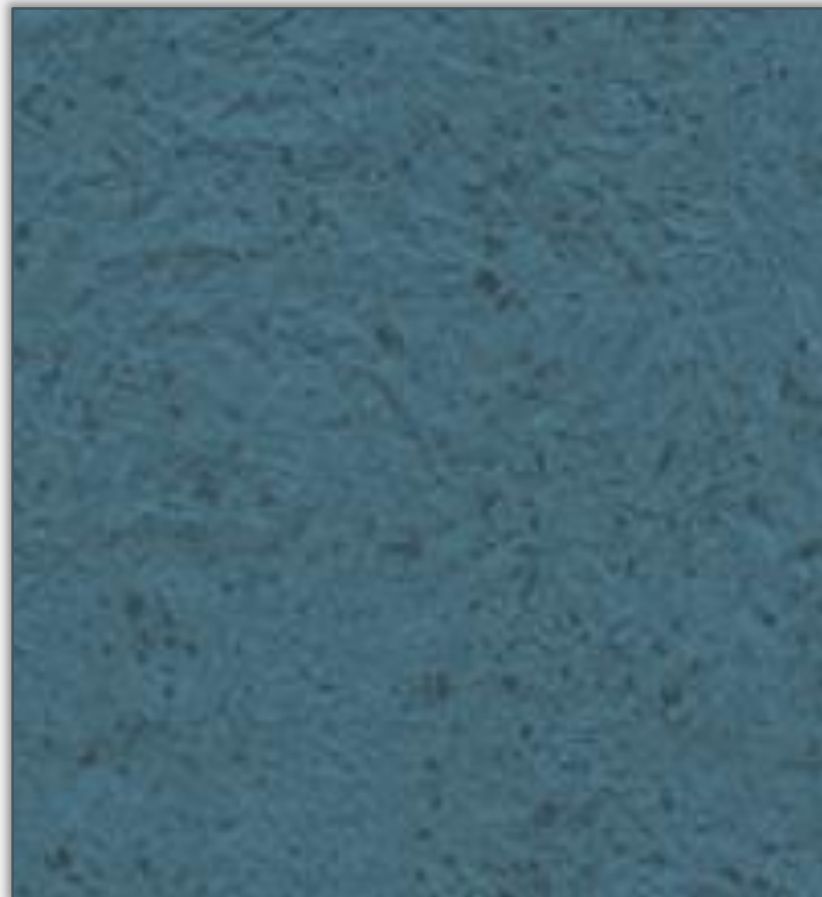


# Atributos de Calidad del Software

# Funcionalidad vs. Atributos de Calidad

- Características Ortogonales (a las vistas)
- Atributos de calidad: no necesariamente alcanzables para cualquier funcionalidad.
- Las decisiones arquitectónicas determinarán un nivel relativo de calidad.

# Atributos de calidad



(L,M)

Reduce storage latency on customer DB to < 200 ms.

(M,M)

Deliver video in real time.

(H,H)

Add CORBA middleware in < 20 person-months.

(H,L)

Change Web user interface in < 4 person-weeks.

(H,H)

Power outage at site1 requires traffic redirected to site2 in < 3 seconds.

(H,H)

Network failure detected and recovered in < 1.5 minutes.

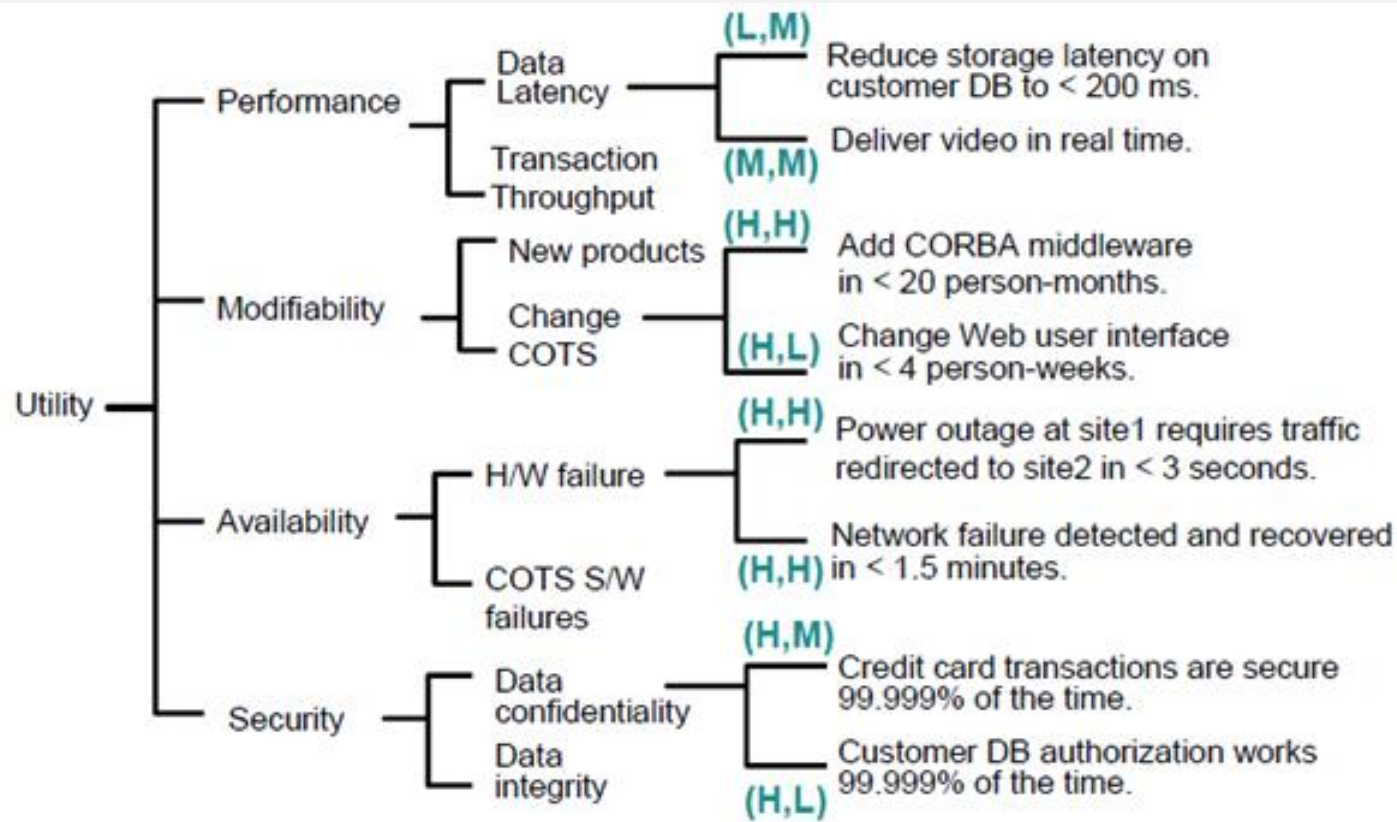
(H,M)

Credit card transactions are secure 99.999% of the time.

(H,L)

Customer DB authorization works 99.999% of the time.

# Atributos de calidad





# Overwhelmability?

Access Control	Coordination Time	Informativeness	Planning Cost	Space
Accessibility	Correctness	Installability	Planning Time	Boundedness
Accountability	Coupling	Integrity	Portability	Space Performance
Accuracy	Customer Evaluation	Internal Consistency	Precision	Specificity
Adaptability	Customer Loyalty	Interoperability	Predictability	Stability
Additivity	Customizability	Intuitiveness	Process Mgmt. Time	Subjectivity
Adjustability	Data Space*	Learnability	Productivity	Supportability
Affordability	Decomposability	Leveragability	Project Stability	Surety
Agility	Degradation of	Main Memory*	Project Tracking Cost	Survivability
Auditability	Service	Maintainability	Promptness	Susceptibility
Augmentability	Dependability	Maintenance Cost	Quality	Sustainability
Autonomy	Development Cost	Maintenance Time	Reconfigurability	Tankness
Availability	Development Time	Maneuverability	Recoverability	Testability
Buffer Space*	Distributivity	Maturity	Recovery	Throughput
Capability	Diversity	Mean Performance	Reliability	Timeliness
Capacity	Ease of use	Measurability	Repeatability	Traceability
Clarity	Efficiency	Mobility	Replaceability	Trainability
Code Space*	Elasticity	Modifiability	Replicability	Transferability
Cohesiveness	Enhanceability	Modularity	Response Time	Transparency
Commonality	Evolvability	Multi-organizational	Responsiveness	Un-parochialness
Compatibility	Execution Cost	Multiness	Retirement Cost	Understandability
Completeness	Extendability	Multi-lingual	Reusability	Uniform*
Comprehensibility	Extensibility	Naturalness	Risk Analysis Cost	Uniformity
Conceptuality	External Consistency	Nomadicity	Risk Analysis Time	Usability
Conciseness	Fault Tolerance	Observability	Robustness	User-Friendliness
Confidentiality	Feasibility	Off-Peak Period*	Safety	Validity
Configurability	Flexibility	Operability	Scalability	Variability
Connectivity	Formality	Operating Cost	Security	Verifiability
Consistency	Generality	Parochialness	Sensitivity	Versatility
Controllability	Human Engineering	Peak Period*	Similarity	Visibility
Coordination Cost	Independence	Performability	Simplicity	*Performance

# Atributos de calidad

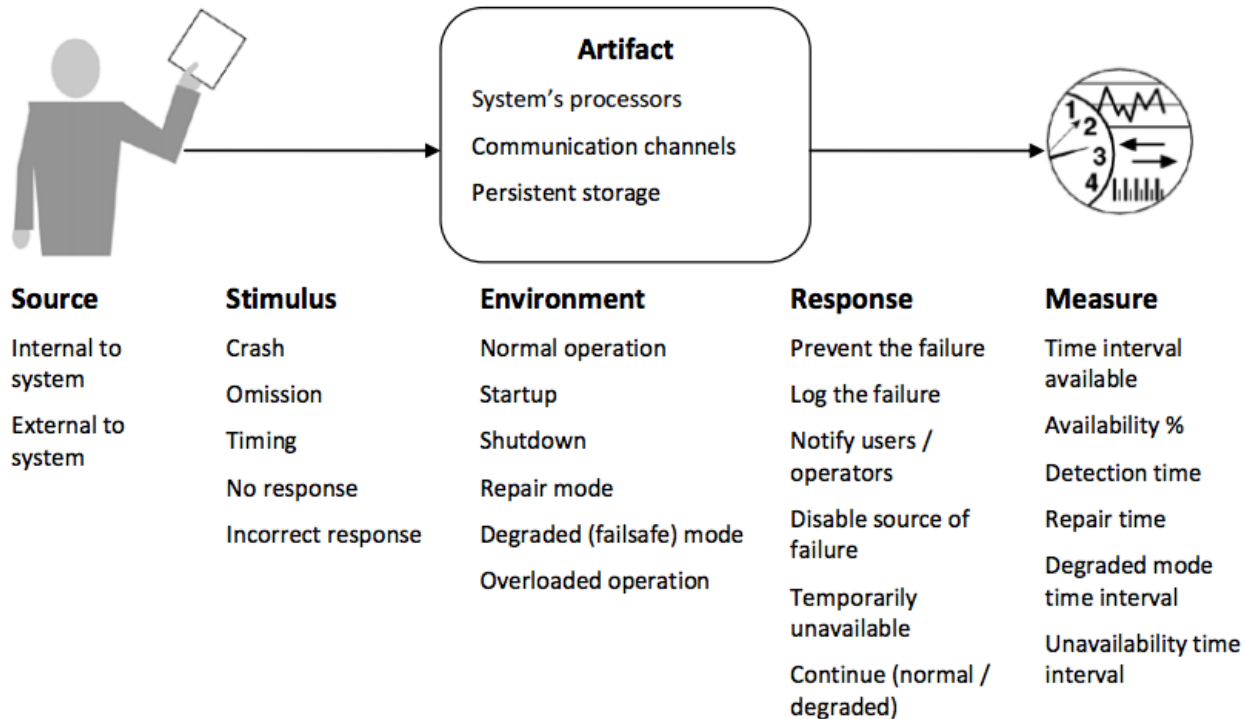
- Especificación de requerimientos
  - Fuente: Origen del estímulo.
  - Estímulo: Evento dentro del sistema o dentro del proceso de desarrollo.
  - Entorno: Contexto/estado en el que se presenta el estímulo.
  - Respuesta: Cómo el sistema maneja (o permite manejar) el evento.
  - Métrica de la respuesta: resultados esperados (o encontrados) tras la aplicación de la respuesta.

# Disponibilidad

- Asociado a las fallas del sistema y sus consecuencias asociadas.
- Dicha falla es observable por los usuarios del sistema (humanos, otros sistemas).

# Disponibilidad

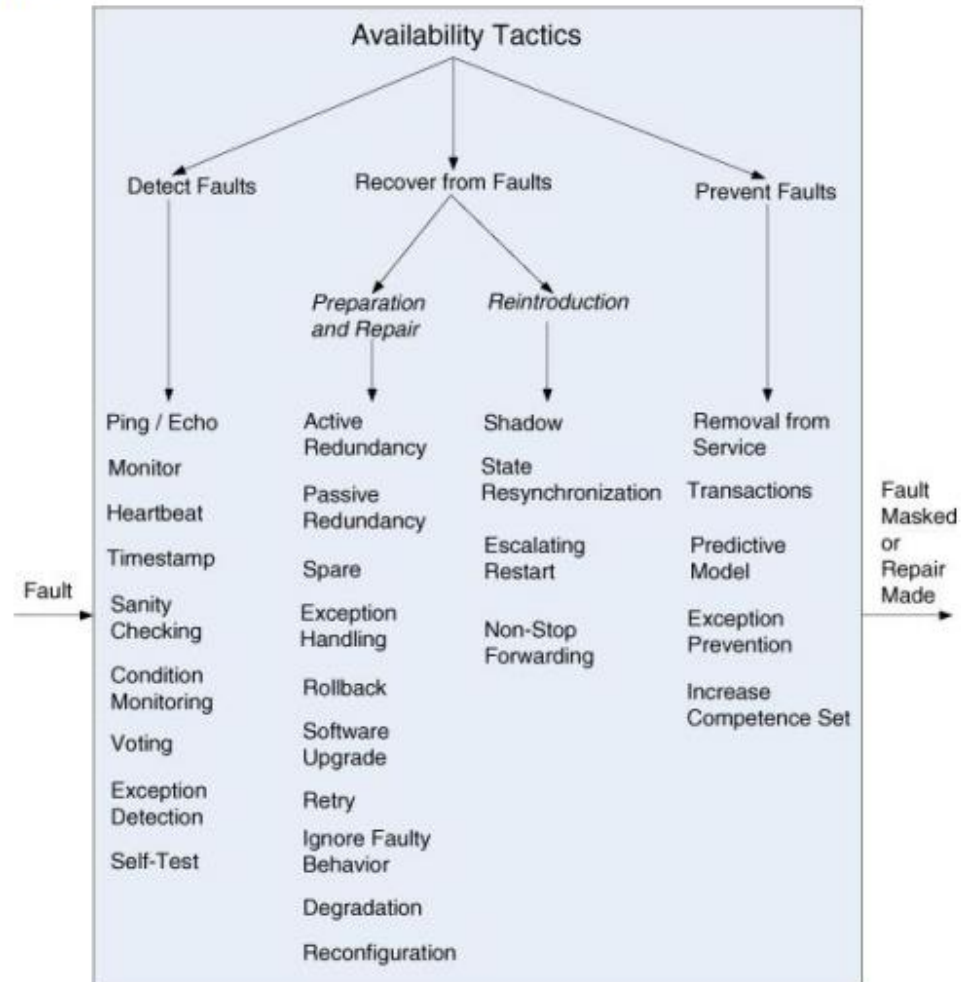
## General availability scenario





# Disponibilidad

## Availability tactics

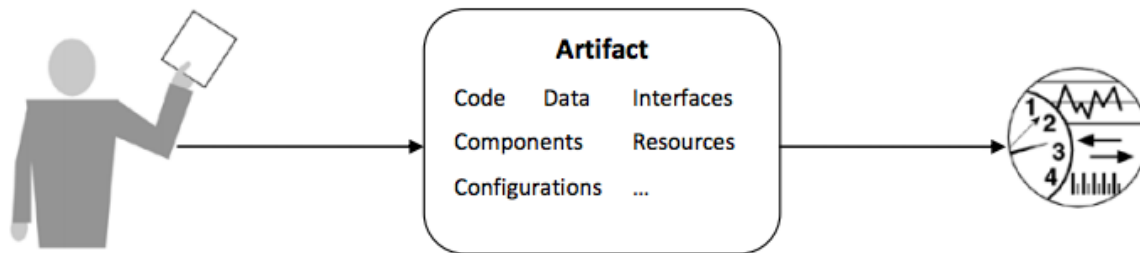


# Modificabilidad

- Asociado al costo del cambio:
  - Qué puede cambiar?
  - Cuando se realiza en cambio y quien lo puede realizar?

# Modificabilidad

## General modifiability scenario



### Source

End-user  
Developer  
System-administrator

### Stimulus

Add / delete /  
modify  
functionality,  
quality attribute,  
capacity or  
technology

### Environment

Runtime  
Compile time  
Build time  
Initiation time  
Design time

### Response

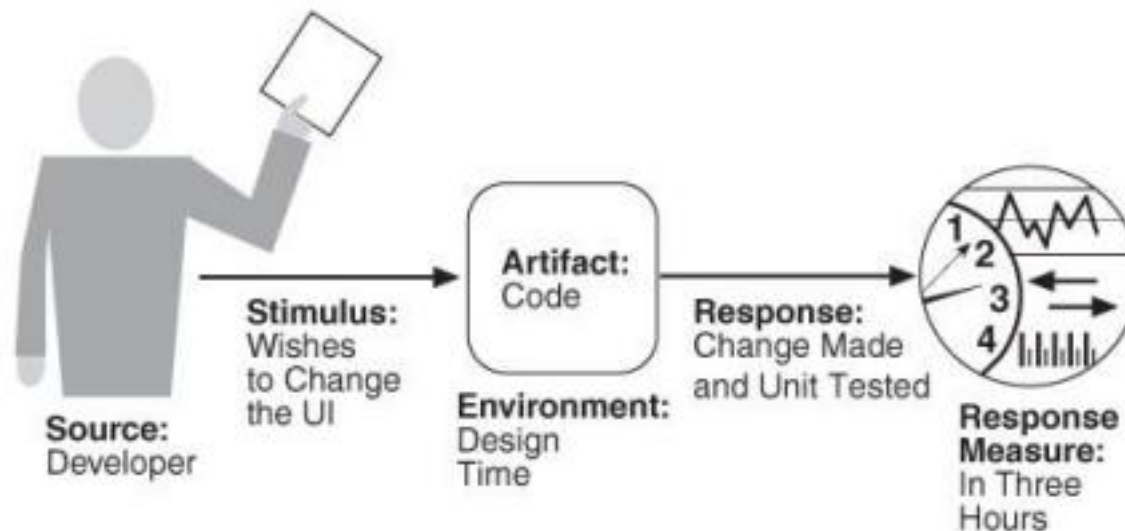
Make modification  
Test modification  
Deploy  
modification

### Measure

Cost in effort  
Cost in money  
Cost in time  
Cost in number,  
size, complexity of  
affected artifacts  
Extent affects other  
system functions or  
qualities  
New defects  
introduced

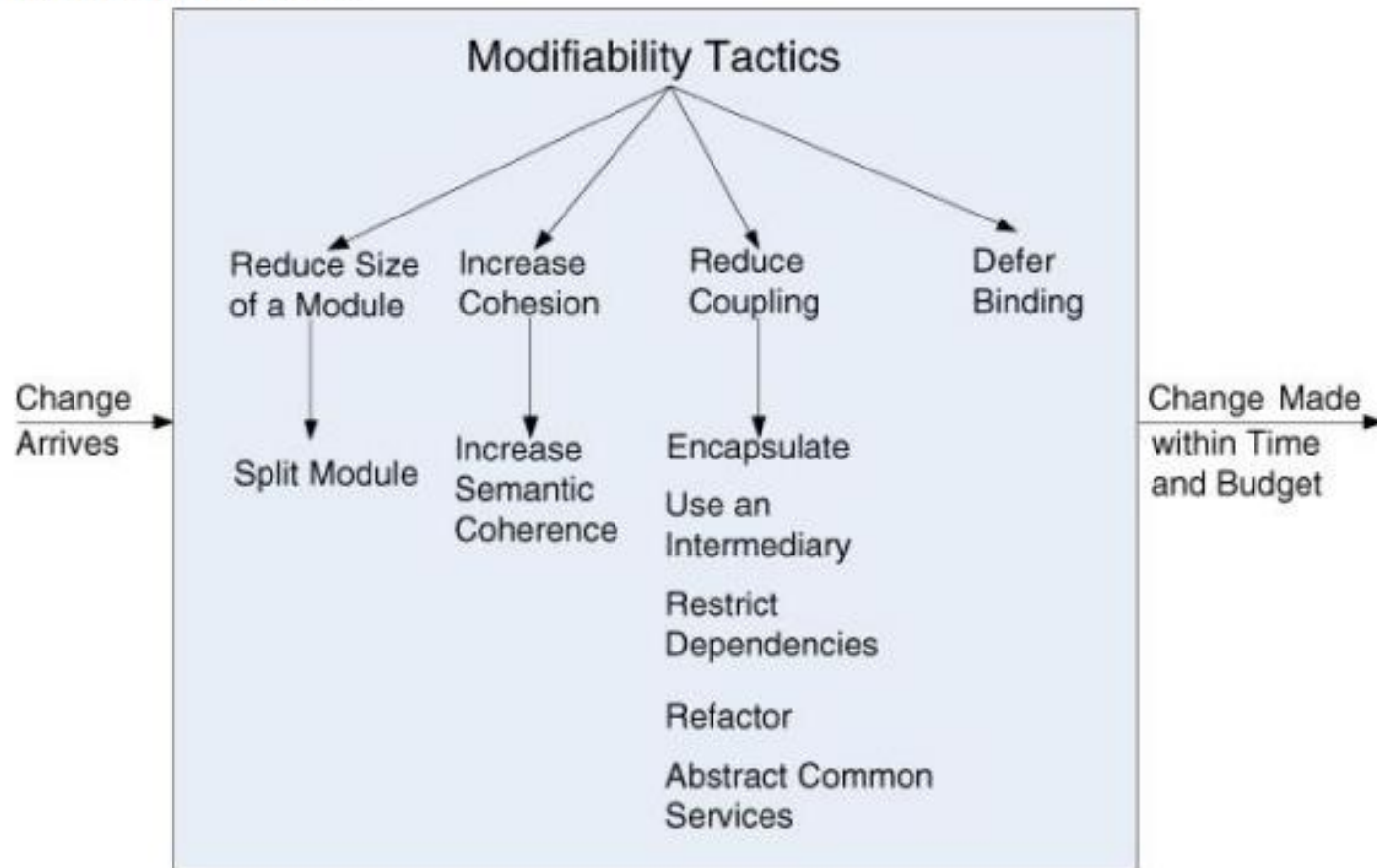
# Modificabilidad

## *Sample modifiability scenario*



# Modificabilidad

## ***Modifiability tactics***





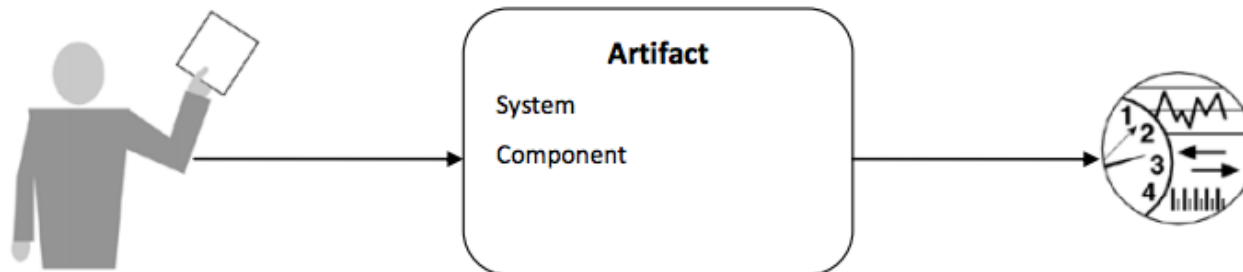
# Desempeño

- Referente a tasas de respuesta, latencia, tiempos límite, tasa de pérdidas de datos, etc, frente a eventos ocurridos en el sistema.
- Se debe considerar que los eventos pueden ser periódicos, esporádicos, o de carácter estocástico.

# Desempeño

## Performance

### *General performance scenario*



#### **Source**

Internal to the system  
External to the system

#### **Stimulus**

Periodic events  
Sporadic events  
Bursty events  
Stochastic events

#### **Environment**

Normal mode  
Overload mode  
Reduced capacity mode  
Emergency mode  
Peak mode

#### **Response**

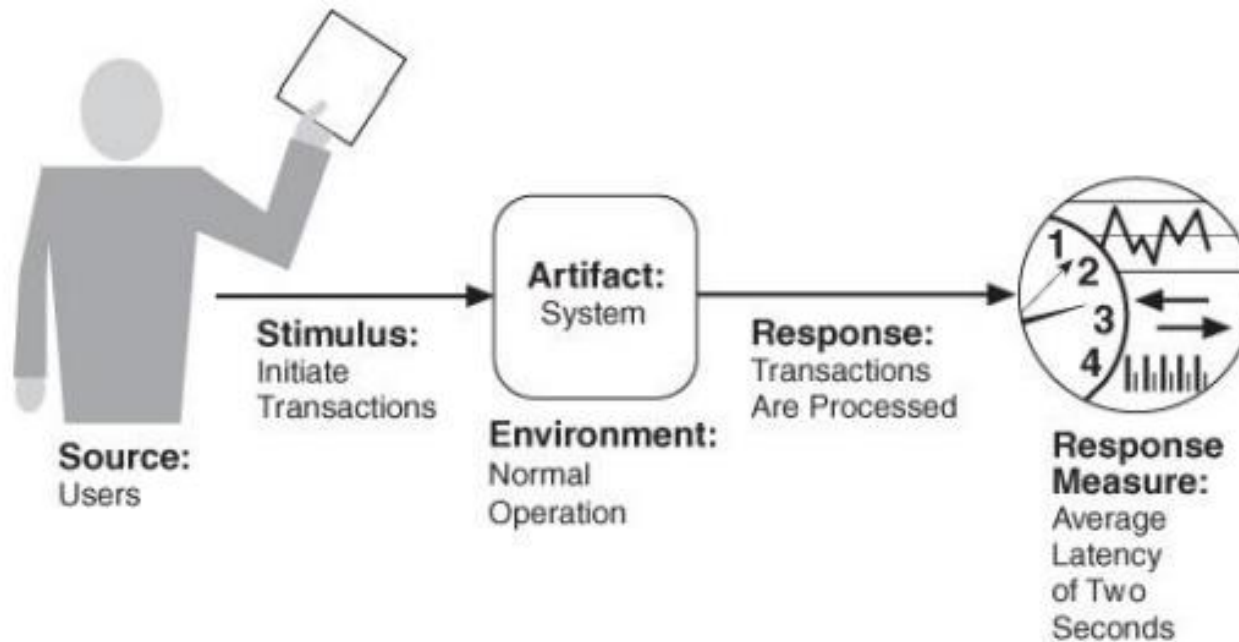
Process events  
Change level of service

#### **Measure**

Latency  
Deadline  
Throughput  
Jitter  
Miss rate  
Data loss

# Desempeño

## *Sample performance scenario*



# Desempeño

## *Performance tactics*



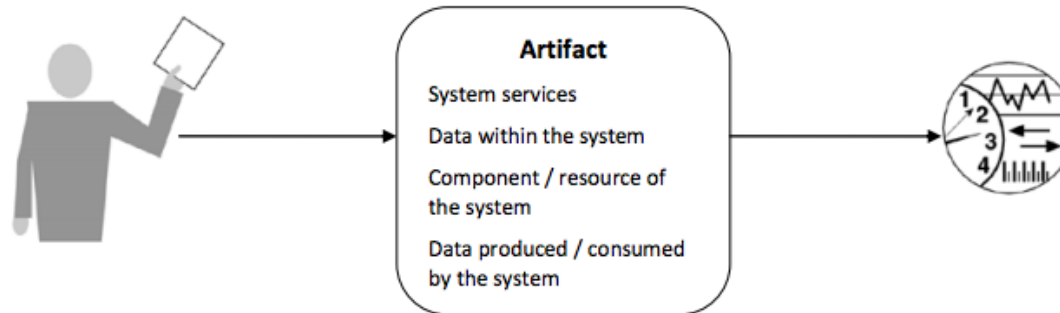
# Seguridad

- Medida de la habilidad del sistema para resistir usos no autorizados, sin dejar de prestar el servicio a los usuarios legítimos.
- Ejemplo de uso no autorizado: acceder a datos o servicios para modificar datos, o denegar el servicio a los usuarios legítimos.



# Seguridad

## General security scenario



### Source

Identified user  
Unknown user  
Hacker from outside the organization  
Hacker from inside the organization

### Stimulus

Attempt to display data  
Attempt to modify data  
Attempt to delete data  
Access system services  
Change system's behavior  
Reduce availability

### Environment

Normal mode  
Overload mode  
Reduced capacity mode  
Emergency mode  
Peak mode

### Response

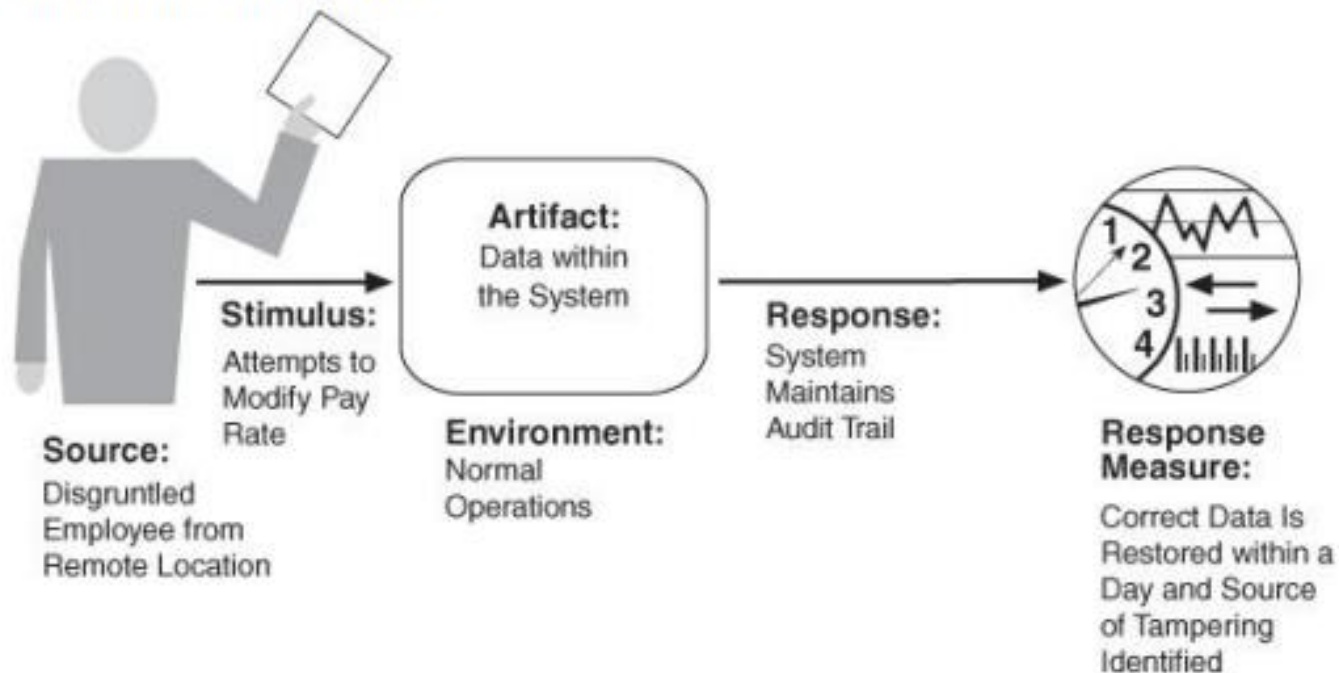
Process events  
Change level of service

### Measure

Latency  
Deadline  
Throughput  
Jitter  
Miss rate  
Data loss

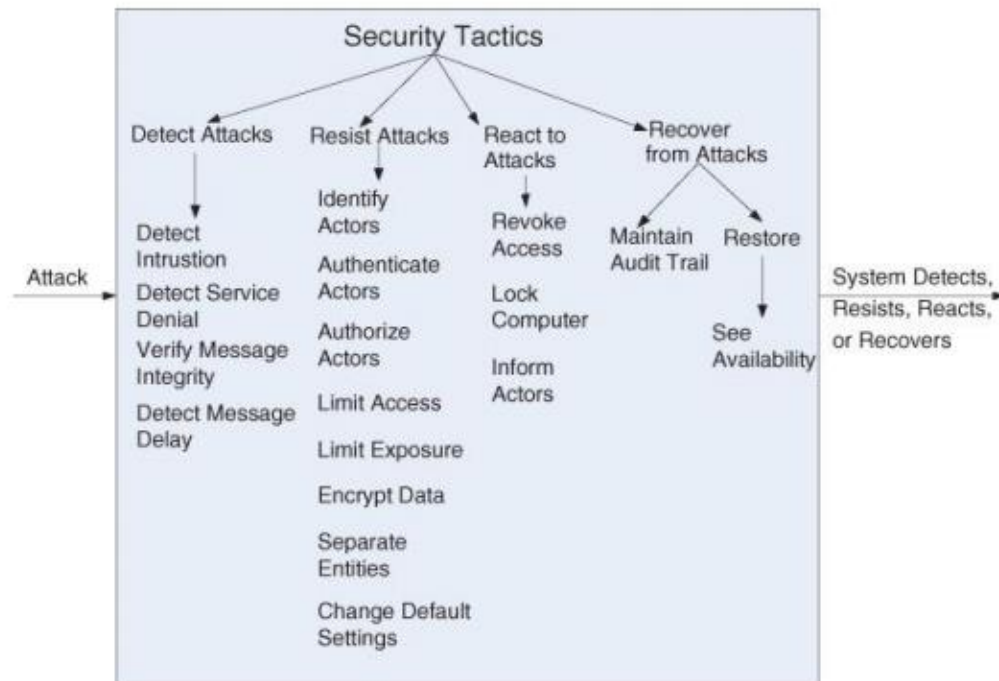
# Seguridad

## *Sample security scenario*



# Seguridad

## ***Security tactics***

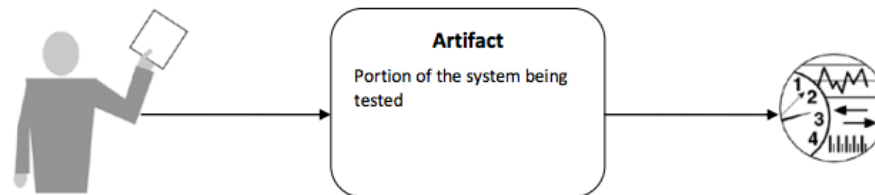


# “Testeabilidad”

- Facilidad con la cual se pueden ejecutar las pruebas en cada fase del desarrollo.
- Facilidad con la cual se pueden implementar pruebas que permitan detectar defectos.

# "Testeabilidad"

## General testability scenario



Source	Stimulus	Environment	Response	Measure
Unit tester	Execution of tests due to completion of code increment	Design time	Execute test suite & capture results	Effort to find fault
Integration tester		Development time	Capture cause of fault	Effort to achieve coverage %
System tester		Compile time	Control & monitor state of the system	Probability of fault being revealed by next test
Acceptance tester		Integration time		Time to perform tests
End user		Deployment time		Effort to detect faults
Automated testing tools		Run time		Length of longest dependency chain
				Time to prepare test environment
				Reduction in risk exposure



# Portabilidad

---

- Nivel de usabilidad del software en diferentes ambientes/entornos.

# “Trade off” de atributos de calidad

- Plataforma MMORPG on-line (1)
  - Arquitectura de una sola capa (servidor).
  - Cliente/Servidor - UDP sobre TCP/IP.
  - Protocolo propio.
- Plataforma MMORPG on-line (2)
  - Arquitectura multi-capa.
  - Cliente-servidor – HTTPS /REST

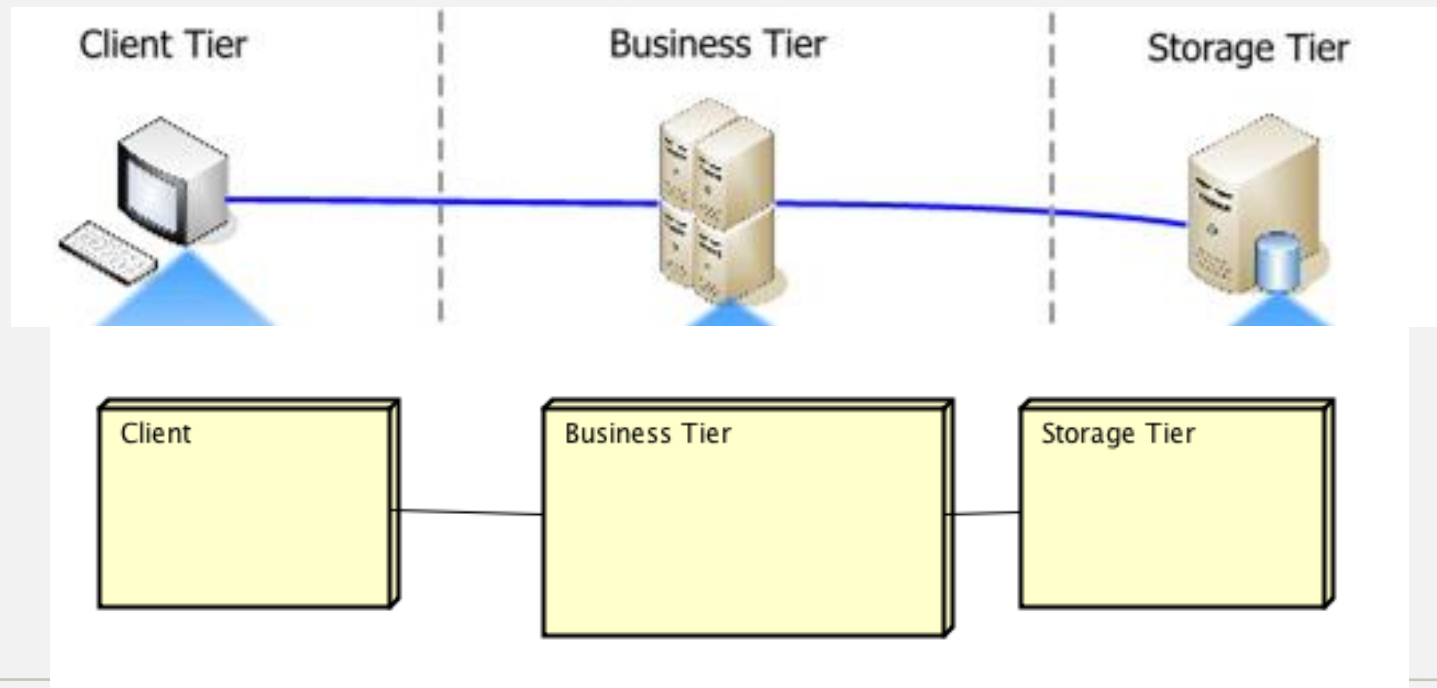
# Atributos de calidad

Escalabilidad:

- Propiedad de un sistema de manejar cantidades cada vez mayores de trabajo, o de ser fácilmente expandida, en respuesta a una demanda creciente de tráfico, procesamiento, acceso a base de datos, etc.

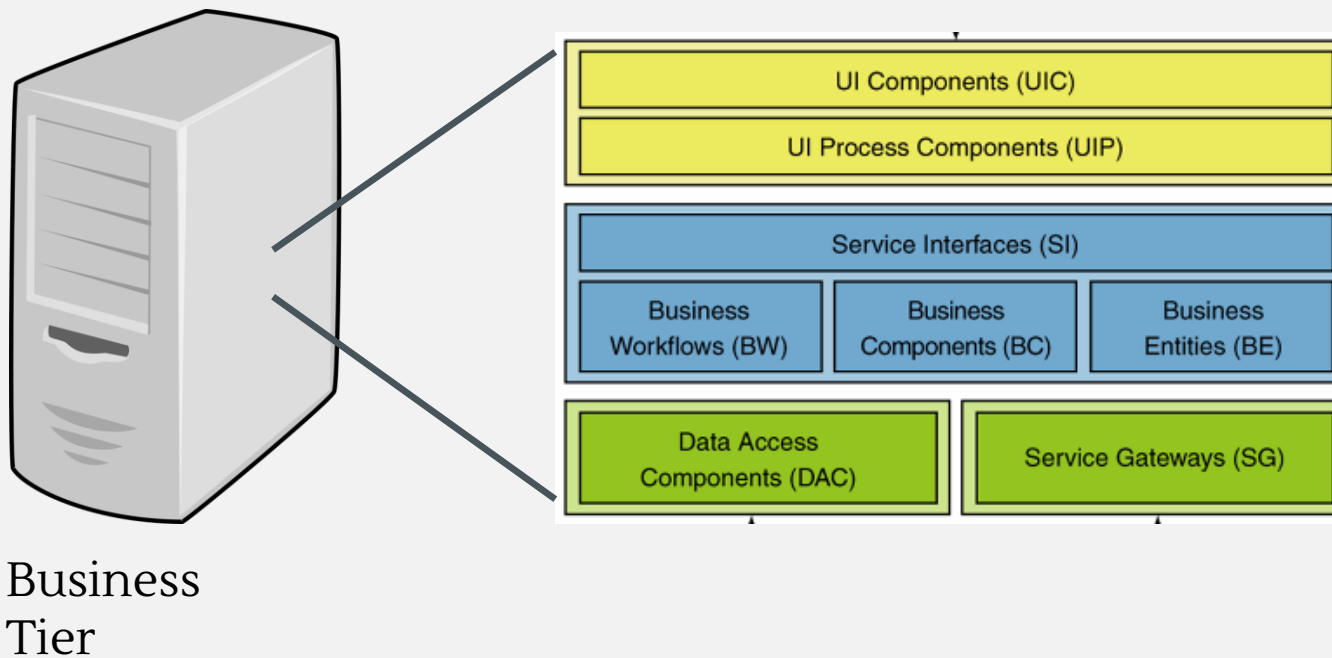
# Escalabilidad

- Capas (Layers) vs Niveles (Tiers)
- N-Tier Architecture



# Escalabilidad

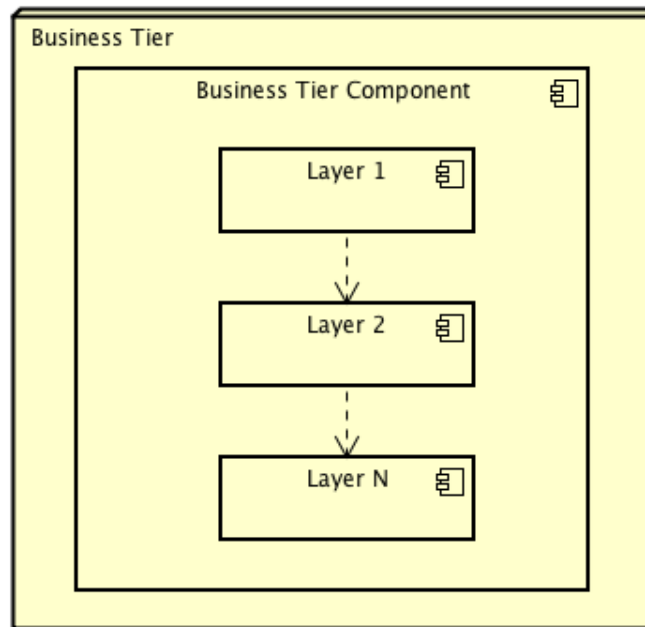
- Capas (Layers):





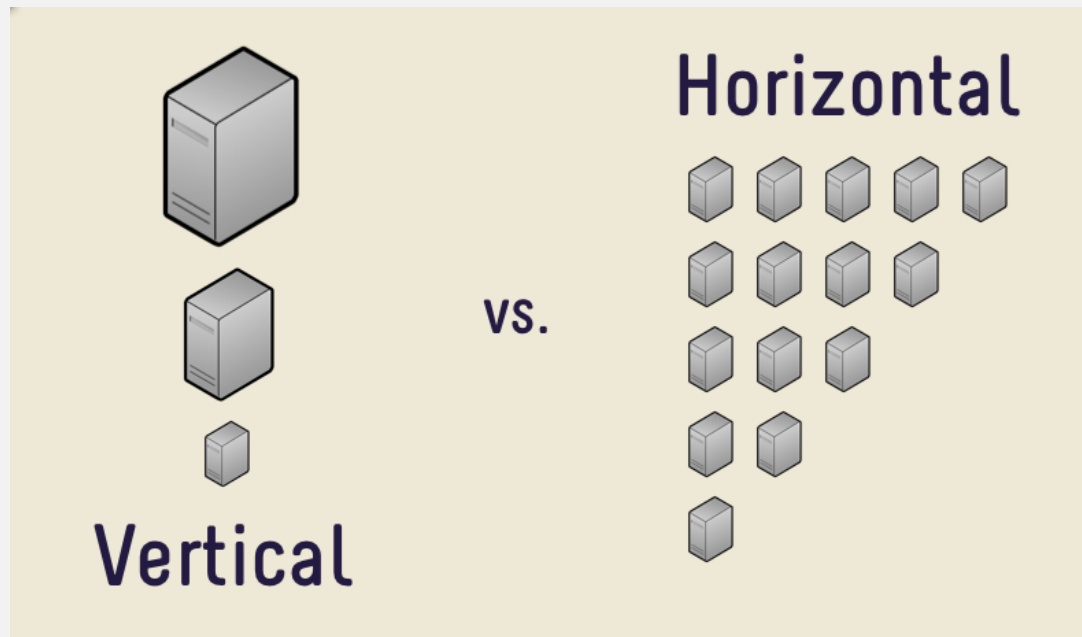
# Escalabilidad

- Capas (Layers)



# Escalabilidad

- Escalabilidad Horizontal vs Escalabilidad Vertical



# Escalabilidad Vertical

- Un sistema escala verticalmente o hacia arriba, cuando al añadir más recursos a un nodo particular del sistema, este mejora en conjunto. Por ejemplo, añadir memoria o un disco duro más rápido a una computadora puede mejorar el rendimiento del sistema global.

# Escalabilidad Horizontal

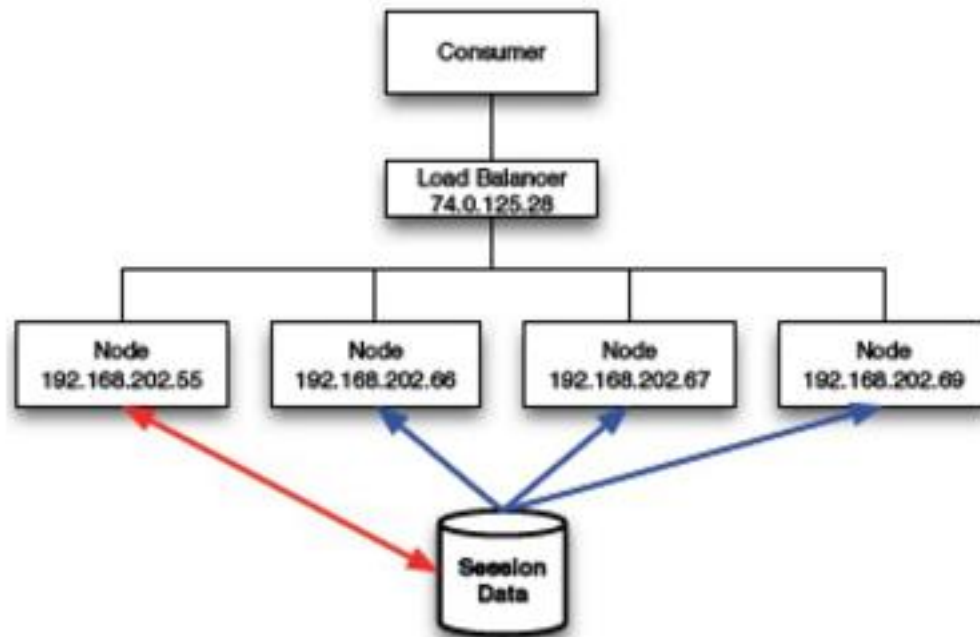
- Un sistema escala horizontalmente, cuando su capacidad puede ser extendida agregando nuevos nodos con una funcionalidad idéntica a la existente, redistribuyéndola entre todas estas.

# Escalabilidad Horizontal

- Tácticas – Escalabilidad Horizontal
  - Balanceo de Carga
  - Clustering
  - Paralelización / Procesamiento asíncrono

# Escalabilidad Horizontal

- Balanceo de carga



# Escalabilidad Horizontal

Algunas estrategias:

- Round Robin
- Por número de conexiones.
- Por tiempos de respuesta.
- Por peso (prioridad).



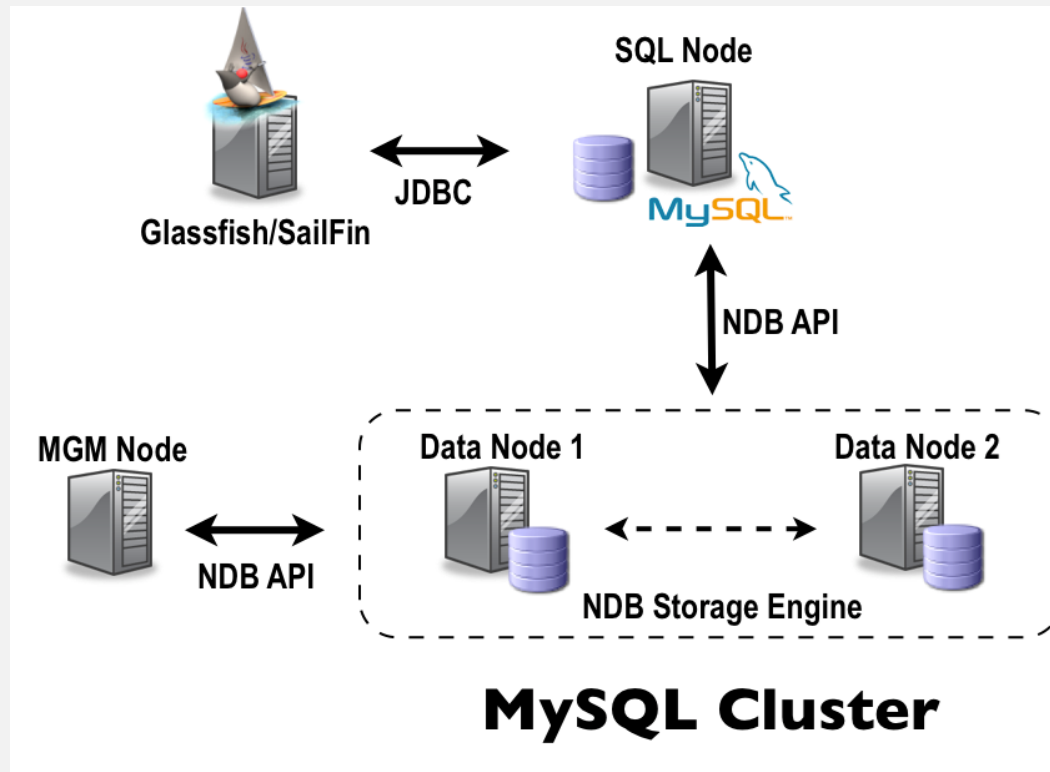
# Escalabilidad Horizontal

## Clustering.

- Cluster: Grupo de sistemas que trabaja de forma conjunta como una única unidad. Cada unidad está consciente de la existencia de las demás.
- Por lo general, maneja balanceo de carga.

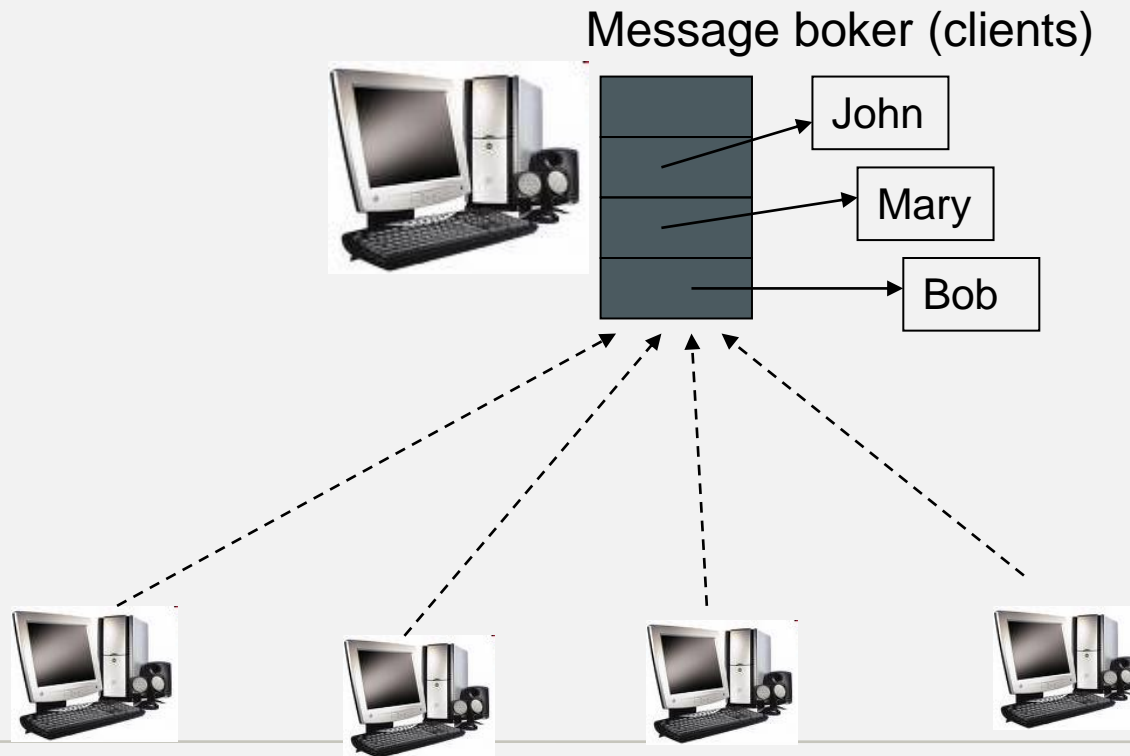
# Escalabilidad Horizontal

- Clustering.



# Escalabilidad Horizontal

- Paralelización / Procesamiento asíncrono
- Capacidad de cómputo: Colas de trabajo (procesamiento asíncrono)



# Disponibilidad

- Capacidad de un servicio, datos o sistema, de ser accesible y utilizable por los usuarios (o procesos) autorizados cuando estos lo requieran.
- Asociado a las fallas del sistema y sus consecuencias asociadas.
- Dicha falla es observable por los usuarios del sistema (humanos, otros sistemas).

# Disponibilidad vs Escalabilidad Horizontal

- SPA: Single Point of Failure?

