UML Class Diagram: Final Copy Revised
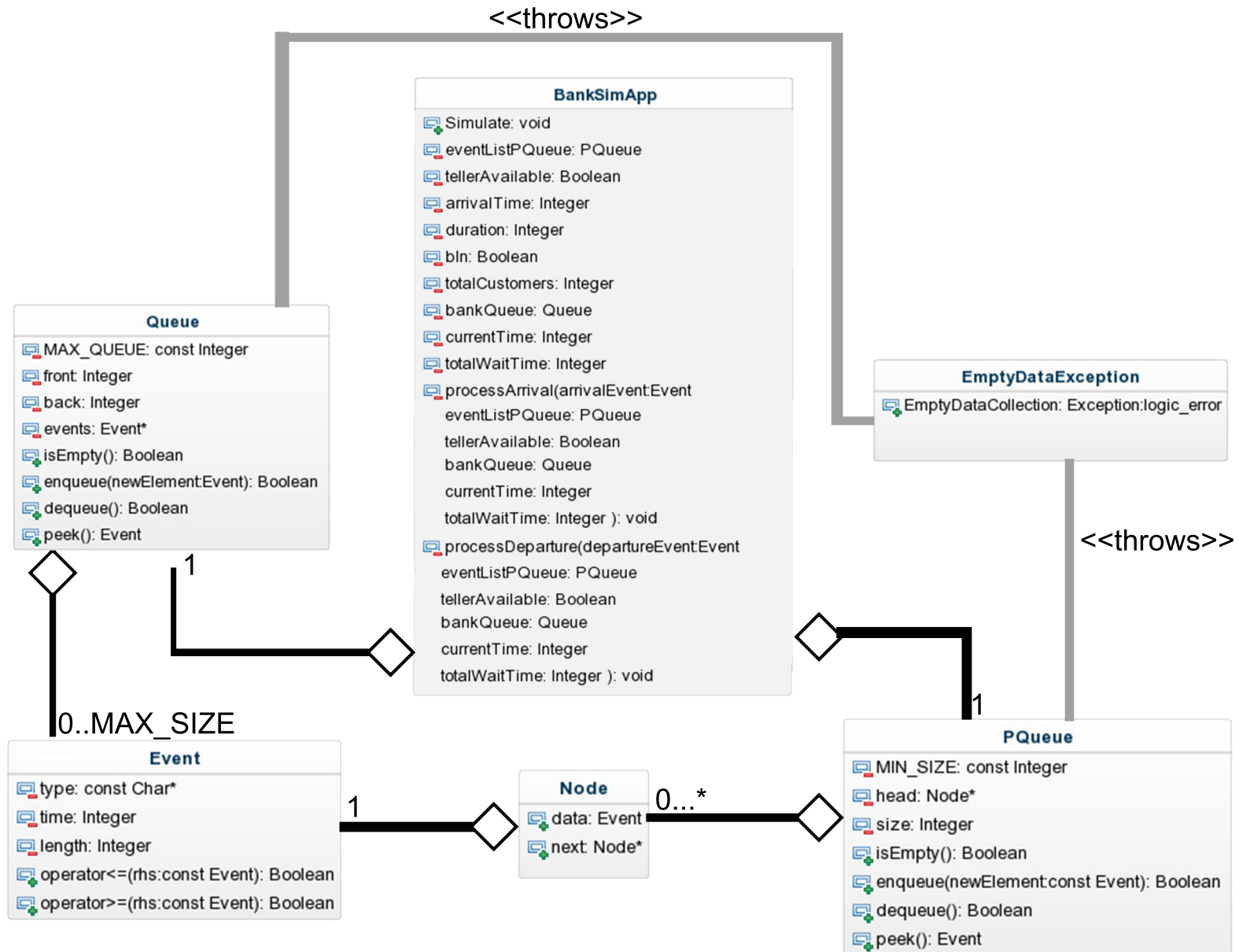## Does not include: Getters, Setters, Destructors, Constructors (intentional)
For Unrevised, see page 2.

<<throws>>

BankSimApp
- Simulate: void
- eventListPQueue: PQueue
- tellerAvailable: Boolean
- arrivalTime: Integer
- duration: Integer
- bln: Boolean
- totalCustomers: Integer
- bankQueue: Queue
- currentTime: Integer
- totalWaitTime: Integer
- processArrival(arrivalEvent:Event
  - eventListPQueue: PQueue
  - tellerAvailable: Boolean
  - bankQueue: Queue
  - currentTime: Integer
  - totalWaitTime: Integer): void
- processDeparture(departureEvent:Event
  - eventListPQueue: PQueue
  - tellerAvailable: Boolean
  - bankQueue: Queue
  - currentTime: Integer
  - totalWaitTime: Integer): void

Queue
- MAX_QUEUE: const Integer
- front: Integer
- back: Integer
- events: Event*
- isEmpty(): Boolean
- enqueue(newElement:Event): Boolean
- dequeue(): Boolean
- peek(): Event

EmptyDataException
- EmptyDataCollection: Exception:logic_error

<<throws>>

Event
- type: const Char*
- time: Integer
- length: Integer
- operator<=(rhs:const Event): Boolean
- operator>=(rhs:const Event): Boolean

Node
- data: Event
- next: Node*

PQueue
- MIN_SIZE: const Integer
- head: Node*
- size: Integer
- isEmpty(): Boolean
- enqueue(newElement:const Event): Boolean
- dequeue(): Boolean
- peek(): Event

1

0..MAX_SIZE

1

1

0...*

UML Class Diagram for Bank Simulator

By: Raj Mahey
Salman Siddiqui

EmptyDataException

+ EmptyDataCollection
Exception: logic_error

BSAPP

+ Simulate() : void
- ProcessArrival(arrival Event: Event,
     eventList PQueue: PQueue,
     BankQueue: Queue,
     tellerAvailable: Bool,
     currentTime: const int,
     totalWaitTime: float): Void
- processDeparture(departureEvent: Event,
     eventList PQueue: PQueue,
     BankQueue: Queue,
     teller available: bool,
     currentTime: const int
     totalWaitTime: float): Void

- eventList PQueue: PQueue
- tellerAvailable: bool
- arrivalTime: int
- duration: int
- bln: bool
- totalCustomers: int
- bankQueue: Queue
- currentTime: int
- totalWaitTime: int

Queue

- MAX_QUEUE: const int
- front: int
- back: int
- events: Event*
+ Queue:
+ ~Queue
+ isEmpty(): bool
+ enqueue(newElement: Event): bool
+ dequeue(): bool
+ Peek(): Event

Event

- type: const char*
- time: int
- length: int
+ Event(): void
+ Event(newtype: const char*, newTime: int)
+ Event(newtype: const char*, newTime: int, newLength: int): void
+ ~Event
+ getType(): const char*
+ setType(newtype: const char*): bool
+ getTime(): int(const)
+ setTime(newTime: int): bool
+ getLength(): int(const)
+ setLength(newLength: int): bool
+ operator<=(rhs: const Event): bool
+ operator>=(rhs: const Event): bool

PQueue

- MIN-SIZE: const int
- head: Node*
- size: int
+ PQueue: void
+ ~PQueue: void
+ isEmpty(): bool (const)
+ enqueue(newElement: const Event): bool
+ deQueue(): bool
+ getSize(): int
+ Peek(): Event

NODE

+ data: Event
+ next: node*
+ Node(): void
+ Node(theData: Event): void
+ Node(theData: Event, theNextNode: Node*): void
+ ~Node: void

<<throws>>

<<throws>>

0..1

0...MAX_SIZE

1!

1

1

0...*