

UNIVERSITA' DEGLI STUDI DI PADOVA
FACOLTA' DI INGEGNERIA

CORSO DI LAUREA TRIENNALE IN INGEGNERIA
GESTIONALE

TITOLO TESI

SERVIZIO DI PRONTO SOCCORSO, SIMULAZIONE
TRAMITE LINGUAGGIO ARENA

RELATORE: Prof. GIORGIO ROMANIN JACUR

LAUREANDO: ALESSANDRO SARDINI

ANNO ACCADEMICO: 2010 - 2011

INDICE

| | |
|--|-----------|
| Introduzione ed esposizione del problema..... | 3 |
| | |
| Capitolo 1: Servizio di Pronto Soccorso | |
| 1.Caratteristiche generali del Pronto Soccorso..... | 5 |
| 2.Comportamento del sistema in dettaglio..... | 5 |
| | |
| Capitolo 2: Il modello | |
| 1.Caratteristiche e utilizzo..... | 8 |
| 2.Inputs del modello..... | 8 |
| | |
| Capitolo 3: Implementazione del modello di simulazione | |
| 1.La simulazione al computer..... | 10 |
| 1.1.Popolarità e vantaggi..... | 10 |
| 1.2.Gli aspetti negativi..... | 11 |
| 1.3.Quando è opportuno utilizzare la simulazione..... | 11 |
| 1.4.Differenti tipi di simulazione..... | 12 |
| 2.Software di simulazione Arena..... | 13 |
| 2.1.Concetti fondamentali per la simulazione con Arena..... | 14 |
| 3.Costruzione del modello di simulazione di un servizio di Pronto Soccorso con l'utilizzo di Arena..... | 18 |
| 4.Outputs del modello..... | 29 |
| | |
| Conclusioni..... | 34 |
| | |
| Bibliografia..... | 35 |

INTRODUZIONE ED ESPOSIZIONE DEL PROBLEMA

Il Pronto Soccorso è una struttura ospedaliera che fornisce assistenza medica e chirurgica ai pazienti con malesseri e/o infortuni, che arrivano in ospedale per bisogno di immediate cure. Pertanto lo si può definire l'interfaccia tra il territorio e l'ospedale, nei casi in cui è necessario un intervento veloce ed efficace.

Quando un paziente arriva al Pronto Soccorso, un infermiere dell'ufficio accettazione gli pone alcune domande per determinare la natura e la gravità del malore o infortunio. In base alle informazioni avute, viene decisa la priorità di intervento mediante l'assegnazione di un codice colore che rappresenta il grado di urgenza. Questa classificazione è effettuata seguendo il metodo denominato "triage", dal francese "trier", che significa smistamento, classificazione.

Il tempo di attesa di un paziente è strettamente associato alla categoria del triage di cui fa parte, dalla quantità di altri pazienti che lo precedono, ma anche dalla disponibilità di medici che può variare a seconda dell'orario e del giorno (ad esempio nelle ore notturne di un giorno festivo è facile che ci sia minor assistenza medica rispetto alle ore giornaliere di un giorno feriale).

Gli arrivi dei pazienti e la durata delle visite sono caratterizzati da casualità, inoltre ci possono essere delle eventuali necessità di esami approfonditi o controlli specialistici. Il comportamento del servizio di emergenza non può essere valutato in media, ma deve essere accuratamente descritto nel tempo.

Un modello analitico tradizionale impone sostanziali semplificazioni e può dare solo risultati medi, senza mostrare le vette e le criticità che ci possono essere in una struttura ospedaliera (ad esempio carenze del servizio medico tali da comportare pericolo per la salute dei pazienti).

Di conseguenza un modello di simulazione, in grado di fornire una descrizione dettagliata nel tempo e che non trascuri nessun aspetto essenziale del sistema studiato è l'unico valido.

Il Pronto Soccorso ha una struttura complessa; un modello che simula i suoi servizi in modo specifico può richiedere una grande quantità di tempo per poter essere costruito e implementato e, una volta validato, è utile per descrivere solo i servizi specifici.

Invece un modello di simulazione flessibile e generalizzato, in grado di descrivere diversi servizi di Pronto Soccorso per mezzo di piccole modifiche, ma che non alterino in maniera sostanziale la struttura del modello, permetterebbe un considerevole risparmio di tempo.

Inoltre, se tale modello fosse implementato in un linguaggio comprensibile e leggibile, magari anche con animazione, tutte le comunicazioni con gli utenti finali del modello, come medici e dirigenti ospedalieri, sarebbero più facili.

In questo articolo è stato costruito un modello generalizzato e flessibile nei riguardi di tutte le

comuni caratteristiche strutturali e funzionali di ogni ambiente di emergenza reale. Questo modello di simulazione pur avendo una struttura estremamente semplice, prende in considerazione ogni caratteristica essenziale del comportamento di ogni reparto. Può essere facilmente adattato a diversi dipartimenti di emergenza per mezzo di poche e opportune modifiche della struttura del modello, sia per regolare strutture che accolgono grandi quantità di pazienti e che per strutture più piccole, tutto attuato con un linguaggio facile e comprensibile per l'utente finale.

CAPITOLO 1

Servizio di Pronto Soccorso

1. Caratteristiche generali del Pronto Soccorso

Un servizio di Pronto Soccorso è dedicato a fornire il primo soccorso a pazienti, alternativamente può servire:

- un ospedale,
- un gruppo di ospedali, in genere a poca distanza fra l'uno e l'altro,
- un reparto ospedaliero o un gruppo di dipartimenti ospedalieri, in base sia alle patologie (ad esempio, ostetricia, chirurgia) che al genere di pazienti (per esempio, neonatologia, pediatria, geriatria).

Il servizio è caratterizzato da elevata variabilità di arrivo del paziente, in funzione sia del tempo (ad esempio, in diverse ore di un giorno, giorni della settimana, mesi dell'anno) e sulla casualità (ad esempio, incidenti, meteo, epidemie); nonostante queste possibili variabili, si deve garantire l'importante requisito di generare un responso veloce (a volte immediato), anche in caso di congestione di pazienti.

A causa delle posizioni sopra citate, un servizio di emergenza deve essere correttamente progettato e gestito tenendo in considerazione le risorse a disposizione e il relativo utilizzo. Al fine di fornire un servizio di alta qualità con un accettabile costo, in modo che nessuna risorsa sia sottoutilizzata, dobbiamo considerare:

- le strutture (sale di cura principali o secondarie, sale d'attesa, camere ad ammissione a breve termine),
- le risorse tecnologiche (strumentazioni specifiche),
- le risorse umane (medici, infermieri, personale ausiliario, ingegneri e le relative norme di lavoro: orari, turni di servizio, ecc),

Come sottolineato in precedenza, bisogna ottenere un dimensionamento corretto del servizio sia nella fase di progettazione, quando è necessario costruire il giusto numero di strutture e fornire la giusta quantità di strumenti, sia in fase di lavorazione, quando il servizio è operativo e il sistema si deve comportare nel modo adeguato per sopperire alle necessità.

2. Comportamento del sistema in dettaglio

Tutti i pazienti che arrivano sono selezionati dallo stesso unico filtro, il "triage", governato da infermieri specializzati. La selezione si basa sull'analisi dei parametri vitali, predefiniti e universali, ossia: coscienza, respirazione e circolazione del sangue.

Il personale preposto effettua questa prima rilevazione e assegna ad ogni paziente un codice colore secondo l'urgenza:

- codice rosso: quando almeno uno dei tre parametri di vita è alterato, l'intervento deve essere immediato e ogni attesa può essere causa di decesso;
- codice giallo: l'aiuto non deve essere ritardato oltre i 10 - 20 minuti; aspettare può influenzare la stabilità del paziente e aumentarne la gravità;
- codice verde: l'aiuto può essere ritardato; è consentita un'attesa limitata (fino a due-tre ore) senza pericolo, ma l'assistenza non può essere fornita al di fuori dell'ospedale;
- codice bianco: nessun pericolo di vita; qualsiasi attesa non può causare danno e l'assistenza può essere fornita al di fuori dell'ospedale (ad esempio, da un medico generico o in ambulatori specialistici)

Si noti che la classificazione con quattro codici colore, con possibili piccole variazioni, è universalmente adottata.

Se durante l'operazione di triage l'infermiere specializzato decide che il paziente è un codice rosso, questo viene immediatamente inviato a una sala di cura per essere assistito esercitando diritto di prelazione (interruzione di servizio) su pazienti di qualsiasi altro codice ogni volta che tutte le risorse sono impegnate.

Se il paziente non è un codice rosso, viene effettuata una breve intervista per raccogliere dati personali e informazioni circa l'incidente, quindi riceve, se necessario, una prima assistenza, (ad esempio una fasciatura). Possono essere compiuti inoltre alcuni controlli, come la pressione arteriosa, la temperatura corporea, l'elettrocardiogramma e infine, se queste misurazioni risultano adeguate, il paziente è messo in una coda di priorità e attende in una stanza sotto l'osservazione dell'infermiere di triage, in modo che qualsiasi eventuale aumento di gravità venga immediatamente rilevato. Non appena le risorse per l'assistenza (in genere una stanza, un medico e un infermiere) sono a disposizione, e secondo la priorità (regola FIFO in caso di codice di colore identico) i pazienti sono chiamati per il controllo medico.

I pazienti vengono così assistiti secondo la seguente programmazione:

- prima ispezione con stabilizzazione delle condizioni del paziente e assegnazione della terapia quindi si decide se dimettere il paziente, ricoverarlo in ospedale, ammetterlo in un ricovero a breve termine o se è opportuno che sostenga degli ulteriori esami.
- Esami interni e/o esterni, possono venire richiesti dall'ispezione. Questi sono, per lo più, esami di laboratorio (ad esempio, l'urina, ematochimico, ecc), esami strumentali specializzati (come radiografie, ecografie, tomografia assiale computerizzata, ecc) e infine esami di consulenza e/o assistenza (come consulenza cardiologica, assistenza ortopedica, ecc). Questi possono essere richiesti ed eseguiti, sia all'interno del reparto, sia in un laboratorio ospedaliero, sia in un dipartimento esterno.
- Seconda ispezione dopo i risultati degli esami qui si decide tra la dimissione del paziente, l'ammissione ad un reparto ospedaliero o a un ricovero a breve termine e sulla necessità di

effettuare ulteriori esami di accertamento.

Ogni servizio di Pronto Soccorso è dotato di un reparto di ammissione a breve termine, dove i pazienti possono essere accettati per un breve periodo di tempo, generalmente fino a 36 ore, durante il quale il paziente viene periodicamente osservato. La stanza è generalmente controllata da un medico o da un infermiere.

- Terza ispezione dopo l'ammissione a breve termine e successiva decisione su dimissione, ricovero in ospedale o necessità di effettuare ulteriori controlli.

La morte può interrompere la sequenza in qualsiasi momento, in particolare per i pazienti con codice rosso.

Le risorse impegnate per le cure ai pazienti (medici, infermieri, eventuali ingegneri, stanze e strumenti necessari) vengono assortite in modi diversi a seconda della situazione. Ogni reparto ospedaliero può richiedere più o meno personale e attrezzature strumentali diverse. La scelta di risorse può dipendere dalla gravità del paziente o dal tipo di malessere (se derivante da un malore o se in seguito a un trauma). Il numero totale delle risorse a disposizione può variare nel corso di una giornata, in corrispondenza alle diverse richieste (tipicamente minori durante la notte): di norma alcune stanze possono essere aperte o chiuse, e il numero di medici aumentato o diminuito, a seconda dei turni di servizio.

Alcuni dipartimenti di emergenza si occupano anche della gestione del servizio ambulanze. Un medico in servizio è inviato a soccorrere i pazienti codice rosso sul luogo dell'incidente, mentre un infermiere è inviato per i pazienti codice giallo: in questi casi la prima visita di controllo si compie durante il viaggio in ambulanza.

CAPITOLO 2

Il modello

1. Caratteristiche e utilizzo

Lo scopo dell'articolo è quello di costruire un modello generalizzato e flessibile, in grado di descrivere correttamente molti servizi di Pronto Soccorso esistenti, attraverso alcuni adeguamenti della struttura e dei parametri del modello.

La sua flessibilità permette inoltre di poter regolare i diversi servizi di emergenza reale sia nelle condizioni di normalità sia in condizioni di grande variabilità del numero di pazienti in ingresso e/o della regolamentazione del lavoro.

Gli outputs del modello possono essere utilizzati per descrivere il comportamento del sistema, per verificare le sue prestazioni e per evidenziare eventuali difetti, suggerendo così utili modifiche nell'uso delle risorse al fine di aumentarne efficacia ed efficienza.

Se consideriamo che il modello:

- ha un numero elevato di variabili stocastiche (per esempio, tempi di interarrivo, durate delle ispezioni, probabilità delle decisioni, ecc),
- deve tener conto delle complesse interazioni tra le attività del modello (ad esempio, priorità, prelazione, ecc),
- deve tener conto di un numero elevato di variabili di controllo (ad esempio, numero di medici, infermieri, personale ausiliario e ingegneri in servizio nei loro relativi turni e del numero di camere a disposizione per ospitare i pazienti),
- deve fornire diverse modalità di risultati (ad esempio, tempi di attesa del paziente, carico di lavoro del personale, utilizzo sale, ecc) e di questi risultati si devono conoscerne l'andamento in media ma anche l'andamento nei picchi,

allora è immediato rendersi conto che un modello analitico non è sufficiente e, pertanto, si deve adottare un modello di simulazione.

Tale simulazione deve essere caratterizzata da un alto livello di dettaglio, al fine di non trascurare alcun aspetto fondamentale del comportamento del sistema, inoltre deve essere implementata in un linguaggio comprensibile, in modo da agevolarne l'utilizzo finale.

2. Inputs del modello

Per l'implementazione del modello si devono tenere in considerazione tre categorie di input:

- caratteristiche dell'arrivo dei pazienti,
- caratteristiche di funzionamento del servizio,
- comportamento del servizio risorse.

Gli arrivi dei pazienti possono venir definiti a partire dai dati effettivi di arrivo per un certo

numero di intervalli, durante i quali la media del tempo di interarrivo è costante. Quindi è sufficiente fornire una matrice in cui, per ogni codice, vi sono riportati tutti gli intervalli di giorno e dei relativi tempi di interarrivo.

Questi parametri dipendono dal numero di abitanti della regione che potrebbero far uso del servizio e, eventualmente, dalla stagione dell'anno (ad esempio, ci saranno maggiori arrivi durante l'inverno o durante le epidemie).

E' ovvio che ogni servizio di emergenza è caratterizzato da una matrice di arrivi diversa.

Per quanto riguarda le caratteristiche funzionali di servizio, abbiamo da descrivere tutte le regole di servizio, i tempi e le probabilità che definiscono i movimenti dei pazienti.

Per ogni gruppo di pazienti (rosso, giallo, verde, bianco) viene solitamente dato:

- il tempo medio e le risorse di triage,
- la durata media e le risorse richieste per l'ispezione,
- la probabilità di morte durante tutte le fasi del flusso,
- la probabilità di rilascio e di ricovero ospedaliero,
- probabilità di ammissione a breve termine e durata media del ricovero,
- probabilità che venga richiesta la necessità di sostenere esami interni e/o esterni e la loro durata media .

Per quanto riguarda le risorse di un servizio di Pronto Soccorso bisogna considerare:

- il numero e il tipo di camere a disposizione nei vari intervalli della giornata,
- il numero di medici, infermieri, ingegneri a disposizione nei vari intervalli di una giornata,
- gli orari e la durata dei turni di servizio.

Tutti questi dati possono essere raccolti in matrici che dipendono, ovviamente, dal servizio specifico che si vuole analizzare.

I parametri di distribuzione di probabilità e i valori di probabilità sono ottenuti dalle registrazioni effettuate sul dipartimento, in generale sulla base di dati informatici raccolti dai servizi precedenti.

CAPITOLO 3

Implementazione del modello di simulazione

1. La simulazione al computer

Una simulazione al computer si riferisce ai metodi per lo studio di una grande quantità di modelli di sistemi reali, per i quali è necessaria una valutazione numerica mediante l'utilizzo di software che imitano le operazioni e le caratteristiche del sistema.

Da un punto di vista pratico, la simulazione è il processo di progettazione e creazione di un modello computerizzato di un sistema reale, effettuata in termini di *relazioni logico-matematiche* tra le variabili, allo scopo di eseguire esperimenti numerici che ci diano una migliore comprensione del funzionamento di quel sistema, a partire da un dato insieme di condizioni.

Benchè possa essere utilizzata anche per lo studio di sistemi semplici, la vera forza di questa tecnica è raggiunta pienamente per l'analisi di sistemi complessi.

Sebbene la simulazione non debba essere l'unico strumento da utilizzare per lo studio di un modello, esso è frequentemente il metodo preferito. La ragione di ciò deriva dal fatto che il modello di simulazione può diventare abbastanza complesso se necessita di rappresentare fedelmente il sistema, ma nonostante ciò si può ancora velocemente analizzare la simulazione. Il vantaggio principale della simulazione è che gli esperimenti possono essere completamente controllati e le prestazioni di un sistema tutte osservate.

1.1. Popolarità e vantaggi

Negli ultimi 30 anni, la simulazione è divenuta la metodologia di analisi più apprezzata per lo studio di sistemi semplici e complessi. La principale ragione della popolarità della simulazione è da ricercarsi nella capacità di trattare complicati modelli di sistemi reali altrettanto complicati. Questo lo rende uno strumento molto potente e versatile. La simulazione è inoltre uno strumento che aiuta a risolvere problemi, aiutando a prendere la "decisione giusta" evitando inutili e/o costosi tentativi di realizzazione.

Un'altra ragione dell'aumento di popolarità della simulazione è certamente l'incremento del rapporto qualità/prezzo dei computer, il che ha permesso di ridurre drasticamente i costi effettivi di ciò che solo pochi anni prima era senza dubbio molto oneroso. Alla fine, i passi avanti fatti nella potenza, nella flessibilità e nella facilità di utilizzo dei software di simulazione hanno mosso l'approccio dal regno del basso livello di programmazione, incline all'errore, al campo del rapido e valido *decision making*. La previsione è che la popolarità e l'efficacia della simulazione cresca sempre più, proporzionalmente allo sviluppo dei sistemi hardware e software.

1.2.Gli aspetti negativi

La simulazione non è però il paradiso, anzi. Poiché molti sistemi reali sono affetti da variabili d'ingresso incontrollabili, queste si propagano nel modello di simulazione fino a raggiungere l'uscita (*modello stocastico*). Ad esempio, il modello di un centro di distribuzione avrebbe arrivi, partenze e quantità di lotti aleatorie, in accordo alle particolari distribuzioni di probabilità, che si propagheranno lungo il modello logico fino all'output, causando misurazioni di performance (come il tempo di ciclo) anch'esse aleatorie. Così, avviando la simulazione stocastica, una volta eseguendo un esperimento fisico aleatorio, una volta guardando il centro di distribuzione per un singolo giorno, è come se vedessimo qualcosa di diverso ogni volta, pur senza cambiare nulla all'interno del modello.

In molte simulazioni, quando il tempo di simulazione diventa più lungo (ad esempio mesi piuttosto che giorni), molti risultati tendono a stabilizzarsi verso un valore medio, divenendo in tal modo meno variabili. Il problema è che risulta molto difficile determinare quanto lungo possa essere tale periodo, inoltre il modello dovrebbe imporre un termine alla simulazione (ad es. una banca apre alle 9 e chiude alle 17), per cui imporre un periodo di simulazione più lungo può risultare inappropriato.

Quindi si deve prestare molta attenzione alle fasi di progettazione ed analisi degli esperimenti di simulazione per tener conto delle incertezze dei risultati, soprattutto se il tempo adatto per il modello è relativamente breve.

Anche se i risultati della simulazione sono spesso incerti, possiamo comunque trattare, quantificare e ridurre questa incertezza. L'incertezza potrebbe essere completamente eliminata assumendo forti semplificazioni per il sistema, in modo da ottenere un modello di semplice fattura che ci dia risultati non aleatori. Sfortunatamente, però, un simile modello semplificato non può essere assunto come una valida rappresentazione del sistema reale e l'errore dovuto a tale modello non può essere facilmente calcolato e ridotto.

È dunque sempre preferibile avere una risposta sbagliata al problema giusto piuttosto di una risposta giusta al problema sbagliato.

1.3.Quando è opportuno utilizzare la simulazione

Non sempre è opportuno utilizzare la simulazione per l'analisi del sistema in studio.

I motivi che spingono alla scelta della simulazione piuttosto che all'utilizzo di modelli fisici o matematici, sono:

- _ Adattabilità
- _ Sicurezza
- _ Flessibilità
- _ Minor costo rispetto ai modelli fisici (l'uso di prototipi, quando realizzabili, è in genere più

- costoso, più rischioso e richiede più tempo)
- _ Il sistema reale non esiste oppure è costoso (in tempo e/o denaro), pericoloso, o impossibile effettuare esperimenti mediante prototipi. (Ad es., progetto di computer, sistema solare, reattori nucleari)
 - _ Il training con sistemi reali è costoso, pericoloso o può causare seri danni (Ad es., sistemi di trasporto, sistemi di produzione, macchinari complessi, aeromobili, impianti nucleari, etc...)
 - _ Maneggevolezza (è lo stesso utente che introducendo i parametri può verificare immediatamente gli effetti sui vari scenari)
 - _ Necessità di studiare il comportamento passato, presente e futuro del sistema in tempo reale, tempo espanso o tempo compresso (sistemi di controllo real-time, crescita della popolazione, effetti collaterali di nuovi medicinali, ecc.)
 - _ Il modello matematico del sistema non esiste (esplorazioni petrolifere, meteorologia, economia mondiale, conflitti internazionali, reti di calcolatori, ecc.)
 - _ I modelli matematici non hanno una soluzione analitica o numerica semplice e praticabile (equazioni differenziali non lineari, problemi stocastici, ecc.)

1.4. Differenti tipi di simulazione

Le simulazioni si possono classificare nei seguenti modi:

Statici vs. Dinamici; Nei primi abbiamo che il tempo non ha nessun ruolo, mentre nei secondi il tempo è di primaria importanza in quanto il sistema stesso dipende dalla configurazione degli istanti di tempo prima. Molti simulatori si concentrano su questi ultimi.

Continui vs. Discreti. Nei modelli continui lo stato del sistema può cambiare con continuità (portata di liquido in un tubo). Nei modelli discreti invece può cambiare solo in determinati istanti di tempo (sistema manifatturiero con ingressi ed uscite di merci in determinati istanti di tempo). Si possono avere anche modelli continui-discreti ossia aventi sia elementi continui che discreti.

Deterministici vs. Stocastici: I primi non hanno ingressi casuali, i secondi invece lo hanno (esempio i clienti che arrivano ad uno sportello di servizio).

Un modello può avere sia ingressi casuali sia ingressi deterministici nelle differenti componenti: quali siano gli elementi trattati come deterministici e quali siano trattati come casuali dipende dal realismo del modello. Visto che l'aleatorietà delle variabili in un qualsiasi sistema reale è sempre presente, i modelli stocastici sono più idonei a modellare sistemi complessi dove il fenomeno aleatorietà è consistente.

Il processo di costruzione di strutture sanitarie porta con sé un'esigenza prioritaria: la razionalizzazione delle proprie attività. La non comprensione profonda del comportamento

del sistema implica perdere l'opportunità di utilizzare al meglio le risorse e di fornire livelli di servizio adeguati agli utenti.

Perché in ambito sanitario è necessaria l'ottimizzazione e strumenti operativi di supporto alle decisioni?

- Ospedali, ambulatori, laboratori sono sistemi costosi complessi, soggetti ad emergenze.
- La gestione richiede scelte che tengano conto di molti fattori non facilmente valutabili.

Attraverso i modelli di simulazione è possibile:

- valutare i potenziali risparmi nell'adozione di nuove tecniche gestionali;
- stimare il fabbisogno di posti letto;
- valutare la congestione della struttura ospedaliera all'aumentare dei ricoveri;
- ottimizzare gli spazi a disposizione;
- schedulare gli interventi a seconda delle disponibilità delle sale operatorie;
- dimensionare in modo adeguato le sale d'attesa;
- testare le procedure di emergenza;
- l'assenza di rischi per i pazienti;
- la possibilità di mettere in pratica la gestione di procedure ed eventi routinari, così come il management di eventi meno comuni ma pericolosi per la vita dei pazienti;
- Garantire livelli di servizio;
- Utilizzare al meglio macchinari e sale operatorie;
- . Schedulare attività e personale;
- Gestire laboratori e macchinari;
- Fornire risposte tempestive e giustificare azioni e visioni strategiche;
- . Gestire liste d'attesa;
- Anticipare problemi e valutare rischi legati a cambiamenti e situazioni di emergenza;
- Introdurre nuove procedure o modificare procedure esistenti;
- . Disporre delle informazioni in modo aggregato, sintetico e velocemente;

2. Software di simulazione Arena

L'implementazione del modello è stata effettuata al calcolatore mediante il pacchetto software Arena Simulation della Rockwell Automation.

Arena è un potente strumento che permette di creare modelli ed eseguire simulazioni del modello stesso, con un buon compromesso tra livello di semplicità e velocità di comprensione. Qualsiasi ambiente di business, dal servizio clienti alla produzione

all'assistenza sanitaria, possono trarre vantaggio dalla simulazione. Se si sta analizzando una catena di fornitura o un nuovo servizio di Pronto Soccorso, è necessario seguire cinque semplici passaggi:

1. Creare un modello di base. Arena offre un'interfaccia intuitiva, basata sulla costruzione di un diagramma di flusso. È sufficiente trascinare i moduli Arena (le forme nel diagramma di flusso) nella finestra del modello e collegarli a definire il flusso di processo.
2. Perfezionare il modello. Aggiungere i dati reali (ad esempio, tempi di processo, requisiti delle risorse, livelli di personale, ecc) con un doppio clic sui moduli e aggiungere informazioni con i moduli *data* di Arena. Si può inoltre creare un quadro più realistico del sistema aggiungendo degli elementi grafici di animazione che Arena già fornisce automaticamente.
3. Simulare il modello. Eseguire la simulazione per verificare che il modello rifletta adeguatamente il sistema attuale. Identificare i “colli di bottiglia” e controllare visivamente l'andamento delle entità e risorse attraverso le dinamiche di animazione grafica di Arena.
4. Analizzare i risultati della simulazione. Arena fornisce report automatici sui criteri decisionali comuni, come l'utilizzo delle risorse e tempi di attesa. Questi vengono presentati con tabelle e grafici in modo da poter essere consultati e analizzati facilmente.
5. Selezionare l'alternativa migliore. Apportare modifiche al modello al fine di valutare tutti i possibili scenari alternativi, quindi confrontare i risultati per trovare la migliore soluzione.

2.1. Concetti fondamentali per la simulazione con Arena

Esistono varie componenti che strutturano un modello concepito per la simulazione.

Esamineremo ciascun elemento singolarmente.

Le Entità

In gran parte delle simulazioni sono presenti le entità. Questi oggetti si spostano all'interno del sistema durante la simulazione variando i valori e lo stato del sistema stesso. Essi rappresentano gli oggetti dinamici della simulazione. Dopo un periodo di tempo queste entità devono lasciare il sistema. Tutte le entità devono essere definite, o dall'utente o dal software automaticamente, e la loro presenza è fondamentale.

Esistono situazioni dove sono presenti dell'entità fittizie in quanto non esiste concretamente nel sistema l'oggetto rappresentato dalle entità; tipico esempio può essere un guasto, anche

se nella realtà non è sempre presente va tenuto conto nel modello da simulare. Nel simulare un modello il primo passo è capire e distinguere quali sono le entità in gioco.

Attributi

Per distinguere le varie entità presenti all'interno di un sistema dobbiamo associare a ciascuna tipologia di essi un attributo: all'interno della stessa classe di attributi ciascuna entità avrà un valore specifico. Per esempio ipotizziamo che le entità del nostro modello sono 3: la data, il colore, la priorità; all'interno di ciascuna entità avremo attributi univoci tipo: 10 novembre, rosso, basso, ecc...

Variabili Globali

Una variabile globale è un'informazione che rispecchia una caratteristica dell'intero sistema. Le variabili possono essere sia costanti che funzioni di entità, ad esempio pezzi presenti in una sub-area del modello. Le variabili possono essere anche vettori o matrici.

Risorse

Le entità del nostro sistema sono spesso in conflitto tra loro per l'ottenimento dei servizi offerti dalle risorse del sistema, tipo personale, attrezzatura, spazio in un magazzino limitato, ecc... Un'entità occupa una risorsa per poi liberarla quando ha terminato di sfruttare i suoi servizi. È la risorsa ad essere ceduta all'entità e non viceversa perché un'entità può usufruire di più risorse (attrezzatura e personale assieme).

Una risorsa rappresenta una tipologia di server, ognuno dei quali costituisce un'unità di quella determinata risorsa.

Code

Quando un'entità non può occupare un'unità di risorse perché quest'ultima è occupata da un'altra entità, l'entità necessita di un posto dove attendere e questo posto è detto coda. Durante il processo di sviluppo di un modello bisogna definire il comportamento del sistema nel momento che un'entità arrivando ad una coda la trovi piena.

Accumulatori Statistici

Per il calcolo della prestazione del sistema è necessario tenere traccia di diverse variabili di tipo accumulatori statistici mentre la simulazione procede. Queste variabili possono essere:

- Il numero di pezzi prodotti fino ad un dato momento
- Il tempo di attesa totale nelle code
- Il tempo massimo di attesa
- Il tempo totale all'interno del sistema di tutti i pezzi che sono usciti dal sistema.

Tutti gli accumulatori statistici devono essere iniziati al valore 0. Quando accade un evento straordinario (guasto, manutenzione, fine processo, inizio processo, ecc.) gli accumulatori ne devono tener conto secondo la logica prefissata.

Eventi

Un evento è qualcosa che può succedere in un istante (di simulazione) e che può cambiare attributi, variabili o accumulatori statistici; ecco degli esempi:

Arrivo - quando un nuovo pezzo entra nel sistema

Allontanamento - quando un pezzo lascia il sistema

Fine - fine della simulazione.

Tempo di simulazione

Il valore di tempo di una simulazione è mantenuto in una variabile dal nome simulation clock.

Il tempo in una simulazione non è continuo, ma considera solo istanti di tempo dove accade un evento, siccome nel tempo intermedio il sistema rimane inerte.

Per la costruzione del modello si fa ricorso a due tipologie di moduli:



-Moduli *Flow-chart*: vengono inseriti nella finestra del modello e collegati a formare un diagramma di flusso descrivendo la logica del processo. Questi a loro volta si dividono in:

- *modulo create*, inteso come punto di partenza per le entità in un modello di simulazione. Le entità sono create utilizzando una pianificazione o sulla base di un tempo tra gli arrivi; queste allora lasceranno il modulo per essere poi processate lungo il sistema;
- *modulo dispose*, questo modulo è inteso come il punto finale per le entità in un modello di simulazione. Qui vengono registrate le statistiche di processo per le entità prima che queste vengano smaltite;
- *modulo process*, ovvero il processamento di un'entità. Rappresenta un'attività svolta dalle entità che lo attraversano.

Può essere di tre tipi:

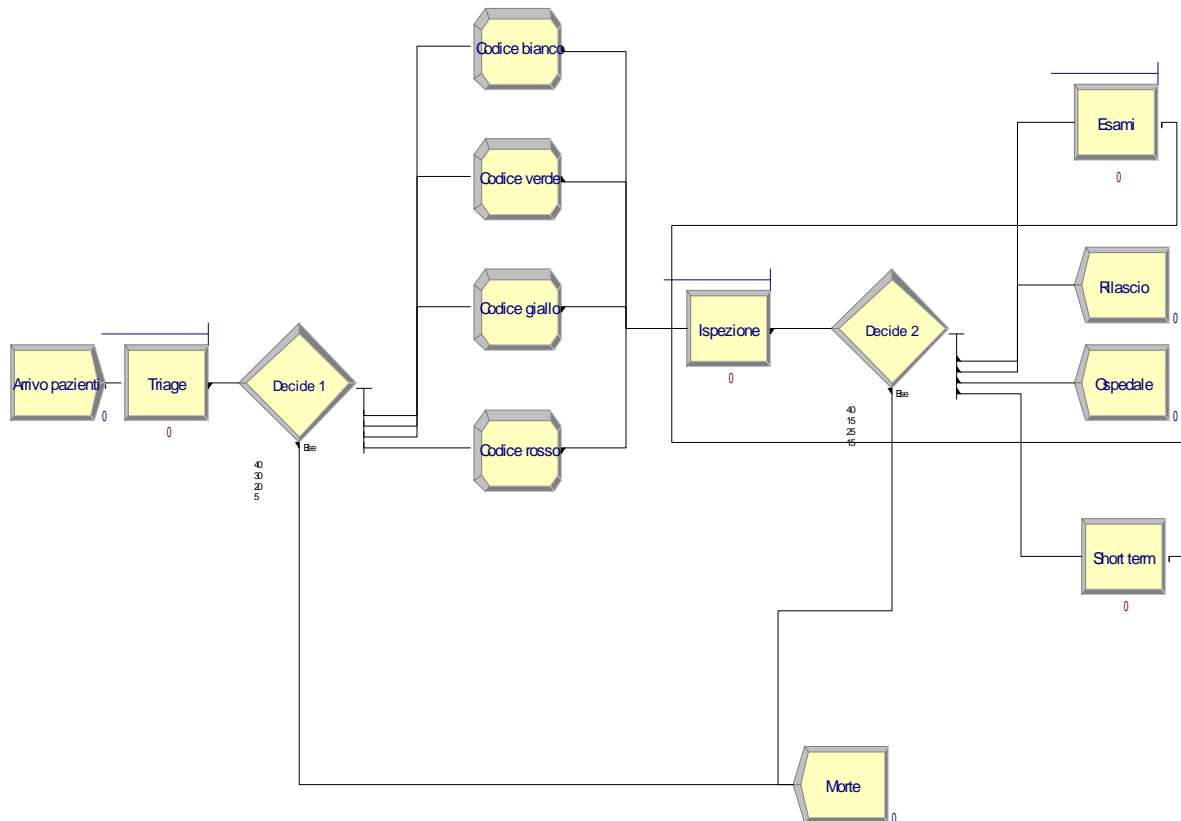
- 1.delay: è richiesto un tempo di processamento ma nessuna risorsa.
- 2.seize delay: è richiesto un tempo di processamento e una risorsa che viene allocata ma non rilasciata;
- 3.seize delay release: è richiesto un tempo di processamento, trascorso il quale una risorsa precedentemente allocata è rilasciata;
- *modulo assign*. Questo modulo viene utilizzato per assegnare nuovi valori alle variabili o nuovi attributi e immagini alle entità.
- *modulo decide*. Permette di implementare processi di decisione nel sistema. Esso include opzioni per prendere decisioni sulla base di una o più condizioni o probabilità.



-Moduli *Data*. Non vengono inseriti nella finestra del modello. Descrivono il sistema staticamente e si distinguono in:

- modulo *attribute*. Questo modulo data viene usato per definire il tipo di attributo per entità, dimensioni e valore iniziale. Si possono riassegnare nuovi valori agli attributi con il modulo *assign*;
- modulo *entity*. Viene usato per definire i vari tipi di entità e la loro immagine raffigurante durante la simulazione;
- modulo *queue*. È possibile utilizzarlo per modificare le regole di disciplina di attesa di una specifica coda. La coda si presenta all'imbocco di un modulo process e può seguire la logica first in first out, last in last out, più piccolo valore di attributo first, più grande valore di attributo first;
- modulo *resource*. Questo modulo data definisce le risorse nel sistema di simulazione, comprese informazioni sui costi e sulla disponibilità;
- modulo *schedale*. Questo modulo viene utilizzato in collaborazione con il modulo *resource* per definire un programma operativo per una risorsa o con il modulo *create* per definire un programma di arrivo;
- modulo *variable*. Usato per definire le dimensioni e valori di una variabile.

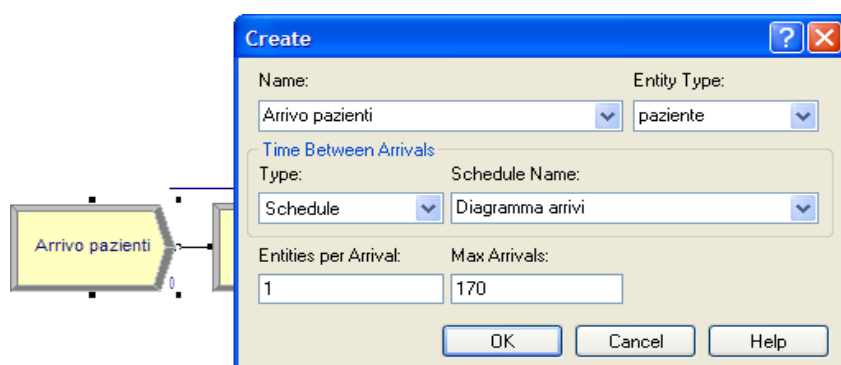
3.Costruzione del modello di simulazione di un servizio di Pronto Soccorso con l'utilizzo di Arena



Modello di un servizio di Pronto Soccorso costruito con Arena.

Analizziamo ora di seguito la costruzione dettagliata del modello di un servizio di Pronto Soccorso tramite l'utilizzo del software Arena Simulation.

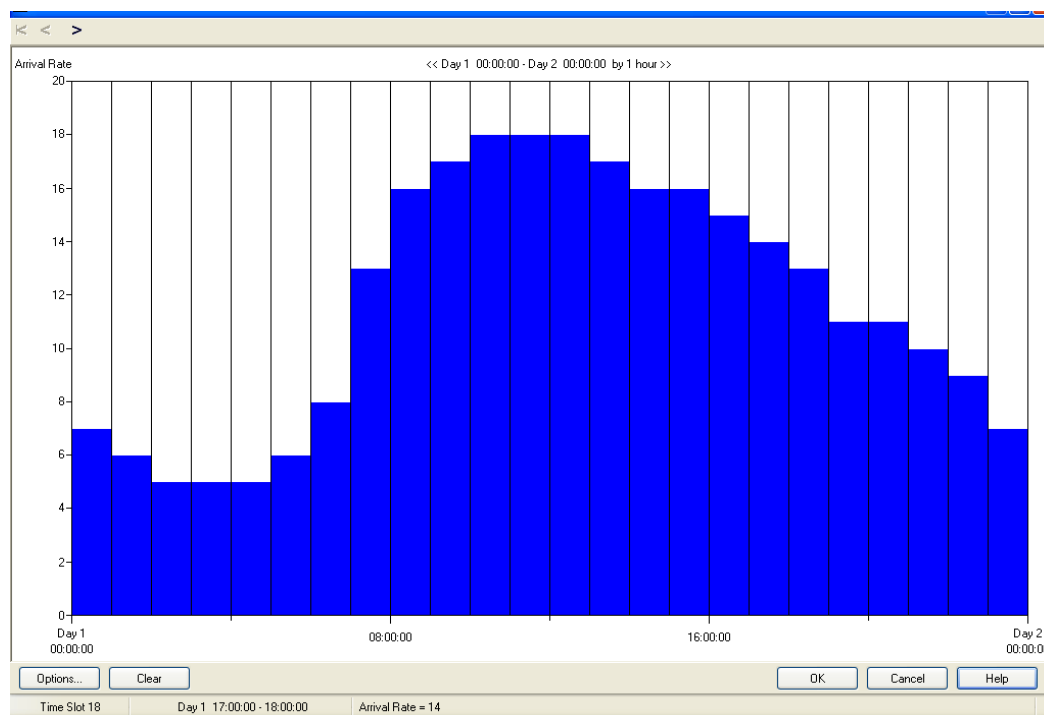
Inizialmente è stato inserito il modulo "Arrivo pazienti" che rappresenta il punto d'ingresso delle entità pazienti nel sistema.



Facendo doppio clic col mouse sul *modulo create* inserito appare la finestra in figura. Qui è inserito il nome del modulo “Arrivo pazienti” e viene creato il nome delle entità che si vogliono generare, nel nostro caso “paziente”. Di seguito viene richiesto come deve essere gestito il tempo di interarrivo; noi utilizziamo una logica “Schedule”, in alternativa si può scegliere anche tra logica random, costante o ricavabile da un’espressione matematica. Ne indichiamo il nome “Diagramma arrivi” e stabiliamo che per ogni arrivo corrisponde un’entità in entrata. Poniamo poi un massimo di arrivi che non può essere superato, nel nostro caso esemplificativo inseriamo che il max è 170 per non appesantire eccessivamente la simulazione.

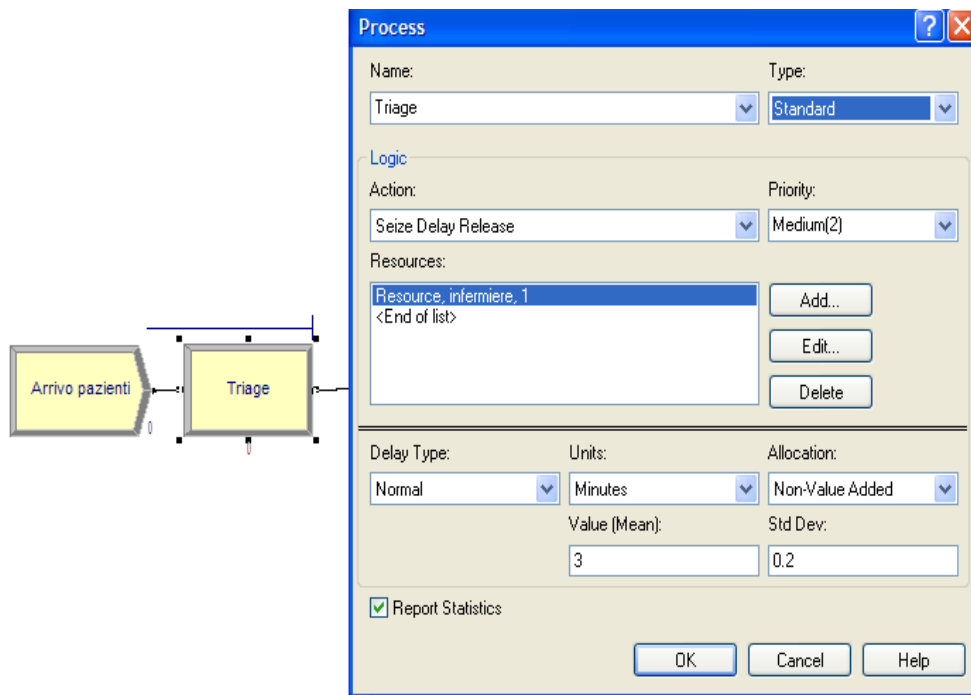
Abbiamo appunto supposto che l’arrivo dipende dall’orario giornaliero secondo una pianificazione in tabella costruita con il *modulo data schedule*.

Con questo modulo è possibile costruire in diagramma del genere riportato in figura, in cui in ascissa sono riportate le ore e in ordinata il numero di ingressi di pazienti. La costruzione prevede che ci sia un numero di arrivi maggiori di giorno rispetto alla notte.



Dopo l’arrivo dei pazienti, questi vengono tutti selezionati dal triage, governato da infermieri specializzati, dove ricevono un codice colore. Per i pazienti più gravi c’è da considerare anche l’eventualità negativa che non tutti riescano a sopravvivere alle cure.

In Arena questo passaggio viene rappresentato inserendo per primo il modulo “Triage”, un *modulo process* in cui l’entità viene processata dalla risorsa infermiere.

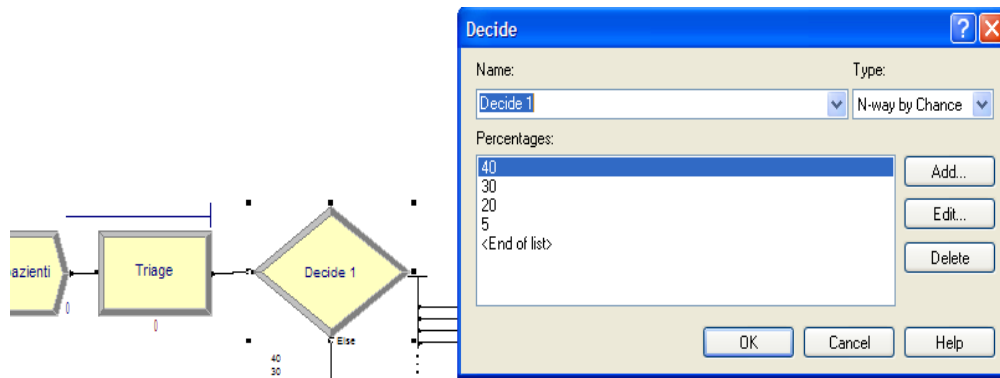


Apriamo la finestra del modulo e inseriamo quindi il nome “Triage”, applichiamo la logica *Seize delay Release* (ovvero che è richiesto un tempo di processamento, trascorso il quale una risorsa precedentemente allocata è rilasciata), assegniamo una priorità media e inseriamo la risorsa infermiere nelle apposite caselle.

In seguito viene poi chiesto il tipo di tempo di attesa per svolgere questa attività, la scelta è tra una distribuzione costante, normale, triangolare, uniforme o se ricavabile da un'espressione. Supponiamo di utilizzare nel nostro esempio una distribuzione normale con una media che si aggira sui 3 minuti e con una deviazione standard dello 0,2 e assegniamo questo tempo di processo al tempo di *Non-Value Added*.

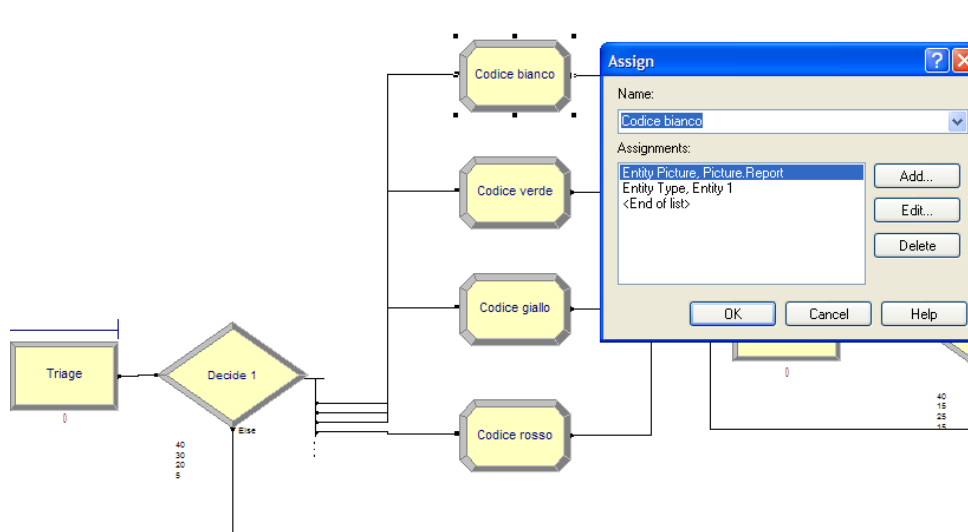
Segue dopo il modulo “Triage” il modulo “Decide 1” in cui i pazienti vengono divisi in cinque rami, ognuno con una sua probabilità di essere percorso. I pazienti che seguono il primo ramo percorrono il modulo di assegnazione “Codice Bianco”. Qui viene assegnato all'entità paziente l'attributo appunto di codice bianco. Accade lo stesso per i successivi tre rami in cui vengono assegnati i codici verde, giallo o rosso. L'ultimo ramo riguarda quei pazienti che terminano il loro percorso nel modulo “Morte”: si tratta di un modulo *dispose* conclusivo per le entità.

Per modulo “Decide 1” appare la finestra di figura successiva.



Nel nostro caso si fa riferimento a un tipo di decisione regolata da delle probabilità a più vie; inseriamo quindi nella casella *Type* la voce *N-way by Chance* e successivamente le probabilità di accadimento di ogni ramo. Nell'esempio in figura è stato riportato che il 40% dei pazienti seguirà il primo ramo e quindi diverranno dei codici bianco, il 30% saranno codice verde, il 20% codice giallo e il 5% codice rosso. I restanti 5% saranno quei pazienti che seguiranno il ramo che porta al modulo “Morte”.

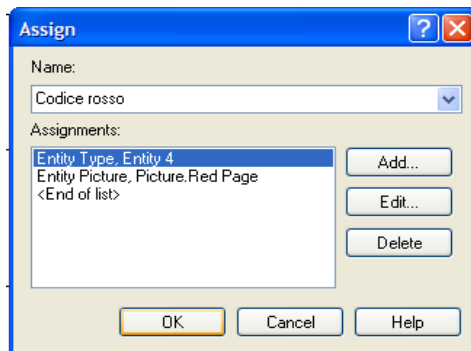
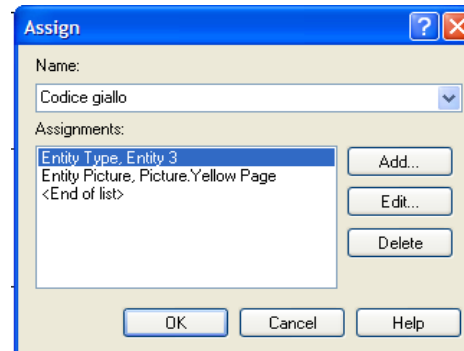
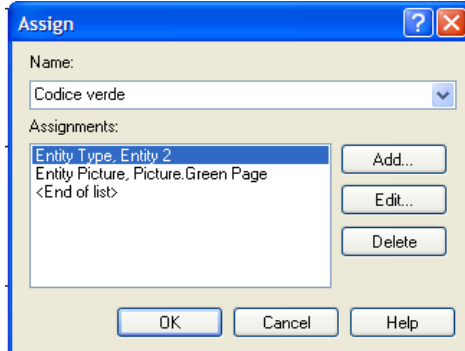
Nella figura sottostante è riportata la finestra riguardante il *modulo Assign* denominato “Codice bianco”.



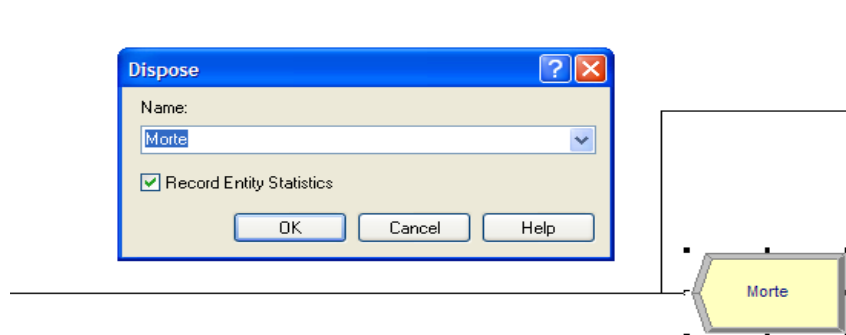
Nella casella degli assegnamenti abbiamo aggiunto una nuova *Entity picture* chiamata in *Arena Picture Report*, tale modifica permette di rappresentare nella simulazione i pazienti codice bianco con la nuova immagine di un foglio bianco stilizzato.

Abbiamo poi caratterizzato il tipo di entità (*Entity type*) passante per questo modulo con il nome di “Entity 1”. Il nome non è scelto a caso, tale denominazione permette di creare una coda di priorità secondo cui le “Entity 1” saranno le ultime ad avere la precedenza e le “Entity 4” le prime, sull’argomento ci ritorneremo successivamente quando analizzeremo la coda del modulo “Ispezione”.

Per gli altri codici colore accade lo stesso procedimento, solo che i codici verdi avranno un foglio verde che li raffigura nella simulazione e un nuovo nome “Entity 2”, i codici giallo un foglio giallo e il nome “Entity 3” e i codici rosso un foglio rosso e il nome “Entity 4” come rappresentato dalle immagini sottostanti.

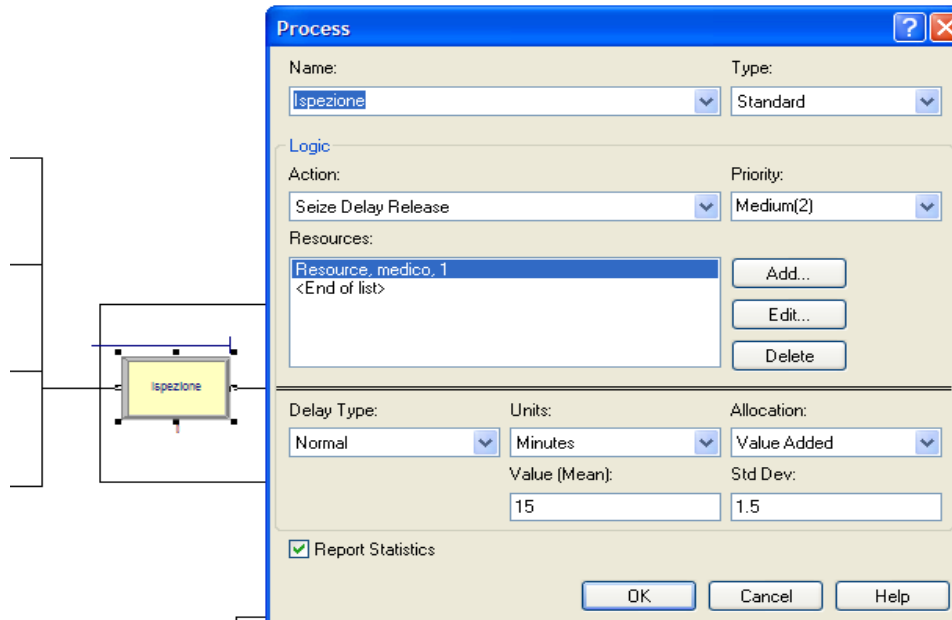


La finestra del *modulo dispose* “Morte” è rappresentata qui di seguito.



I pazienti, dopo aver ricevuto il loro codice colore, vengono chiamati nella stanza d’ispezione dove un medico effettua le prime cure per stabilizzare le condizioni del paziente per poi decidere se dimmetterlo, ricoverarlo in ospedale o in una sala d’ammissione a breve termine o, se è necessario, effettuare degli esami ulteriori. In Arena viene così inserito un modulo *process* denominato “Ispezione”. Qui i pazienti entrano in ordine seguendo le norme di

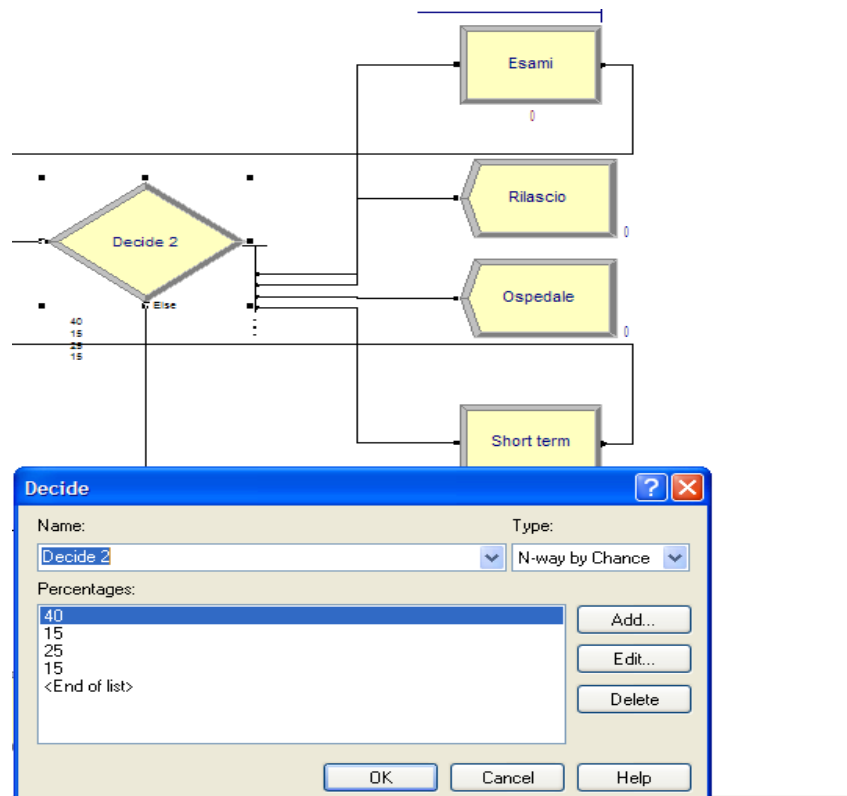
priorità e prelazione stabilite dal codice colore precedentemente assegnato dal triage. Entreranno così per primi i codici rossi, poi i gialli, i verdi e i bianchi. Nella sala d'ispezione è presente la risorsa medico che processa i pazienti per una durata media di tempo prefissata.



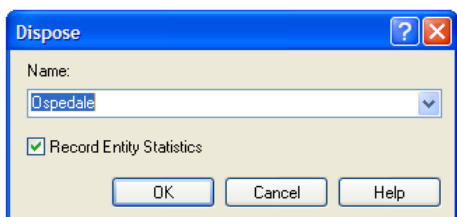
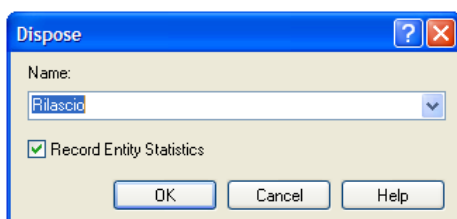
Eseguendo un doppio clic sul modulo si apre la finestra che definisce le sue proprietà. Come si vede della figura di sopra è stato inserito il nome “Ispezione”, la logica di processamento *Seize Delay Release* e la nuova risorsa medico. È stato poi stabilito nell'esempio di far riferimento a un tempo di processamento che segua una distribuzione normale con valore medio 15 minuti e deviazione standard 1.5. Il tempo di tale processo viene assegnato al tempo di *Value Added*.

Segue poi la decisione da parte del medico. Nel modello in Arena rappresentato dal modulo “Decide 2”, il medico decide se dimettere i pazienti convogliandoli nel modulo *dispose* “Rilascio”, se ricoverarli in ospedale e quindi indirizzandoli nel modulo *dispose* “Ospedale”, o se è necessario invece condurre i pazienti a sostenere degli esami interni e/o esterni oppure se tenerli in stretta osservazione in un reparto d'ammissione a breve termine.

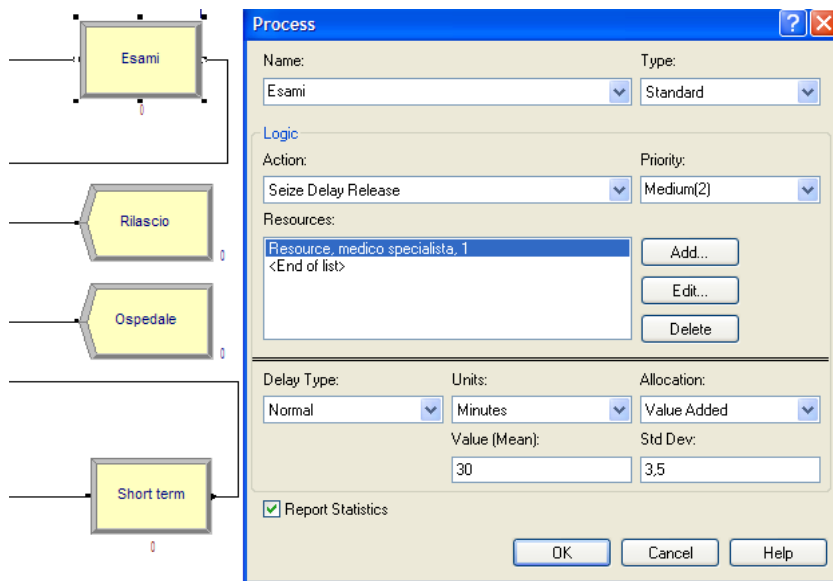
Nel modulo “Decide 2” sono state inserite le quattro probabilità di accadimento. Nell'esempio qui sotto riportato si nota che la probabilità che i pazienti effettuino ulteriori esami è del 40%, la probabilità che vengano rilasciati è del 15%, che vengano ricoverati in ospedale 25% e che vengano ammessi in un ricovero a breve termine 15%. La percentuale rimanente riguarda quei pazienti indirizzati nel modulo “Morte”.



I moduli “Rilascio” e “Ospedale” e le loro relative finestre di proprietà sono qui sotto raffigurati.

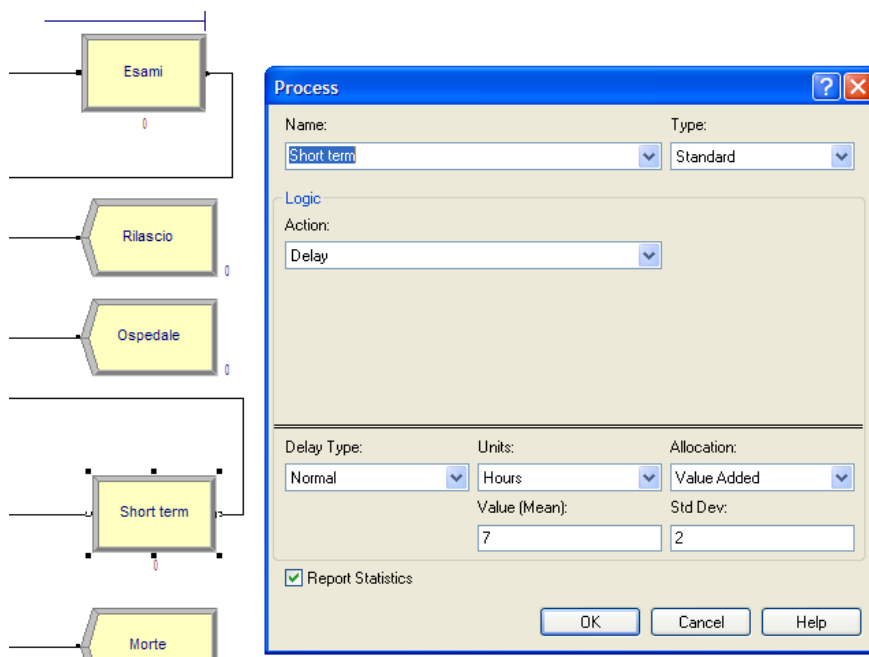


Per sostenere gli esami un paziente passa per il modulo *process* di Arena denominato “Esami”. Qui è presente la risorsa medico specialista che effettua dei controlli sui pazienti per poi inviarli di nuovo, con i risultati degli esami alla stanza d’ispezione.



Nell'immagine si nota che è stato inserito il nome "Esami" al modulo, che si riutilizza la logica *Seize Delay Release*, che è stato introdotta la risorsa "medico specialista" e che si suppone che il tempo per svolgere gli esami segua una distribuzione normale di valore medio 30 minuti e deviazione standard 3,5. Il tempo intercorso in questo processo va allocato al tempo di *Value Added*.

I pazienti ammessi in un ricovero a breve termine invece entrano nel modulo *process* chiamato qui "Short term", dove resteranno in osservazione per alcune ore per poi essere di nuovo condotti alla camera d'ispezione.



Per questo modulo si fa ricorso alla logica *Delay*, secondo cui un'entità entrante trascorre un certo tempo e poi viene liberata senza l'utilizzo di risorse che la processino.

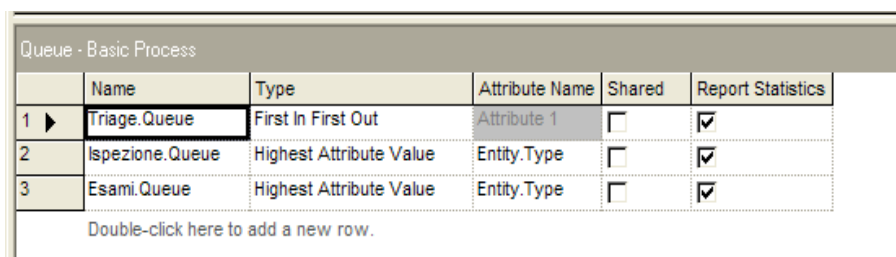
Assumiamo nell'esempio che un paziente trascorra nel "Short term" in media 7 ore con una deviazione standard di 2 ore. Questo tempo viene assegnato al tempo di *Value Added*.

Per i pazienti ritornati dal medico d'ispezione e provenienti dal "Short term" e dagli "Esami", si valuta nuovamente se indirizzarli nei moduli "Rilascio", "Ospedale", "Esami" o "Short term" e si ripete così il ciclo.

Da "Decide 2" come per "Decide 1" parte poi sempre un ramo che riguarda quei pazienti che non sopravvivono alle cure e che terminano il loro percorso nello stesso modulo *dispose* "Morte".

Code

È necessario ora gestire le code. In Arena si utilizza il modulo *data Queue* per analizzarle; cliccandolo si apre una schermata in cui compare una tabella formata di tre righe, ognuna atta a gestire la coda di un diverso *modulo process*. Vedi figura.



| | Name | Type | Attribute Name | Shared | Report Statistics |
|---|-----------------|-------------------------|----------------|--------------------------|-------------------------------------|
| 1 | Triage.Queue | First In First Out | Attribute 1 | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| 2 | Ispezione.Queue | Highest Attribute Value | Entity.Type | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| 3 | Esami.Queue | Highest Attribute Value | Entity.Type | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

Double-click here to add a new row.

Sulla prima riga viene gestita la coda al "Triage", qui si fa ricorso a una coda del genere FIFO.

Sulla seconda riga si analizza la coda all' "Ispezione" e sulla terza la coda agli "Esami". Per queste ultime due si attua la logica *Highest Attribute Value First* (il più alto attributo prima) facendo riferimento al *Entity Type* che è stato introdotto nel modello nei moduli di assegnazione. Infatti per ogni *modulo assign* era stato definita il nuovo tipo di entità: per il codice bianco "Entity 1", per il codice verde "Entity 2", per il codice giallo "Entity 3" e per il codice rosso "Entity 4".

Entità 4 risulta perciò avere il valore più alto e sarà quindi la prima in ingresso, segue Entità 3 e così via.

Grazie a questa modifica si riesce quindi a creare delle code che diano la precedenza prima ai codici rossi, poi ai codici gialli, poi entreranno i codici verdi e per ultimi i codici bianchi.

Risorse

Per procedere all'analisi delle risorse del modello si fa ricorso al modulo *data resource*.

Aperto questo modulo vengono visualizzate tutte le risorse in gioco e le loro caratteristiche. Nel nostro esempio consideriamo che le risorse infermiere e medico specialista sia fissa per tutto l'arco della simulazione e che invece il numero di risorse medico si basi su una pianificazione ottenuta col modulo *data schedule*.

| Resource - Basic Process | | | | | | | | | | | |
|--------------------------|--------------------|-------------------|----------|---------------|---------------|-------------|-------------|---------|---------------|----------|-------------------------------------|
| | Name | Type | Capacity | Schedule Name | Schedule Rule | Busy / Hour | Idle / Hour | Per Use | StateSet Name | Failures | Report Stati |
| 1 | medico | Based on Schedule | Diagra | Diagramma | Wait | 0.0 | 0.0 | 0.0 | | 0 rows | <input checked="" type="checkbox"/> |
| 2 | medico specialista | Fixed Capacity | 1 | 1 | Wait | 0.0 | 0.0 | 0.0 | | 0 rows | <input checked="" type="checkbox"/> |
| 3 | infermiere | Fixed Capacity | 1 | 1 | Wait | 0.0 | 0.0 | 0.0 | | 0 rows | <input checked="" type="checkbox"/> |

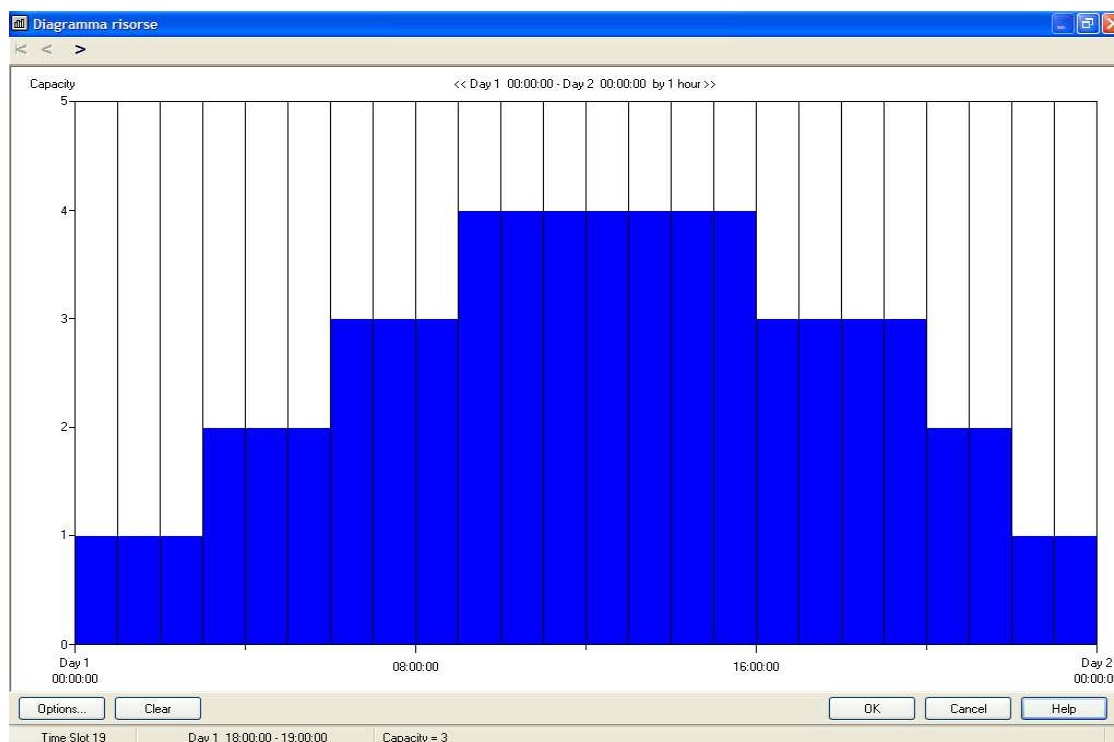
Double-click here to add a new row.

Costruiamo quindi un "Diagramma risorse" che si va ad aggiungere al "Diagramma arrivi" nel *data schedule* costruito in precedenza per gestire il numero di ingressi di pazienti al Pronto Soccorso.

| Schedule - Basic Process | | | | | |
|--------------------------|-------------------|----------|------------|--------------|-----------|
| | Name | Type | Time Units | Scale Factor | Durations |
| 1 | Diagramma arrivi | Arrival | Hours | 1.0 | 18 rows |
| 2 | Diagramma risorse | Capacity | Hours | 1.0 | 7 rows |

Double-click here to add a new row.

Nel nostro esempio il "Diagramma risorse" è costruito come appare in figura, con un massimo di 4 risorse medico tra le ore 9 e le ore 16 e un minimo di una risorsa medico tra le ore 23 e le ore 3 del giorno successivo.



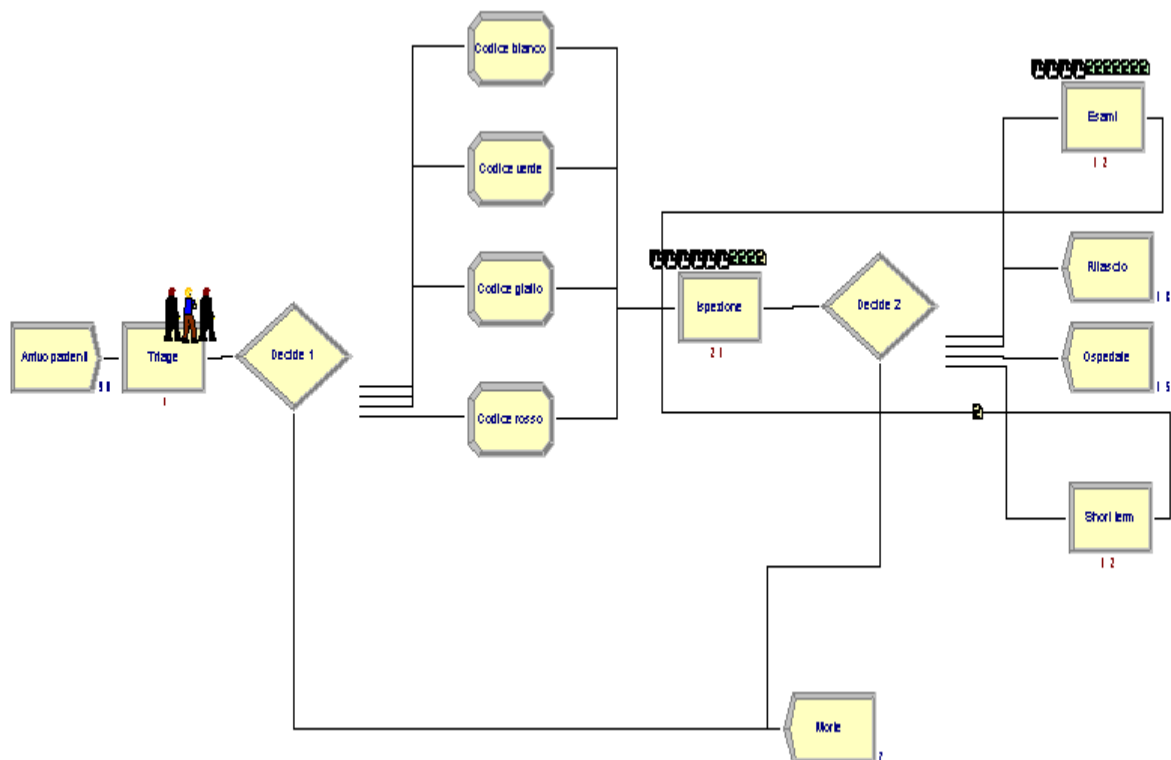
La costruzione del “Diagramma risorse” è stata attuata tenendo in considerazione che prevedibilmente saranno necessarie più risorse medico durante il giorno rispetto la notte per far fronte ad un numero di arrivi al Pronto Soccorso maggiore di giorno che di notte.

Simulazione

Dopo aver finito con la costruzione del modello è possibile lanciare la simulazione.

In Arena si attua cliccando col mouse sulla voce *Run* presente sulla barra di controllo di inizio pagina. Qui è possibile anche regolare la velocità di simulazione e la sua durata nel tempo reale; nel nostro esempio abbiamo impostato una simulazione che analizzi il servizio di Pronto Soccorso per la durata di 48 ore, al termine del quale vengono riportate le statistiche e i risultati.

La simulazione, nel nostro esempio, appare come in figura.



Si possono osservare durante la simulazione i percorsi svolti dalle entità. Si notano inoltre la coda in ingresso di pazienti al “Triage” rappresentata in figura dalle tre immagine di persone appena sopra al modulo e le code in ingresso all’ “Ispezione” e agli “Esami” rappresentate dalla sequenza di fogli colorati, secondo il codice colore, in ordine di priorità.

4.Outputs del modello

In seguito alla costruzione del modello, il software Arena offre, come già accennato in precedenza, la possibilità di ottenere una simulazione veloce del processo di studio per verificarne la correttezza. Terminata la simulazione, Arena restituisce immediatamente i risultati. Ci vengono date informazioni su più campi; le più importanti riguardano i tempi di percorrenza delle entità nel modello, la lunghezza delle code e i loro rispettivi tempi di attesa e l'utilizzo effettivo delle risorse impiegate.

I risultati vengono forniti in media evidenziando anche gli scostamenti e le criticità che si possono registrare in alcuni intervalli della giornata. Da questi dati si possono così effettuare delle valutazioni e si possono attuare delle modifiche per migliorare il modello. Per esempio potrebbe accadere che, in un certo periodo del giorno, la coda dei pazienti in attesa di ricevere le cure mediche risulti essere troppo lunga. In questo caso sarebbe opportuno intervenire nel modello per aumentare il numero di risorse mediche in modo da ridurre i tempi di attesa. Al contrario, se si dovesse notare che alcune risorse risultino sottoutilizzate bisognerebbe allora intervenire sottraendone alcune, così da poter ridurre al minimo i costi.

Per l'esempio da noi considerato finora, vengono mostrati in Arena seguenti risultati.



The screenshot displays the 'Category Overview' page in Arena. At the top left is the version number '11.37.11', at the top center is the title 'Category Overview', and at the top right is the date 'settembre 23, 2011'. Below this is a header for 'Unnamed Project'. Underneath, it shows 'Replications: 1' and 'Time Units: Hours'. The main section is titled 'Key Performance Indicators' and contains a table with two columns: 'System' and 'Average'. The row for 'Number Out' shows a value of 135.

| System | Average |
|------------|---------|
| Number Out | 135 |

Nella prima pagina di output viene indicato il numero di uscite medie durante il processo simulativo, nel nostro caso 135.

Entity

Time

| VA Time | Average | Half Width | Minimum Value | Maximum Value |
|----------|---------|----------------|---------------|---------------|
| Entity 1 | 1.5452 | (Insufficient) | 0.2145 | 12.9594 |
| Entity 2 | 1.9832 | (Insufficient) | 0.2153 | 14.0842 |
| Entity 3 | 3.6143 | (Insufficient) | 0.2179 | 16.3503 |
| Entity 4 | 6.3976 | (Insufficient) | 0.2415 | 22.1840 |
| paziente | 0.00 | (Insufficient) | 0.00 | 0.00 |

| NVA Time | Average | Half Width | Minimum Value | Maximum Value |
|----------|------------|----------------|---------------|---------------|
| Entity 1 | 0.04970262 | (Insufficient) | 0.04223383 | 0.05904678 |
| Entity 2 | 0.05079447 | (Insufficient) | 0.04445243 | 0.06000703 |
| Entity 3 | 0.04997383 | (Insufficient) | 0.04004909 | 0.05805092 |
| Entity 4 | 0.05108477 | (Insufficient) | 0.04394715 | 0.05417706 |
| paziente | 0.04989937 | (Insufficient) | 0.04623491 | 0.05496645 |

| Wait Time | Average | Half Width | Minimum Value | Maximum Value |
|-----------|---------|----------------|---------------|---------------|
| Entity 1 | 4.0614 | (Insufficient) | 0.2628 | 40.6829 |
| Entity 2 | 11.2812 | (Insufficient) | 0.05790671 | 36.6795 |
| Entity 3 | 2.2300 | (Insufficient) | 0.04291201 | 9.8412 |
| Entity 4 | 0.4924 | (Insufficient) | 0.01265928 | 1.3309 |
| paziente | 0.0980 | (Insufficient) | 0.00 | 0.4045 |

Nella seconda pagina si riportano i risultati dei tempi di percorrenza nel modello delle diverse entità.

Si noti come i valori dei tempi siano riportati per ogni entità differente in media e si evidenziano anche gli scostamenti riportando i valori massimi e minimi.

Ci si ricordi che nel nostro modello avevamo denominato con Entity 1 i pazienti codice bianco, con Entity 2 i pazienti codice verde, con Entity 3 i pazienti codice giallo e Entity 4 i codici rosso.

Con *VA Time* ci si riferisce al tempo di valore aggiunto trascorso dalle entità nei moduli *process*, nel nostro esempio avevamo allocato questi tempi nei moduli "Ispezione" "Esami" e "Short Time".

Con *NVA Time* ci si riferisce al tempo di non valore aggiunto, tempo che nel nostro esempio avevamo assegnato al processamento delle entità nel modulo "Triage".

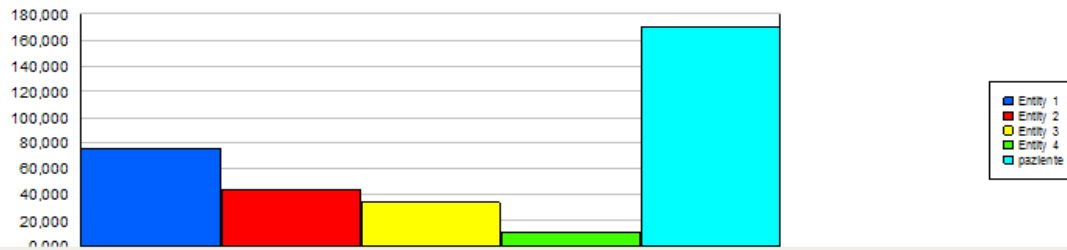
Con *Wait Time* si intende il tempo di attesa alle varie stazioni.

Time

| Total Time | Average | Half Width | Minimum Value | Maximum Value |
|------------|---------|----------------|---------------|---------------|
| Entity 1 | 5.6563 | (Insufficient) | 0.5820 | 41.7468 |
| Entity 2 | 13.3152 | (Insufficient) | 0.3585 | 40.7853 |
| Entity 3 | 5.8943 | (Insufficient) | 0.3242 | 23.1716 |
| Entity 4 | 6.9411 | (Insufficient) | 0.3074 | 23.0497 |
| paziente | 0.1479 | (Insufficient) | 0.05236096 | 0.4507 |

Other

| Number In | Value |
|-----------|---------|
| Entity 1 | 76.0000 |
| Entity 2 | 44.0000 |
| Entity 3 | 34.0000 |
| Entity 4 | 11.0000 |
| paziente | 170.00 |



Nella terza pagina di output, come si vede dalla figura di sopra, è riportato il *Total Time* per ogni entità, si riferisce al tempo calcolato da quando le entità entrano nel sistema fino alla conclusione della simulazione quando vengono riportate le statistiche.

È riportato in seguito in una tabella e attraverso un rappresentazione grafica il numero di ingressi (*Number In*) di entità per tipo nel modello.

Queue

Time

| Waiting Time | Average | Half Width | Minimum Value | Maximum Value |
|-----------------|---------|----------------|---------------|---------------|
| Esami.Queue | 6.8123 | (Insufficient) | 0.00 | 39.9204 |
| Ispezione.Queue | 0.9216 | (Insufficient) | 0.00 | 4.6610 |
| Triage.Queue | 0.1016 | (Insufficient) | 0.00 | 0.4743 |

Other

| Number Waiting | Average | Half Width | Minimum Value | Maximum Value |
|-----------------|---------|----------------|---------------|---------------|
| Esami.Queue | 35.0341 | (Insufficient) | 0.00 | 56.0000 |
| Ispezione.Queue | 5.7985 | (Correlated) | 0.00 | 41.0000 |
| Triage.Queue | 0.3597 | (Insufficient) | 0.00 | 10.0000 |

Nella quarta pagina si parla di code.

Viene presentata la coda in termini di tempo di attesa (*Waiting Time*) e la coda intesa in numero di entità (*Number Waiting*) che attendono di venir processate. Nel nostro esempio sono riportate le code in ingresso agli Esami, all'Ispezione e al Triage. Come si può vedere dall'immagine sono sempre riportati i valori in media e i valori max e min.

Resource

Usage

| Instantaneous Utilization | | | | |
|---------------------------|---------|----------------|---------------|---------------|
| | Average | Half Width | Minimum Value | Maximum Value |
| infermiere | 0.1778 | (Insufficient) | 0.00 | 1.0000 |
| medico | 0.6276 | (Insufficient) | 0.00 | 1.0000 |
| medico specialista | 0.9700 | (Insufficient) | 0.00 | 1.0000 |

| Number Busy | | | | |
|--------------------|---------|----------------|---------------|---------------|
| | Average | Half Width | Minimum Value | Maximum Value |
| infermiere | 0.1778 | (Insufficient) | 0.00 | 1.0000 |
| medico | 1.5742 | (Insufficient) | 0.00 | 4.0000 |
| medico specialista | 0.9700 | (Insufficient) | 0.00 | 1.0000 |

| Number Scheduled | | | | |
|--------------------|---------|----------------|---------------|---------------|
| | Average | Half Width | Minimum Value | Maximum Value |
| infermiere | 1.0000 | (Insufficient) | 1.0000 | 1.0000 |
| medico | 2.6689 | (Insufficient) | 1.0000 | 4.0000 |
| medico specialista | 1.0000 | (Insufficient) | 1.0000 | 1.0000 |

| Scheduled Utilization | |
|-----------------------|--------|
| | Value |
| infermiere | 0.1778 |
| medico | 0.5898 |
| medico specialista | 0.9700 |



Resource

Usage

| Total Number Seized | |
|---------------------|---------|
| | Value |
| infermiere | 170.00 |
| medico | 302.00 |
| medico specialista | 93.0000 |



Nella quinta e sesta pagina sono riportate le statistiche di risorse.

Instantaneous Utilization (utilizzo istantaneo) è determinato per ricavare il grado di utilizzo

della risorsa in ogni istante calcolato con una media ponderata nel tempo. Qui viene tenuto conto del tempo in cui l'utilizzo della risorsa non era stato programmato.

Per *Number Busy* si intende il numero di unità di risorse occupate.

Per *Number Scheduled* si intende il numero di unità di risorsa in programma.

Scheduled Utilization è calcolato dividendo il valore medio del *Number Busy* per il valore medio del *Number Scheduled*. Questo è molto utile quando si è alla ricerca di un valore cumulativo di utilizzo e si tiene conto del tempo di utilizzo attuale della risorsa programmato.

Viene poi in conclusione riportato il *Total Number Seized*, il numero di entità processate da ogni risorsa, riportando anche un grafico annesso.

CONCLUSIONI

Il pacchetto software Arena Simulation della Rockwell Automation si è rivelato uno strumento molto utile e di facile utilizzo per la costruzione e implementazione di un modello simulativo per rappresentare efficacemente il servizio di Pronto Soccorso.

Si è riusciti a creare un modello che è ,allo stesso tempo, flessibile, generalizzato, di semplice lettura e che descriva ogni aspetto essenziale della struttura.

Attraverso Arena si sono potute eseguire simulazioni veritiere in grado di rispecchiare al meglio il funzionamento dei diversi servizi di Pronto Soccorso, da quelli che accolgono un numero ridotto di pazienti a quelli dal bacino di utenza più ampio. Si possono infatti effettuare simulazioni variando il numero di pazienti in ingresso e il numero di risorse per ogni reparto con delle piccole modifiche senza alterare la struttura base del modello.

L'utente finale, seppur non a conoscenza del linguaggio Arena, è in grado di fare suo il modello e analizzare al meglio i risultati ottenuti in seguito a una simulazione.

Bibliografia

Kelton W. David, Sadowski Randall P., Swets Nancy B., 2009, Simulation with Arena, Oxford, OX, United Kingdom: McGraw Hill international edition.

Facchin Paola, Rizzato Elena, Romanin Jacur Giorgio, 2010, "Emergency department generalized flexible simulation model".

S. George, S. Read, L. Westlake, B. Williams, P. Pritty and A. Fraser-Moodie, Nurse triage in theory and in practice, Arch Emerg Med.10(3) September 1993-228.

S. L. Albin, S. Wassertheil-Smoller, S. Jacobson and B. Public Health 65(10) October 1975: 1063-1068.

T. Ruohonen, P. Neittaanmaki, J. Teittinem, Simulation Model for Improving the operation of the Emergency Department of Special Health Care, Proceedings of 2006 Winter Simulation Conference ed. L. F. Perrone, F.P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol and R. M. Fujimoto: 453-458.

G. W. Evans, T. B. Gor, E.Unger, A Simulation Model For Evaluating Personnel Schedules In A Hospital Emergency Department, Proceeding of the 1996 Winter Simulation Conference ed J.M.Charnes, D.J.Morrice, D.T. Brunner and J.J.Swain: 1205-1209.

T. D. Clark, C. W. Waring, A simulation approach to analysis of emergency services and trauma center management, Proceedings of the 1987 Winter Simulation Conference A.Thesen, H. Grant, W. D. Kelton: 925-931.

www.arenasimulation.com