

«typedef»
AccessPattern:
vector<tuple<AccessType, unsigned int>

«typedef»
DevID: unsigned int

«typedef»
Cost: unsigned int

«typedef»
LinkID: unsigned int

«typedef»
DevInfo: tuple<std::string, Cost, Cost,
double, unsigned int>

«enumeration»
AccessType

FREE
BASIC
EXPENSIVE

«enumeration»
NetworkType

PART_CONN_GRAPH
FULL_CONN_GRAPH
STAR
RING
CART

Device

- next_id: DevID
- ID: const DevID
- NAME: const string
- BAC: const Cost
- EAC: const Cost
- CAPACITY: const double
- VECTOR_LENGTH: const unsigned int

+ «constructor» Device(NULL: void*)
+ «constructor» Device(name: string, bac: Cost, eac: Cost, cap: double, veclen: unsigned int)
+ «constructor» Device(source: const Device&)
+ «constructor» Device(source: Device&&)

+ isNull(): bool
+ getID(): DevID
+ getName(): string
+ getBasicAccessCost(N: const unsigned int): Cost
+ getExpensiveAccessCost(N: const unsigned int): Cost
+ getCapacity(): double
+ getVectorLength(): unsigned int

BasicCostModel

hardware: Hardware
known_data_layouts: map<string, DataLayout>

defaultLayouts(): void

+ «constructor» BasicCostModel(hw_info: const vector<DevInfo>&)
+ «constructor» BasicCostModel(Hardware& hw)

+ getHardware(): Hardware&
+ addDataLayout(name: string, extent: unsigned int, ap: AccessPattern&): void
+ rmDataLayout(name: string): void
+ getDataLayout(NAME: const string): const DataLayout&
+ accessCost(DEV_ID: const DevID, LAYOUT: const DataLayout&, AP: const AccessPattern&, COUNT: const unsigned int): Cost
+ accessCost(DEV_ID: const DevID, LAYOUT: const DataLayout&, AP: const AccessPattern&, COUNT: const unsigned int, HARDWARE: const Hardware&): Cost
+ movementCost(DEV_SRC: const DevID, LAYOUT_SRC: const DataLayout&, DEV_DEST: const DevID, LAYOUT_DEST: const DataLayout&): Cost
+ movementCost(DEV_SRC: const DevID, LAYOUT_SRC: const DataLayout&, DEV_DEST: const DevID, LAYOUT_DEST: const DataLayout&, hardware: Hardware&): Cost
+ movementDecision(DEV_SRC: const DevID, LAYOUT_SRC: const DataLayout&, DEV_DEST: const DataLayout&, AP: const AccessPattern&, COUNT: const unsigned int): bool
+ movementDecision(DEV_SRC: const DevID, LAYOUT_SRC: const DataLayout&, DEV_DEST: const DataLayout&, AP: const AccessPattern&, COUNT: const unsigned int, hardware: Hardware&): bool
+ recommendDevice(LAYOUT: const DataLayout&, AP: const AccessPattern&, COUNT: const unsigned int): DevID
+ recommendDevice(LAYOUT: const DataLayout&, AP: const AccessPattern&, COUNT: const unsigned int, HARDWARE: const Hardware&): DevID

Note

BasicCostModel defines trivial responses to these queries.

It can be inherited from and query functions overridden as we see fit.

