U.S. Army Research, Development and Engineering Command

Executable Scenario Definition Using Datalog to Describe Simulation Capabilities

TECHNOLOGY DRIVEN. WARFIGHTER FOCUSED.

US Army Research Laboratory (ARL)

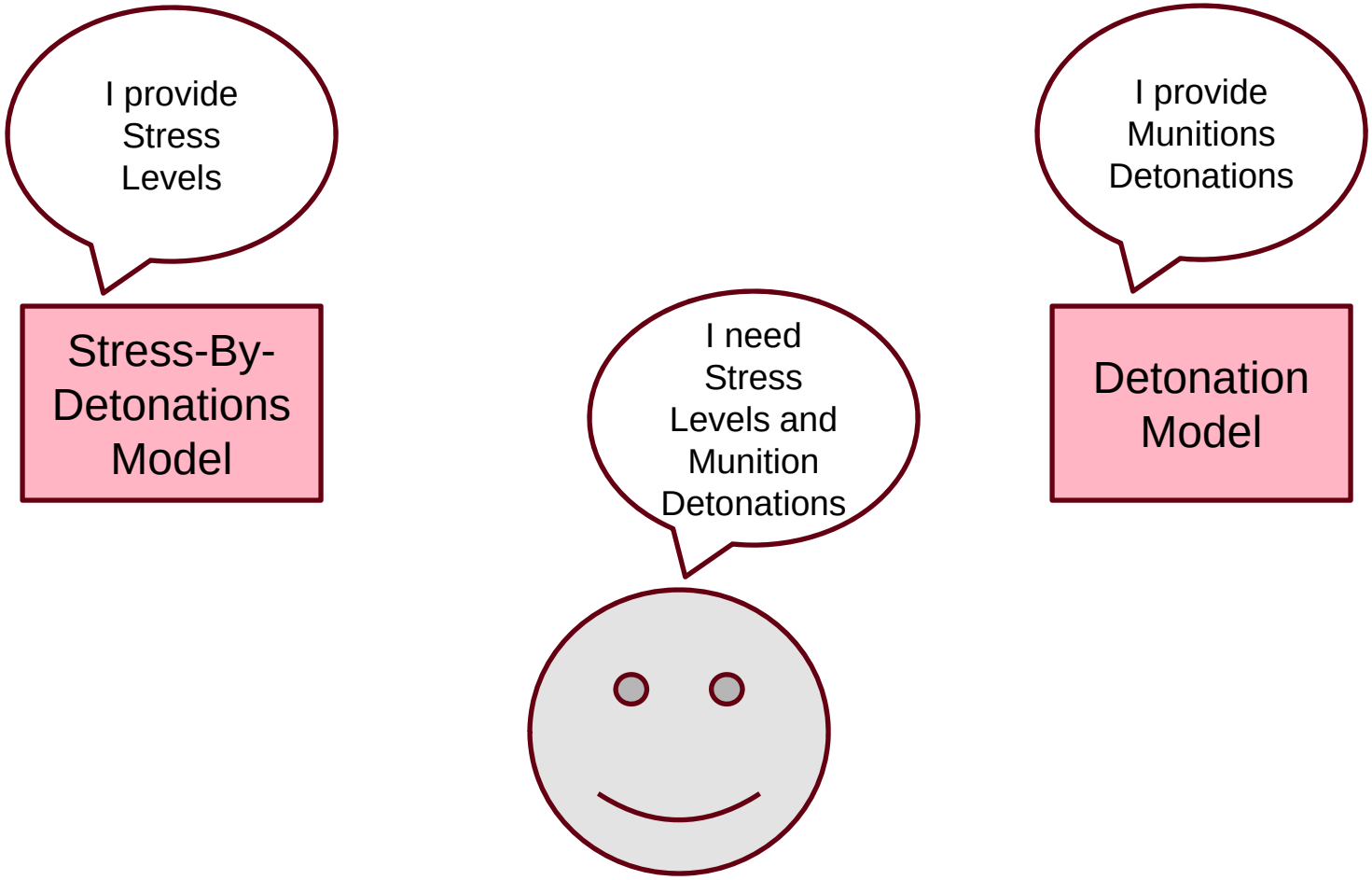
Human Research and Engineering Directorate (HRED)
Simulation & Training Technology Center (STTC)

- The Next Generation Architectures (NGA) for Modeling & Simulation (M&S) research project refines and demonstrates advances in computer science that support the development of M&S architectures required for future training, experimentation and acquisition decisions
- This presentation discusses an initial effort under this project area to identify scenario primitives that can link analytical data elements to simulation execution objects in a way that supports automated model and simulation selection based on the content of the scenario

The Situation

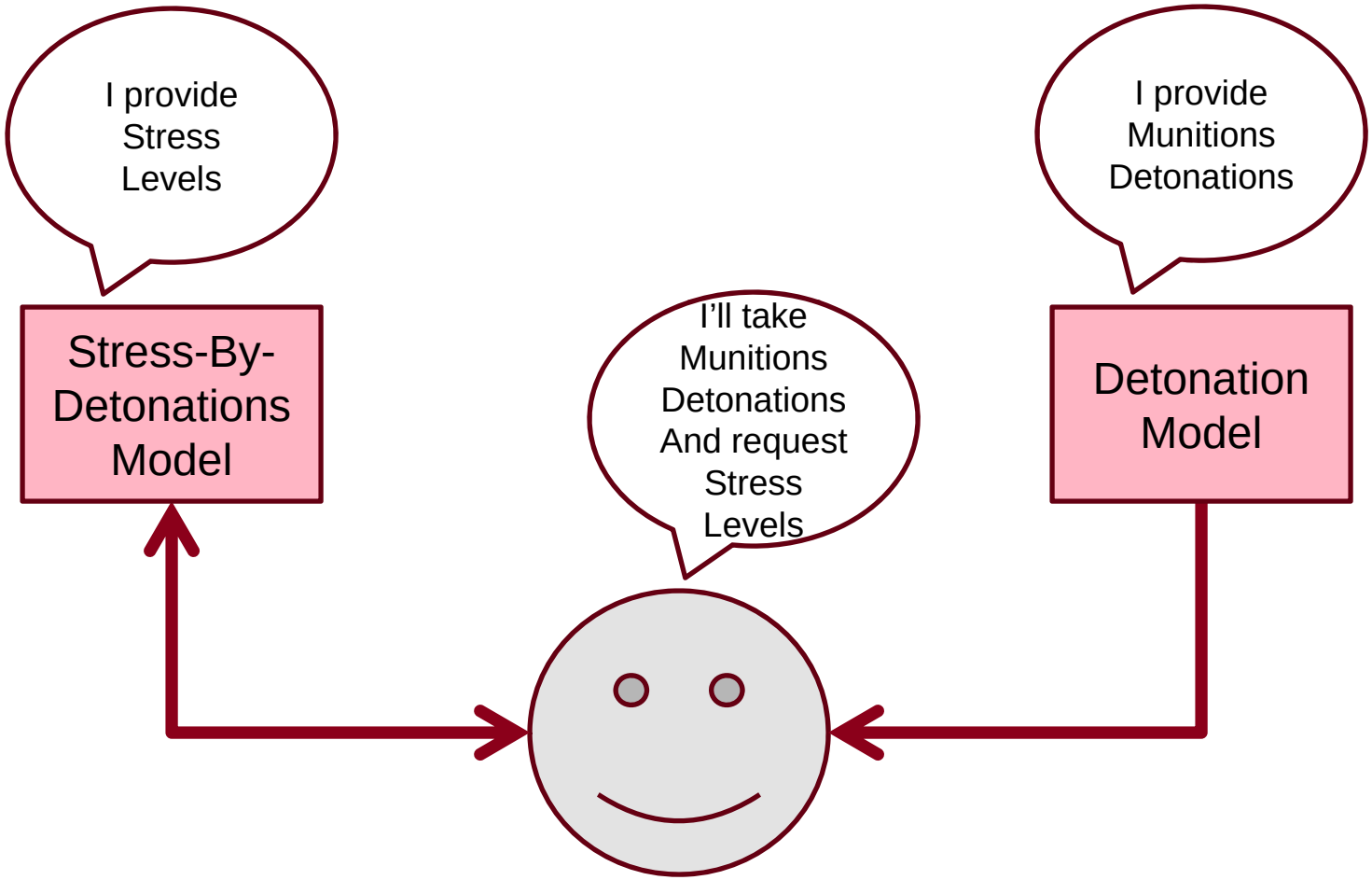
- Simulation models with fixed interfaces
- Explicit integration
- Failing events due to models and simulation not meeting expectations

What we have



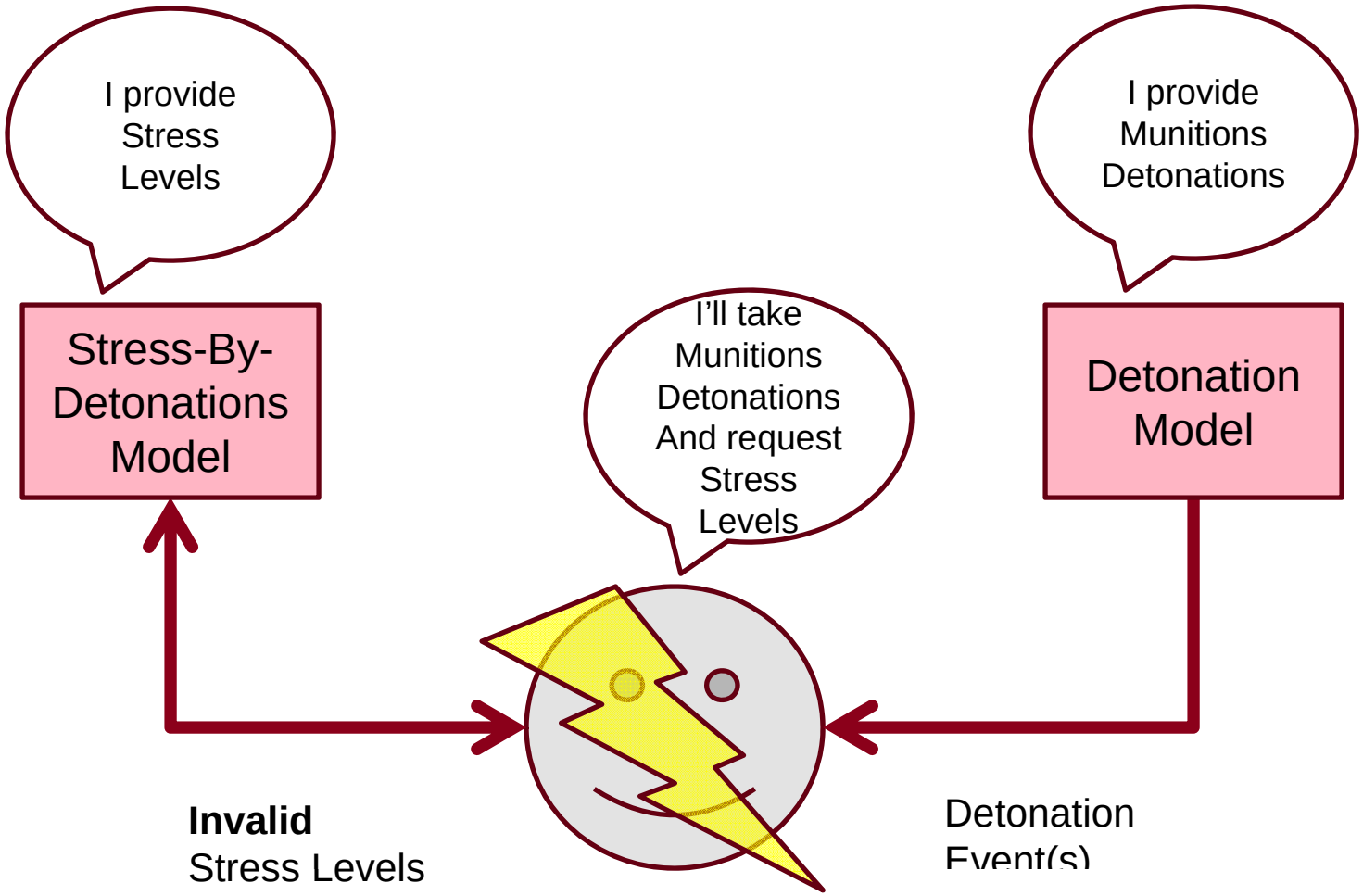
Simulation models with fixed interfaces

What we have



Explicit integration

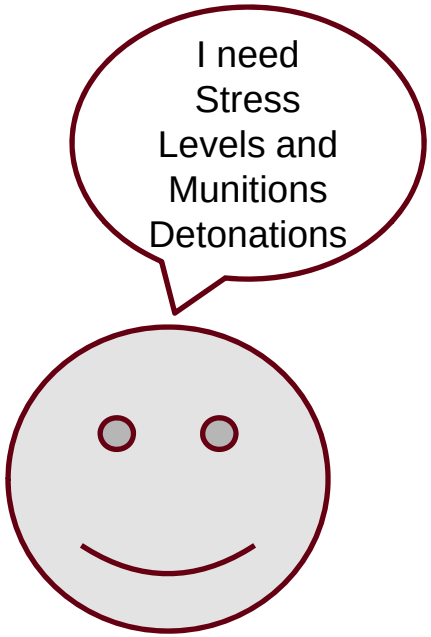
What we have


I provide Stress Levels
I provide Munitions Detonations
Stress-By-Detonations Model
I'll take Munitions Detonations And request Stress Levels
Detonation Model
Invalid
Stress Levels
Detonation
Event(s)
Failing events

What we want

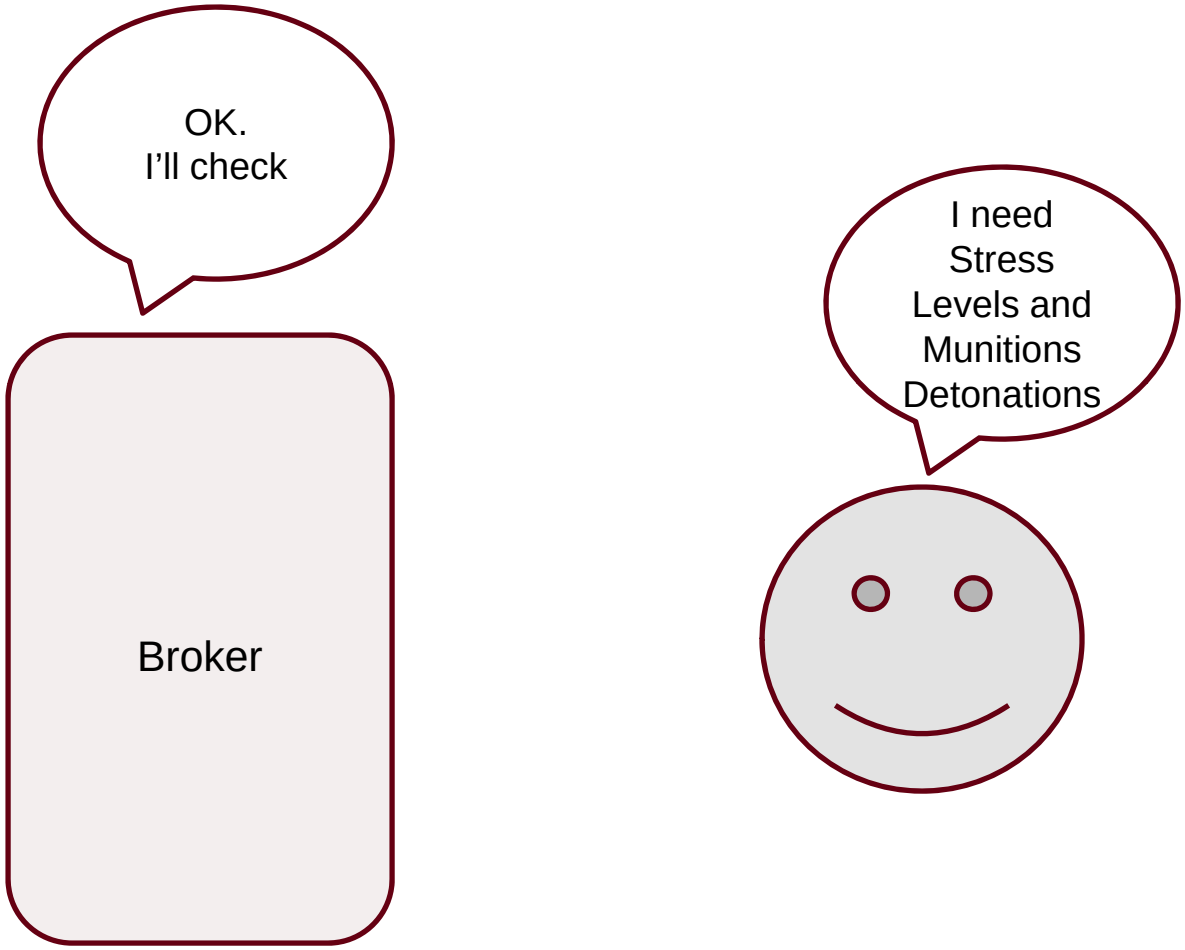
- Fluid, discoverable simulation models
- Model sandboxing
- Integration via composition

What we want



Fluid, discoverable simulation models

What we want



Fluid, discoverable simulation models

What we want

Anyone provide Stress and Detonations ?

Stress-By-Detonations Model

Broker

Detonation Model

Fluid, discoverable simulation models

What we want

I provide Stress Levels

Stress-By-Detonations Model

I provide Munitions Detonations

Detonation Model

Broker

Fluid, discoverable simulation models

What we want

Do these Work for You?

Stress-By-Detonations Model

Broker

Detonation Model

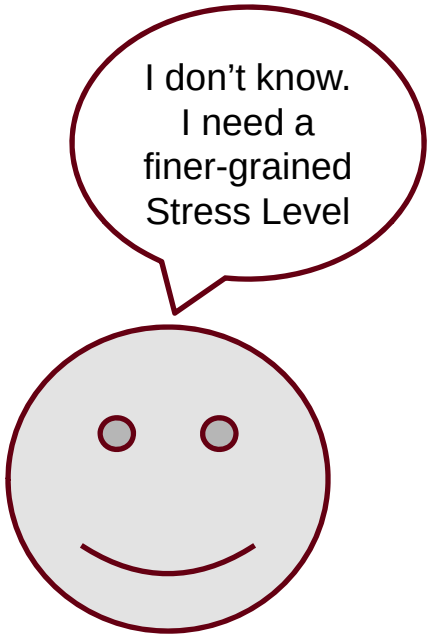
Fluid, discoverable simulation models

What we want

Stress-By-Detonations Model

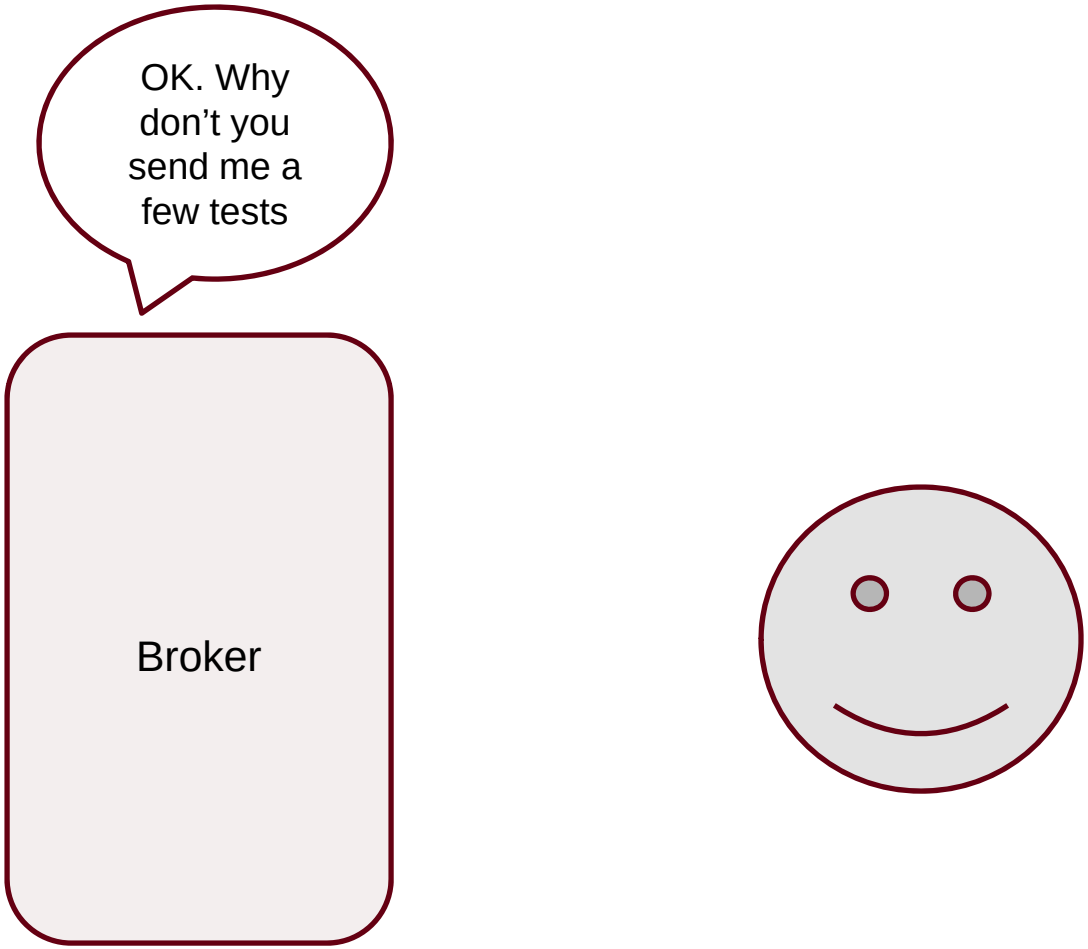
Broker

Detonation Model



Fluid, discoverable simulation models

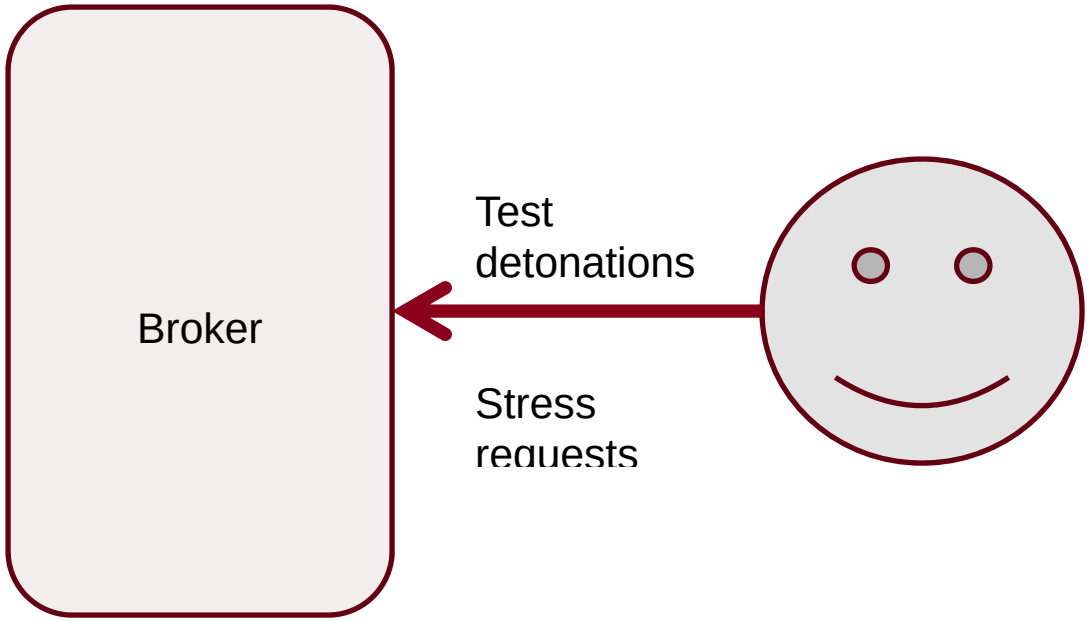
What we want



Model sandboxing

What we want

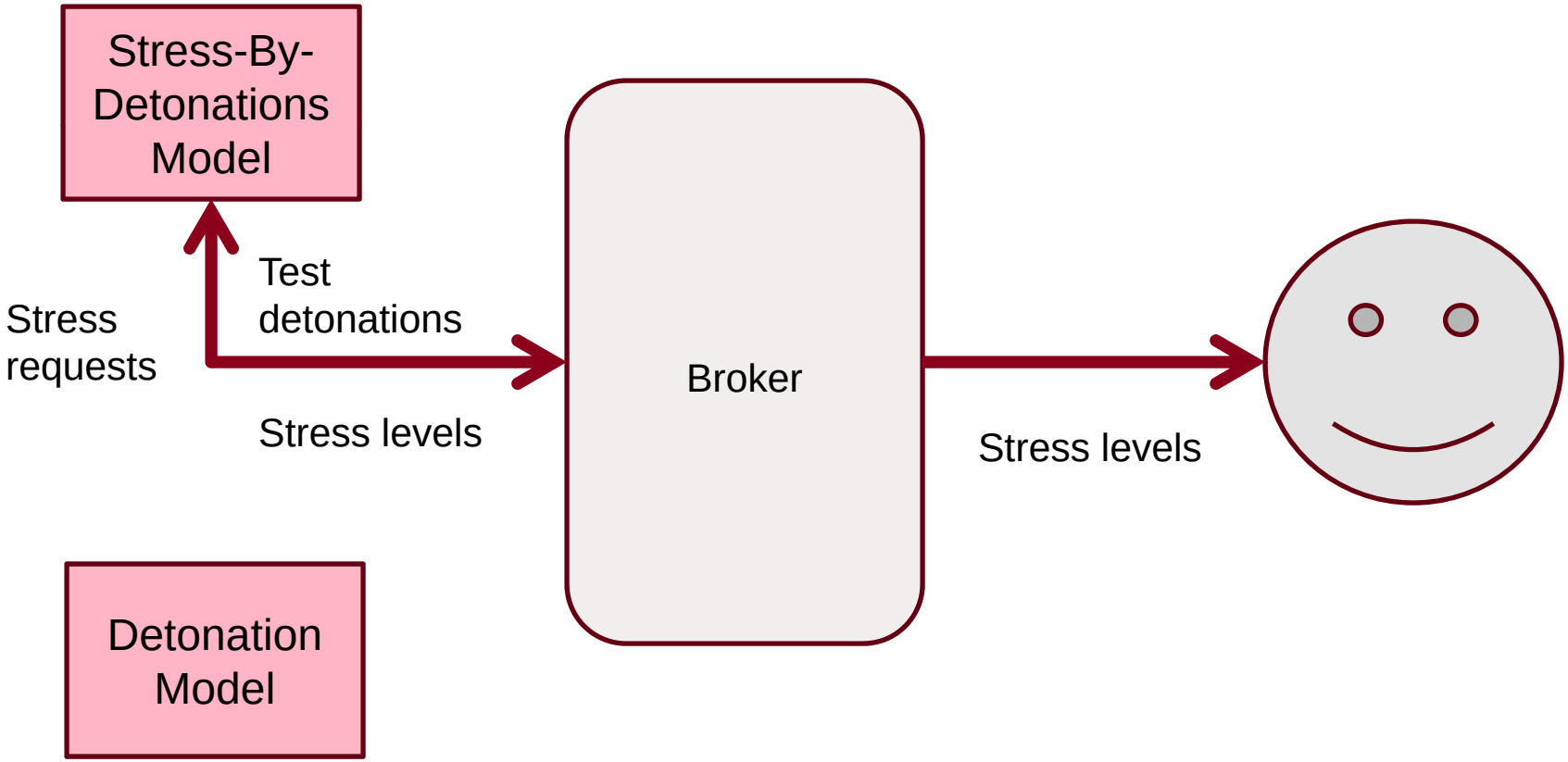
Stress-By-Detonations Model



Detonation Model

Model sandboxing

What we want



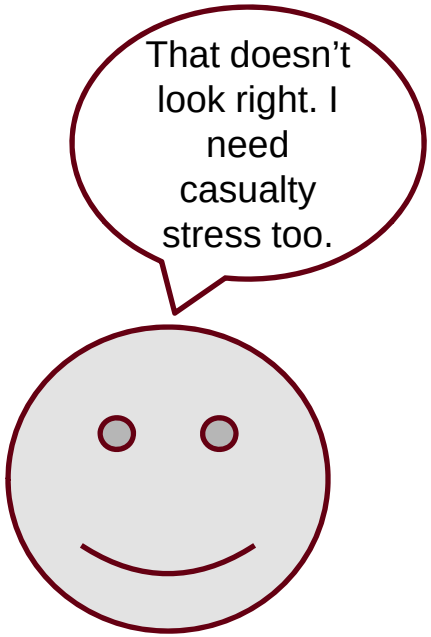
Model sandboxing

What we want

Stress-By-Detonations Model

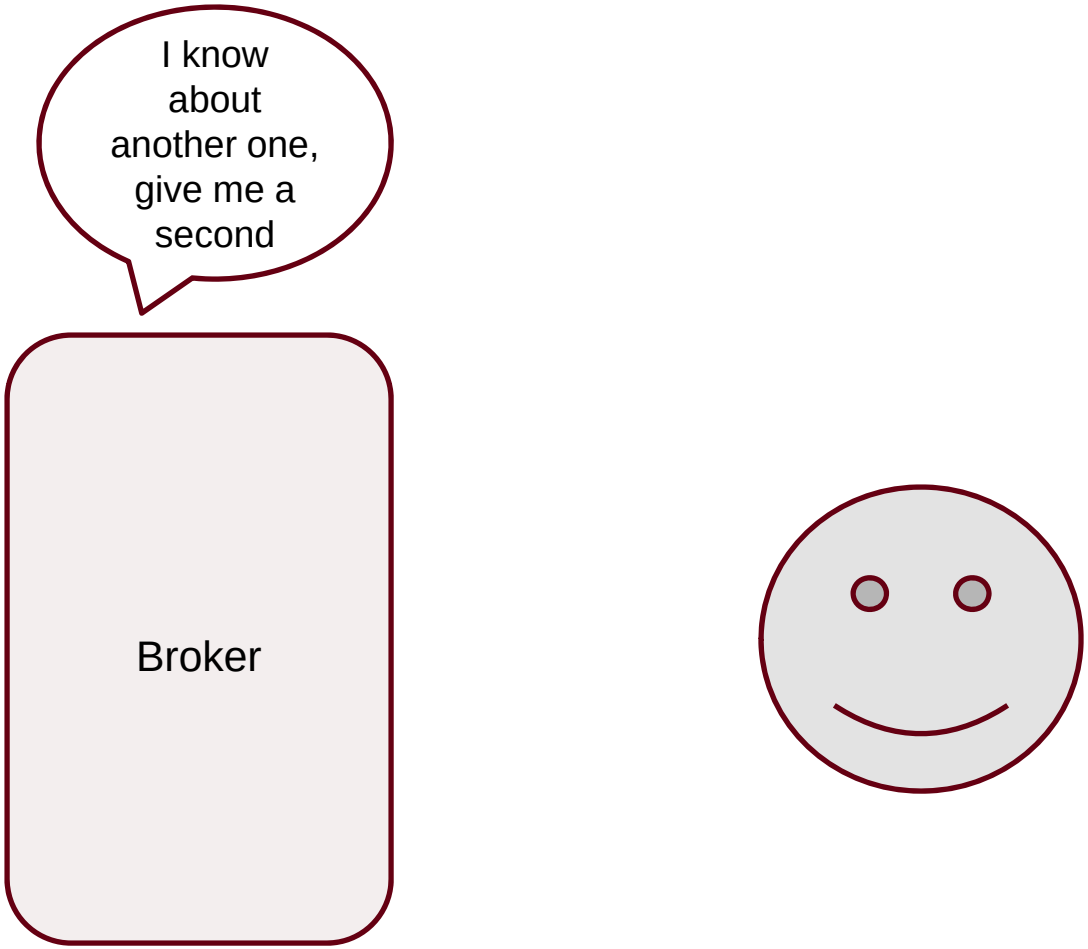
Detonation Model

Broker



Model sandboxing

What we want



Integration via mix-ins

What we want

Start Stress-By-Casualties Model

Stress-By-Detonations Model

Broker

Detonation Model

Integration via mix-ins

What we want

Yes

Stress-By-Detonations Model

Can you two compose*?

*Always a tricky question!

Yes

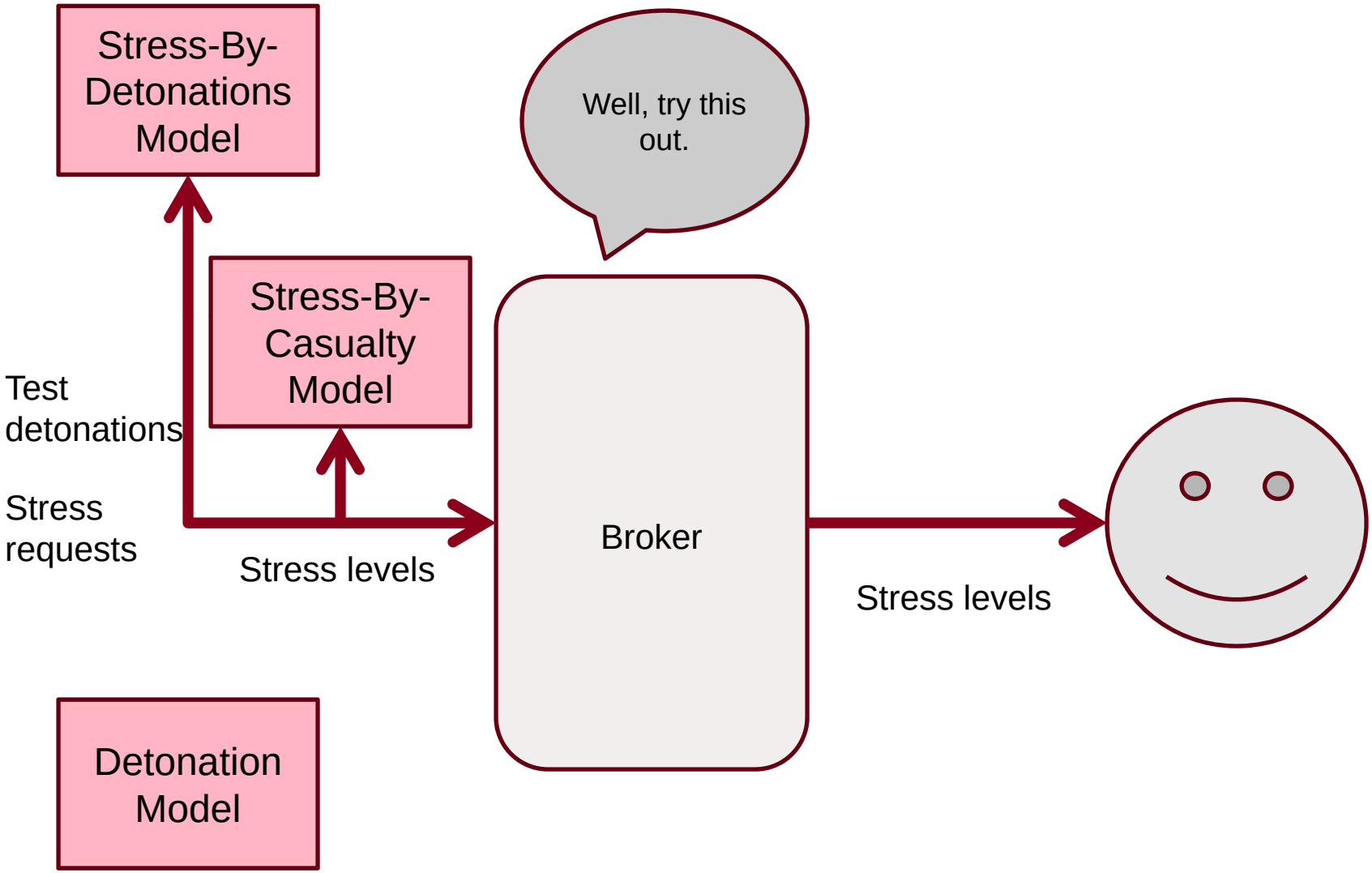
Stress-By-Casualty Model

Broker

Detonation Model

Integration via mix-ins

What we want



Model sandboxing

What we want

Stress-By-Detonations Model

Stress-By-Casualty Model

Detonation Model

How did that look?

Broker

Integration via mix-ins

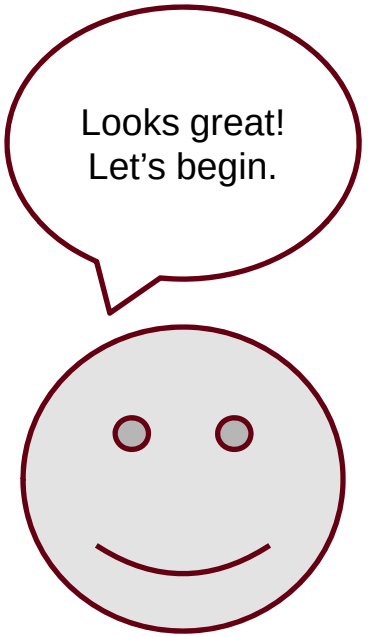
What we want

Stress-By-Detonations Model

Stress-By-Casualty Model

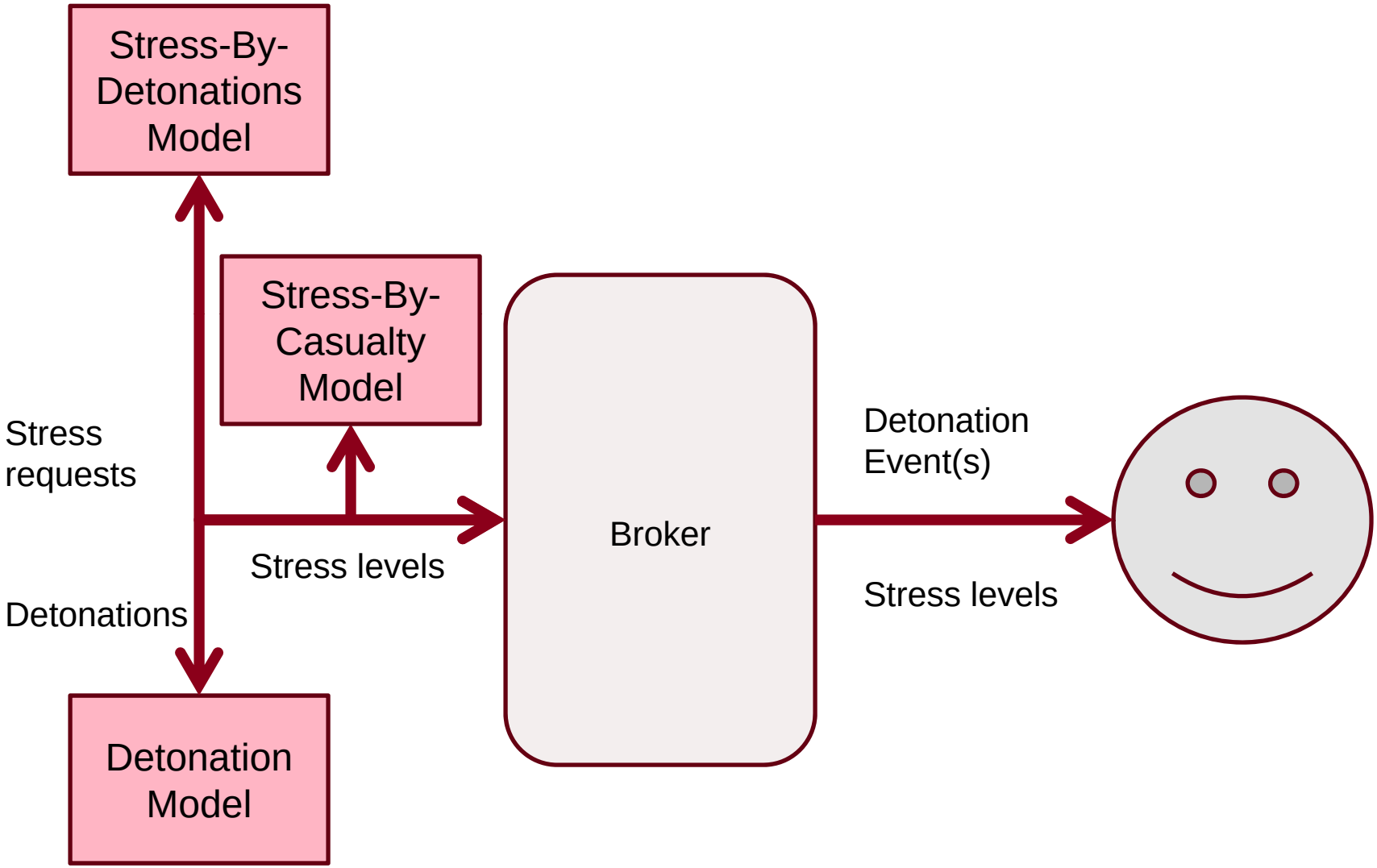
Detonation Model

Broker



Integration via mix-ins

What we want


Stress-By-Detonations Model
Stress-By-Casualty Model
Stress requests
Stress levels
Detonations
Broker
Detonation Event(s)
Stress levels
Detonation Model
Successful events

What we want

- Fluid, discoverable simulation models
  - Models need to be able to describe what they “think” that they can do
- Model sandboxing
  - Consumers and models need a way to test their interactions in a safe way to determine viability
- Integration via composition
  - Model composability needs to be expressible

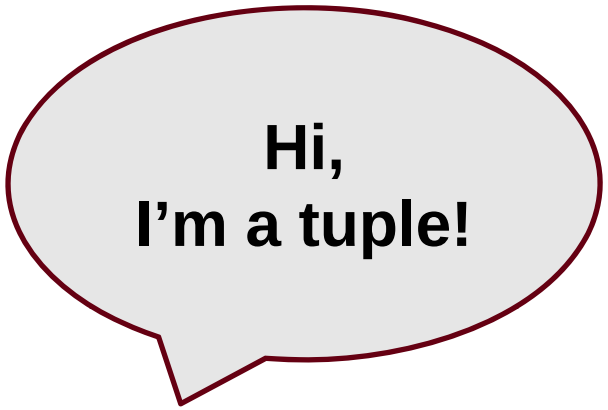
We need a "primitive" language

- To express wants and needs
- To express capabilities
- To express composition of capabilities
- That can "host" existing systems

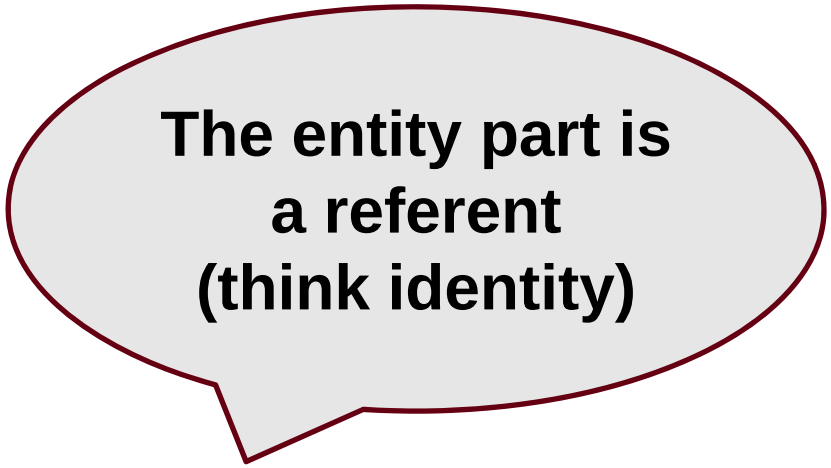
We need a primitive "primitive" language

- Simple
- Syntactically composable
- Syntactically inferential

Simple solution: Datalog



[<entity> <attribute> <value>]



[<entity> <attribute> <value>]

[<entity> <attribute> <value>]

The value part is a property's value for an entity

[<entity> <attribute> <value>]

Stress-By-Detonations Model

[Stress-By-Detonations Model :provides Stress]

Stress-By-Detonations Model

[Stress-By-Detonations Model :provides Stress]

[User :needs Stress]

Datalog composability

Stress-By-Detonations Model

[Stress-By-Detonations Model :provides Stress]

[Stress-By-Detonations Model :stress/component Detonations]

+

Stress-By-Casualty Model

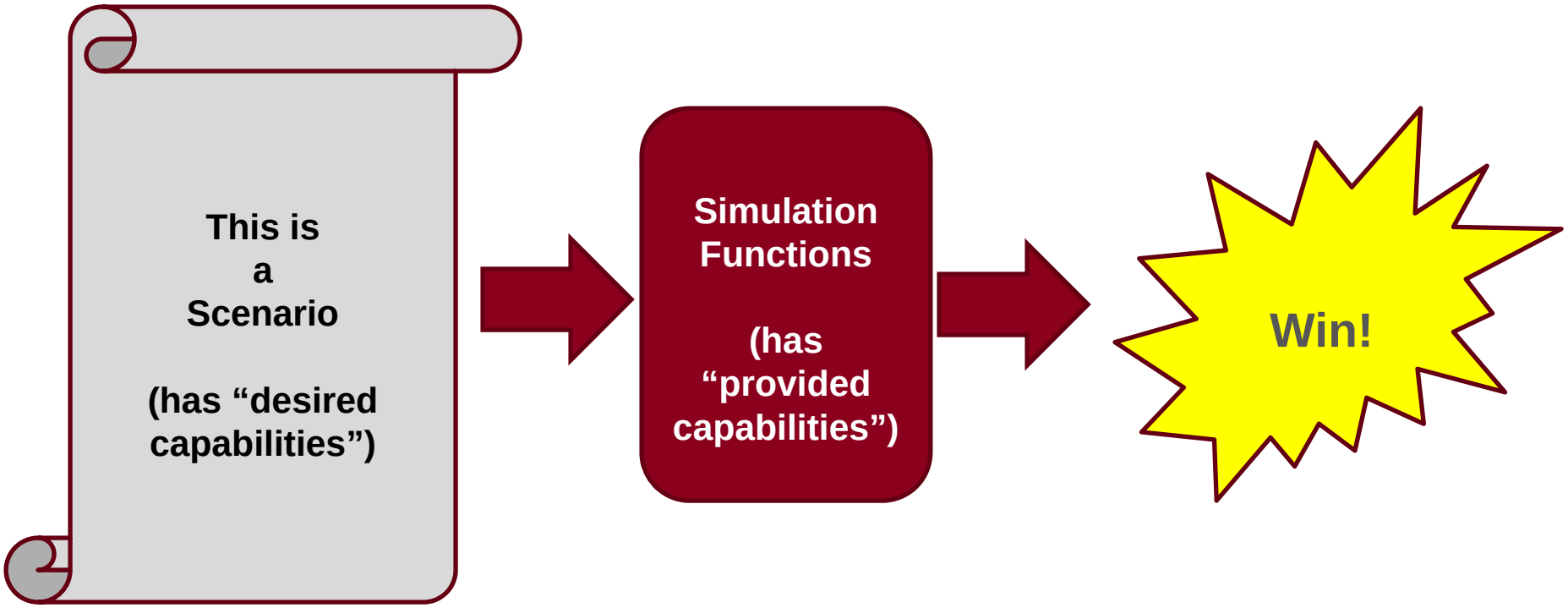
[Stress-By-Casualty Model :provides Stress]

[Stress-By-Casualty Model :stress/component Casualty]

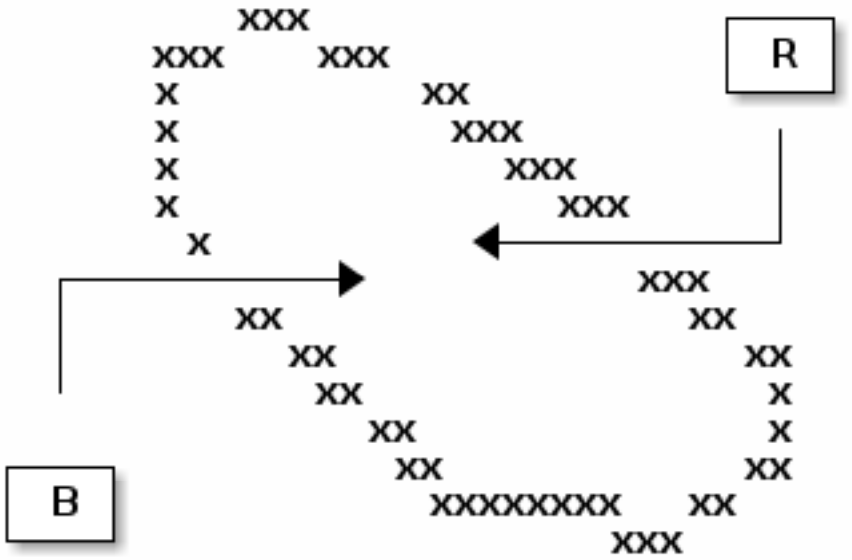
Datalog

- Allows inference and querying
- Hosts primitive “model” AND Object Models
- Simple
- Self-describing / self contained

Executable scenarios

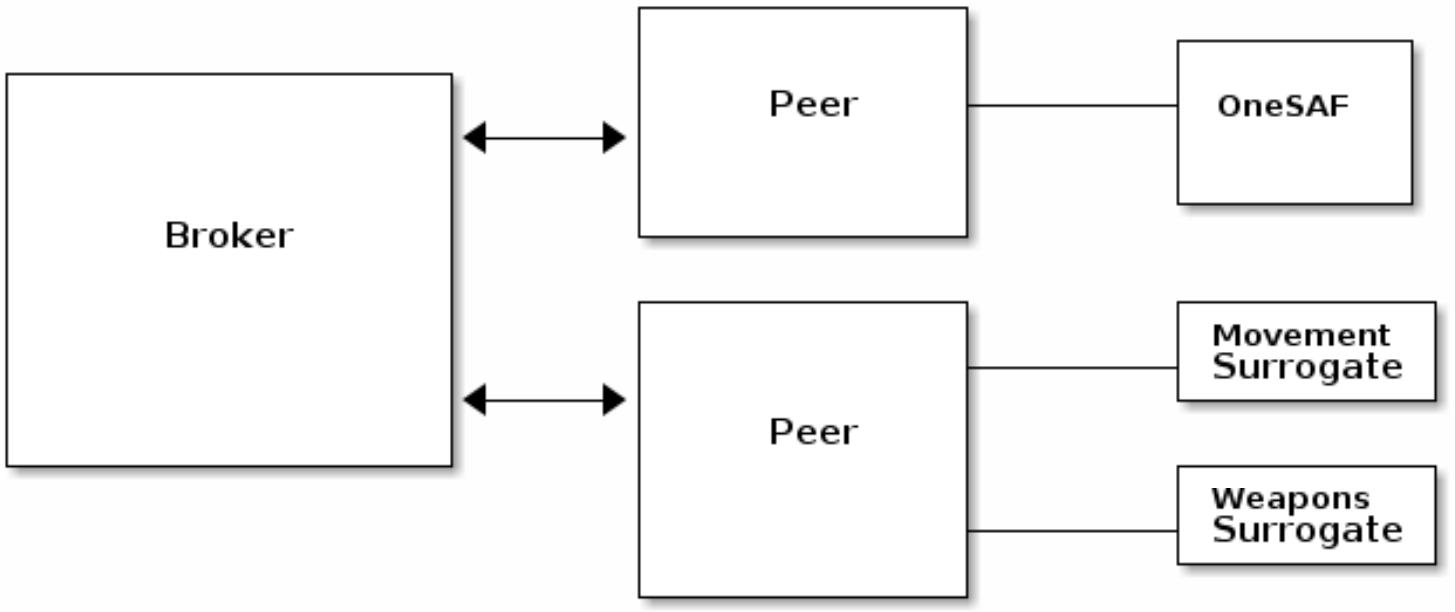


"Petri-dish" scenario



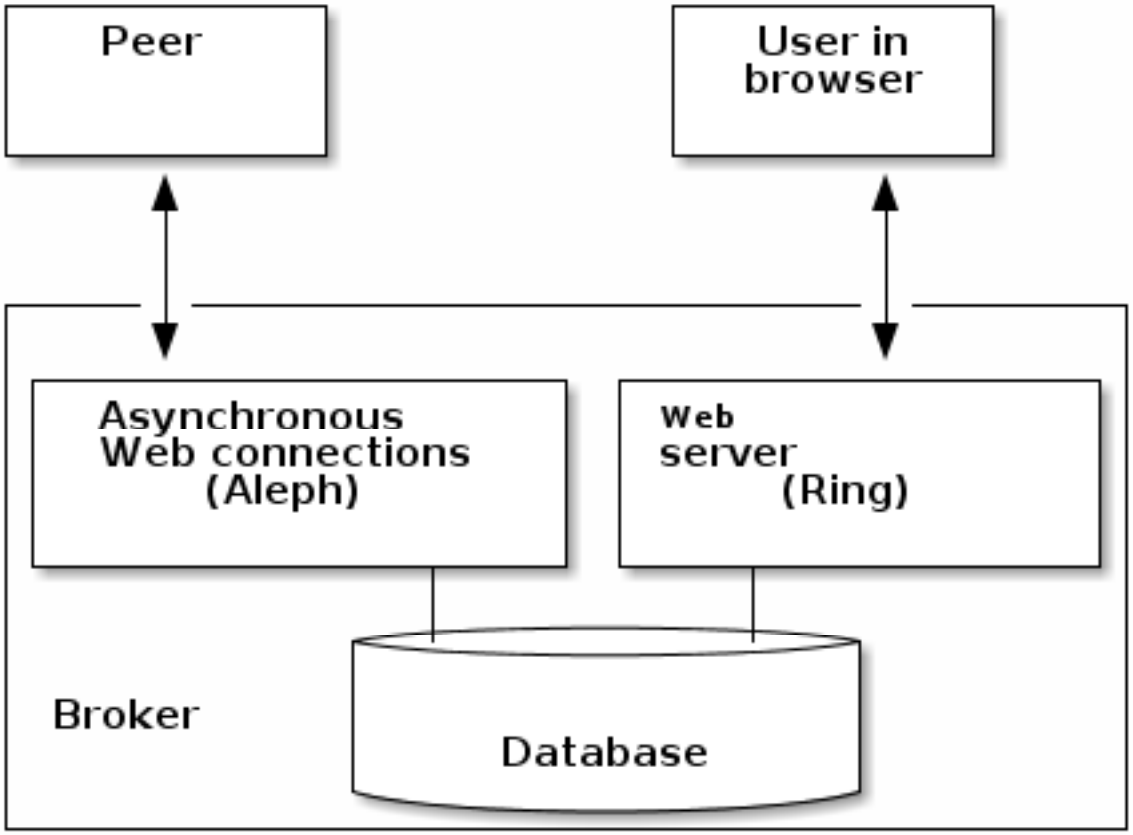
Red and Blue start with an obstruction between them. They move forward until they see each other and engage.

System overview



Capability	Primitives	Notes
Run-time infrastructure	n/a	This is a special category of capability
Plan View Display	n/a	This is a special category of capability
Creation	Time, Location, Heading, Units, Platforms	
	Side, Orientation	
Movement	Latitude, Longitude, Velocity, Orientation	
Damage	Health	
Fires	Weapons Control	
Vision	Sensation, Orientation	

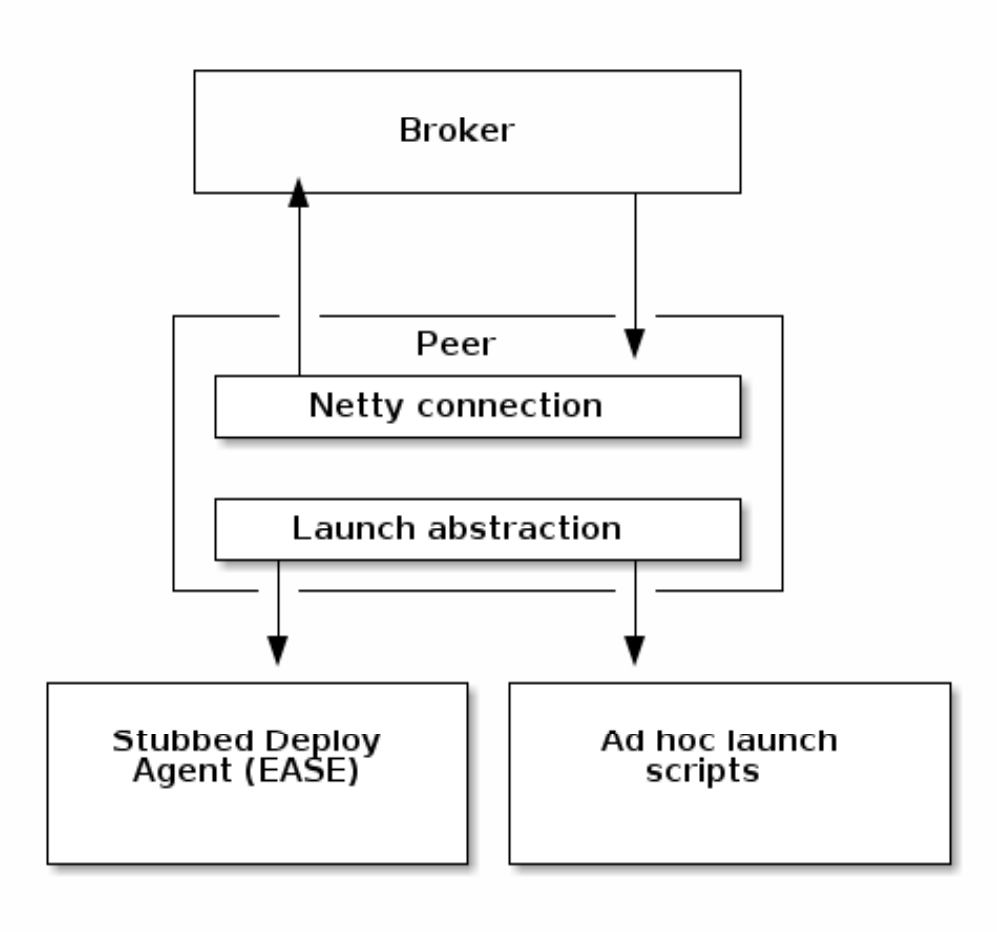
Broker architecture



Broker architecture

- Persistent Peer architecture
- Inferential database
- Browser-based user interface
- Hyper Text Transfer Protocol (HTTP)-based system-to-system interface
  - Messages containing collections of tuples
    - I.E. [<entity> <attribute> <value>]

Peer architecture



EASE: Executable Architecture Systems Engineering – contact authors for more information

Peer architecture

- Persistent connections
- Capabilities declaration
- Launch logic

Lessons learned

- We can use the Broker/Peer technologies now, and in the future
  - Coordinator (persistent connections, launch)
  - EASE (meta-model and inference)
- We need to think harder about simulation primitives
- We need to talk to some analysts
- We need to explore sandboxing in depth
- We need to explore auto-extraction of desired capabilities from scenarios

Chris Gaughan
Christopher J. Metevier
Simulation and Training Technology Center,
Human Research and Engineering Directorate,
Army Research Laboratory,
Orlando, Florida

Mike Fogus
Joseph S. McDonnell, Ph.D.
presenter – joe.mcdonnell@d-a-s.com
703-474-7038
Dynamic Animation Systems, Inc.
Fairfax, Virginia

Scott Gallant
Effective Applications Corp.,
Orlando, Florida

Questions/Comments?

ARL-HRED-STTC
Public Affairs Office
(407) 384-5227

Additional Reference