

Aspect Formulation for Developing Small-Footprint CORBA Event Channels

<http://www.cs.wustl.edu/~doc/RandD/PCES/facet/>

Ravi Pratap, Frank Hunleth and Ron K. Cytron

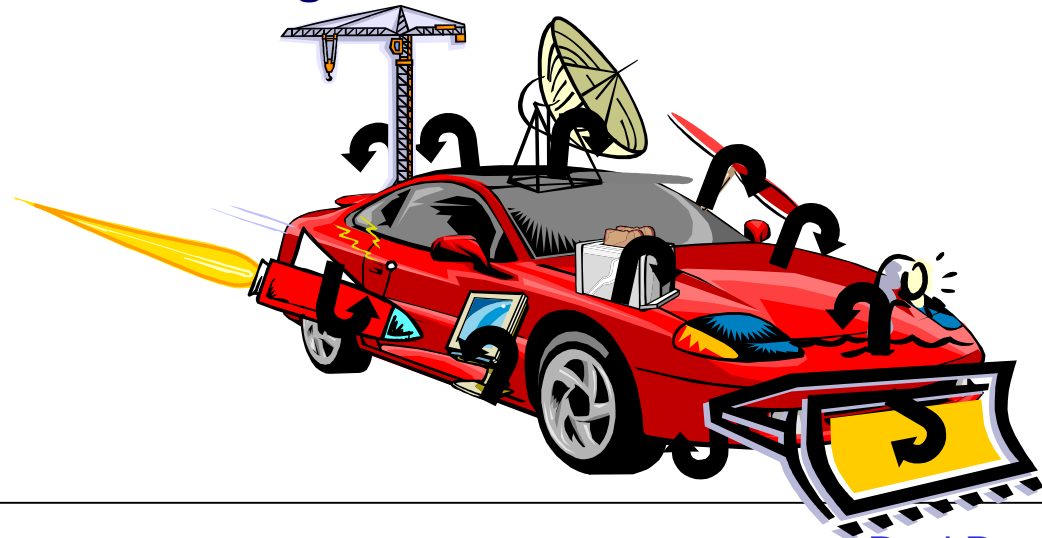
Center for Distributed Object Computing
Department of Computer Science
Washington University



Sponsored by DARPA under contract F33615-00-C-1697

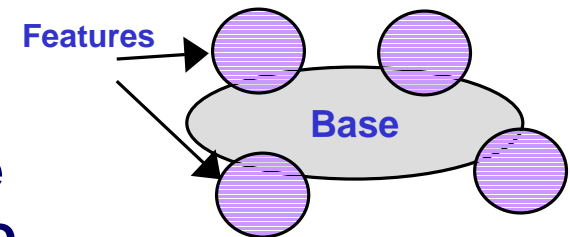
Traditional Middleware Development

- **First release**
 - Clean and well decomposed design – minimal set of features
- **Subsequent releases**
 - More features are added to the core
 - As core bloats, functionality is re-factored into features
- **Problems**
 - Programs typically need only a subset of the features
 - Re-factoring can be difficult and time consuming
 - Hooks need to be scattered throughout the code
- **Case study**
 - ACE and TAO



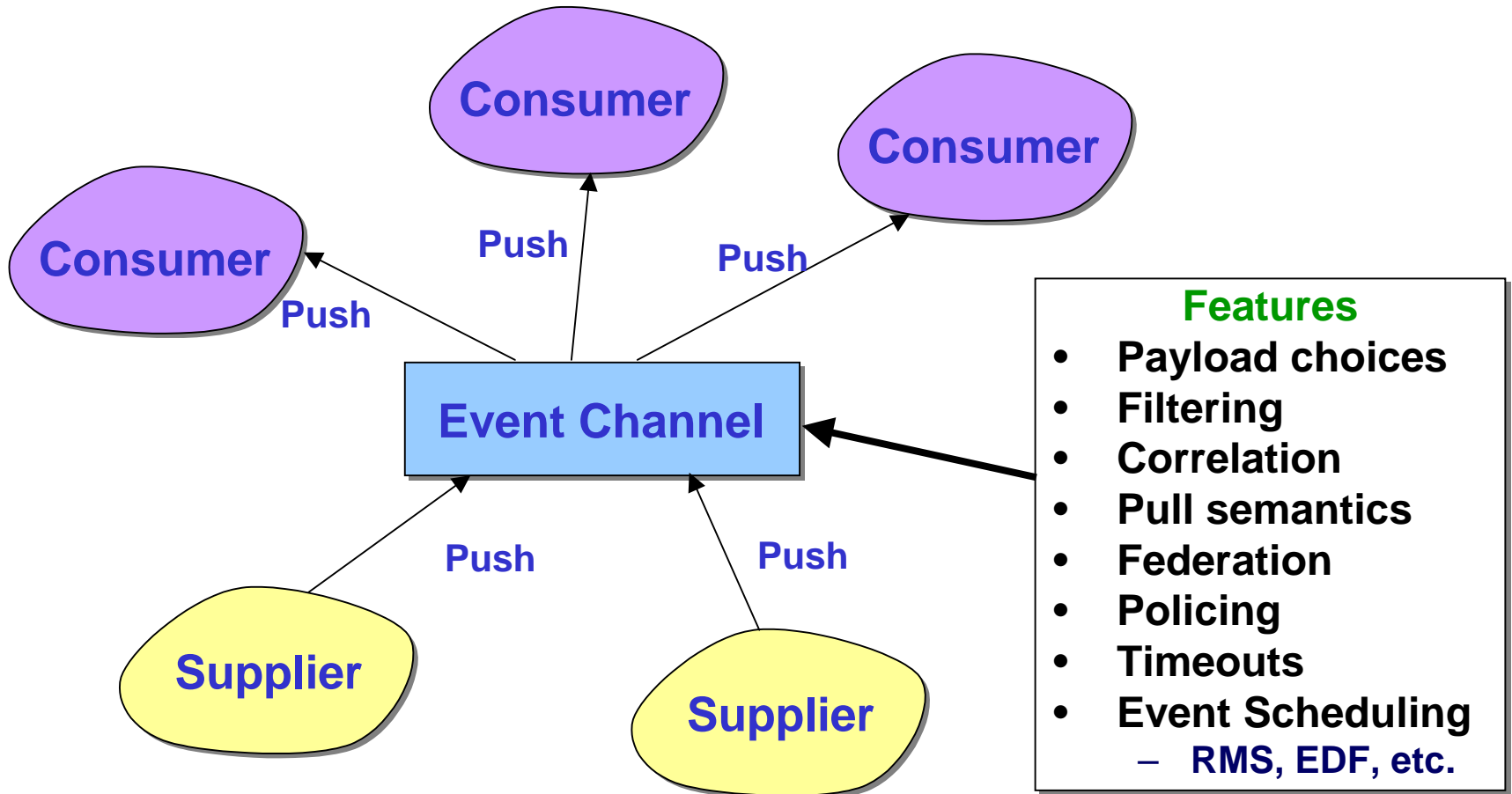
A Compositional Approach Using Aspects

- **Develop a simple base**
 - Contains minimal code common to all configurations
- **Add functionality with features**
 - Code is introduced using aspects
 - User selectable
- **Results**
 - What You Want Is All You Get – no code bloat or performance degradation due to redundant features
 - Flexible and simple – no need to strategize base code



FACET

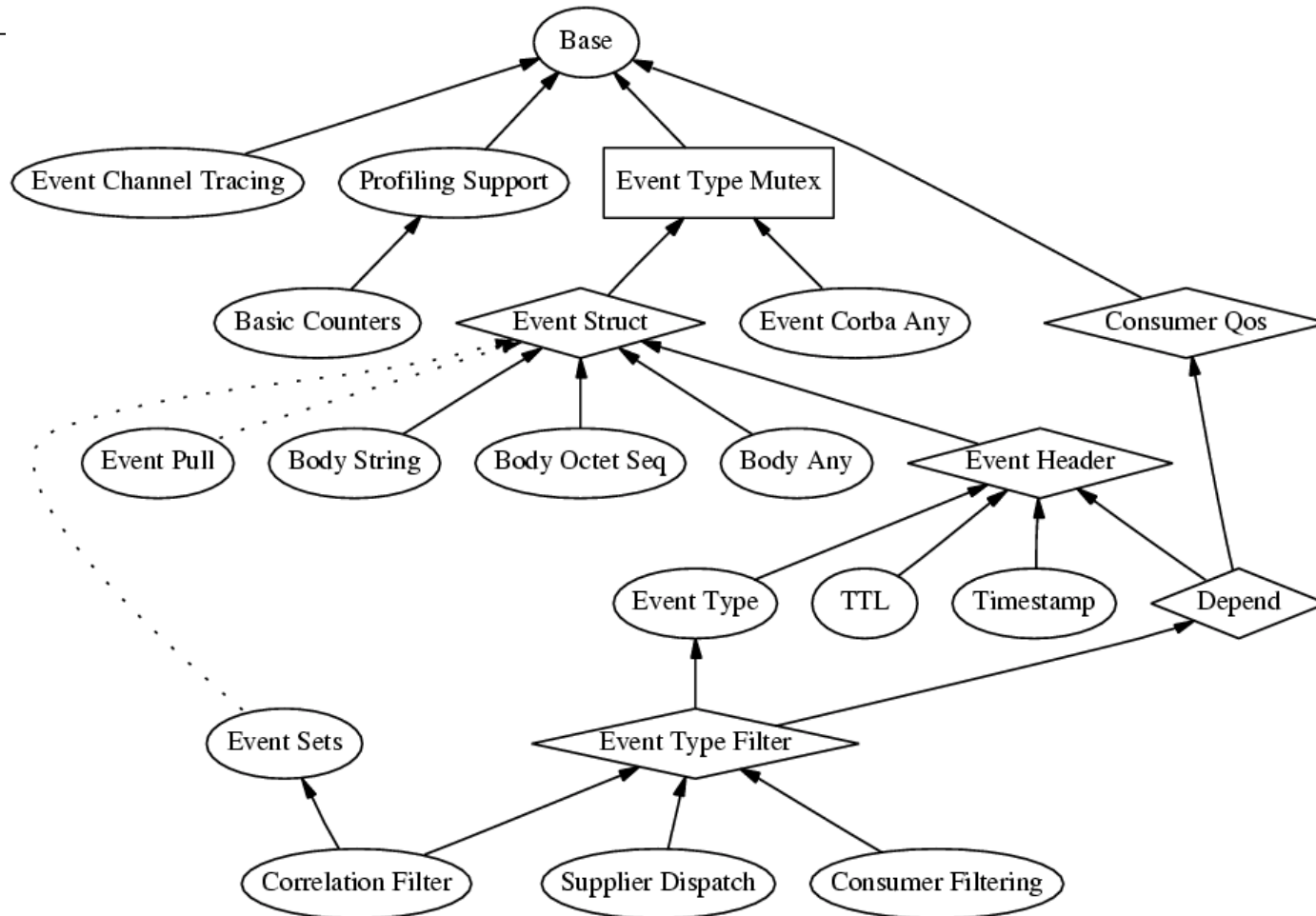
- Framework for **A**spect **C**omposition for an **E**ven**T** channel
- Modeled after features from the TAO Real-time Event Service



The Event Channel Base

- **Provides a basic CORBA “Interrupt” Service**
- **Contains**
 - **IDL specifications for simple “push” consumers and suppliers**
 - **No event data is passed**
 - push () method call takes no arguments
 - **Event channel forwards all events to all consumers**
 - No filtering

FACET Feature Dependence Graph



**Automatic testing and validation through
build system – 9192 valid combinations
out of the 2^{21} possibilities**

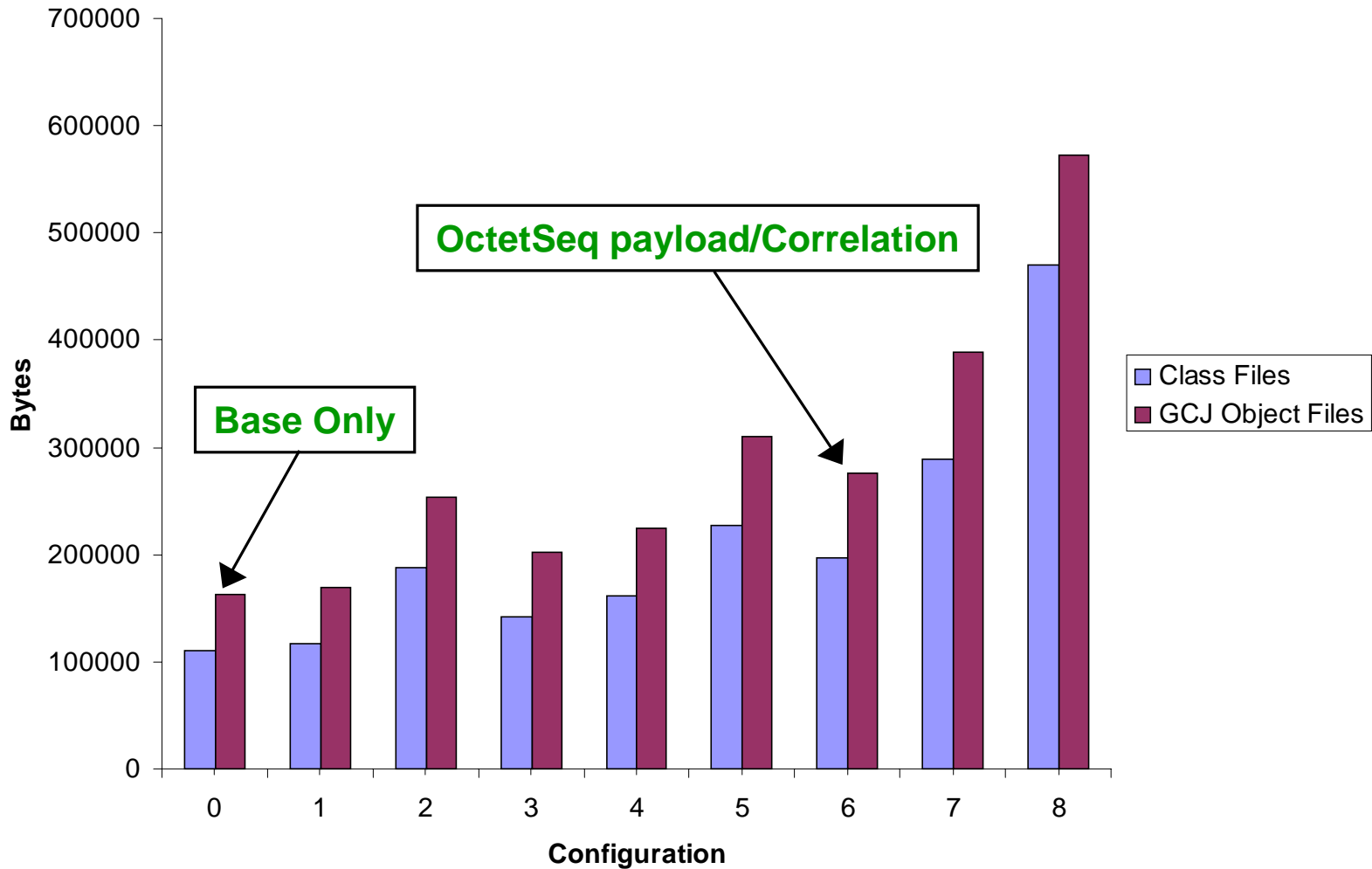
Common Event Channel Configurations

- TAO-users poll was conducted

	Tracing	Any Payload	Octet Seq Payload	Event sets	TTL	Time stamp	Type filtering	Event correlation	“Pull” Semantics
Config 0									
Config 1	X	X							
Config 2		X							
Config 3		X		X	X				
Config 4			X				X		
Config 5		X		X	X		X		X
Config 6			X			X		X	
Config 7		X		X	X	X	X	X	X
Config 8	X	X		X	X	X	X	X	X

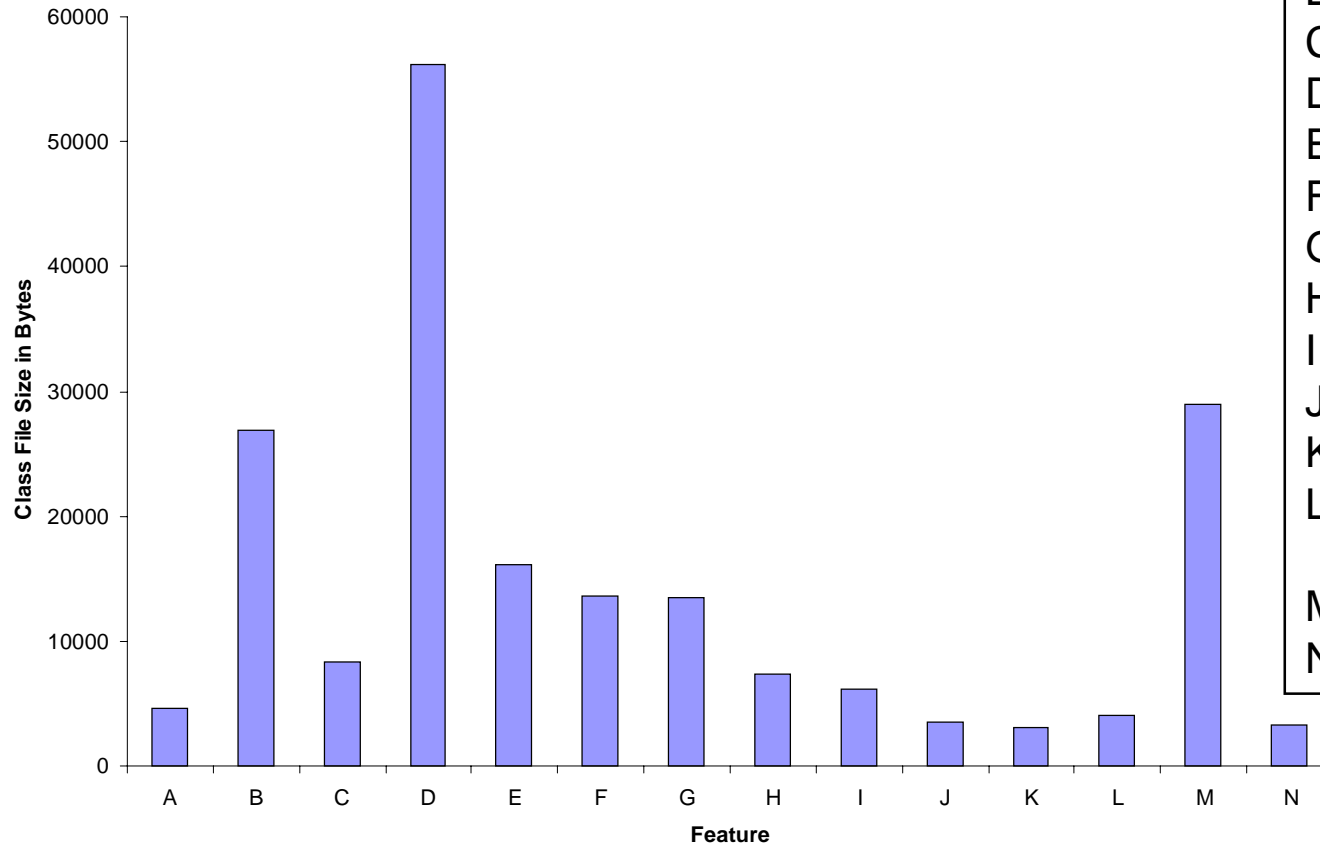
Library Footprint

FACET Footprint vs. Configuration



Footprint Results

Average Feature Sizes

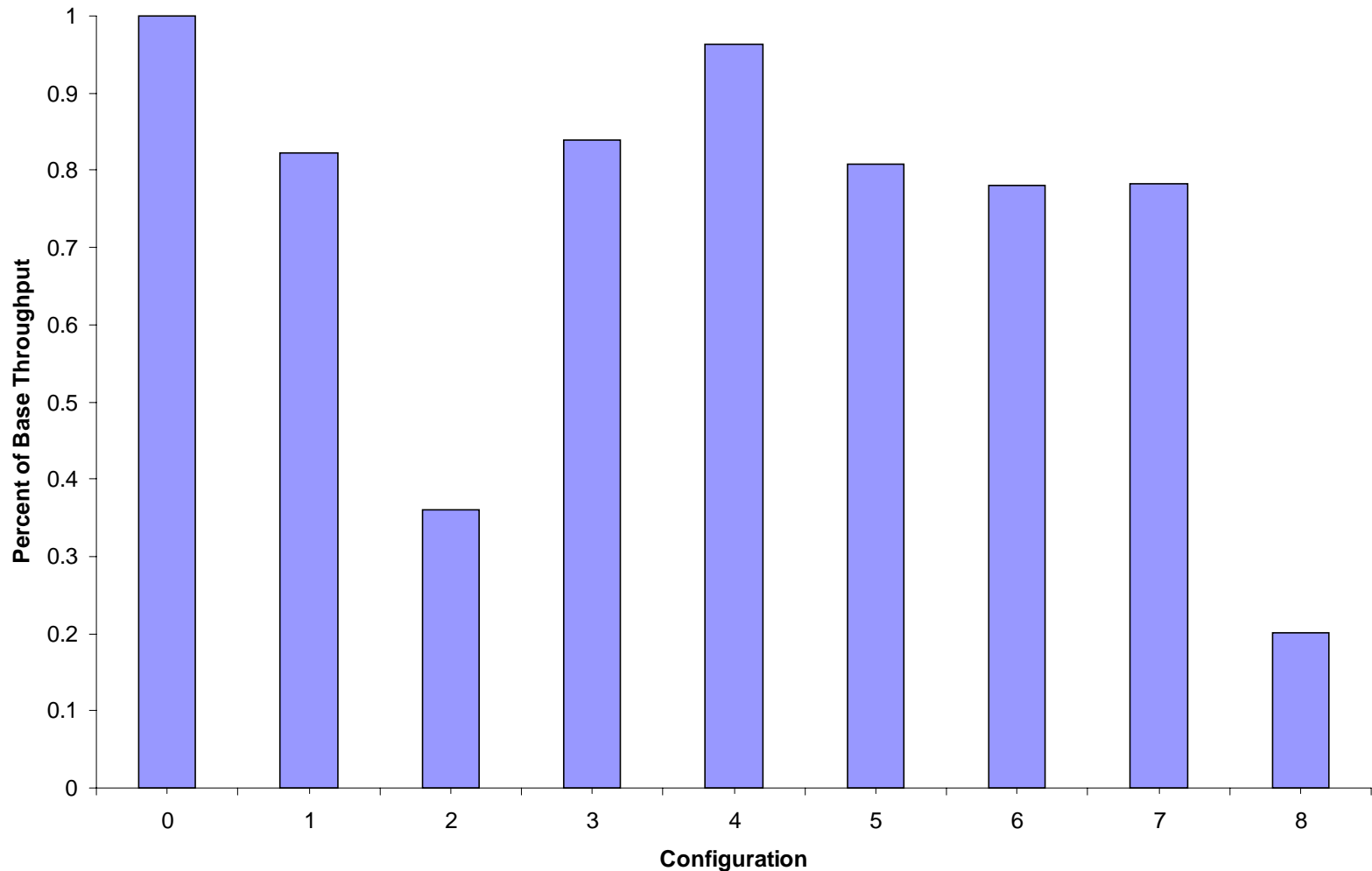


Key

- A: Simple event filtering
- B: Event correlation
- C: Event Set support
- D: Event Pull support
- E: Octet sequence payload
- F: Any payload
- G: String payload
- H: CORBA Any events
- I: Calculated: Event header
- J: Event type
- K: Timestamp
- L: Time to live field and processing
- M: Profiling support
- N: Event statistics counters

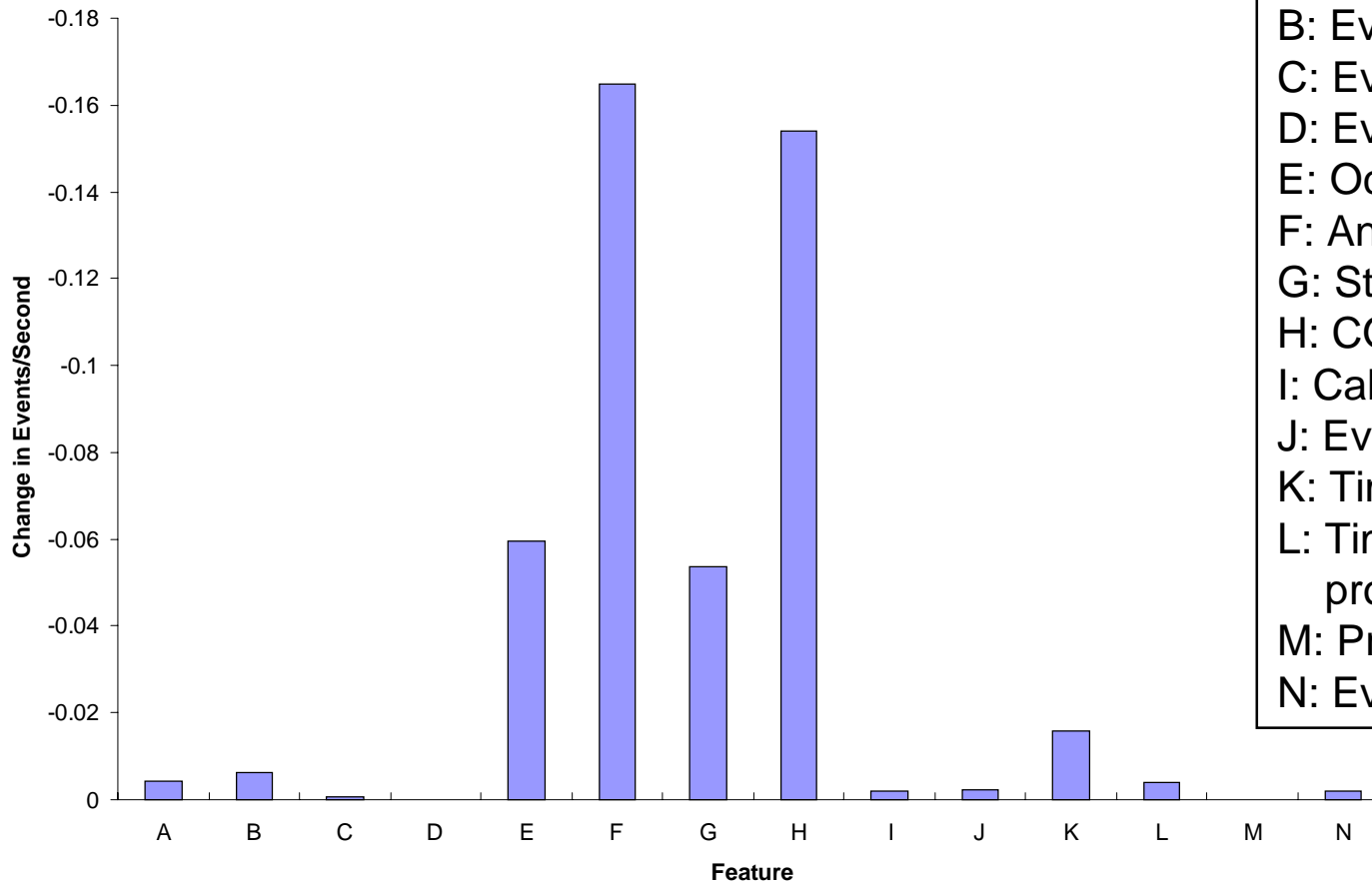
Throughput for Common Configurations

Event Throughput vs. Configuration



Feature Performance Results

Average Feature Effect on Raw Throughput

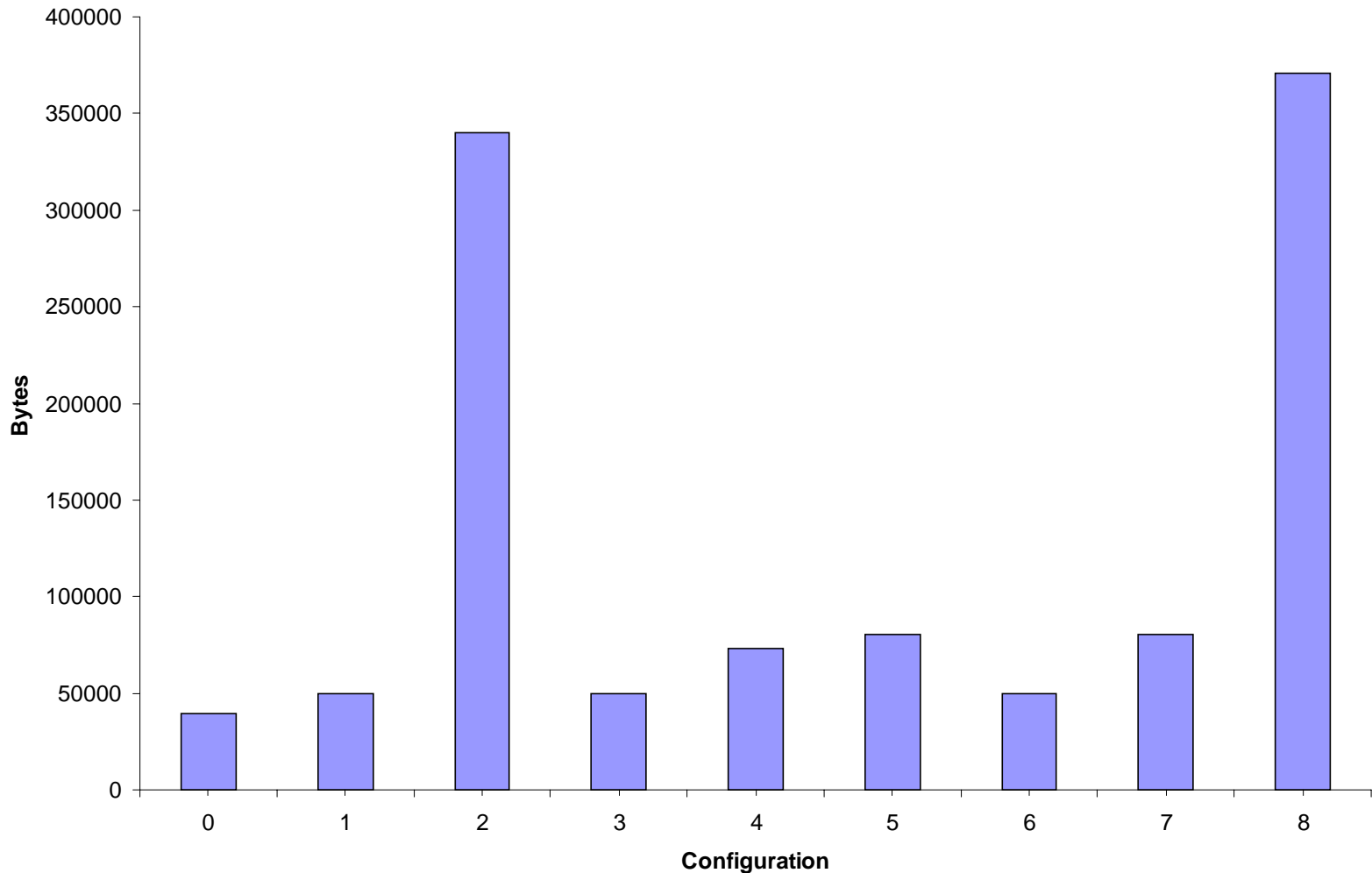


Key

- A: Simple event filtering
- B: Event correlation
- C: Event Set support
- D: Event Pull support
- E: Octet sequence payload
- F: Any payload
- G: String payload
- H: CORBA Any events
- I: Calculated: Event header
- J: Event type
- K: Timestamp
- L: Time to live field and processing
- M: Profiling support
- N: Event statistics counters

Additional Code From External Libraries

Additional External Library Footprint vs. Configuration



Non-CORBA Contexts

- **Boeing's *Bold Stroke* architecture**
 - Requires complex features – filtering, correlation etc.
 - Needs events to be dispatched in real-time based on various parameters
 - JacORB is “too heavy” and not decomposable
- **FACET was found to be ideal for *Bold Stroke***
 - Real-time event dispatching added through a feature since the dispatcher is pluggable
 - Simple Java objects in place of CORBA objects

Current Work

- **FACET needs to be used in wide contexts**
 - Make the use of CORBA optional by making it a feature
 - Enable support for CORBA replacements – Java RMI for example
- **Challenges**
 - Aspects cannot be applied to CORBA IDL
 - Using CORBA's standard naming conventions to apply advice
 - Ensure there is no predication towards CORBA – something which currently exists

Conclusions

- **Highly customizable CORBA event channel**
- **AOP is a novel and practical approach to building customizable middleware**
- **Footprint and throughput results support the use of AOP techniques**

More Information

- **FACET is available from :**
 - <http://www.cs.wustl.edu/~doc/RandD/PCES/facet/>
- **AspectJ – AOP bindings for Java :**
 - <http://www.aspectj.org>
- **The DOC Group at Washington University**
 - <http://www.cs.wustl.edu/~doc/>

Thanks

- **Advisor - Dr. Ron K. Cytron**
- **Original FACET Author - Frank Hunleth**
- **The DOC Group at Washington University**
- **The audience for bearing with me !**