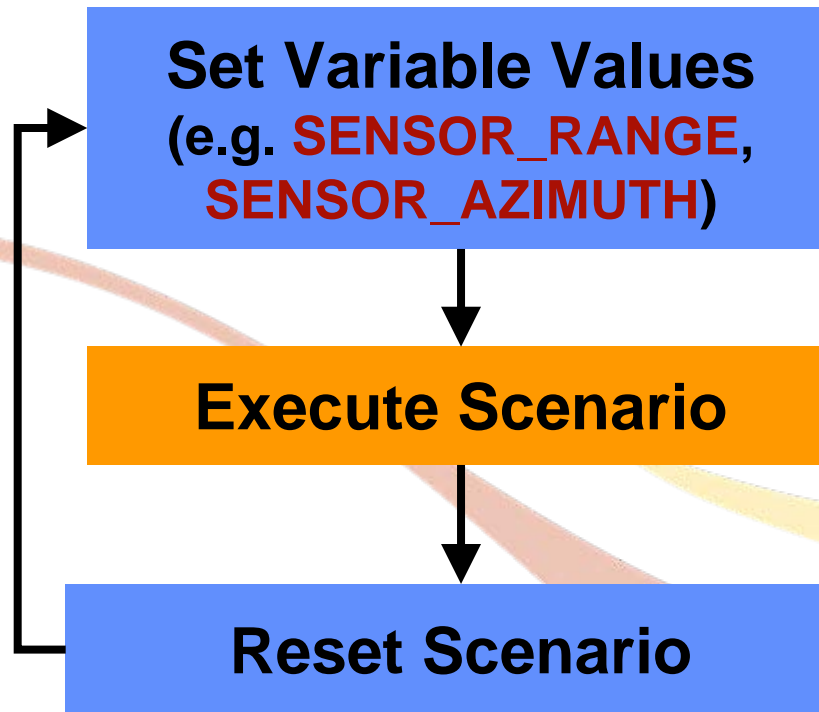
Sensor Parameters as Scenario Variables

RANGE = &SENSOR_RANGE

AZIMUTH = &SENSOR_AZIMUTH

Scenario Multiple Execution

A scenario can be executed multiple times with different scenario variable values



FLAMES and ModelCenter

ModelCenter can specify variable values, control FLAMES execution, and analyze scenario results

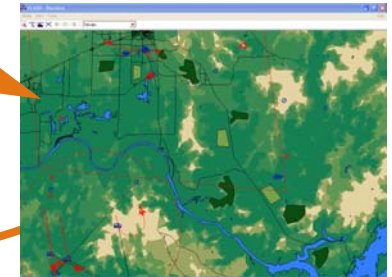
ModelCenter
Automate · Integrate · Optimize



Specify experiment inputs

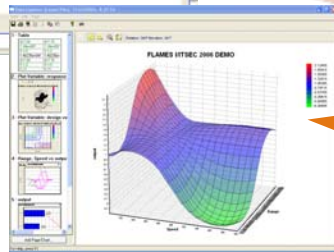
Scenario Variable Values

FLAMES
Simulation Framework



Execute the scenario

Scenario Results



Plot and analyze the results



The ModelCenter interface development was sponsored by the US Air Force

More Information

More information on Ternion and FLAMES is available in the following places

www.ternion.com

Mark Davies, Technical Sales
(flames_sales@ternion.com)

Questions?

